
Read the Docs Template Documentation

Release 1.0

Read the Docs

Dec 07, 2020

Contents

| | | |
|----------|--|-----------|
| 1 | About S-Drive project | 3 |
| 1.1 | What is S-Drive project? | 3 |
| 1.2 | What does S-Drive project consist of? | 3 |
| 1.3 | S-Drive BLDC driver board | 3 |
| 1.4 | S-Drive BLDC driver firmware | 3 |
| 1.5 | S-Drive BLDC driver control software | 3 |
| 1.6 | Table of contents | 6 |
| 1.7 | licence | 6 |
| 1.8 | Support the project | 6 |
| 2 | Arduino IDE setup for S-Drive firmware | 7 |
| 2.1 | Troubleshooting | 8 |
| 3 | S-Drive control software | 11 |
| 3.1 | About | 11 |
| 3.2 | GUI guide - left side of screen | 12 |
| 3.3 | GUI guide - Middle of screen | 13 |
| 3.4 | GUI guide - Right side of screen | 13 |
| 3.5 | Requirements to work! | 13 |
| 3.6 | Troubleshooting | 13 |
| 3.7 | Simple “How it works” | 13 |
| 4 | Driver versions | 15 |
| 4.1 | S-Drive BLDC driver Version 3.1 (Latest version) | 15 |
| 4.2 | S-Drive BLDC driver version 1 (August 2019) | 15 |
| 4.3 | S-Drive BLDC driver version 2 (December 2019) | 15 |
| 4.4 | S-Drive BLDC driver version 3 rev1 (July 2020) | 15 |
| 4.5 | S-Drive BLDC driver version 3 rev2 (In making) | 20 |
| 5 | Tested Motors | 21 |
| 5.1 | GB4114 Brushless Motor | 21 |
| 5.2 | GM7008H Brushless Motor | 23 |
| 5.3 | 2804 Brushless Motor | 23 |
| 5.4 | 5010 150T Gimbal Brushless Motor | 24 |

Contents:

1.1 What is S-Drive project?

S-Drive project is open-source project with aim to make good BLDC control for robotics cheap and available to everybody. Primary use for this project is in: Compliant robotic arms, dynamic walker robots, camera gimbals ... Goal is to make everything simple to use and integrated but at the same time allow a lot of experimentation and ability for advanced development!

1.2 What does S-Drive project consist of?

1.3 S-Drive BLDC driver board

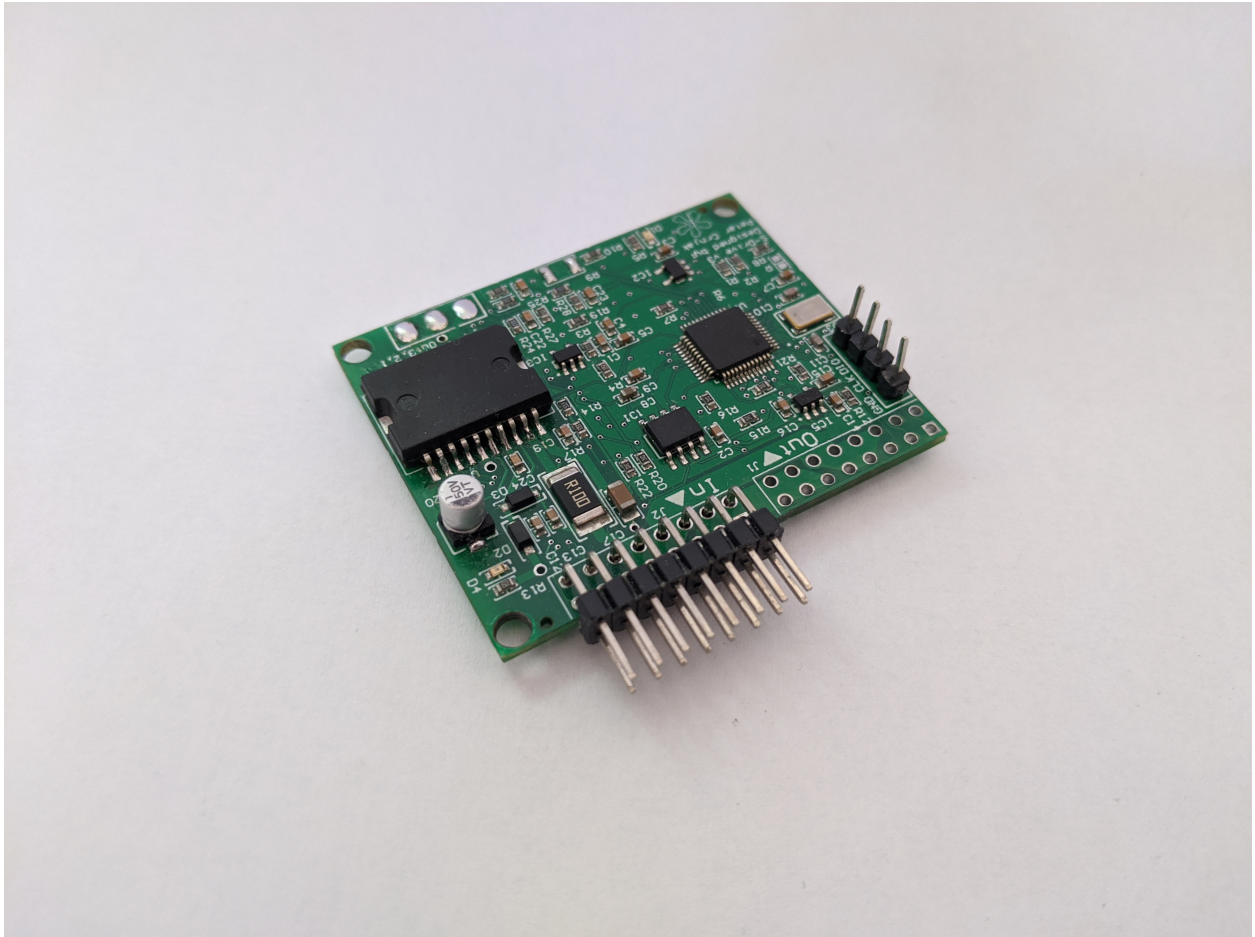
- is small BLDC driver board with all needed functions integrated in!

