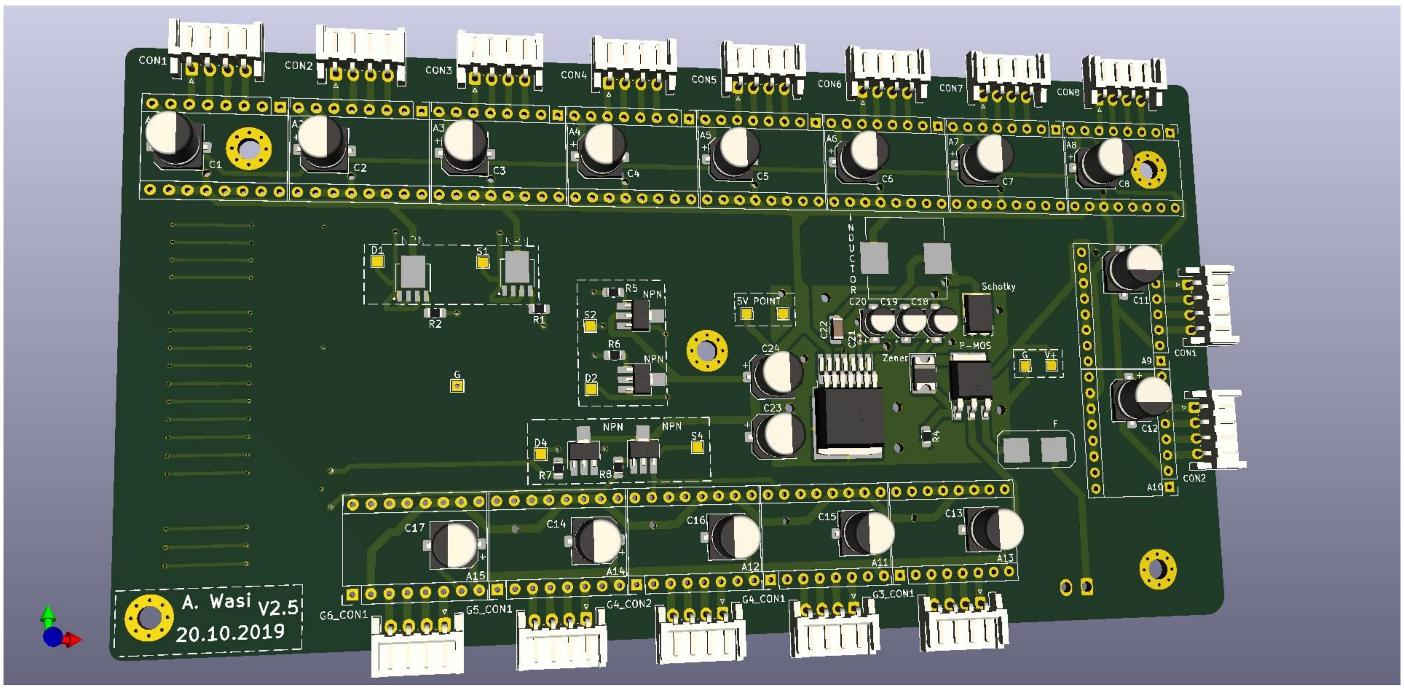
SENSA

Multi Channel Synchronous Stepper Motor Controller

aka „TonsOfSteppers“



Verzion: 2.0

Autor: Ariyan Wasi (ariyan.wasi@sensa-group.net)

