WROCŁAW UNIVERSITY OF SCIENCE AND TECHNOLOGY

Micromouse

Authors:
Barłomiej OSTROWSKI
Sebastian MRÓZ

Supervisor:
Witold PALUSZYŃSKI
Ph.D.

Faculty of Electronics,
Department of Cybernetics and Robotics

January 29, 2019

Wrocław University
of Science and Technology

Abstract

The aim of the project is to build a Micromouse robot able to take part in competitions. Task of the robot is to pass maze and find the way out as quickly as it can.

1 Goals

- Design and build a small robotics platform that is capable of exploring within the size constraints of a standard Micromouse maze.

- Explore the use of DC motors with feedback in a precision control application.

- Find sensors suitable for close range distance detection for use on the Micromouse.

- Explore navigation strategies for mobile robots including the use of odometery and sensor feedback

- Design all of the Micromouse hardware including chassis parts and electrics systems for the mouse

2 Assumptions

- Robot will be wheeled platform type (2,0)

- Navigation through maze will be done with distance sensors and odometry

- Available sensors for robot are distance sensors, encoders and inertial measurement unit

- For the diagnostic purposes: communication to the robot will be available via the bluetooth protocol

3 Work program

- Design electronics and mechanics

- Robot able to move

- Robot able to move in straight lines and rotate an angle using odometry

- Quantization of movement - robot can divide it's path into short operations

- Readings from distance sensors

- Mapping of the maze

- Designing algorithm to solve the maze

- Maybe further optimization of algorithm [speed vs length of track]

- Robot solving the maze

4 Mechanics

4.1 Constraints

- There was decided to make robot as small as possible, maximum size was set to 10cm square.

- Size of wheels was chosen to be 32mm in diameter.

- Main electronic board of the robot will be base of all mechanics elements.

- Height is not limited but center of mass have to be near to the ground

- Center of the mass should be located in middle of the robot, between the wheels

- Battery used in robot is LiPo battery with nominal voltage 7.2V

- Weight is not limited but should be as low as possible

- Programming connector and reset have to be accessible at all time and cannot interfere with robot motion

- Encoder magnets have to be installed on outer face of the wheel on it's center

4.2 Design

3D model of the board was exported with use of Eagle to Fusion conversion (embedded into software). After this we added 3D models of the motors and brackets to ensure proper size and spacing. After this we decided to add spacer between motors and board to lower the profile of the robot and ensure good accessibility to reset button and programming connector. Designing process took place in few iterations, on every step there was an evaluation of the dimensions compatibility and spacing and necessary modifications were done.
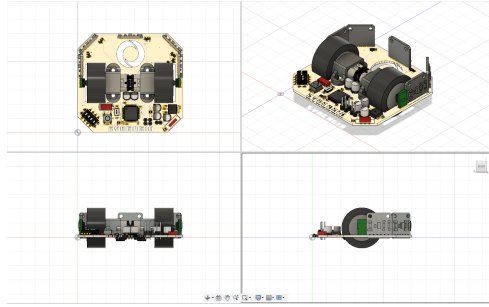

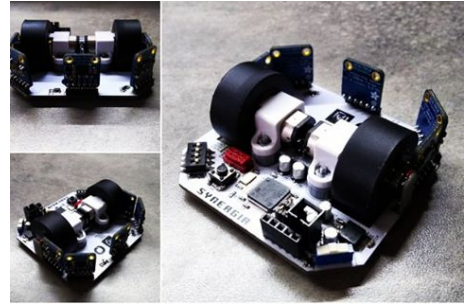Figure 1: 3D model of the robot, project in Autodesk Fusion 360

Figure 2: Pictures of real robot

4.3 Additional elements

As part of the design there was need to make additional spacer between the motors and main board of robot to lower the profile. It was Designed based on 3D model and then printed on FDM machine. Additionaly we created 3D printed template to ensure proper mount of encoder magnets on wheel face.
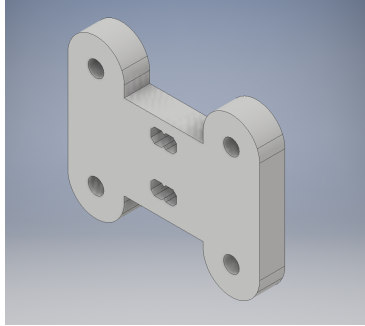
Figure 3: 3D model of spacer

Figure 4: Picture of designed centering tool after mounting the magnets

5 Electronics

Using Auodesk Eagle software we designed main board of the robot.