1.4 S-Drive BLDC driver firmware

- is firmware specially developed for S-Drive BLDC driver board.

1.5 S-Drive BLDC driver control software

- lets you control and monitor motor connected to S-Drive BLDC driver!



```

VI_0 Base Control Globals.h Magic Motor_parameters.h Open_loop SVPWM_table.h Setup Temperature_table.h communication
// This is not really a bug since EEPROM saving works great, it is not being used with full potential.
// Write proper EEPROM lib. ATM I am using only 1 sector of EEPROM.
// This is not priority

#include <filters.h>
#include <Arduino.h>
#include <PID_v1.h>
#include <AS5040.h>
#include <QuadratureEncoder.h>
#include "SVPWM_table.h"
#include "Globals.h"
#include "Motor_parameters.h"
#include "Temperature_table.h"
#include <CircularBuffer.h>
#include <Wire.h>
#include "SparkFun_External_EEPROM.h"

AS5040 enc (CLK, CSn, DO) ;
ExternalEEPROM myMem;
TwoWire Wire2 (2, I2C_FAST_MODE);

CircularBuffer<int, 8> buffer; // 3*0.5s if motor is drawing max current for 1.5s it will shut down

IIR::ORDER ORDER_ = IIR::ORDER::OD3; // Order (OD1 to OD4)
// Low-pass filter
Filter f(CUTOFF_FREQ, SAMPLING_TIME, ORDER_);

PID PID_speed(&Input, &Output, &Setpoint_speed, Kp_speed, Ki_speed, Kd_speed, DIRECT);
PID PID_position(&Input, &Output, &Setpoint, Kp_position, Ki_position, Kd_position, DIRECT);

HardwareTimer pwmTimer4(4); // We use timer 4 and pins that have access to that timer to set PWM frequency

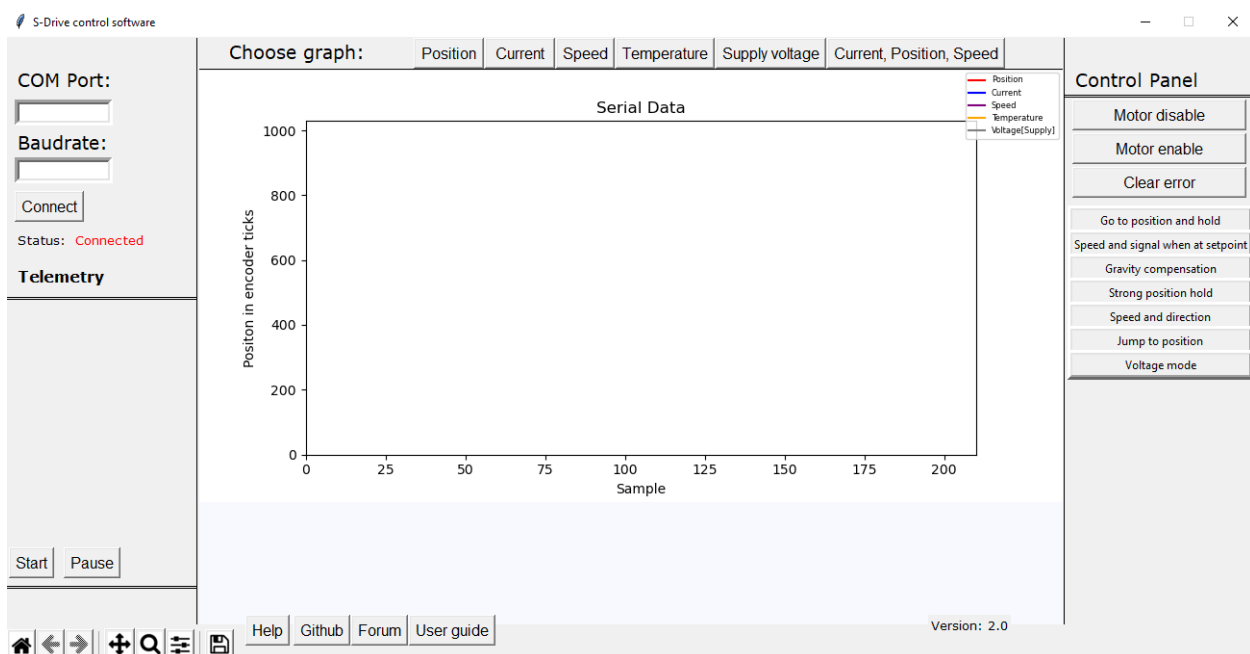
Encoders BLDC_encoder(ENCA, ENCB); // Input pins of Quadrature encoder

Done Saving.
Reset mode: Software reset
Device ID: 0x410
Device flash Size: 128 Kbytes
Device family: STM32F10xx Medium-density

Loading file...
Flash Programming:
File : C:\Users\RopePC\AppData\Local\Temp\arduino_build_353057\VI_0.ino.bin
Address : 0x08000000
Memory programming...
0%XXXXXXXX 11%XXXXXXXX 22%XXXXXXXX 34%XXXXXXXX 45%XXXXXXXX 57%XXXXXXXX 68%XXXXXXXX 80%XXXXXXXX 91%XXXXXXXX 100%
Memory programmed in 5s and 235ms.
Programming Complete.

MCU Reset.

