Tevo Tarantula i3 3D Printer

I have recently purchased an incredibly cheap 3D printer that supposedly should be quite capable and produce Prints of acceptable quality. The brand name is Tevo, the model is called Tarantula i3. That's because it is one of many Prusia i3 clones.

The printer costs only about $200 from AliExpress but what you save on money you will have to invest in time. While the assembly manual seems to be improved over earlier versions, it is only nearly complete for about half the build. All of a sudden a lot of details and steps are missing and you have to google around for images of the finished product to figure out what goes where. There are some pretty good assembly videos on Youtube, but following videos definately takes alot longer than following a decent assembly manual. Anyway, it was alot of fun putting the machine together. It somehow reminded me of the days when I as a little boy sat in my parents house in the livin groom with all the parts of my RC cars layed out, carefully following the instructions to put it together. The Tarantula delivers much of a similar experience, so that's fun, just take your time.

Also it could use some improvement here and there. One of the most annoying short comings of the kit is the absence of washers to go on the mounting plates where the rollers are attached to. the simple addition of a washer would help to keep the wheels straight and to keep the nuts from diggin into the plexy glass when tightening the assembly. This is something i will modify pretty soon, but first I want to get the printer runnin in its original state and then continue from there.

I have purchased the model with the standard 20x20cm bed and single extruder. The only optional extra I have purchased is the inductive z-axis sensor to auto-level the bed. The only 3D Printer I Played around with so far was the XYZ Printing Da Vinci 1.0 which was everything but a joy when it came to levelling the bed. And having your bed level or compensated is very important to get a decent print quality from the first layer on, so i decided i'd get this option to save some time.

Despite owning that XYZ Printer for some years now i still consider myself a novice to 3d printing as i have used it very rarely. It was quite frustrating to spend more time getting the printer calibrated and set up than i spent time printing or designing things to print. Also i was a complete novice to CAD which ment it took forever to come up with my own designs.

Assembly

I won't write anything about the assembly here except for this: search on youtube for Tevo tarantula assembly videos. there are good tutorials that will help you and plan to spend about 12 Hours on your first one. The second will probably be a lot faster, as most of that time goes into figuring out what goes where and which way around :) you can do the first half of the assembly by just following the included manual but from then on you need to google and research

Axis

just to get this clear right up front: everybody talks about X Y and Z axis. But which is which? The X axis is the one where the Extruder is attached. When you look at your printer from the Front it is the

axis that moves left to right.

The Y axis is the one on which the printbed moves, so from front to back

And Finally the Z axis is up and down.

The 0,0,0 point on my printer is in the front left corner of my bed, so the X axis all the way to the left, The print bed all the way to the back and the Z axis touching the bed.

