Dot.
Team Description Paper

Dmitriev Vladimir
Presidential Physics and Mathematics Lyceum №239,
Kirochnaya street 8, 191028, Saint Petersburg, Russia
dot.robocup@gmail.com
@dot.robocup

Abstract. This paper presents details of the hardware and software systems of DOTs in the Soccer Open category. Team DOT. intending to participate in the RoboCup World Championship 2021. The document structure was taken from the tdp team TIGERs Mannheim from the RoboCup Major SSL category. Our team learns from the best adult soccer team.

1. Introducing Robot Generation v2021

Fig. 1. CAD rendering of robot.

The v2021 generation presents our second iteration of robots for the RoboCup Junior Soccer Open. A CAD rendering of v2021 is shown in figure 1. With 5 years of experience in designing such robots, many mistakes were made and many lessons learned. The main robot systems (transmission, impact mechanism, vision module, etc.) have been redesigned шn the new iteration of robots, and new elements have also appeared. More details on the powertrain can be found in section 2.1.The specifications for v2018 and v2021 can be found in table 1. Electronics has also gone through a lot of changes. A large number of boards appeared in the robot, and the density of their assembly increased. The details of the different boards, their purpose, and interconnections are outlined in section 3

Table 1. Robots specifications

Year of creation	2018	2021
Dimension	⌀218 x 210 мм	⌀198 x 198 мм
Total weight	2.39 kg	2.29 kg
Driving motors	Pololu 25D 12 V HP 47:1	Maxon EC45 flat with Hall sensors 30 Watt (200142)
Encoders	Pololu 48 CPR encoder	Maxon Encoder MILE 1024 CPT 2 channels with Line Driver (462004)
Driving motors drivers	DRV8870	КРВТ.22.80.02.00ПС
Wheel diameter	50 mm	58.5 mm
Number of rollers	18	20
Dribbling motor	Used Maxon RE13 (344516)	SURPASS C2208 1350KV
Kicker charge	2200 μF @ 50 V	1500 μF @ 250 V
Main microcontroller	ATmega2560	STM32F407
Vision Module	Pixy cam 1	Logitech C920 + Khadas Vim 3
Line sensors	32 x KTIR0711S	32 x TCRT5000
Additional sensors	Ball Presence Sensors, SparkFun 9DoF Razor IMU M0 (Gyroscope, Accelerometer, Compass).	
Power Supply	Multistar Li-Po 4S1P 4000 mAh	4S1P INR18650HG2L 3000 mAh

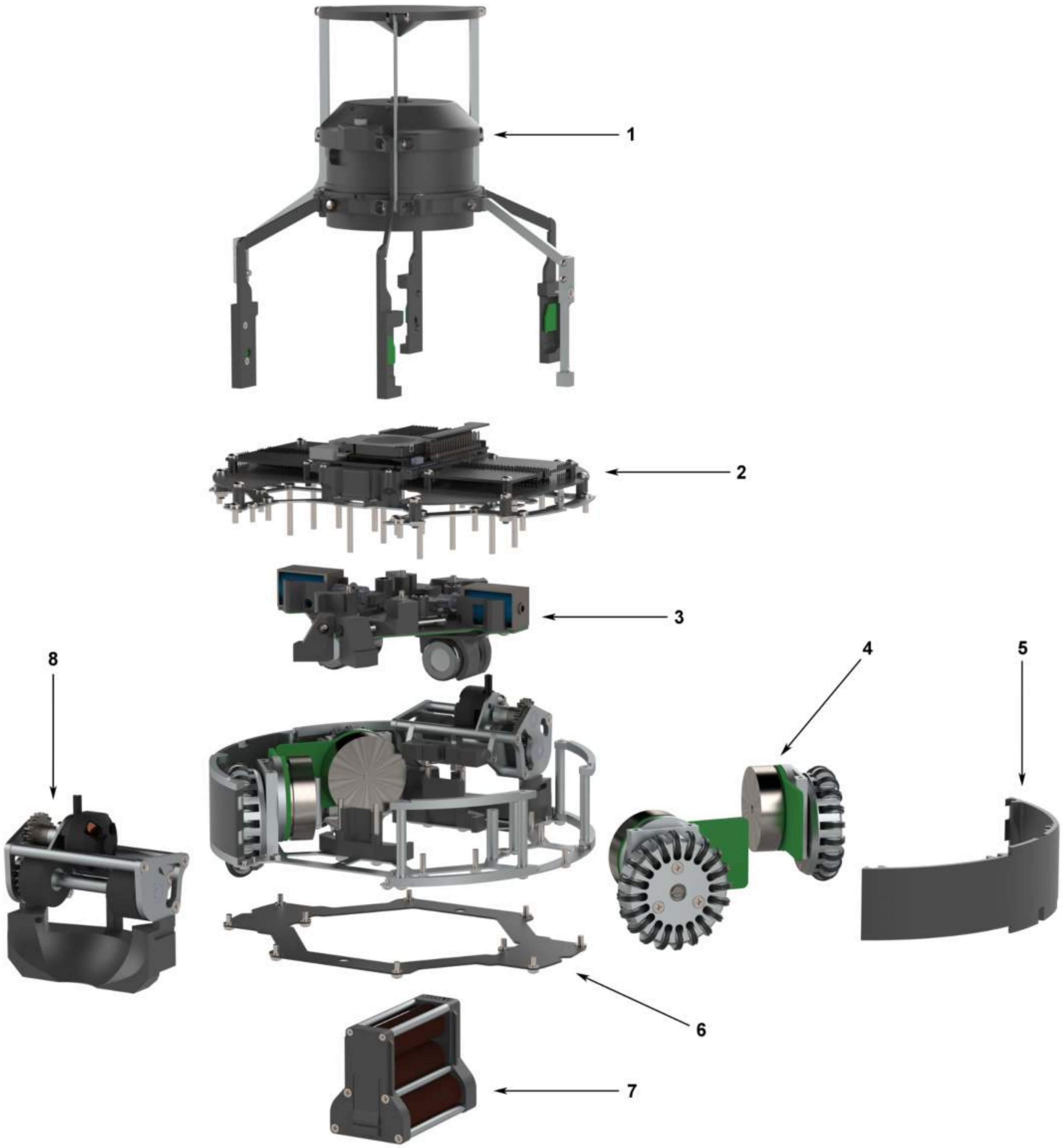


Fig 2. Exploded view of major robot modules. Each interchangeable individually. 1)Vision module 2) Motherboard with VIM3 3) Kicker module 4) Powertrain 5) Snap on protection 6) Line sensors board 7) Battery 8) Dribbler

2. Mechanical Design

Open-Source Dot. v2021 robot design will be made available as open-source design after the RoboCup World Championship 2021
The main change compared to v2018 was the reduction in the size of the robots to 200mm. This was done for future changes in the rules. In addition, the reduction in size allowed us to reduce the total weight of the robots, which allowed us to accommodate all our ideas.

2.1. Powertrain and Wheels

v2018 - GTF Robots omni wheels The v2018 wheels were purchased from robotshop.com. It was cheap aluminum wheels that fit the usual SEMI-POLOLU motors. Wheels could not be retrofitted to be used with the new Maxon motors. The wheels had an aluminum body, aluminum rollers and a steel fitting for the shaft. The rollers were equipped with 18 hard rubber O-rings. The wheels performed well at the Robocap 2018 in Canada.

v2021 - Solid Aluminum Wheels To be able to use new Maxon motors, purchased wheels are replaced with manufactured wheels with a built-in gearbox. Omni-wheels are rotated by the motor via a 3:1 gear set. The production wheel made it possible to increase the diameter of the wheel, install more rollers, and also replace the old rubber O-rings with X-rings. X-rings allow the wheels to have more grip on the field. Overall, wheel size increased slightly compared with v2018 to a total diameter of 58.5 mm.
Omni wheel consists of a steel shaft with two bearings, 20 rollers, steel driven and pinion gears, aluminium wheel base and aluminium wheel cover. Steel gears were purchased from the American gear manufacturing company KNK. Subsequently, these gears were processed by a lathe and milling machine in production. The transmission was produced at the St. Petersburg factories, but the roller shafts were produced on a lathe in our club.