```



1.6 Table of contents

1.7 licence

1.8 Support the project

This project is completely Open source and free to all and I would like to keep it that way, so any help in terms of donations or advice is really appreciated. Thank you!

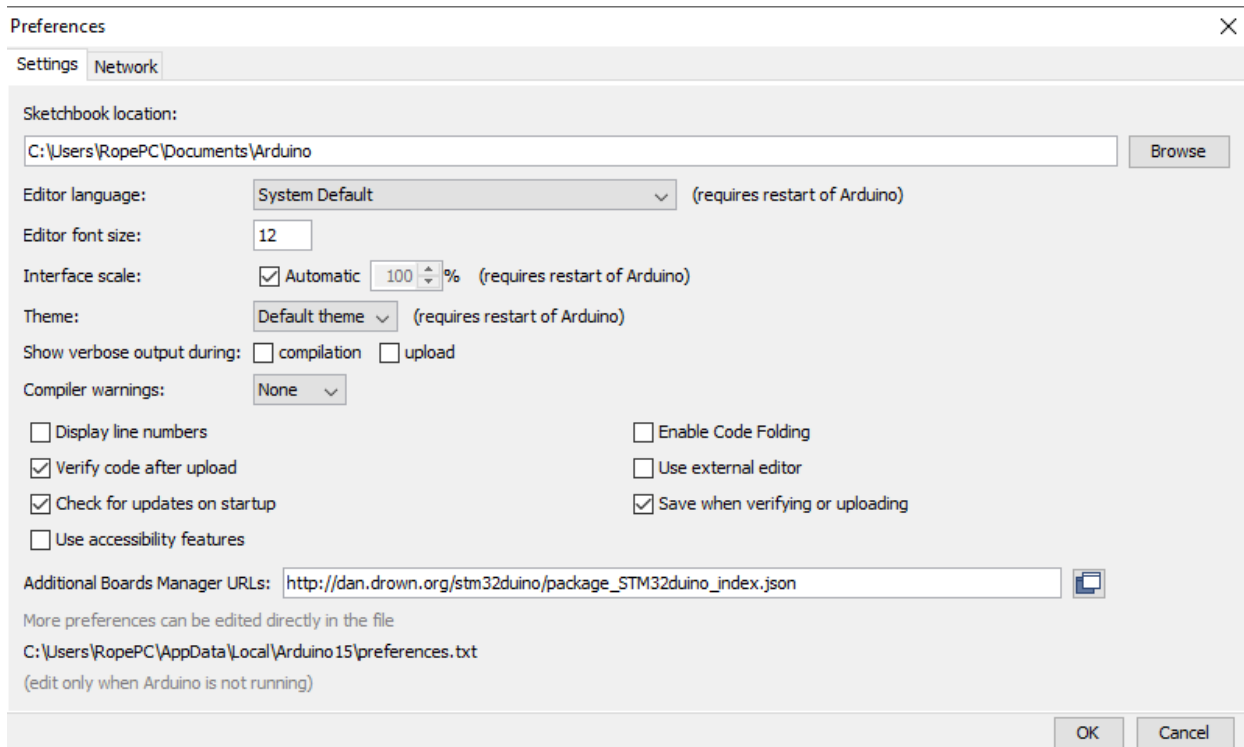


CHAPTER 2

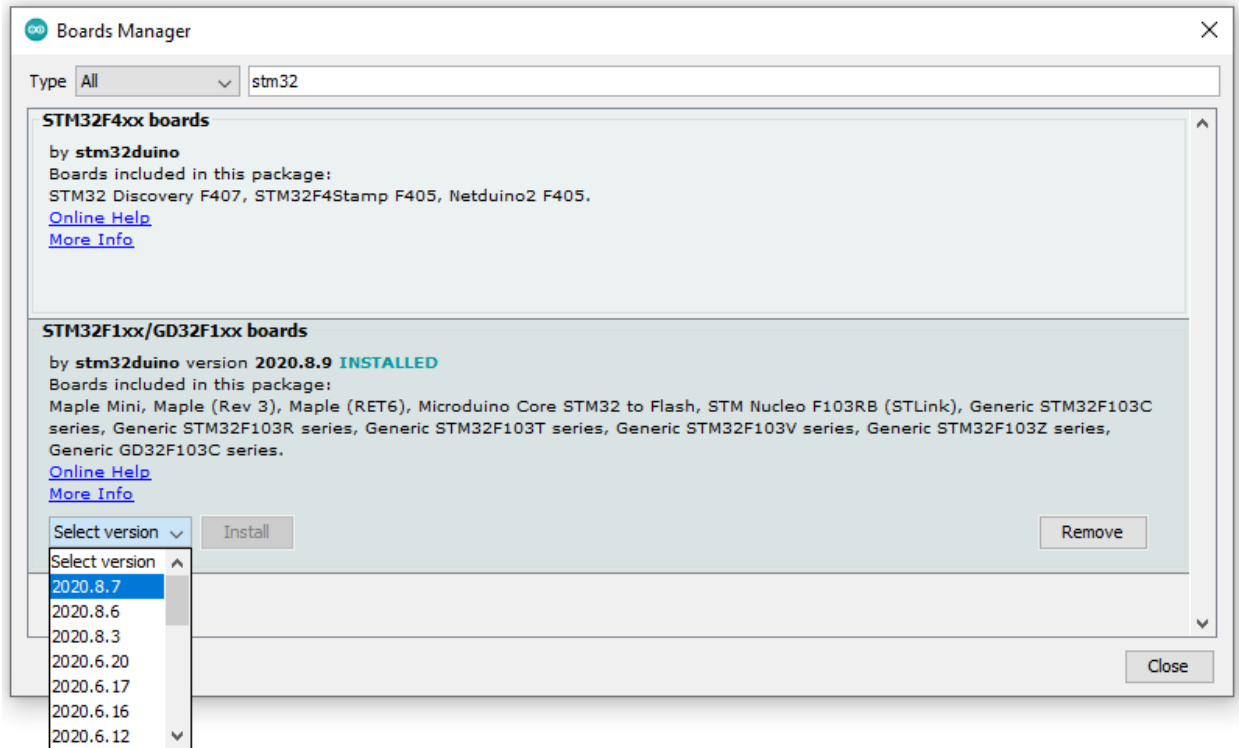
Arduino IDE setup for S-Drive firmware

First download Arduino IDE from official website: (Version we are running is 1.8.13)

Run Arduino IDE, go to file/preferences/additional boards manager urls: Enter : http://dan.drown.org/stm32duino/package_STM32duino_index.json (Any additional json file is separated with ,)



Now go to tools/board/board manager and search for STM32 Select STM32F1XX/GD32F1XX BOARDS (We used version 2020.8.9.)



Now download needed libraries from S-Drive firmware repo: We recommend using versions of libraries from this repo for compatibility. Using new versions can cause issues!!

Copy needed libs to libraries folder of your Arduino IDE installation.

We are almost done!

To program S-Drive BLDC driver board use ST-link. Windows 10 will usually download drivers alone, but if it does not download them from here: Note (STM32cube programmer is used for ARDUINO FOC firmware) https://my.st.com/content/my_st_com/en/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-programmers/stm32cubeprog.license=1598927090654.product=STM32CubeProg.version=2.5.0.html

or just drivers here: <https://www.st.com/en/development-tools/stsw-link009.html>

To access serial port you need FTDI TO USB board. Windows 10 will usually download drivers alone, but if it does not download them from here: <https://www.ftdichip.com/Drivers/VCP.htm>