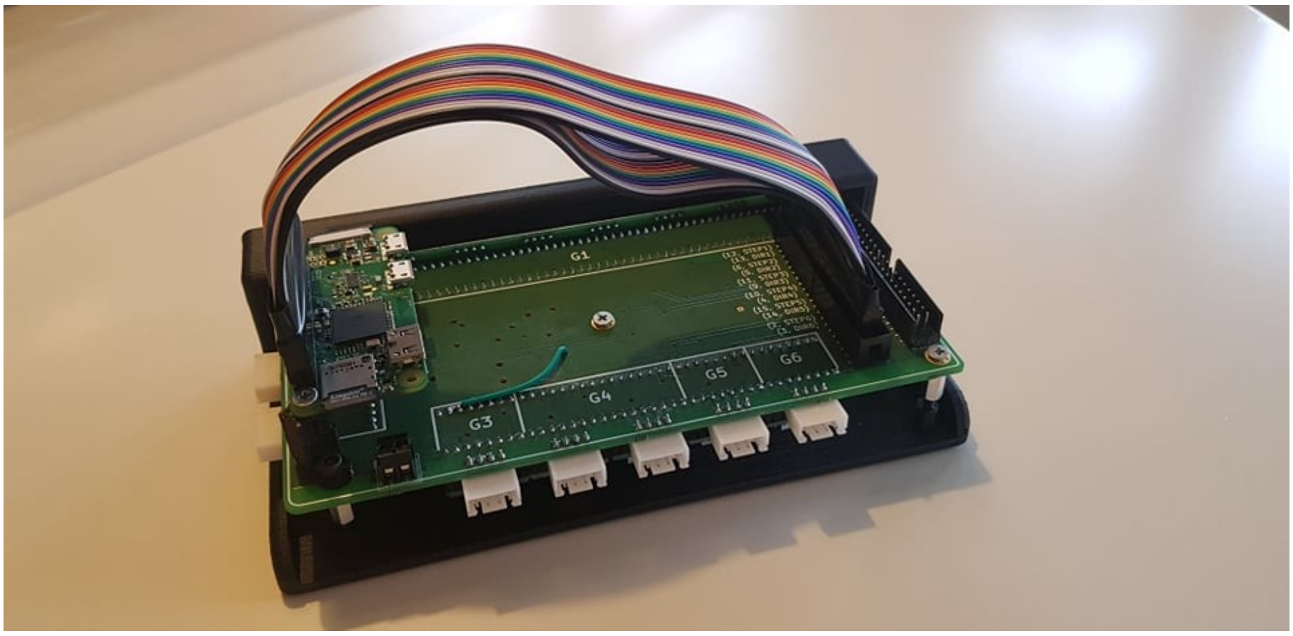
SENSA, 2020

Table of Contents

1 Hardware history

Ver	**Rev**	**Datum**	**Autor**	**Opis**
1	1	01.11.2019.	Ariyan Wasi	Initial version
2	1	01.02.2020.	Ariyan Wasi	Updagred Hardware and looks

2 Project Principle



The controller is made for the simultaneous control of large groups of stepper motors that need to work at the same time synchronously. The driver board has the ability to drive one motor on its own, up to eight motors at a time. The driver is made to be controlled via a raspberry pie with the standard forty pin header. However, it can be run easily with any other mass-market development board. The board can drive up to fifteen motors at once.