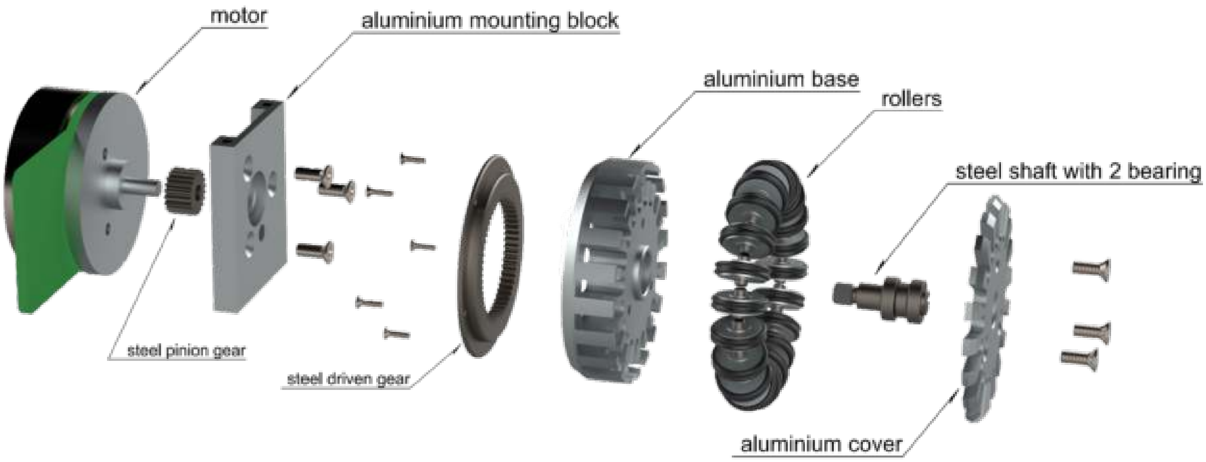
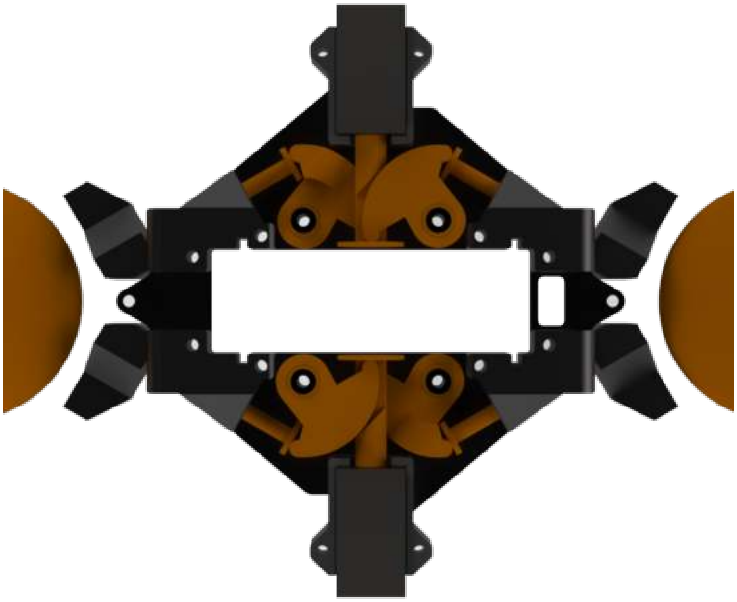
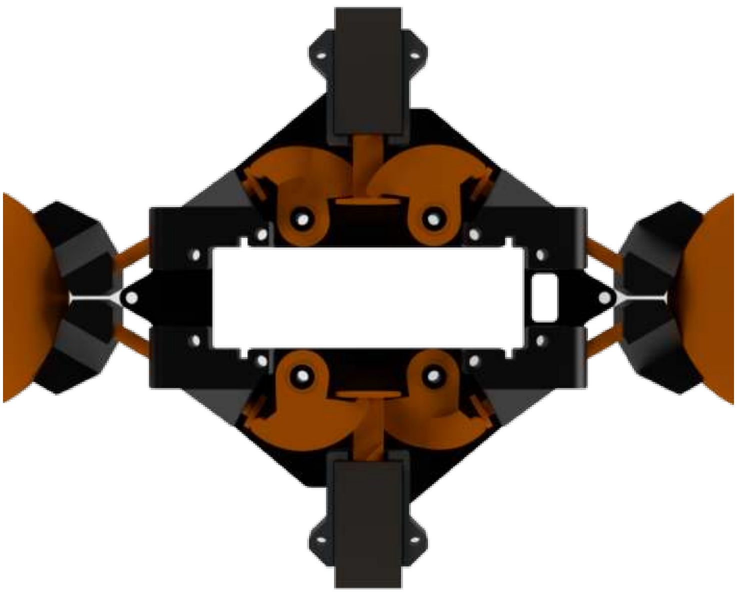


Fig. 3. Cad render of exploded view of powertrain.

2.2. Kicker





The impact mechanism consists of two solenoids (Gangbei-0630B), two steel rods for the solenoids, four plastic swings, 4 aluminum guides and 4 plastic kickers. Plastic parts are printed on a 3D printer. Steel and aluminum parts are turned on a lathe in our club. For better sliding of the guides, fluoroplastic tubes are inserted in the guide holders.
In v 2018 used a three-way mechanism to hit the ball. This was the main feature of our robots at the world stage of Robocup 2018. In the new version, robots have the ability to approach the ball from both sides, so the hitting mechanism has been modified. To avoid weight gain, only 2 solenoids were used for 2 sides. So that 2 solenoids can hit in both directions, we have developed a swing-type mechanism. During a voltage supply, the solenoid pushed the rod, thereby turning a special moving part, which was mounted on 2 bearings to reduce friction. This part, in turn, pushed the kicker rods and a kick was made.

Fig. 3 Cad render of Kicker mechanism before kick.
Fig. 4 Cad render of Kicker mechanism after kick forward.

2.3. Dribbler

Over the years we have tried several versions of dribbling. Dribbling with two separate rollers proved to be optimal for the soccer open ball. The robot has 2 dribblers in the front and back in the new robot iteration. The dribbler rollers are driven through a gear transmission by CJL90514303B motor. A bucket is installed at the bottom of the dribbler so that the ball has 3 points of support: the bucket, the first and second rollers of the dribbler. The rollers were molded from silicone with a hardness of 30 Shore. During the casting, a black dye was added to the color silicone. We installed ball detection sensor boards to determine the position of the ball in the dribbler. One board is an infrared LED, and the other board is a photodiode.