2.1 Troubleshooting



S-Drive control software

Latest version: Version 2.0

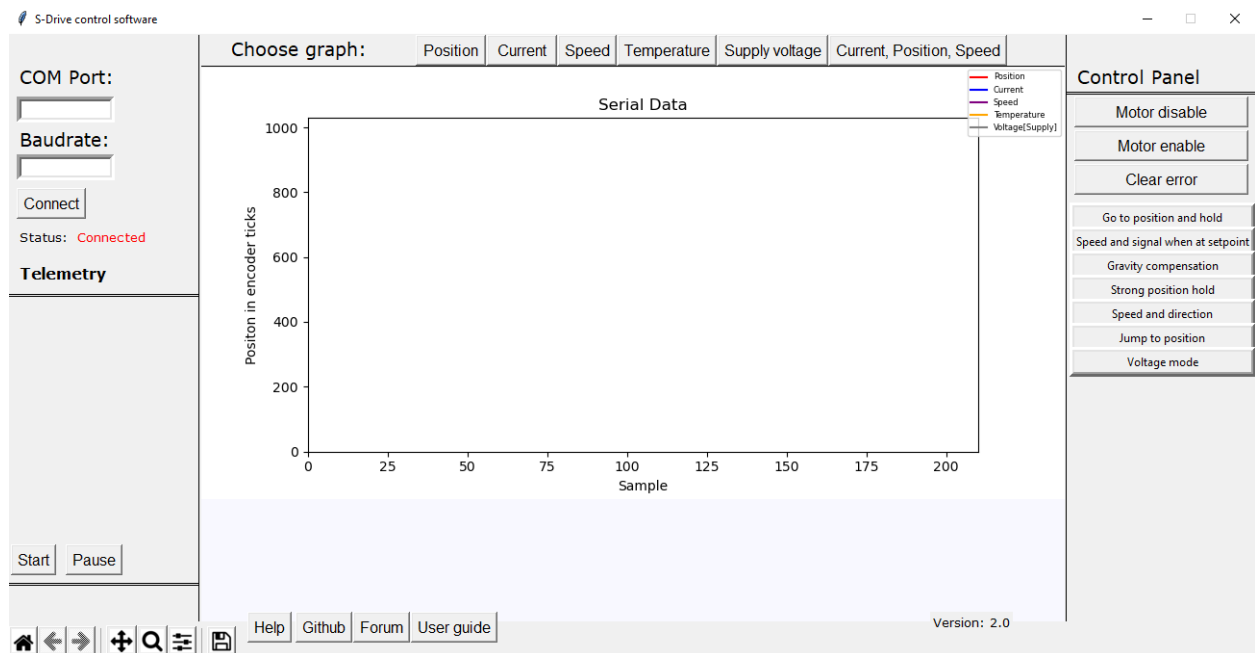
3.1 About

S-Drive control software lets you control and monitor S-Drive motor from PC. This version supports plotting, showing data and controlling of only one motor at the time. S-Drive control software allows you to see live data from your motor like:

- speed
- current
- position
- temperature
- BEMF

AND at the same time control the motor in few different modes like:

- Go to position and hold
- set speed and flag at setpoint
- Gravity compensation
- Strong position hold
- Speed and direction
- Jump to position
- Voltage mode



3.2 GUI guide - left side of screen

On the left side you can see:

- COM PORT: here you enter at what COM port your FTDI to USB board is connected to!
- BAUDRATE: here you enter baudrate at wich your S-Drive board and control software communicate at.

Note: Note BAUDRATE is usually 1000008 or 1000000 in S-Drive Firmware

- Status: Connected - GREEN for connected . RED for NOT connected
- Telemetry is will print useful data about the motor.
- Start will start serial communication, it can also be used to start wiout entering COM PORT and BAUDRARE values

To use it like that in python script edit bottom lines to COM PORT and BAUDRATE you are using!

```
1 s.baudrate = 1000000
2 s.port = 'COM3'
```

To COM PORT and BAUDRATE you are using!

- Now if you use that edited script you can always connet to that com port with that baudrate when pressing START.
- PAUSE pauses serial communication. To resume it PRESS start again!
- On bottom is tkinter tool to navigate graphs!

3.3 GUI guide - Middle of screen

On the middle you can select what graph to view! Only one can be active at the time, or you can view multiple by selecting “Current,Position and Speed”.

3.4 GUI guide - Right side of screen

Here you can:

- DISABLE MOTOR - turns off MOSFTETS
- ENABLE MOTOR - Goes to POSITION HOLD on CURRENT POSITION!
- CLEAR ERROR - clears ERROR from motor.

Note: if motor sends it still has error usually it means that temperature is still to high and it will disable itself again

After CLEAR ERROR are motor modes:

- Go to position and hold - Goes to position with specific speed and holds desired position with desired Kp and current treshhold
- set speed and flag at setpoint - Rotate at specific speed to desired setpoint at wich driver will send done flag
- Gravitiy compensation - Gravitiy compensation mode
- Strong position hold - Strong position hold on current position with desired Kp and current treshhold
- Speed and direction - Move with desired speed in desired direction
- Jump to position - Jump to desired position
- Voltage mode - Move motor in desired direction my setting voltage to the driver

Note: voltage is in value from 0 - 1000 , 0 being zero volt 1000 being power supply voltage (peak)

3.5 Requirements to work!

In S-Drive firmware in Motor_parameters file you NEED to adjust :

```
uint16_t Main_serial_output_time = 50000;
```

Usually it is 10000, but that is to fast for S-Drive control firmware!