We decided to use an STM32F4 microcontroller because it have 128kB of SRAM that can be used to store large mazes and multiple communication interfaces.

As distance sensors we choosen VCNL4010 from Adafruit for the sake of stable readings and easy read thanks to internal signal processing.

For encoder we decided to use standard AS5045B form AMS.

Schematic of the main board of the robot is presented in appendix A.

Electronics part was designed simultaneously with mechanics part. It went through many iterations of design and evaluate process.
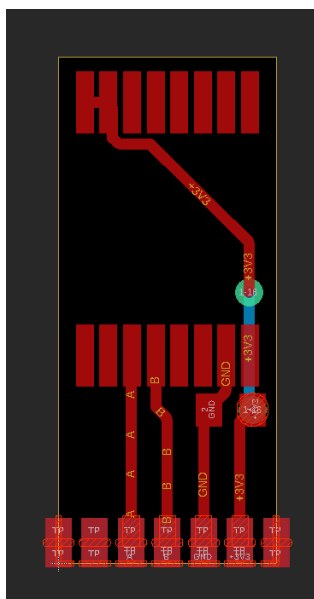
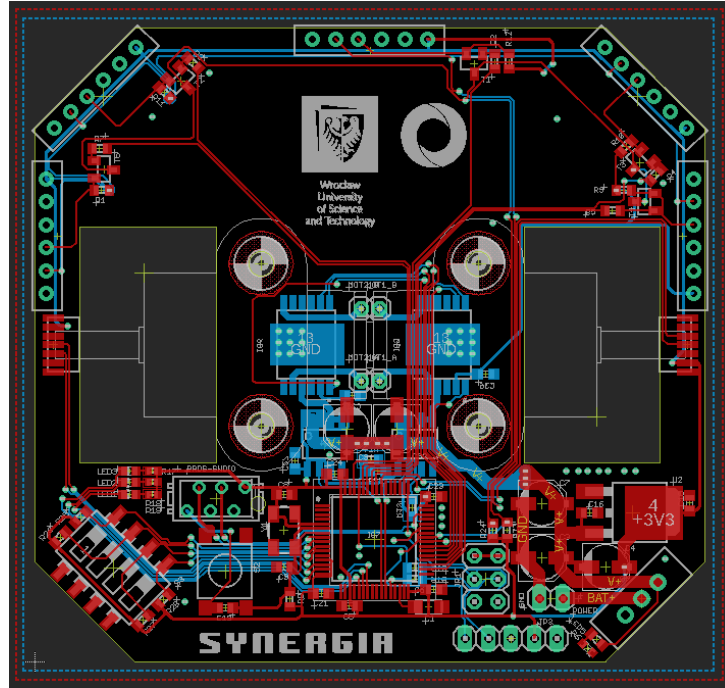Figure 5: Designed encoder board in Autodesk Eagle program

Figure 6: Top view of main board in Autodesk Eagle program

6 Programming and testing

Programming tasks was splitted between two people. We used Github as version control software. STM32CubeMX as peripherial initializer and Atollic True Studio as programming environment.

Configuration of the microcontroller is presented in appendix B

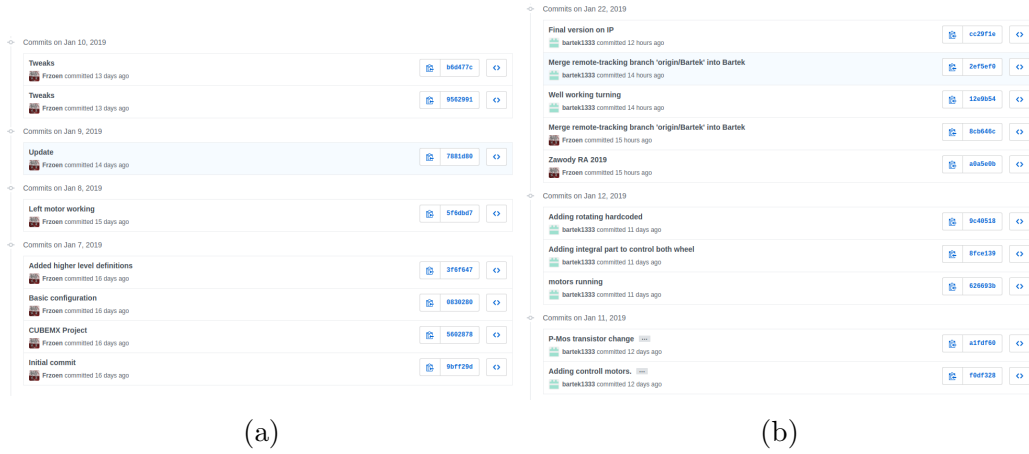Workflow of the programming part is presented in figure 7.