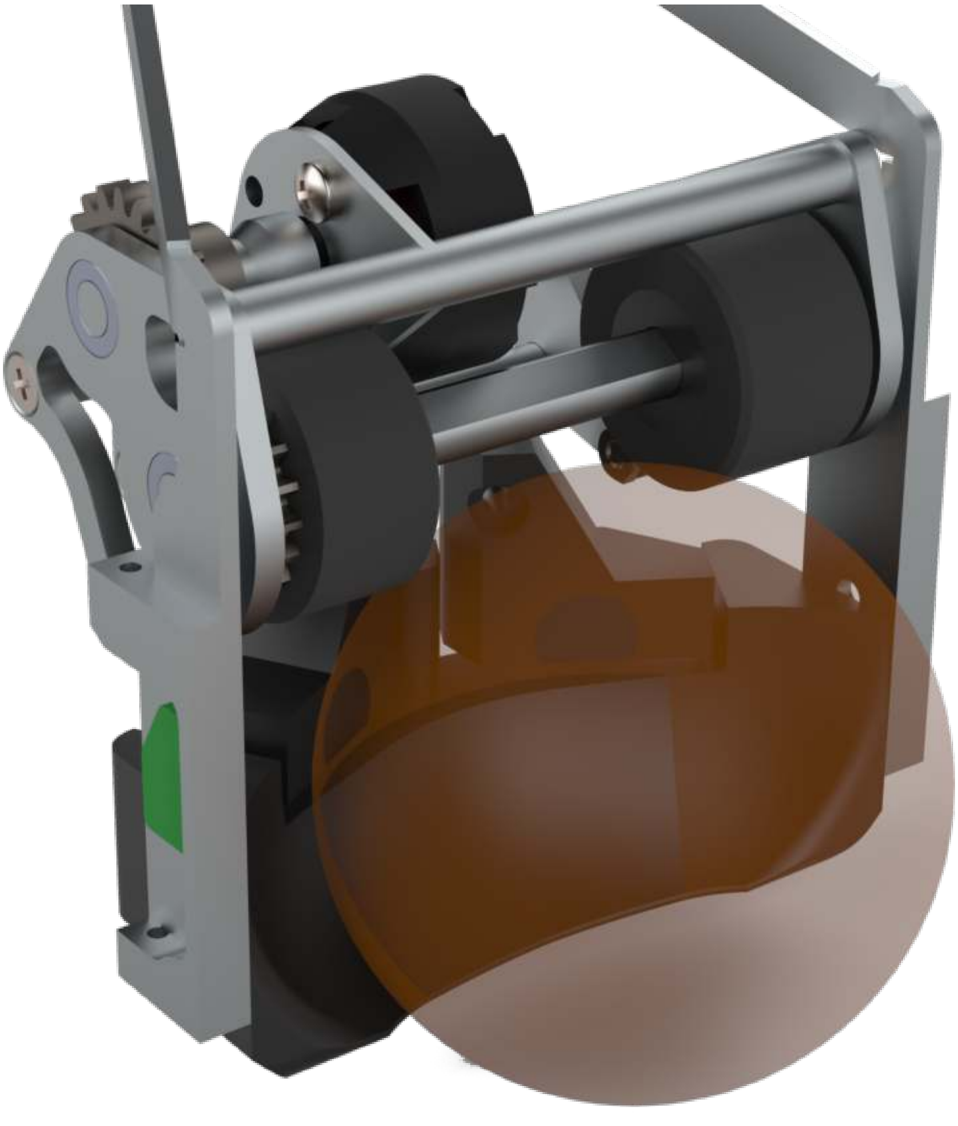


Fig. 5. Cad render of dribbler.

2.4. Vision module

The vision module consists of a camera (Logitech C920), lidar (ILIDAR A0602), and a cone-shaped mirror. The tapered shape allows you to see the ball well both close and at a distance, and also allows you to easily calculate the approximate distance to the ball. The mirror is turned on an aluminum lathe in our club. After turning, the product is polished to a mirror state of the aluminum surface. To reduce the weight of the shearing module, instead of the usual acrylic tube, we used three aluminum strips that were perpendicular to the center of the chamber. The camera image and lidar values are processed on a single board computer.

Fig. 6. Cad render of vision module.

2.5. Battery

This year we have made our own battery, which consists of four 18650 batteries (INR18650HG2L). Welded nickel strips are used to transfer power between the 18650 batteries. In this battery, we used a connector with reverse polarity protection, so we made a special docking station to charge it.

Fig. 7. Cad render of battery.

Fig. 8. Charging dock station for battery.

3. Electronic design

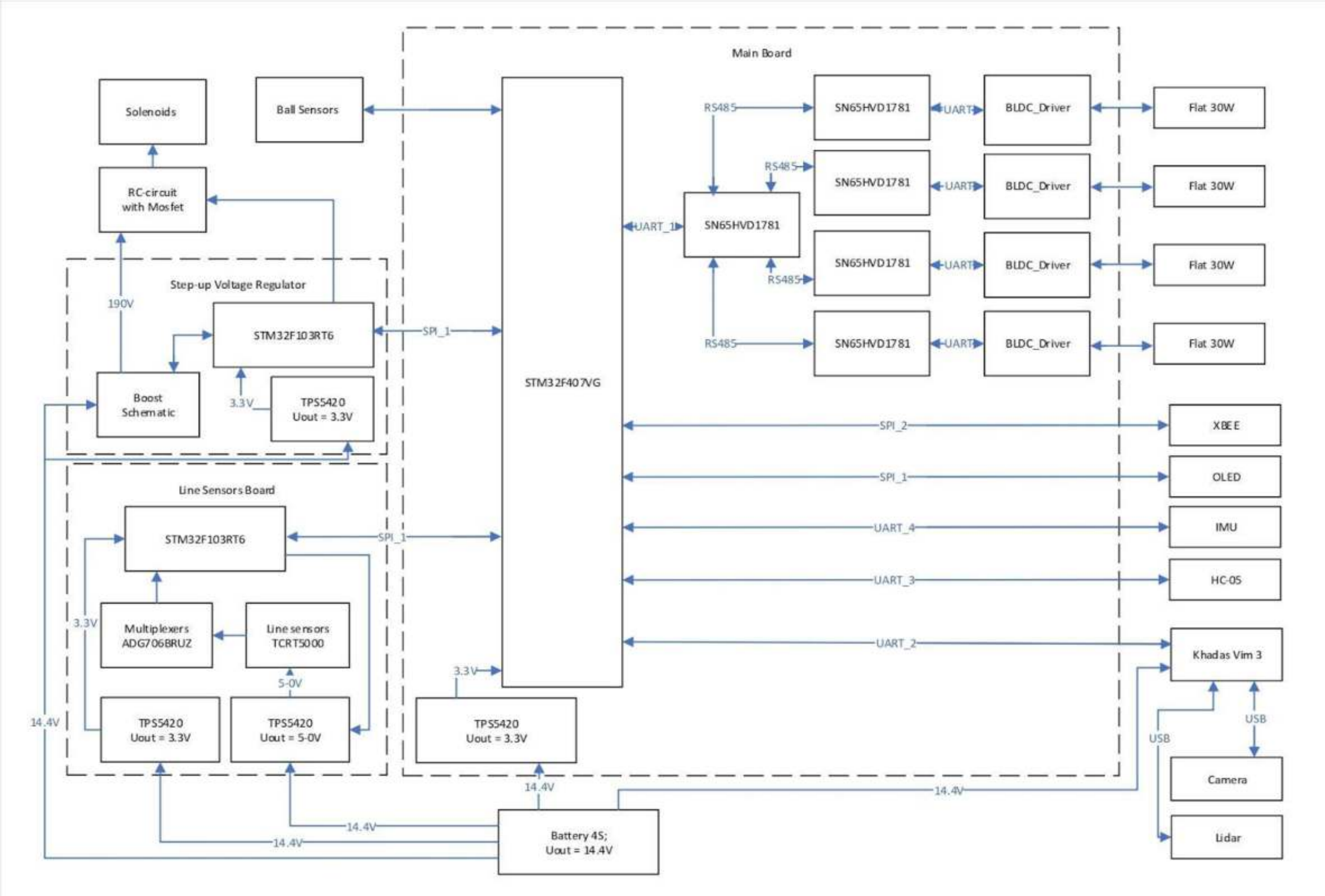


Fig. 9. Functional Schematic.

3.1. Main board (Motherboard)

The main board is the heart of our robot. It is with her that we connect all our components and boards. stm32f407 is an essential link in all processes of our robots. Khadas Vim 3 transmits information to the main board about the operation of the striking mechanism, dribblers and motors. Communication between Khadas Vim 3 and stm32f407 is carried out via usb-uart with a cp104 convector.