Modifications

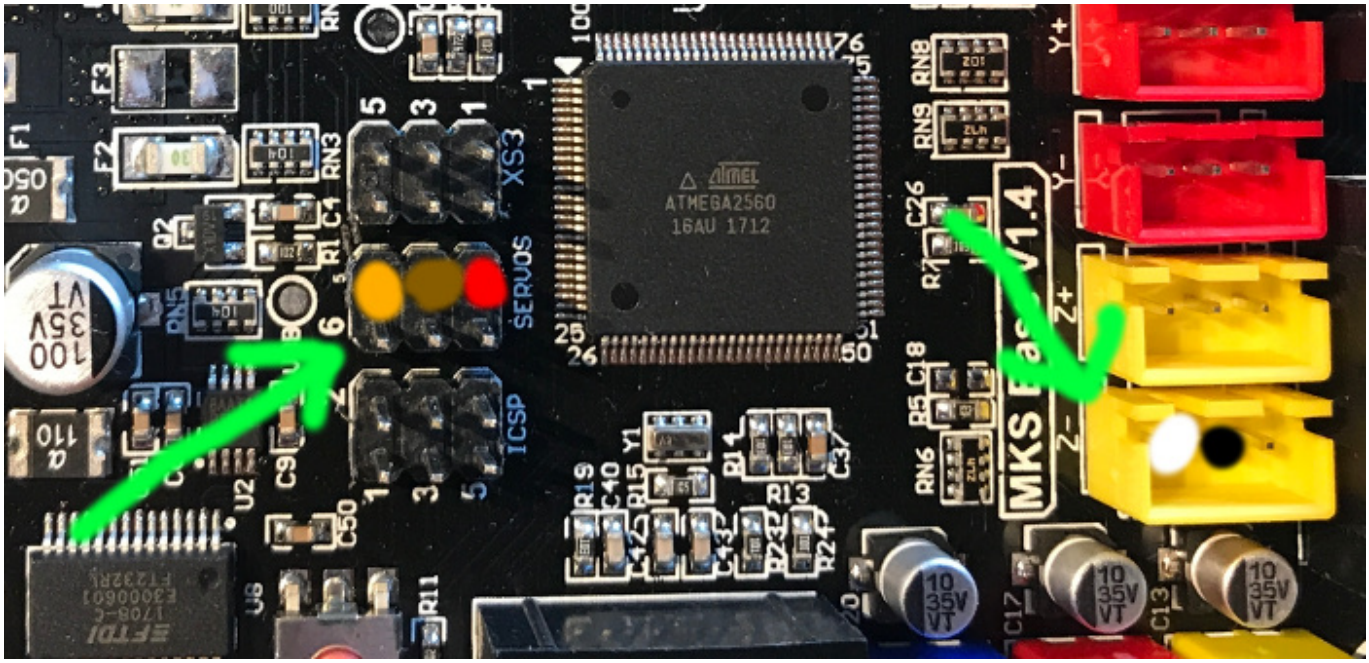
Fan Duct

the biggest let-down of the tevo in my eyes is the fact that there is not sufficient layer cooling. this results in poor print quality giving everything a pretty melted look. I printed the LPA Fanduct v5 which seems to have a pretty good aerodynamic design allowing a single 40mm fan to cool the print excellently. this led to a huge improvment of print quality. Also the fanduct includes a mount for both the SN-05N contact-less bed leveling sensor as well as the BLTouch sensor which i converted to after a few prints

Bed Leveling Sensor

I first ordered my Tevo with the contact-less SN-05N sensor. however i noticed that the sensor needs to be quite close to the print bed in order to detect it and this turned out to be a problem with the offset it has from the print-head in my setup (with the above fanduct. If you mount the sensor too high it will not detect the bed before the filament nozzle hits the bed which leads to the nozzle crashing into your bed and then pushing it down violently as the sensor will never detect the bed in such a case. If you mount it too low it will mess up your prints as it will hit your latest layer. combine this with a slight wrapping of a print and you have a desaster. The problem is, that due to the low range, the gap between too high and too low is very narrow (less than a mm). this means, that if you have a large offset from the printhead (which is the case with most cooling fan setups that distribute the air 360° around the nozzle) your bed needs to be almost perfectly leveled or you risk for the sensor to be too high and too low at the same time, depending on the corner you are measuring in. Last but not least, the tevo's print bed may warp slightly which is why i would like to try to print on a mirror or glass in the future and i would also like to experiment with a print plate that would make it easy to release the print from the surface later on. both these methods increase the distance between the nozzle at 0mm altitude and the metal print bed which won't work with the SN-05N sensor.

For all these reasons I quickly decided to switch to a BLTouch sensor. I ended up ordering a clone called TLTouch. Wiring to your tevo tarantula might be different from mine, depending on the controller version you have. I got mine with the MKS Base v1.4 board, thats the one with the blue,red,yellow and white sockets on it.



connect the BLTouch to the two sockets where the green arrows point to. Orange is `S(ignal)` pin, Brown is `GND`, Red is `5V`, White is `ZMIN` and Black again `GND`. These are the lables you can see on the BLTouch sensor.

if you have an older board, you might find the wiring here:
http://mon.im/2016/06/tevo-tarantula-bltouch-setup.html

BLTouch Settings

set these settings (search for them, they are already in the Configuration.h file, then modify or uncomment what you need) The offsets are for the LPA v5 fanduct, luckily they provide them in the documentation.

```
#define Z_MIN_PROBE_ENDSTOP_INVERTING false
#define BLTOUCH
#define X_PROBE_OFFSET_FROM_EXTRUDER 39  // X offset: -left  +right  [of the
nozzle]
#define Y_PROBE_OFFSET_FROM_EXTRUDER -42 // Y offset: -front +behind [the
nozzle]
#define Z_PROBE_OFFSET_FROM_EXTRUDER 0   // Z offset: -below +above  [the
nozzle]
#define X_MIN_POS 0
#define Y_MIN_POS 42
```

Z axis stabilization

one of the structurally weakest point of the Tevo is the off-center Z-axis leadscrew that is hanging by the flex-coupling of the servo.. there are two solutions to this problem, one is to mount the servo at the bottom and then have the Y-axis push down on it, the other is to add a support at the bottom which will hold the weight of the y-axis so that it is not hanging on the flex coupling. I decided to go with the support, as it seemed more convenient to implment

lower support (needs a 608ZZ bearing) M4 t-nuts

Firmware Setup

Since I have the Auto Level option I had to dig into how to calibrate that thing right away as the auto homing will cause the Z axis to crash into the printing bed with the standard firmware configuration. I decided to get the newest Marlin Firmware and go from there. What you essentially need is the Configuration.h and Configuration_adv.h files that contain the parameters for a Tarantula in your configuration and then you can start from there to tweak the settings.