3.6 Troubleshooting

3.7 Simple “How it works”

4.1 S-Drive BLDC driver Version 3.1 (Latest version)

Whole S-Drive project started in July 2019 with simple breadboard prototype. From the very beginning project was planned to be used for robotic arms and legged robots, but to do that size was needed to be reduced.

4.2 S-Drive BLDC driver version 1 (August 2019)

First version was large and mostly made for prototyping and testing features. It used a lot of large components and had all pins from MCU routed as header pins for testing. Also it had USB plug and bootloader support, but that feature was removed to reduce size. Zener diodes were used on PULS and ENABLE pins for protection, but they were also removed to reduce size. As you can see it used a lot of large components like those blue and green connectors and resistors were definitely oversized. At that time I did not know what type of communication I would so so multiple stuff was tested like : I2C, UART and SPI. This board was hand assembled. It uses 0805 components.

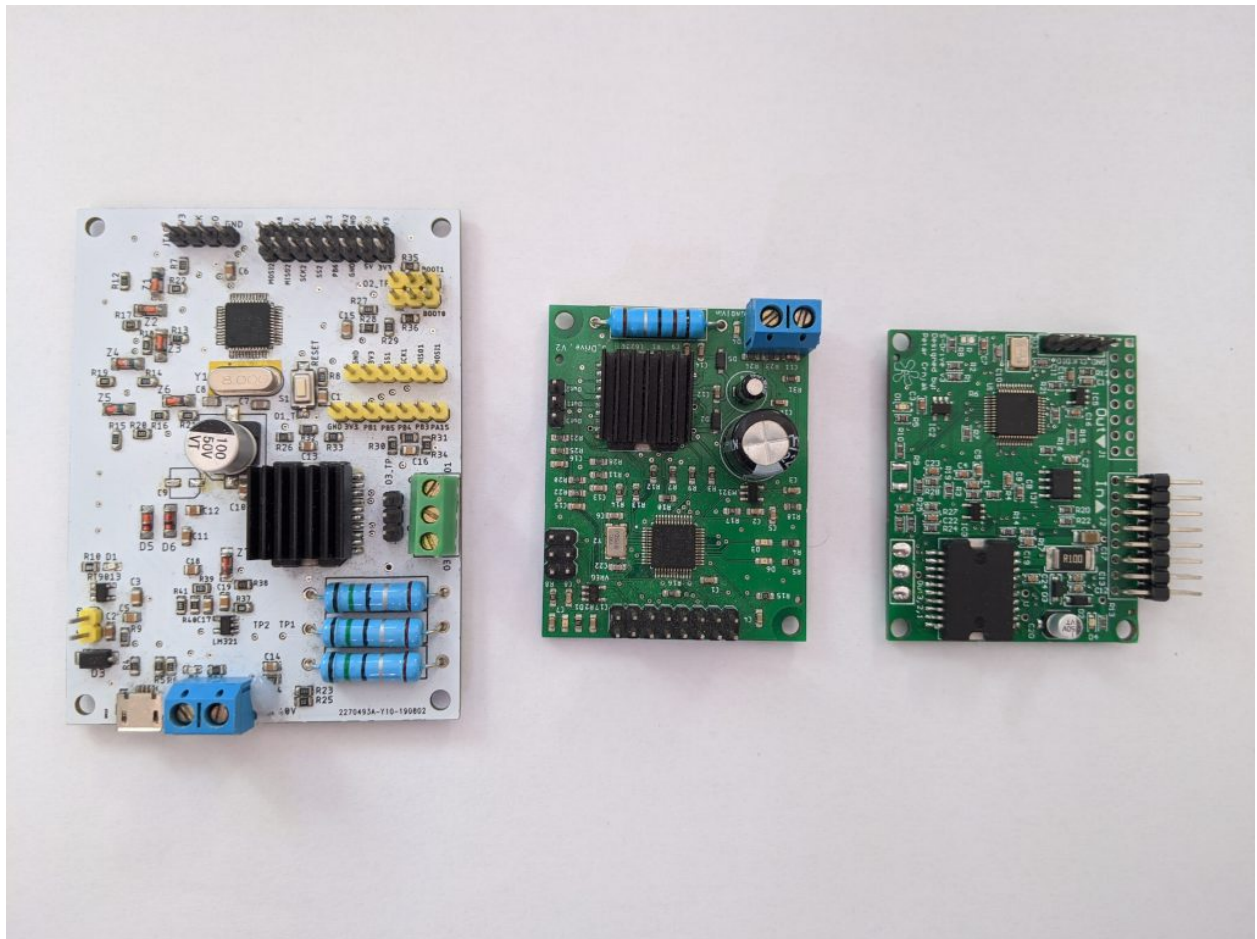
4.3 S-Drive BLDC driver version 2 (December 2019)