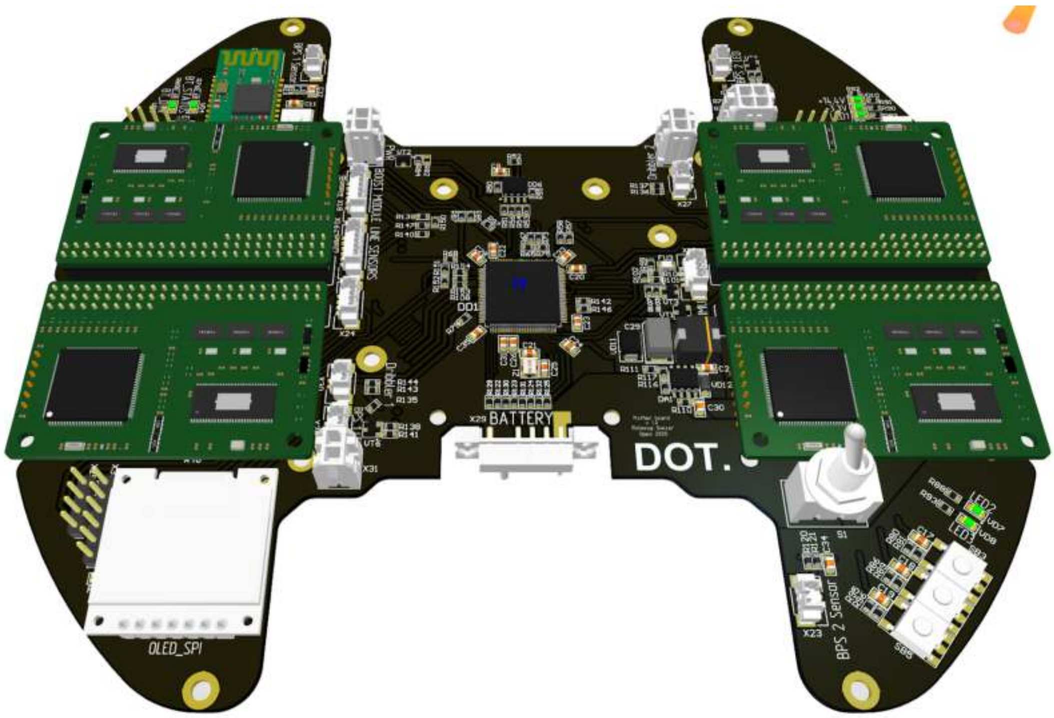


Fig. 10. Render of Main board.

3 buttons are used for control. To display information, we use a spi-connected oled screen. In addition, it is possible to transmit debug information via bluetooth module HC-05.

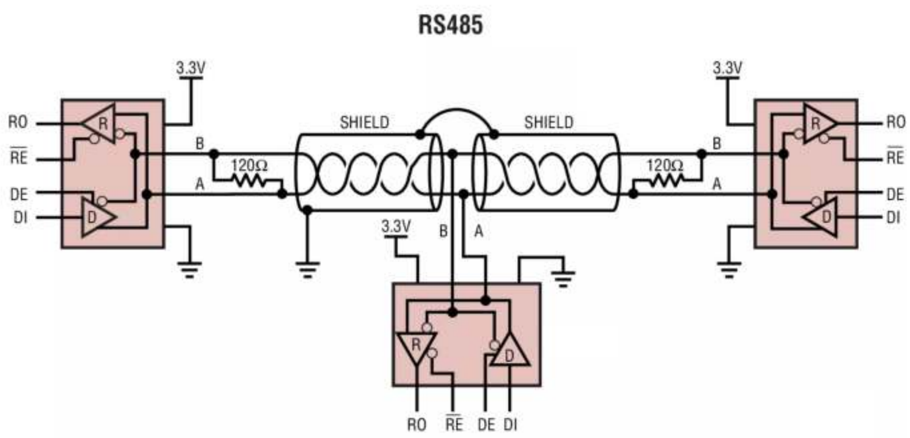


Fig. 11. RS485 connection schematic.

There are 4 brushless motor drivers on the motherboard. We bought these drivers from a small St. Petersburg laboratory, which created them for their own purposes. Communication between the drivers and stm32f407 is carried out via SN65HVD1781 using the RS485 protocol, which is new for us.
To connect two robots with each other, we use the HC-05 bluetooth module, and if the connection between the robots via bluetooth is not stable, then there is the possibility of connecting by xbee.

3.2. Line sensors board

There are 32 TCRT5000 sensors on the line sensor board. We use stm32f103 to read and transfer data to the main board. We use 2 ADG706BRUZ multiplexers for 16 channels to read all 32 sensors.

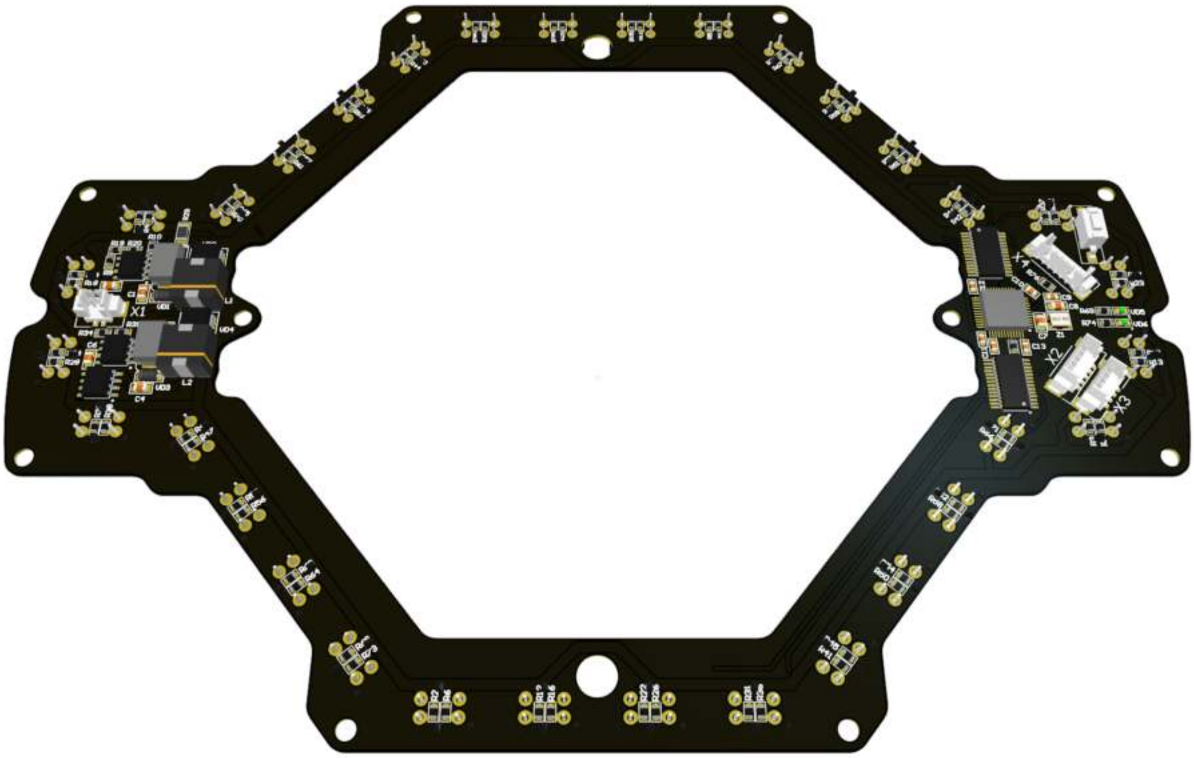


Fig. 12. Render of Line sensors board.

There are different structures and colors of field surfaces in different countries and cities. There is a separate voltage regulator that powers the infrared LEDs of the line sensors. The output voltage of the voltage regulator is controlled through a digital potentiometer via SPI from stm32f103.