2.1 Workings

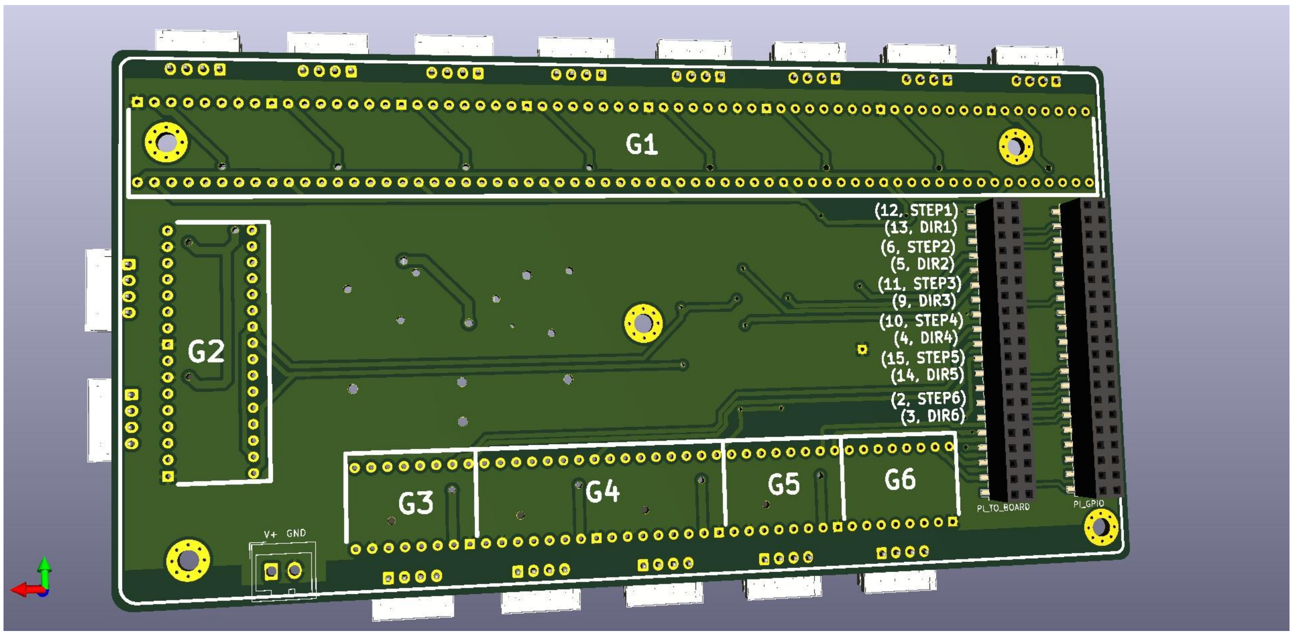
The driver board is capable of supplying a wide array of bipolar stepper motors (like most „nema“ types of stepper motors) due to its large motor supply voltage range from 8 to 40 volts. At any voltage point, the stepper motor will be able to generate 5V on its on to power the logic. It does this via a high efficiency and high amperage switching regulator.

Each of the possible fifteen motors is driven by a switchable A4988 Polulu driver, this driver was chosen as there are many possibilities for upgrading in the future as its a pretty standard footprint and pinout. (https://www.pololu.com/product/1182)

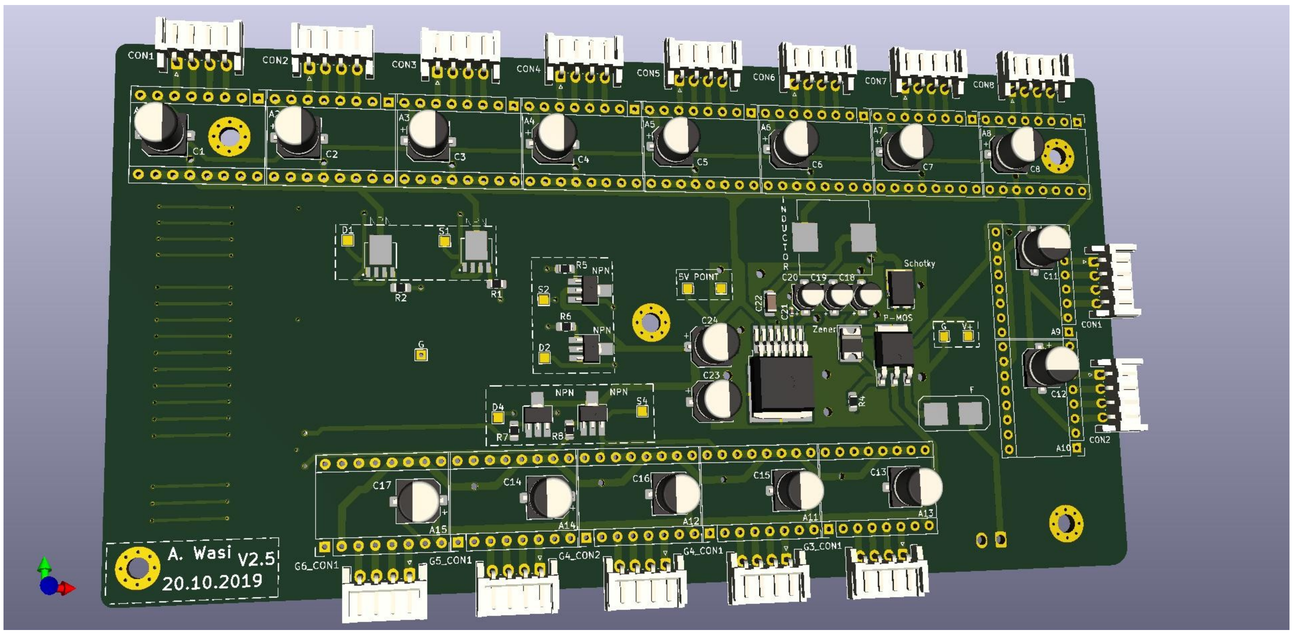
Each group of stepper motors achieves synchronous movements due to a pair of high power transistors. One transistor is to carry the direction signal of the rotation of the stepper motors, the other is to carry the signal that defines the number of „steps“ that the stepper motors should take. The whole backing is installed in a case that hides the sensitive electronics on the bottom while being exposed on the top part for ease of access to connectors and for the ability to read the silkscreen.