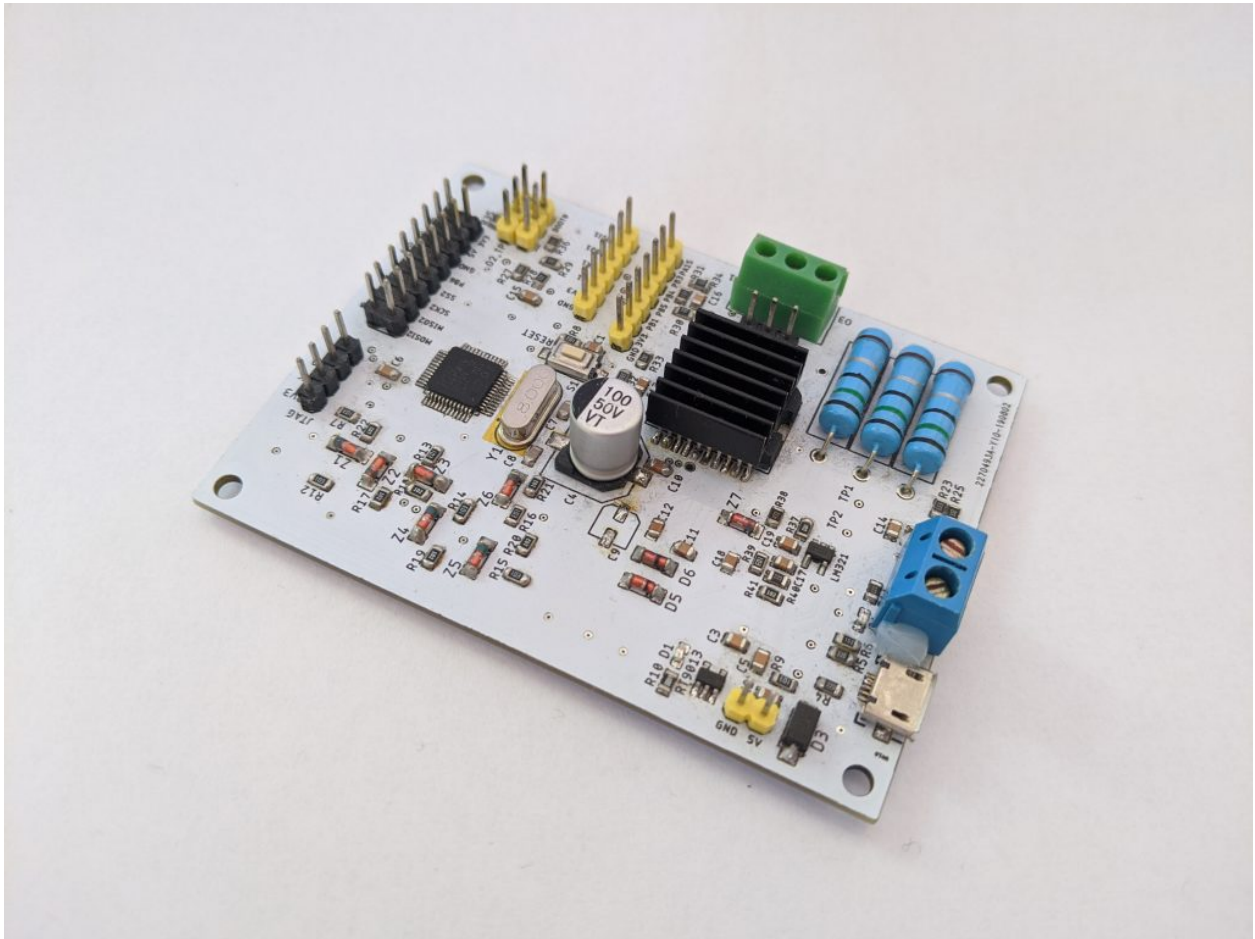
Second version is much smaller and looks similar to V3 but in reality it is more similar to first version. At this board I learned what needed to be made for V3 that is first stable version. From this board I also learned that board needs to have encoder embedded. This version was assembled by machines and uses 0603 components.