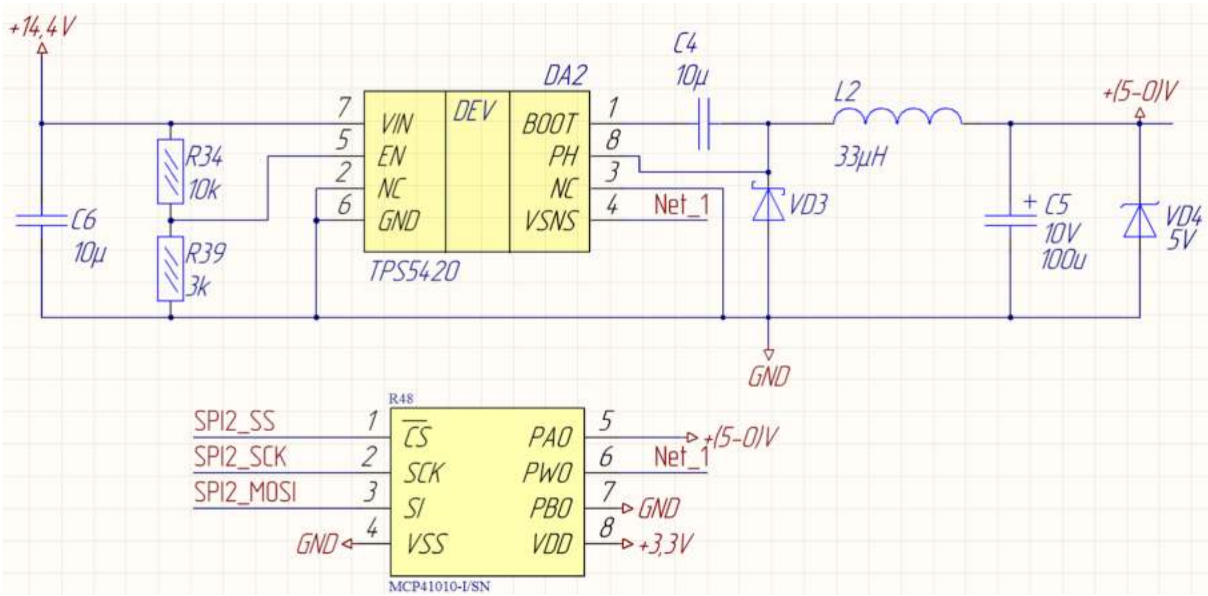


Fig. 13. Schematic of voltage regulator for line sensors LEDS.

3.3. Step-up Voltage Regulator (SVR)

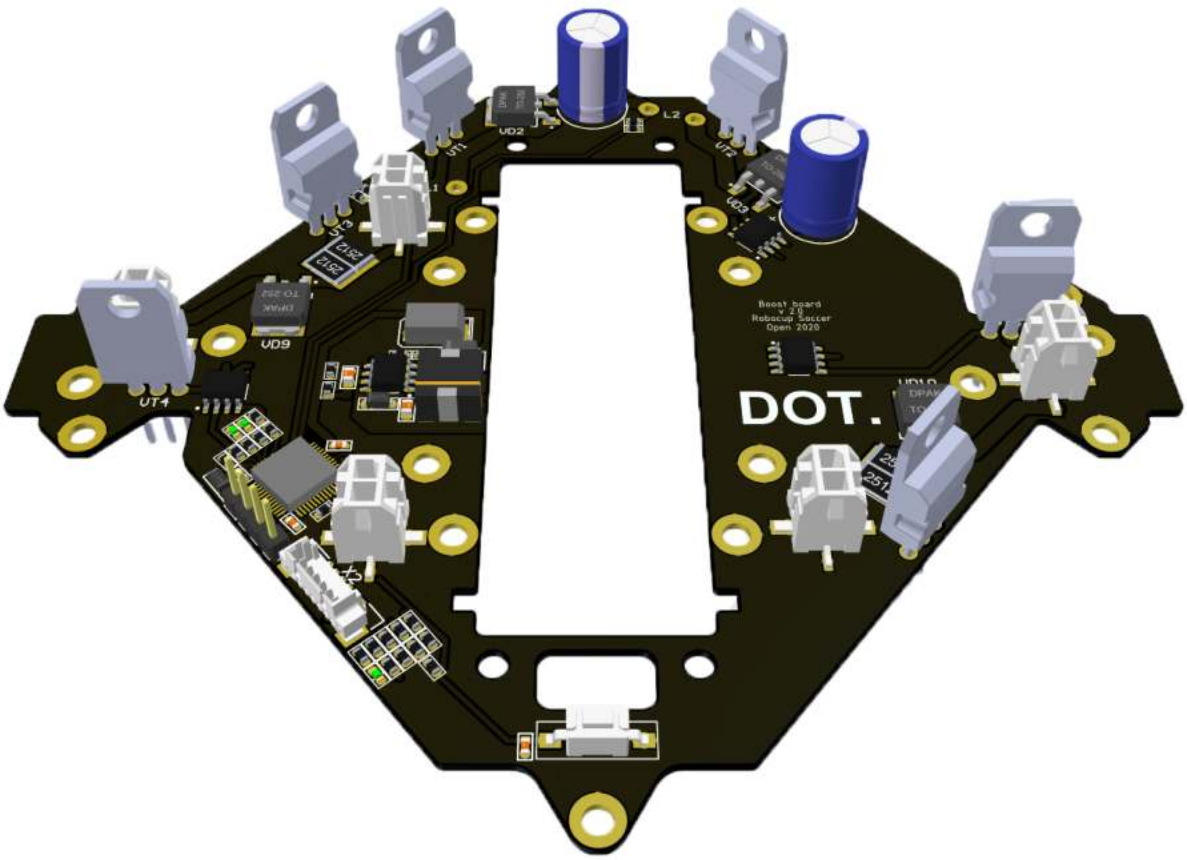


Fig. 14. Render of Step-up Voltage Regulator board.

The step-up voltage regulator is a completely new type of circuit board for our robots. In previous versions, we used ready-made solutions. This year, the ready-made solution of the Step-up voltage regulators did not suit us, because

they were either very large and heavy, or raised at low voltage. Then we decided to develop our own step-up voltage regulator module. There are two ways to step up the voltage: through an inductor and through a transformer (Tesla coil). The way to increase the voltage through the transformer did not suit us because they would be very large. As a result, there was only a way to step up the voltage through the inductor. Idea taken from this site: learnabout-electronics.org/PSU/psu32.php
Our step-up voltage converters use two circuits to raise the voltage from 14.4 volts to 60 volts and 60 to 240 volts. The voltage boost factor is 4, which is the maximum and optimal factor for these circuits. If you put a 3 circuit, then you can get the voltage across the capacitors up to 1000 volts! This scheme works because there is no need to supply voltage to the load constantly. The solenoids hit once every 10-20 seconds, when the very charging of the capacitors with high voltage takes about 5-7 seconds. We made our own inductors for the Boost converter circuit.

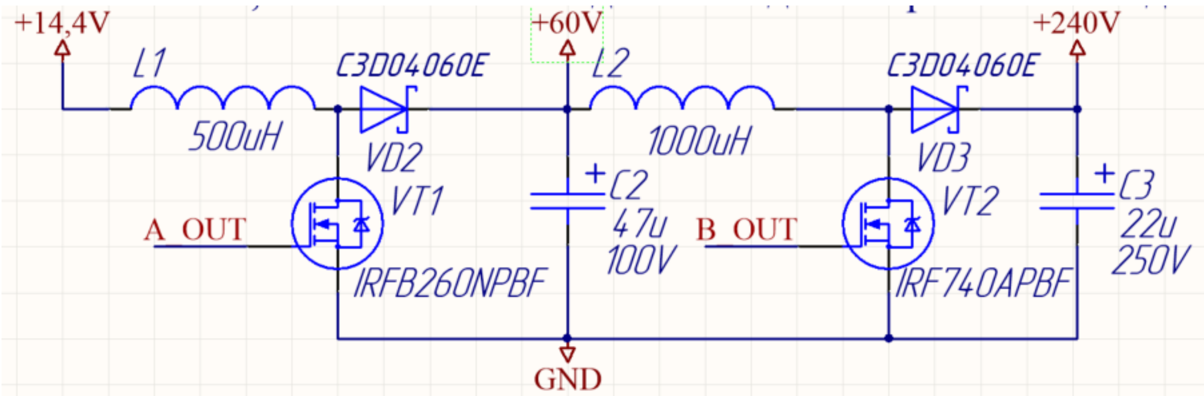


Fig. 15. Boost converter circuit.