3 Visualisation Section

FRONT:



BACK:



3.1 Front Part

On the front of the PCB we have:

- The motor supply input.
- Raspberry pi in and pass through.
- Each of the group labled:
  * Group 1, 8 steppers.
  * Group 2, 2 steppers.
  * Group 3, 1 stepper.
  * Group 4, 2 steppers.
  * Group 5, 1 stepper.
  * Group 6, 1 stepper.

3.2 Back Part

On the back of the PCB we have:

- The input fuse for current spike protection.
- P-Mosfet for reverse polarity protection.
- Switching power supply.
- Two pairs of power transistors for signal switching, for groups with 2 steppers.
- One pair of power transistors for signal switching eight steppers.

- The Whole BOM can be Found At:

https://github.com/kurdish-yoda/TonsOfSteppers/blob/master/Code_and_Documents/tonsOfSteppersBOM.xlsx -

4 Code and Setup

A sample programm can be found in the repository:

https://github.com/kurdish-yoda/TonsOfSteppers/tree/master/Code_and_Documents

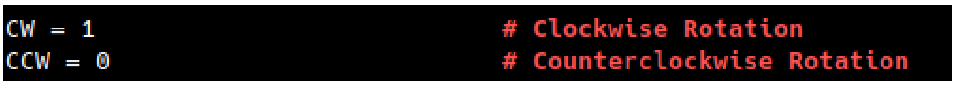
Using this script one could move any of the eight stepper group either a predefined amount or a custom angle. The predefined amount are: 45, 90, 180 and 360 degrees.

Command Syntax:

- move360(group, dir):
    - move+angle(stepperGroup & direction)
- move(group, dir, degree): <-- to move a custom amount
    - move(stepperGroup & direction & amount of degrees)

Direction

The direction is defined by an 1 or a 0.

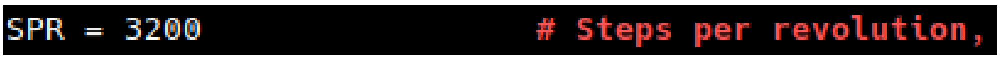


```
CW = 1                                # Clockwise Rotation
CCW = 0                               # Counterclockwise Rotation
```

1 being a clockwise rotation, and 0 being a counter clock wise direction.

Steps Per Revolution

Before using the script one should go in and adjust the SPR (steps per revolution) perameter. This will depend on what motor is used.



```
SPR = 3200                 # Steps per revolution,
```

Each singular driver on the board is setup for 1/16 microstepping. That means that the chips will take each step of the motor normally and devide it up by 16. This means that what ever number of steps the motor has in its datasheets needs to be increased 16 fold.