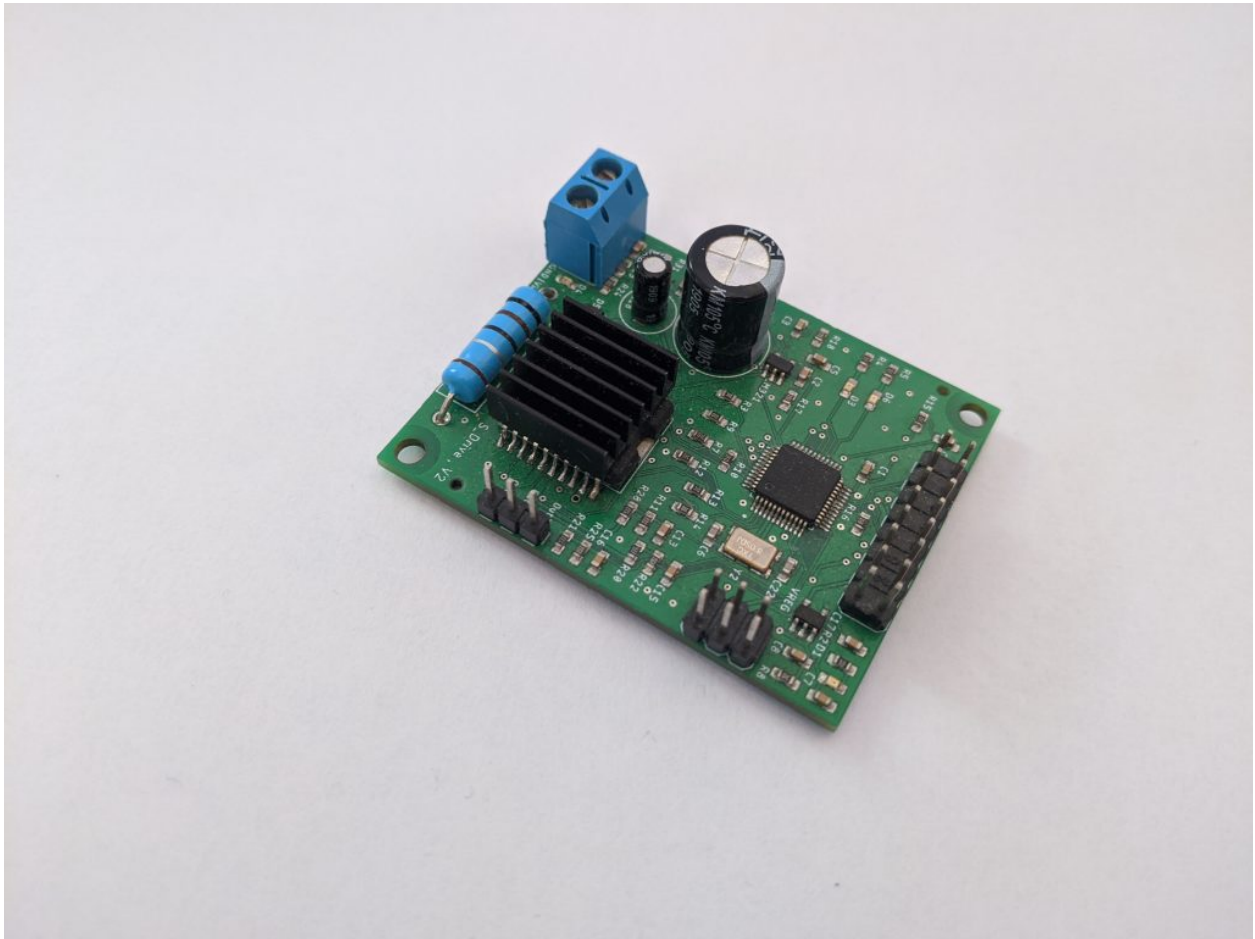
4.4 S-Drive BLDC driver version 3 rev1 (July 2020)

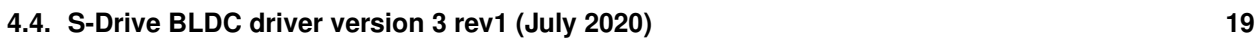
This is third version of S-Drive, it is newest and stable version where everything is working. What is different from V1 and V2:

- RS485 chip for daisy chain ability









- removed all large connectors
- Additional EEPROM chip
- No large connectors
- All connectors were replaced with 2 connection buses where each bus has high voltage lines, UART comms, RS485 comms and low power lines
- thermistor connector
- Small size of 50 x 41 mm
- additional signal LEDs
- Encoder is on bottom side!

4.5 S-Drive BLDC driver version 3 rev2 (In making)

CHAPTER 5

Tested Motors

5.1 GB4114 Brushless Motor



| kv | Pole pairs | Resistance |
|----|------------|------------|
| 42 | 11 | 10.3 Ohm |



5.2 GM7008H Brushless Motor

| kv | Pole pairs | Resistance |
|----|------------|------------|
| 28 | 11 | 10.8 Ohm |

5.3 2804 Brushless Motor



| kv | Pole pairs | Resistance |
|-----|------------|------------|
| 140 | 7 | 10.7 Ohm |

5.4 5010 150T Gimbal Brushless Motor



| kv | Pole pairs | Resistance |
|-----|------------|------------|
| ??? | 7 | 14.6 Ohm |