The stm32f103 is located on the voltage boost converter board. Data between stm32f103 and stm32f407 is transmitted by spi. With the help of stm32, we control the charging process of the capacitors and can change the final charging voltage. At the moment we are using 250 volt capacitors, and we are charging them with 210 volts.

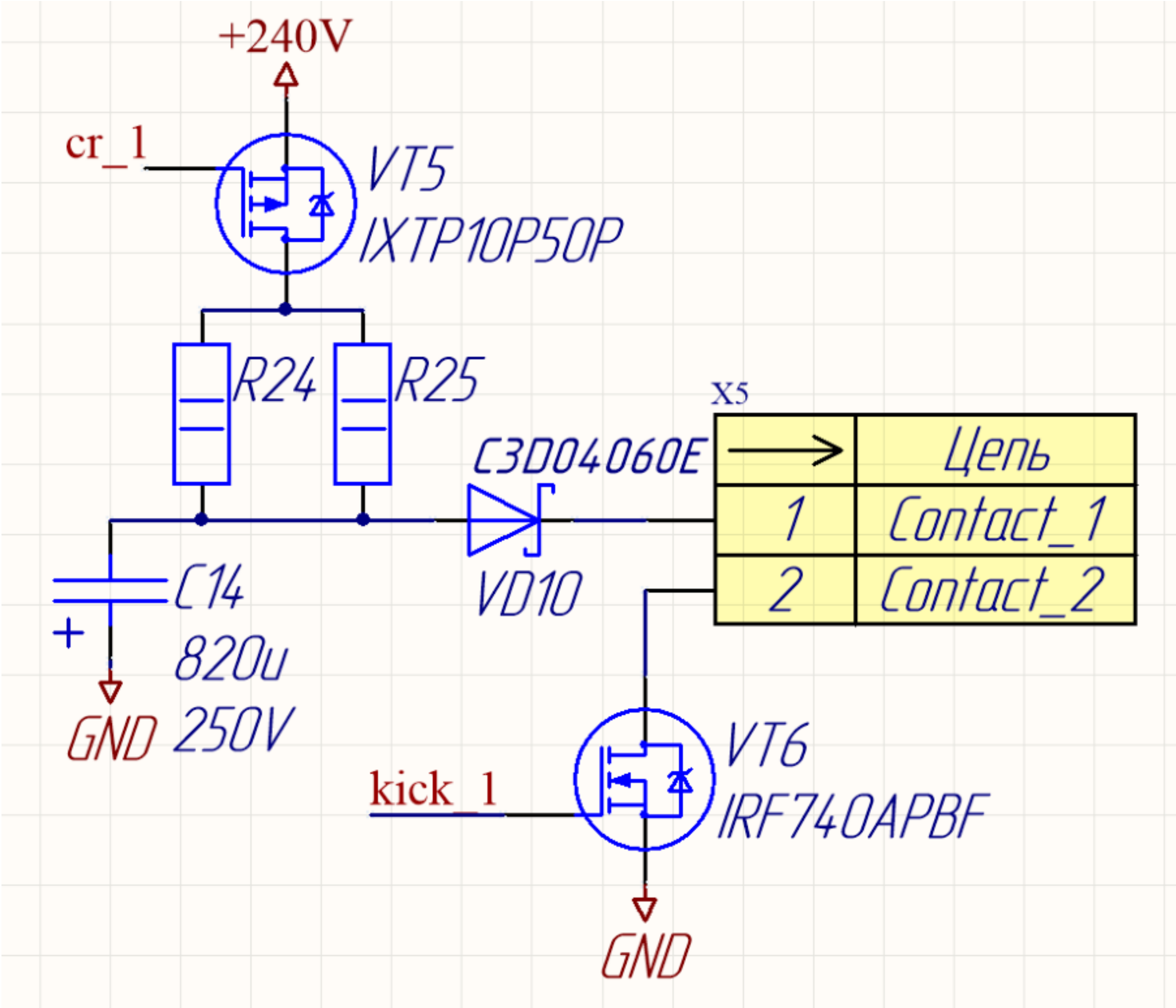


Fig. 16. Solenoid impact circuits.

We have two solenoids, which means two solenoid impact circuits. We can control the charging of each capacitor separately. Schottky diodes are used to prevent reverse current from solenoids.

Due to the fact that high voltages and high currents pass through the transistors (in the case of a solenoid), we used transistor drivers to control them.

3.4. Remaining boards

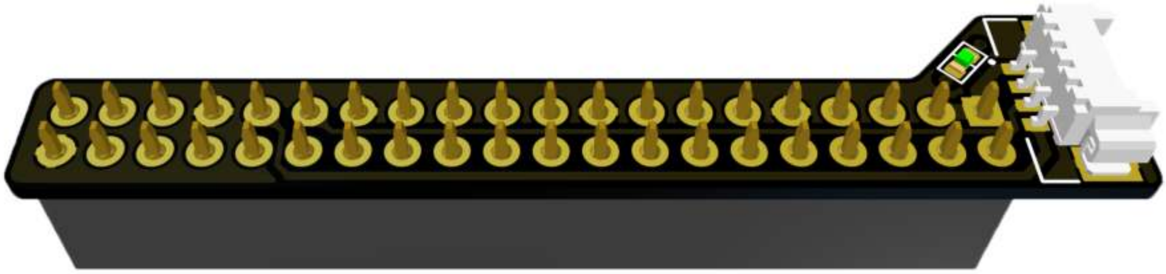


Fig. 17. Render of Vim Connection Board (VCB).

In addition to the camera, a lidar is connected to the khadas vim 3 microcontroller. Lidar transmits information via uart as opposed to a camera. This board was created to reduce the number of wires and make it possible to turn the lidar on and off using a transistor.

Fig. 18. Render of Battery board.

Battery board is the transition between the 18650 batteries and the TE 1473971-2 connector.

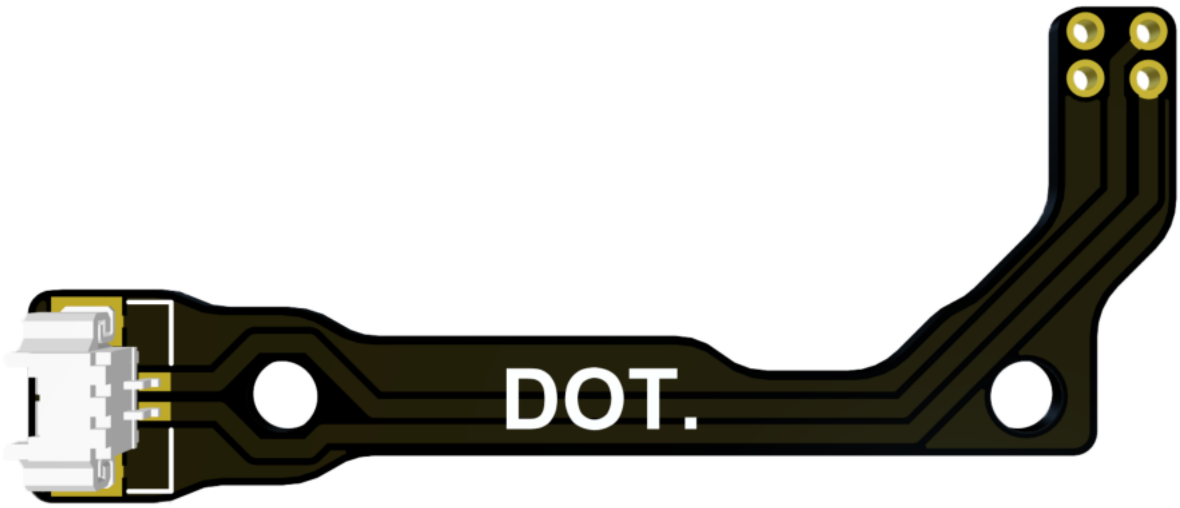


Fig. 19. Render of Camera connection board (CCB).
CCB is designed to reduce the number of wires and make it possible to disconnect the camera without disconnecting the usb from khadas vim 3.