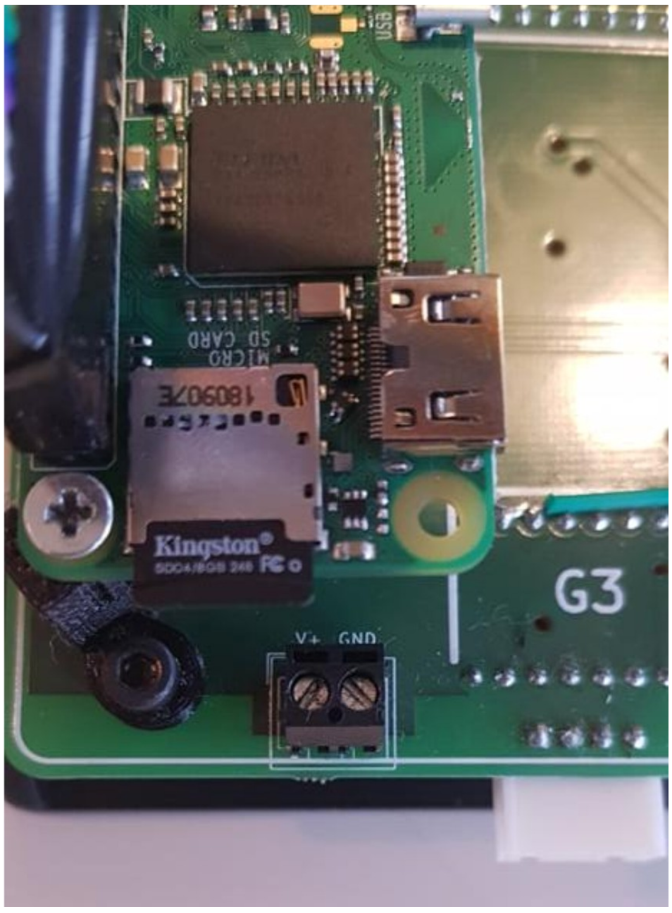
An example would be if you would use a normal nema 17 motor that has about 200 steps per revolution, 200 * 16 = 3200. So 3200 would be the SPR value. As thats how many steps the motor can do hokked up to this driver.

The pinout for the motors is written on the silk screen. Each direction and step pin for each group.

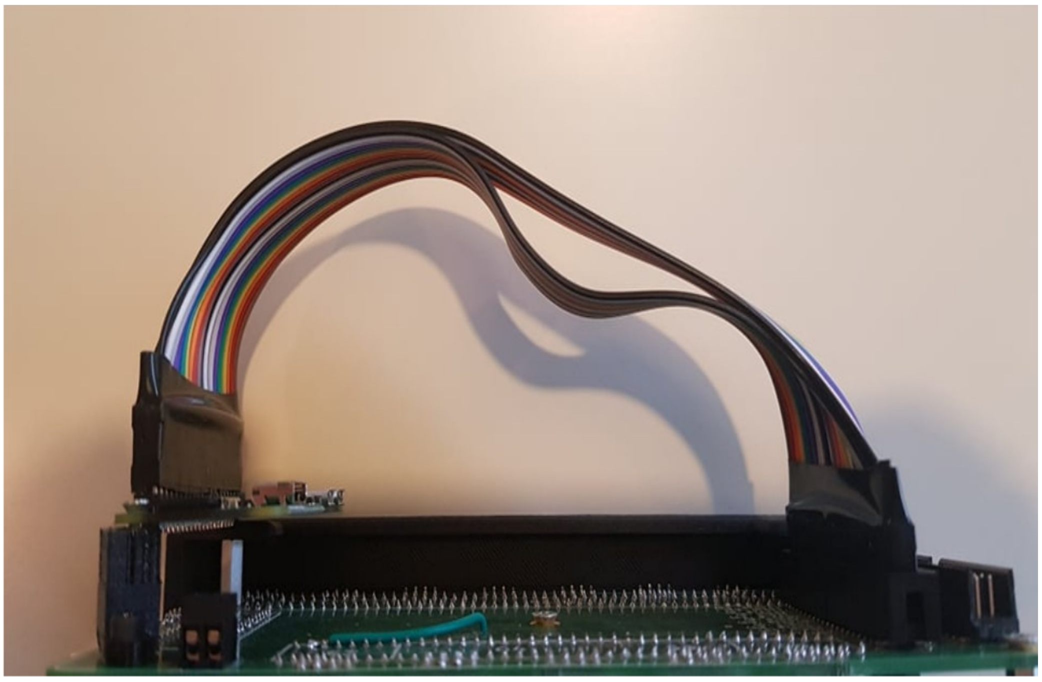
Waringing: Take note that group 4 silk screen is backwards and the pins should be flipped, step and direction.