Figure 7: Workflow presented

7 Summary

We managed to design, build and program the robot according to given assumptions. In the current state of progress robot can navigate through the maze using odometry and readings from distance sensors. We didn't manage to program mapping the maze and finding shortest path.

Robot is hard to rotate because we decided to use wide tyres and that cause inaccuracies in measuring axis of rotation of robot.

We decided to not include IMU on final board, it is an option that can be added later on.

Also used sensors are not ideal because they generate non-linear output of the distance.

8 References

https://www.st.com/resource/en/datasheet/stm32f411re.pdf
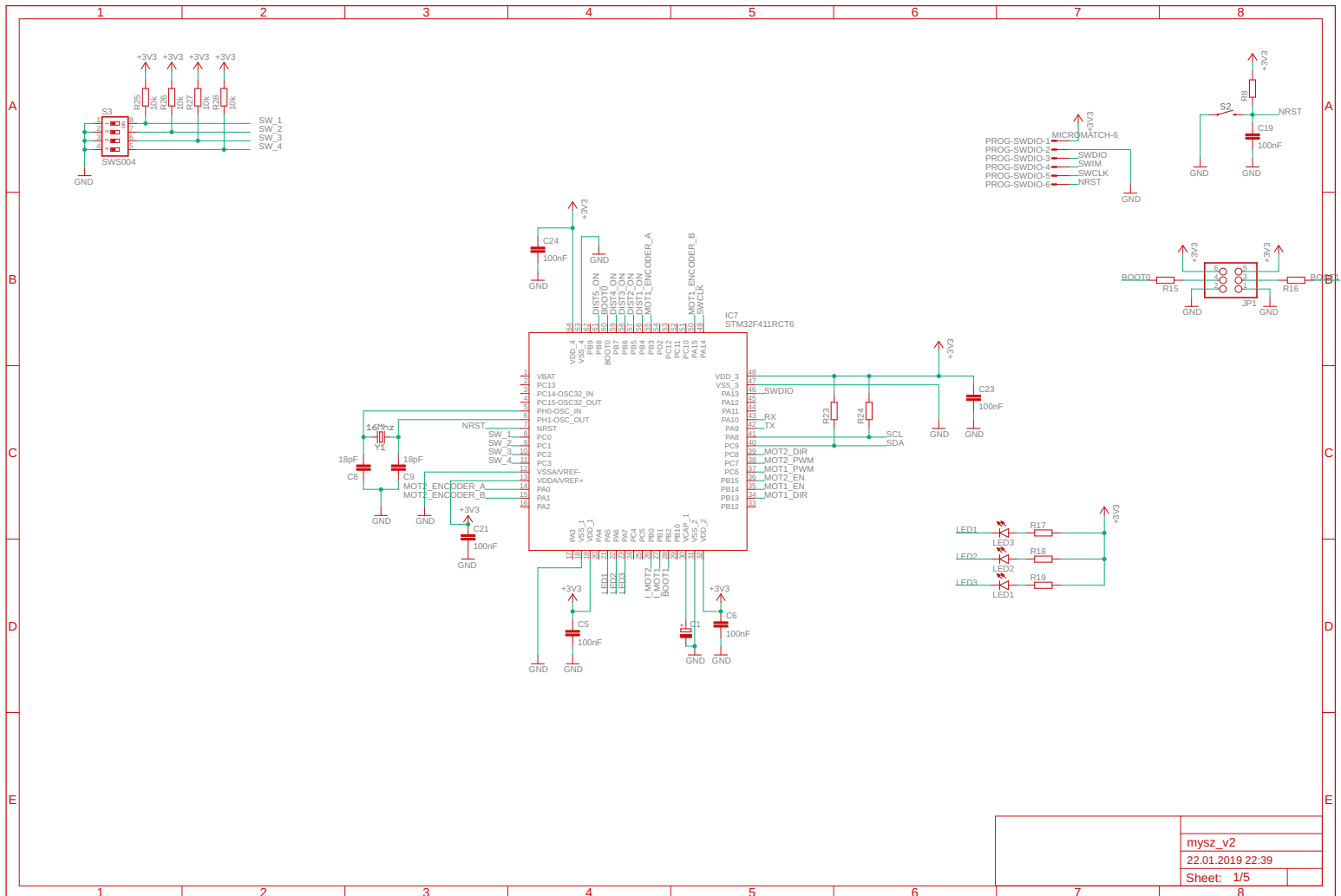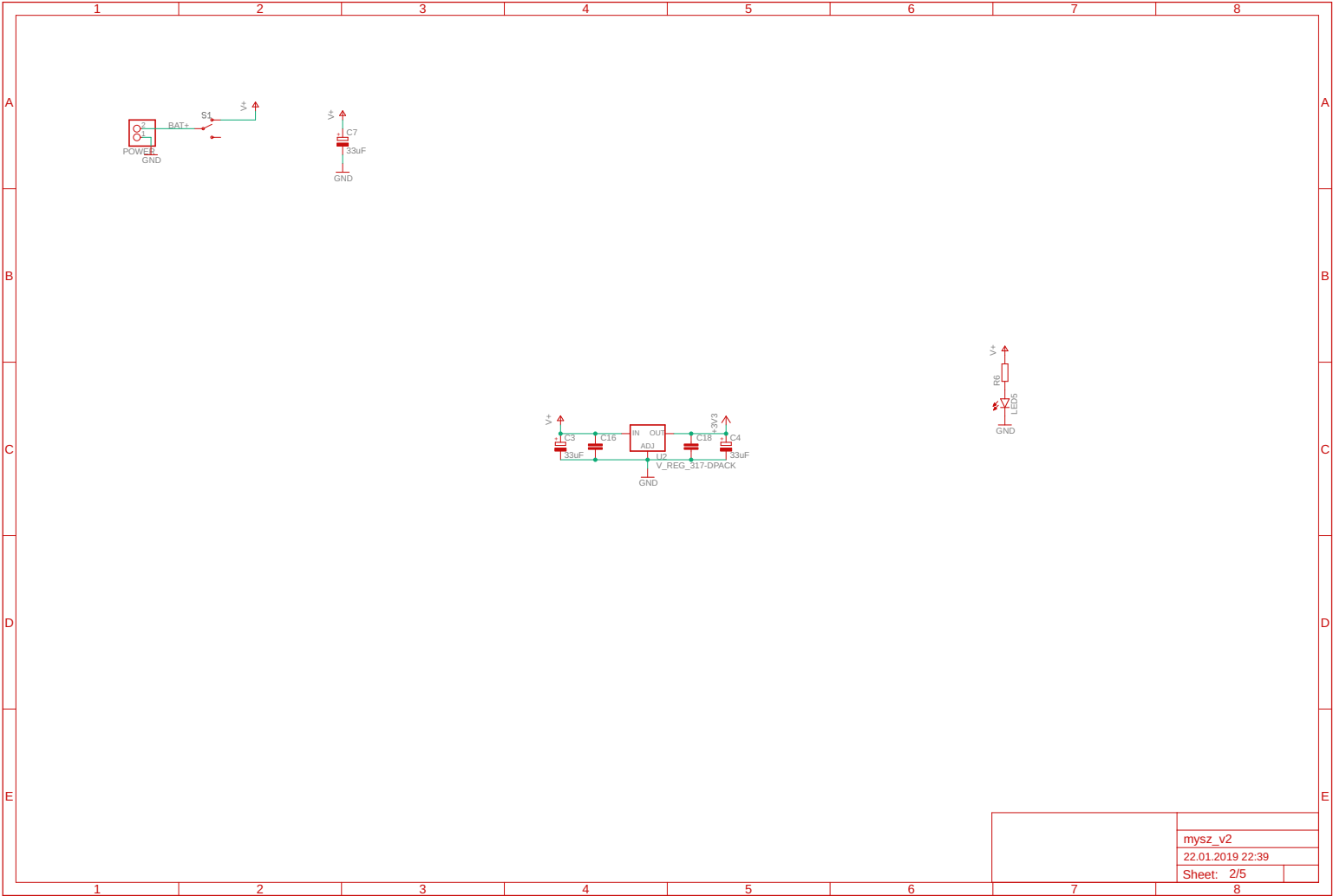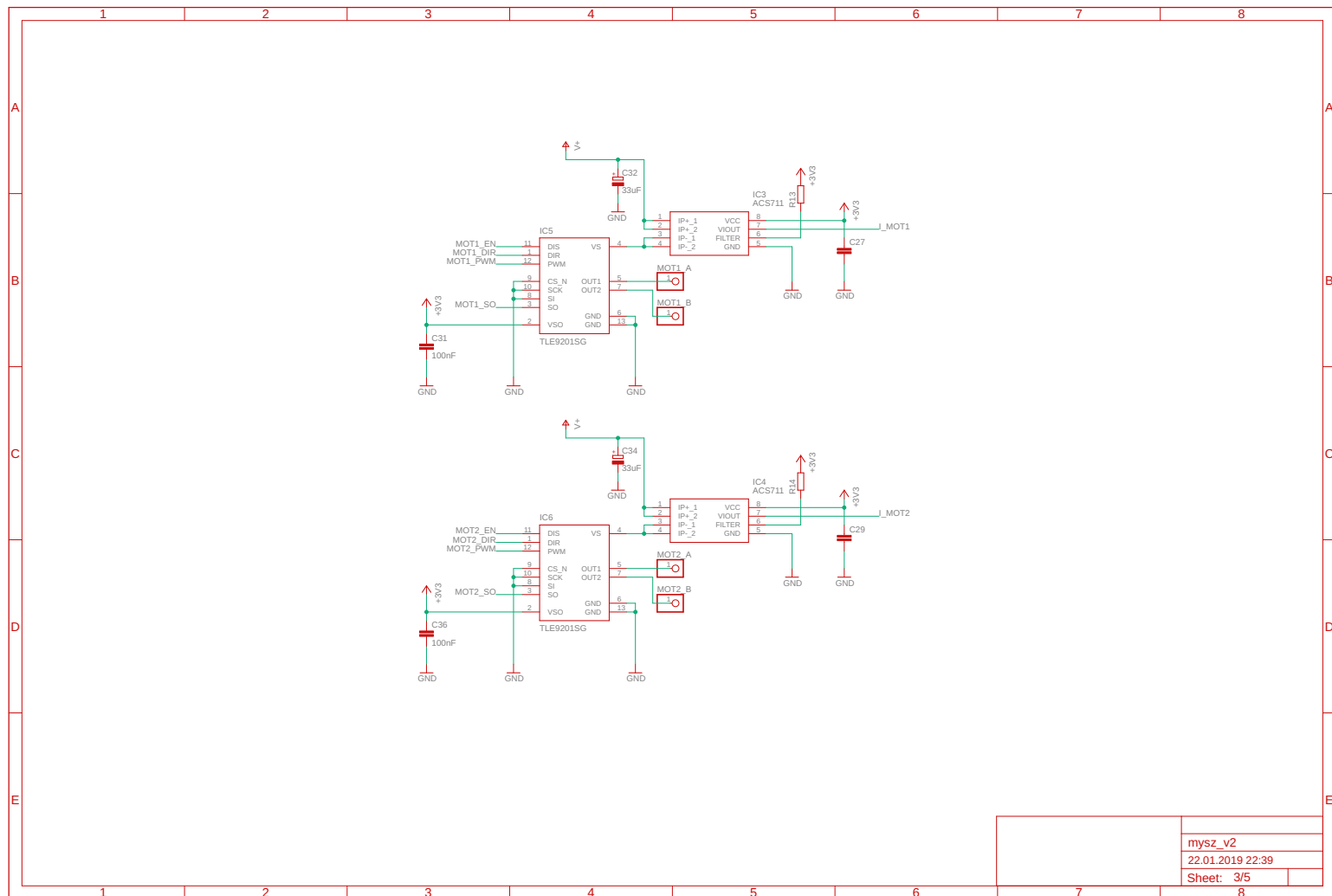https://www.vishay.com/docs/83462/vcnl4010.pdf
https://www.infineon.com/dgdl/Infineon-TLE9201SG-DS-v01_00-en.pdf?
fileId=db3a30434508770901451819f481cec