Compiling and Flashing

that's quite easy: just download the latest Arduino IDE and install it. a little warning here, the one included in the Ubuntu Repos is very old.. for Ubuntu 17.04 there was still a version 1.0.x of the arduino IDE, Marlin currently needs at least 1.6.x to work and you currently can download 1.8.3 from the Arduino webpage. then double check that your control board has the Atmel 2650 chip on it and select the board accordingly in the Arduino IDE. now you can simply open the Marlin.ino file, change your config settings and hit upload.

Settings

To get started, you can download my configuration.h and compare it to the defaults. I recommend you really read the comments and figure out yourself if you need some settings to be different from mine. This file is for Marvin 1.1.9 and a 20x20cm Tevo with BL Touch Sensor and the LP-Fanduct to mount it. I will explain some parameters below i had some difficulty to understand or to get right

Auto Bed Leveling

There are many options as to how the auto bed leveling should work. I went with the Bilinear option which will just check 4 corners and then calculate the bed surface. As for my probe i selected a FIX_MOUNTED_PROBE which is what I have.

to EEPROM or not to EEPROM

in quite a few configs i found it was recommended to disable the EEPROM. The EEPROM will be used to store some settings on the controller without hardcoding them into the firmware. When developing firmware, it might certainly be great to disable the EEPROM as you won't have to deal with clearing it while you make adjustments to the hardcoded settings. However, in order to get some of the parameters right it might be easyer if you can adjust those via some Gcode commands or the display and then copy your final results into your firmware configuration.h file for the final upload. also it seems that some settings are not available in the configuration file, at least i didn't find them there. So i recommend turning the EEPROM on and using it to help you get your settings done.

Sensor Offsets

The endstops for the X Y and Z axis are not mounted at the 0,0,0 point of my printer, they are mounted at a certain offset. So when the bed moves all the way back it will hit the Y limit switch but by that time, the Extruder will be 40mm off the bed. There are two solutions to that: Mechanically you could move your end switch forward so that it hits the bed when it reaches its Y=0 position or you can adjust the offset by Softare. Some people do it one way, some the other, that's why you will always need to fiddle with this setting when you take someone's config files and start from there. When I assembled my Tevo i didn't dive right into getting the software set up due to a lack of time, so it was sitting around for a month or two. by the time got started with the software setup i had forgotten that i could move that Y axis endstop so i solved it by adjusting my software and only now i realized i could have adjusted that enstop :)

So for the homing to work when your endstops are at the physical end of the axis and not at the logical 0 points, you need to configure the probe offsets. here are mine:

```
#define X_PROBE_OFFSET_FROM_EXTRUDER 39  // X offset: -left  +right  [of the
nozzle]
#define Y_PROBE_OFFSET_FROM_EXTRUDER -42 // Y offset: -front +behind [the
nozzle]
#define Z_PROBE_OFFSET_FROM_EXTRUDER 0   // Z offset: -below +above  [the
nozzle]
```

As you can see here, my Y endstop is 40 mm behind the actual 0 point.

X Y Z Axis positions

The problem is, that the probe offsets above are not automatically taken into consideration to calculate the travel limits and the offset for Axis positoins relative to the endstops.

There is a Wiki article in the Marlin Wiki describing how tho set your Axis positions correctly. While I agree withthe Author, that Method 1 (hardcode everything in the firmware) is the best solution, i couldn't figure out the right parameters to do so! What worked for me in the end was Method two which requires setting some offsets in the firmware. Hence we need to enable EEPROM for this.

So here is what i set in my config file:

```
#define X_MIN_POS 0
#define Y_MIN_POS 42
#define Z_MIN_POS -1
#define X_MAX_POS X_BED_SIZE
#define Y_MAX_POS ( Y_BED_SIZE + Y_MIN_POS )
#define Z_MAX_POS 200
```

But this alone seems to be not enough. what happens with this setting is, that now my Y axis will move to the 0 position when i send a `G1 Y0` command (this means, move the y axis to 0). So far so good, but when i send `G1 Y20` i would expect to move now 20mm away from the 0 point, but unfortunately nothing will happen until i move further than 40 away from 0. only then my bed will start moving. So in order to reach the Y=200 (the rear end of the bed) i have to send `G1 Y240` which

needs to be fixed.

so fare i don't know where in the Configuration.h file i could fix that, the only possibilty i have found is by programming the Offest into the EEPROM by running this G-code:

```
M206 X0 Y-42 Z0
M500
```

the first line is to set the X offset to 0 and the y offset to -42, the second line is simply to store this to eeprom.

now my coordinates work, when I move the printer to Y=5 it moves 5mm to the back :)

Setting the Z Axis Sensor offset

In order to get the first layer to a) stick to the bed and b) not be too flat the Z axis needs to have a very precise 0 point. However, the Z probe usually needs some fiddeling until it is mounted correctly so that it won't interfere with your print but still reach the print bed before the nozzle does. So the only viable option is, to configure this in the firmware. Again we will use the eeprom to save that setting. Here it even makes sense, because every time you remove that sensor, i.e. after you have printed the fan duct and mounted it, you will need to play around and adjust this setting.