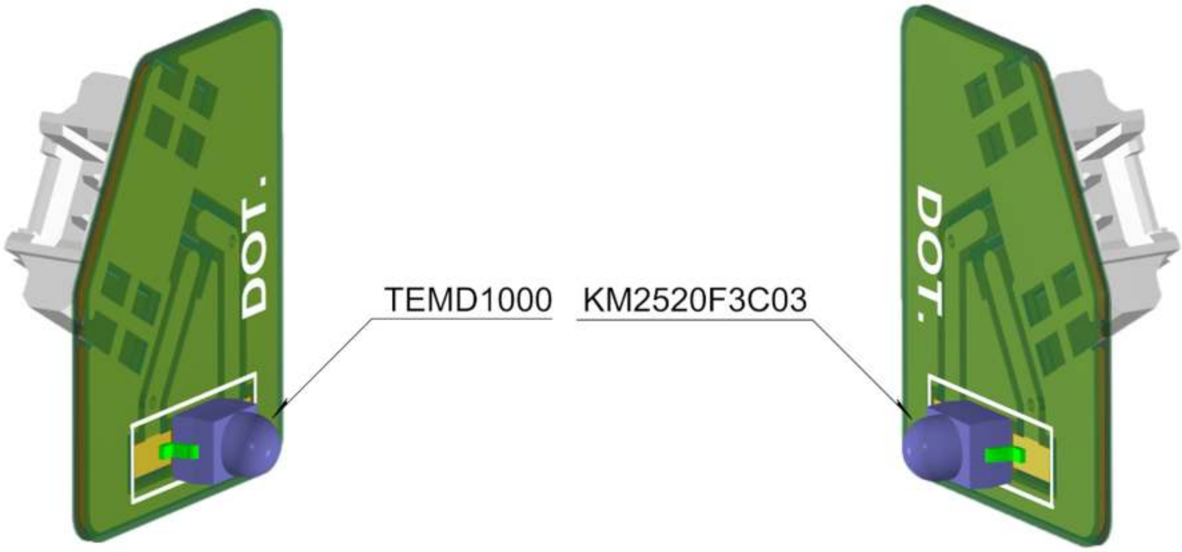


Fig. 20. Render of Ball Presence Sensor boards (BPS).
There are 2 parts in Ball Presence Sensor boards: board with TEMD1000 (Photodiode) and board with KM2520F3C03 (Infrared emitting diode). When the ball is not in the capture zone, the photodiode sees infrared radiation from a specific range of the infrared LED. When the ball is in the capture zone, the photodiode does not see infrared radiation of a certain range from the infrared LED, respectively.

4. Programs and algorithms

4.1. Khadas Vim 3

Khadas VIM3 is the main brain of our robots. It is the single board microcomputer, which was selected for its lightweight, low power consumption and significant performance. In order to further speed up the work of the machine vision libraries, multiple optimizations were used, such as the use of precompiled linear algebra libraries such as Eigen / Lapack, the OpenMP / TBB parallelization libraries, as well as many minor improvements. Most of the code is written in Python 3.8 to reduce development time. But performance is very important too. JIT compilation with llvmlite and the Numba library is used for some important and "heavy" functions, which can increase performance several tens of times, up to the level of C / C ++. In addition, to accelerate matrix operations and work with video streams, the OpenCV and NumPy libraries are used, which also allows achieving performance close to lower-level languages. For debugging we use tigerVNC because it allows you to achieve good speed and stability work with WiFi.

4.2. Strategic

The strategy is based on an advanced positioning system for robots on the field. With its help, we get the ability to calculate the speed of objects on the field, such as enemy robots and the ball. Using this data, robots can build a field model in real time and predict the trajectories of individual objects, which can significantly increase the accuracy of all algorithms. During the game, there is a strict division of the roles of robots, which can nevertheless change. In addition, in the case when one of the robots leaves the game, the remaining one switches the behavior model to a hybrid one, in which the remaining bot not only hits the balls that fly into the goal, but also, capturing them, tries to score into the opponent's goal. Also, several methods are used to determine the position of the opponents' robots, with the help of which our robots cover the ball from the enemy with their body. In a normal situation, the standard attacker's behavior consists of several stages: intercepting the ball, covering the ball from the opponent's robots with the body, and outflanking the opponent's robots at the edge of the field. This strategy allows you to get a significant advantage against opponents who do not know how to identify the opponent's robots. The goalkeeper's strategy is to predict the trajectory of the ball, and to intercept it, for subsequent transfer to the attacker, which allows not to lose the ball in difficult game situations. In addition, any strategy is adaptive, that is, at any moment of the game, one of the possible behavior models is selected for each robot, which is the priority and most effective at the moment.

4.3. Algorithms

Most of the algorithms are written in Python 3.8 and run on a microcomputer. STM microcontrollers perform functions of processing data from the microcomputer, such as feeding the obtained speeds to motors. At each moment of time, the robot calculates its position on the field using a camera, guided by goals. Enemy robots are identified by convolutional neural network, and the obtained values are refined using the analysis of the gradient value. We use a method of least squares to approximate the trajectories. With this algorithm we achieved the maximum accuracy of determining the velocities. In addition, we use the prediction of the positions of objects on the field for a short period of time ahead to improve the stability of work at high speeds.

Fig. 21. convolutional neural network in work (green lines are the centers of the robots).

4.4. Debugging

To simplify robot debugging, we use a cross-platform terminal written in Python 3 using the PyQT5 framework, which simplifies the creation of GUI applications. This terminal allows us to display the position of both robots on the field and their rotation angles. In addition, we can use it to conveniently view logs received from both robots. We can also send commands to robots using buttons or hot keys. The terminal is not used during tournament matches.

Fig. 22. Screenshot of debug terminal.

5. Problems and solutions

5.1. Mechanical part

5.1.1. Motors problem

This is a well-known problem that we have been struggling with for several years and only now have we managed to solve it. All the motors that we tried were either slow or weak, or large or too heavy. We tried motors from companies: Pololu, Gekko and Sumozade. Our solution turned out to be motors from Maxon, namely Maxon Flat EC45. This decision gave rise to several more problems: with too many rpm and because of the price of the motors we don't want to break their shafts due to too high a load. The solution to these problems is the bearing assembly. We borrowed it from the teams from the Robocup Small Size League. The bearing assembly made it possible to reduce the wheel speed from 3000 rpm to 1000 rpm, as well as to minimize the load on the motor shaft itself.

5.1.2. Dribbler problem

Due to the fact that in the old version, dribbling was only on one side, the robots wasted time trying to go around the ball and capture it by dribbling. The solution is very commonplace: use two dribbles (front and back).

5.1.3. Kicker problem

After adding the second dribbler, it was necessary to kick forward and backward while guarding the hitting feature in three directions (left, straight, right). We didn't want to put two more solenoids, which would add more weight, so the solution was “nanotechnology”, thanks to which we were able to keep the ability to strike in 3 directions, but at the same time hit the ball from both dribbles. Through the swing, the solenoid is beaten simultaneously forward and backward.