Appendices

A Schematic

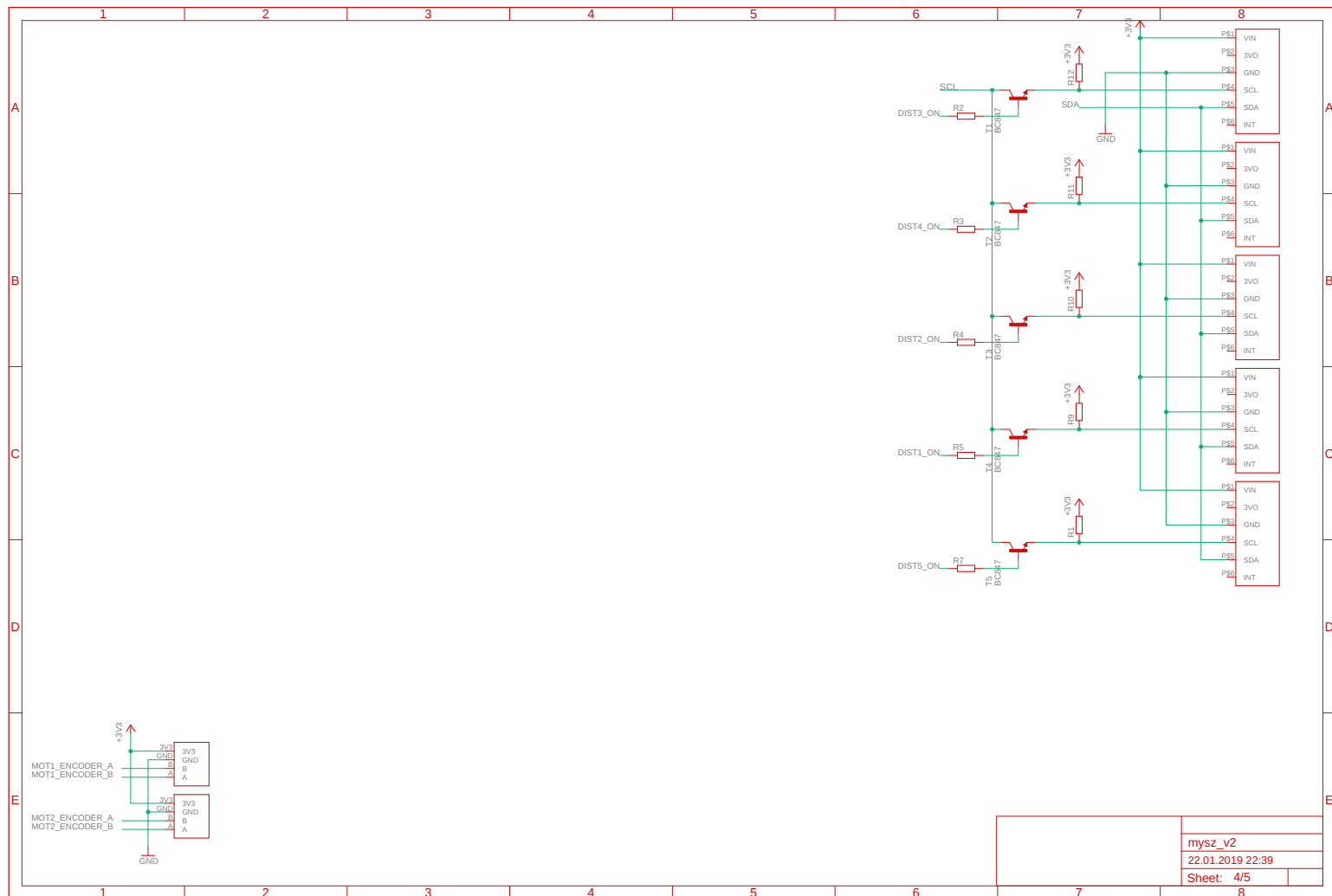+3V3 +3V3 +3V3 +3V3

S3
R25 10k
R26 10k
R27 10k
R28 10k

SW_1
SW_2
SW_3
SW_4

SWS004
GND

PROG-SWDIO-1
PROG-SWDIO-2
PROG-SWDIO-3
PROG-SWDIO-4
PROG-SWDIO-5
PROG-SWDIO-6

MICROMATCH-6
+3V3

SWDIO
SWIM
SWCLK
NRST

GND

S2
R8
+3V3
NRST
C19
100nF
GND GND

C24
100nF
GND
+3V3
GND

BOOT0
R15
+3V3 +3V3
R16
BOOT1
JP1
GND GND

IC7
STM32F411RCT6

VDD_4
VSS_4
PB9
PB8
BOOT0
PB7
PB6
PB5
PB4
PB3
PD2
PC12
PC11
PC10
PA15
PA14

DIST5_ON
BOOT0
DIST4_ON
DIST3_ON
DIST2_ON