4.1 Setup

Adequte motor power should firstly be supplied, everythng is possible between 8 and 40 volts.
Once that is plugged in the driver will take care of the 5 volts for the logic.



After that one should attacth the raspberry pi with one of the included „foots“, in adition to the raspberry pi 3 it is also possible to attatch the zero variant also, and then connect the cable as shown:



After this apply power to the raspberry pi and then the driver will be ready to use.

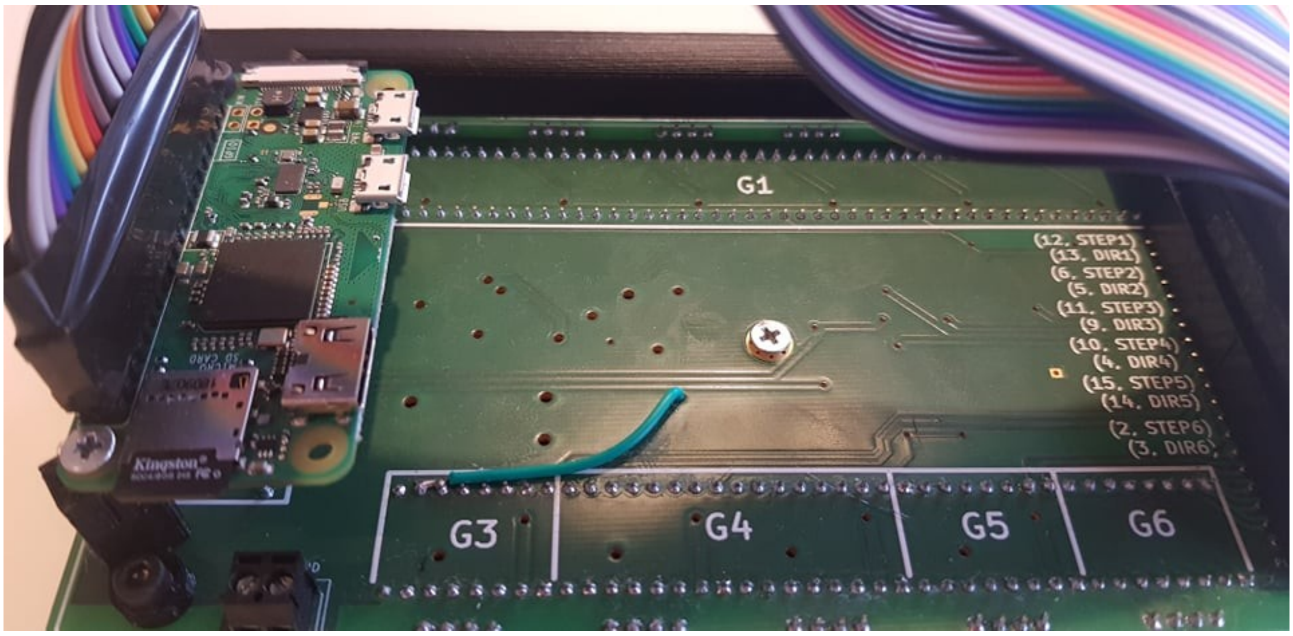
5 For next version

There are a few things that i say would need to be looked at for a final version or next version. (Most of these will be fixed in the comming week and updated on the github).

1. Main Cable Fix (make it possible to use an off the shelf ribbon cable).

2. Group 4 Silk screeen.

3. Missing data valtage on some groups. (This is the one green bodge wire)



4. A cooling Fan should be added (in this case the case should be remodled to suit this)