In order to adjust the Z axis, i first sent G28 to auto home all axis, then G29 to autolevel the bed, i then used `G1 Z0` to move the Z axis to the 0 position. I put a sheet of paper under the Nozzle and set my first guess what the offset could be. it's better if you set your offset too little so that your extruder won't crash into the printbed

```
M851 Z-0.8
```

Now run G28 agin and go again to `G0 Z0`. Repeat this process until the nozzle makes contact with the paper so that the paper can still slide in and out from underneath the nozzle, then increase the offset by another -0.1mm

once done, save your settings

```
M500
```

Now make sure that your Slicer adds these Gcodes to the begining of every job, to make sure your bed leveling has been done:

```
G28 ; auto home
G29 ; auto bed level
```

Now print something and see if you are happy with your first layers

by the way, you can also set this offset in the marlin menu under `Control–>Motion–>Z Offset`

Bed Leveling VS G28

in older versons of the Marlin Firmware it was not possibel to enable RESTORE_LEVELING_AFTER_G28 in `Configuration.h` which means, that as soon as you run G28 to auto-home all axes your auto-bed-leveling mesh is lost and you need to re-run G29. So if your Slicer puts G28 at the begining of a print-job it will kill your bed-leveling efforts! So either configure your slicer to do the auto bed leveling after the homing or set that constant in Configuration.h

Slicer Software and Settings

Whatever software you are using, make sure that the Start GCode contains G28 and G29 for auto height and auto bed leveling.

for more detailed settings see How I 3d Print - Slicer settings

Dead Stepper driver - workaround

i recently printed a quite tall structure over night. it tunred out, that the screws of the z-axis wheels did not clear the mounting bracket for the extruder. as a result of the z-axis missed most of its steps during several hours and was jammend. sadly it turned out that after this the Z-axis stepper driver was broken. when i moved the z-axis the stepper would randomly move forward or backward or just humm. i've tried to adjust the potentiometer but with no luck.

as a workaround i changed the marlin config, so that it's now using the E1 (second extruder, which i don't have) stepper driver in place of the z-axis. i could simply move the molex connector over from Z to E1 and continue to print.

to implement the workaround, i had to start my Arduino IDE, open the Marlin project and edit the `pins_RAMPS.h` file. i searched for the Z- and E1 axis and swapped the values for the pins, they now look like this:

```
#define Z_STEP_PIN         36
#define Z_DIR_PIN          34
#define Z_ENABLE_PIN       30
#ifndef Z_CS_PIN
  #define Z_CS_PIN         44
#endif

#define E1_STEP_PIN        46
#define E1_DIR_PIN         48
#define E1_ENABLE_PIN      62
#ifndef E1_CS_PIN
  #define E1_CS_PIN        40
#endif
```

i then re-compiled and uploaded the firmware and could print right away.

of course i've also trimmed the bracket :)

e1 heating failed

This message is shown when Marlin gets unexpected temperature readings during the heating of the extruder. It expects a certain increase in temperature when it adds a certain amount of power to the heating element. if expected and measured values differ too much, Marlin aborts.

If you see this message it is most likely due to one of the following two:

1.) the heating element has come loose. Try if it still sits tightly in the holder. otherwise re-tighten it and try again

2.) a PID re-tune is necessary. See all the details about it on the PID Tuning Page in the Reprap wiki. To keep it short, here are the GCODE commands to execute on your Marlin powered printer:

```
M303 E0 S200 C8
```

this will run **8 heat cycles** on extruder **E0** at **200°C** and measure how the heating element reacts to the power inputs given

it will then sohow a final result giving you a **Kp, Ki and Kd** value which you should put in your configuration.h file. If you have EEPROM enabled, you can simply set the parameters via GCODE and save them. Here is an example for the following values:

```
Kp: 19.56
Ki: 0.71
Kd: 134.26
```

and this is the GCODE to save it to the printer:

```
M301 P19.56 I0.71 D134.26
```

and now save it

```
M500
```

From:
http://wiki.psuter.ch/ - **pswiki**

Permanent link:
http://wiki.psuter.ch/doku.php?id=tevo_tarantula_i3_3d_printer

Last update: **23.02.2020 12:03**