DIST1_ON
MOT1_ENCODER_A
MOT1_ENCODER_B
SWCLK

VBAT
PC13
PC14-OSC32_IN
PC15-OSC32_OUT
PH0-OSC_IN
PH1-OSC_OUT
NRST
PC0
PC1
PC2
PC3
VSSA/VREF-
VDDA/VREF+
PA0
PA1
PA2

VDD_3
VSS_3
PA13
PA12
PA11
PA10
PA9
PA8
PC9
PC8
PC7
PC6
PB15
PB14
PB13
PB12

SWDIO
RX
TX
SCL
SDA
MOT2_DIR
MOT2_PWM
MOT1_PWM
MOT2_EN
MOT1_EN
MOT1_DIR

NRST
SW_1
SW_2
SW_3
SW_4

16Mhz
Y1
18pF
C8
18pF
C9
GND GND

MOT2_ENCODER_A
MOT2_ENCODER_B

+3V3
C21
100nF
GND

PA3
VSS_1
VDD_1
PA4
PA5
PA6
PA7
PC4
PC5
PB0
PB1
PB2
PB10
VCAP_1
VSS_2
VDD_2

LED1
LED2
LED3
I_MOT2
I_MOT1
BOOT1

+3V3
C5
100nF
GND GND

+3V3
C1
C6
100nF
GND GND

+3V3
C23
100nF
GND GND

R23 R24

+3V3

LED1 R17
LED3
LED2 R18
LED2
LED3 R19
LED1
+3V3

mysz_v2
22.01.2019 22:39
Sheet: 1/5

V+

S1

BAT+

POWER

GND

V+

C7
33uF

GND

V+

R6

LED5

GND

V+

C3
33uF

C16

IN   OUT

ADJ

U2
V_REG_317-DPACK

GND

3V3

C18

C4
33uF

JP3

TX
RX
+3V3
GND
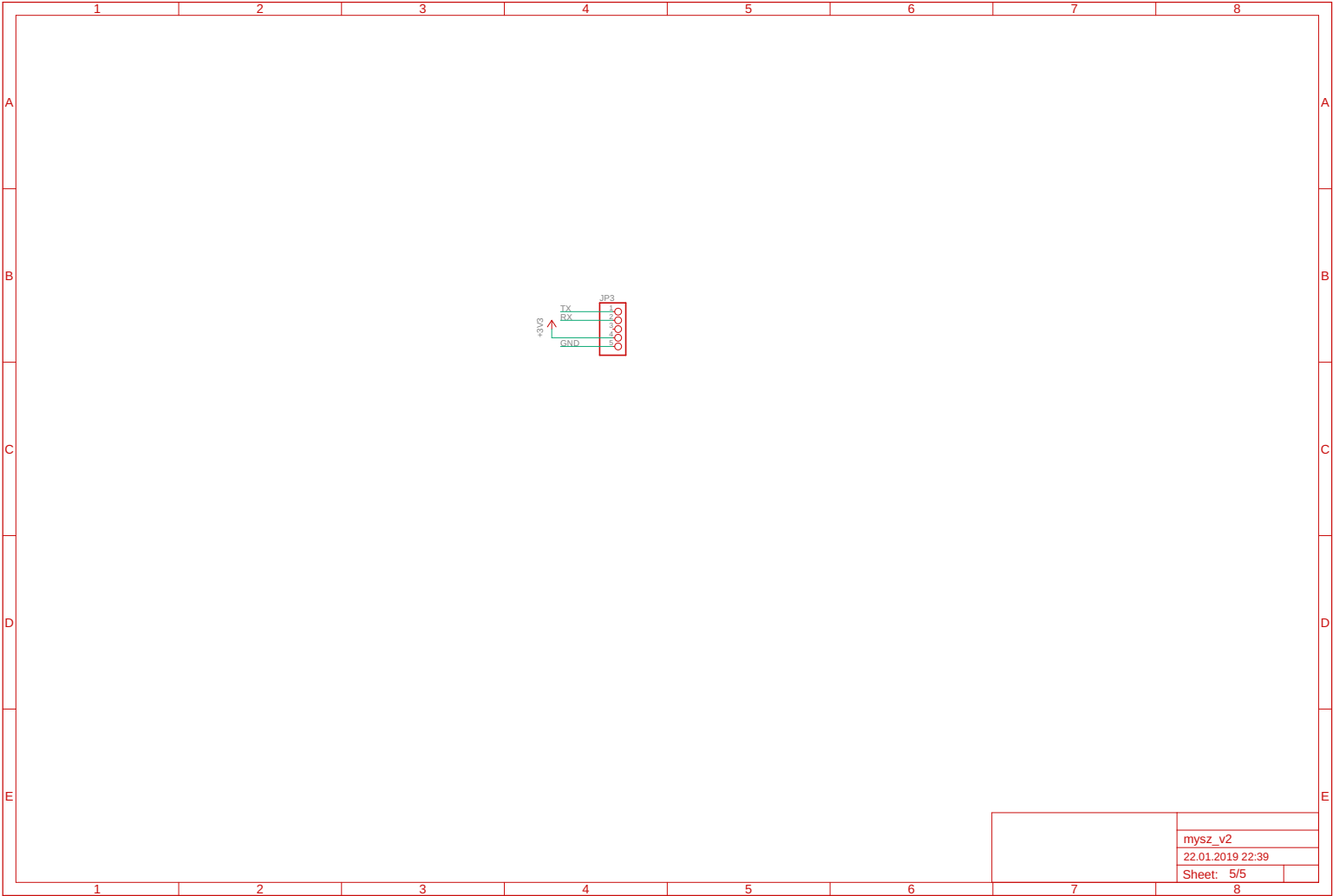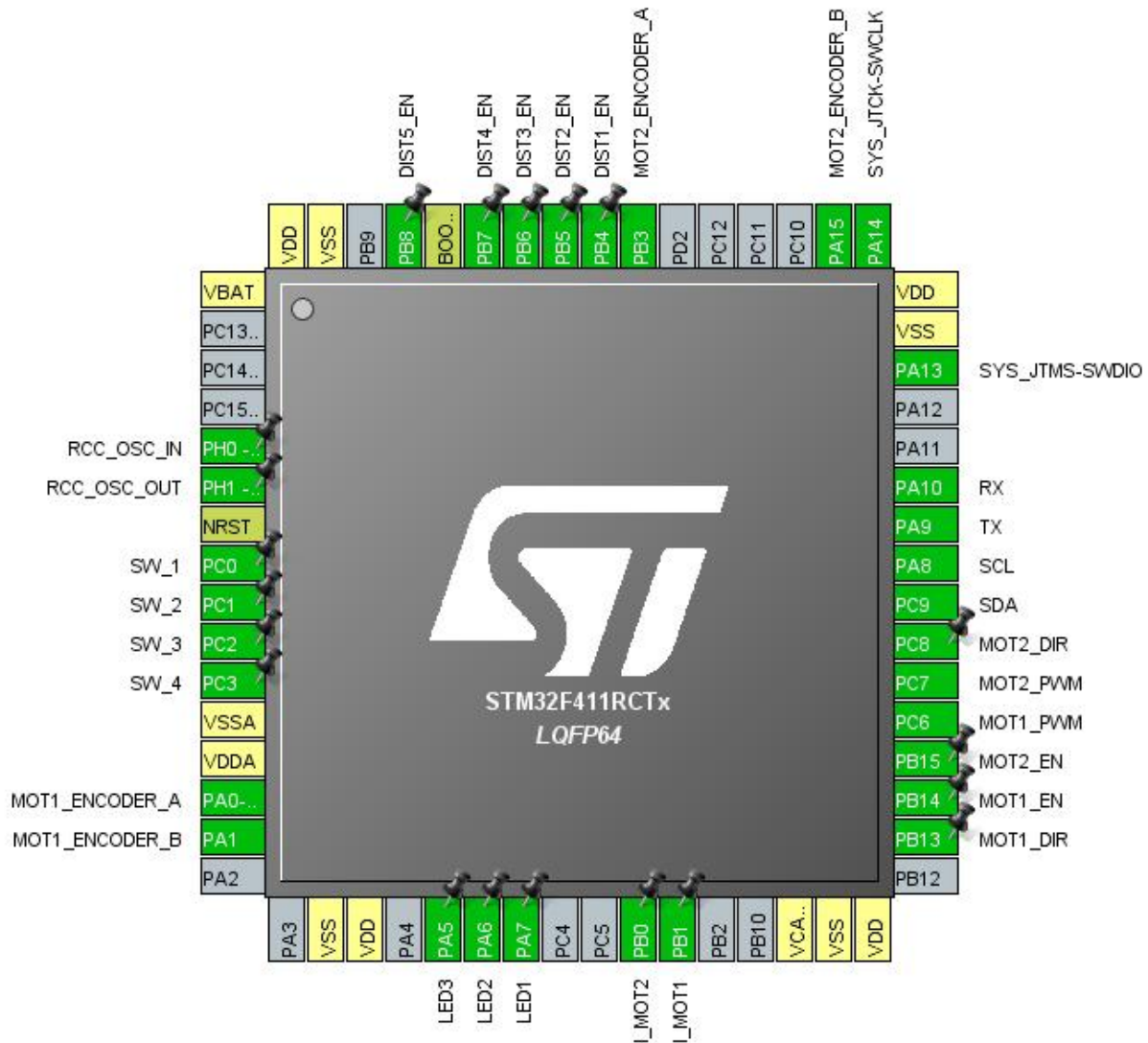
1
2
3
4
5

B Microcontroller configuration

2. Pinout Configuration

4. Clock Tree Configuration