5.1.4. About how the robot lost weight

A large number of new features have appeared in the new version of robots, due to which a large number of parts have been added and, accordingly, the weight of the robots has increased. The solution was to reduce the overall dimensions of the robots from 220 x 220 mm to 200 x 200 mm, as a result of which the robots began to weigh approximately 2300 grams.

5.1.5. Battery problem

Due to the radically redesigned layout of the robots, there was not enough space for standard li-po batteries. The solution was 18650 li-on batteries. 18650 batteries made it possible to configure the battery any way. As a result, the battery has the shape of a Christmas tree, thickening towards the base, where we have two 18650 batteries, when there is one battery in the upper and middle parts.

5.2. Electronic part

5.2.1. Line sensors

In the previous version, we used KTIR0711S to detect the white line. These sensors are good for a flat surface, not for a fuzzy field, which is why we often flew out, because we did not see the white line. Before settling on the TCRT5000, we tried qrd1114 and homemade line sensors. Homemade line sensors proved to be very unreliable, and the qrd1114 depended too much on ambient light. The TCRT5000 hardly noticed any irregularities due to the high viewing angle of the sensors themselves and the high intensity of the LEDs. In addition, we have supplied a separate voltage regulator for the LEDs of the line sensors, so that we can change the glow intensity for different types and colors of the field coverage. More info in 2.2.2.

5.2.2. Wire connectors

In all old projects we used pls connectors for signal wires and xt60 for power wires. These connectors are cheap, but not durable and often fly out of the connectors during games. We borrowed the solution to this problem from our computers. The power cables are connected with Molex connectors. Molex connectors are more durable and hold connections better, but they are more expensive and more difficult to buy. We used Molex Microfit for the power wires and Molex Picoblade for the signal wires. In addition, the battery is connected to the main board via TE Connectivity 6376042-2 and 1473971-2.

5.2.3. Step-up voltage regulator

This year, the ready-made solution of the Step-up voltage regulators did not suit us, because they were either very large and heavy, or raised at low voltage. Then we decided to develop our own step-up voltage regulator module. There are two ways to step up the voltage: through an inductor and through a transformer (Tesla coil). The way to increase the voltage through the transformer did not suit us because they would be very large. As a result, there was only a way to step up the voltage through the inductor. More info in 2.2.3.

5.3. Program and algorithms

5.3.1. Debugging

The main problem with writing programs is checking them for bugs. Previously, our robots traveled with long wires that transferred all information from the robot to the laptop. This year we are using bluetooth protocol to transfer all information between robot and laptop. We created a cross-platform terminal to simplify the debugging of robots programs. It is based on Python3 and pyqt5, which made it possible to reduce the time spent on fixing errors. More info in 4.4.

5.3.2. Performance

In the process of writing algorithms for video vision, we encountered a problem of lack of performance. To solve this problem, we have developed an extremely accurate interpolation algorithm for determining the color match with an asymptotic complexity of O (1) in memory and time in the worst case. This solution has increased the speed of work by 2.5 times.

5.3.3. “Sauron's eye”

Most of our strategies are based on the position of enemy robots. Solving this problem was our top priority. We found many ways to solve this problem, but the convolutional neural network with lidar coworking turned out to be the most effective. Our robots are able to detect the opponent's robots and determine their location using a convolutional neural network written in tensorflow lite framework based on mobileNet architecture, using pruning technology and int8 quantization technologies, which allows us to achieve real-time speed of work.

Once we determine the side where the enemy robot is located, we compare this data with data from lidar and determine the exact location of the robots.

5.3.4. Speed reaction

Any algorithm has a running time, and any motor has inertia. Due to these reasons, the robot cannot work instantly and all actions occur with a certain amount of delay. In order to increase the accuracy and stability of the robots, the calculation of the trajectories of the future movement is used, which allows to increase the positioning accuracy and increase the average speed of the robot, which gives an advantage in the game against fast opponents. Positions are also projected onto the out-bounds in order to minimize the risk of going beyond it. In addition, by penetrating the trajectory of the ball, the goalkeeper is much more efficient in catching the ball, and the attacker can intercept the ball while it is moving at significant speed.

6. Our team

DOT. is a team that represents the robotics club in Presidential Physics and Mathematics Lyceum 239 in Russia, founded in August 2016 for the purpose of participating in RoboCup Junior competitions. Team DOT. was formed as a result of the merger of the DOT. team (RoboCup Junior Soccer Lightweight) and the MVK team (RoboCup Junior Soccer Open) to participate in the RoboCup Junior. Soccer Open category. Current members include:

- Romanko Pavel - romanko69@mail.ru, mentor.
- Dmitriev Vladimir - v0hmly@gmail.com, finished 1st year at ITMO University, team leader, pcb designer, pcb solder and documentation maker.
  Achievements and RoboCup experience:
  - Best Robot Design and SuperTeam 1st Place in RoboCup 2019 in Sydney in Soccer Open category
  - Second place in Robocup Russia Open 2019 in Soccer Open category.
  - First place in Robocup Russia Open 2018 in Soccer Lightweight category.
- Ustinov Ilya - ilust.d@gmail.com, finished 10th grade at FML №344, 3d model designer, maker of documentation for production and fitter.
  Achievements and RoboCup experience:
  - Second place in Robocup Russia Open 2019 in Soccer Lightweight category.
  - SuperTeam 1st Place in RoboCup 2018 in Montreal in Soccer Open category
  - First place in Robocup Russia Open 2018 in Soccer Open category.
- Knyazhytsky Roma - knyaz.roma@gmail.com, finished 9th at Presidential FML №239, main programmer.
  Achievements and RoboCup experience:
  - Second place in Robocup Russia Open 2019 in Soccer Lightweight category.
  - First place in Robocup Russia Open 2018 in Soccer Lightweight category.
- Travin Aleksandr - alexander.travin.01@mail.ru, finished 9th grade at 564 school, pcb solder
  Achievements and RoboCup experience:
  - repeated participant and winner of the robofinist festival.

Fig.1. Photo of the DOT. team. Team members and mentor from left to right:Travin Aleksandr, Knyazhytsky Roma, Ustinov Ilya,Romanko Pavel (mentor), Dmitriev Vladimir.

7. Discussion and Conclusion

7.1. Future plans

For many of us, this year is the last when we can participate in the RoboCup junior league. Ilya and Roma have state exams and university entrance ahead of them this year. The final of our team is victory in the RoboCup world stage in 2021. And of course, we want to continue our work in the RoboCup Major League, but in different teams from different universities.

7.2. Acknowledgements

Team DOT. would like to express gratitude to Pratusevich Maxim Yakovlevich, director of the Presidential Physics and Mathematics Lyceum №239 and Romanko Pavel Nikolaevich, mentor of our team. We also would like to express gratitude to our sponsors: Charitable Foundation of Temur Aminjanov "Finist", CJSC Kirovsky Zavod and the Foundation for the Support of Classical Education "Anabasis". Furthermore we would like to express gratitude to Evgeny Shandarov, RoboCup Junior Russian Regional Representative and International Committees for organizing RoboCup Junior competitions. In conclusion, we want to say a huge thank to all those teams that shared information and technology with us when we were still at the very beginning of our big robotics journey.

7.3. Conclusion

We have done a lot of work this year. Due to the coronavirus, we had less time than we planned at the beginning of the school year. That is why a lot of information was not included in this document. The members of the DOT team are very happy, and we say thank you very much for reading up to this point. If you have any questions, ask us by email or write to us on instagram.

8. References

1. КРВТ.22.80.02.00ПС Datasheet: goo.su/1Ria (short link) drive.google.com/file/d/11dqx-Hxc7nWz4hxk-LhudH-szbRQZNbD/view?usp=sharing (full link)
2. ELP-USBFHD08S-L36: aliexpress.ru/item/32883126314.html
3. ILIDAR A0602: aliexpress.ru/item/32952097259.html
4. INR18650HG2L Specification: www.li-force.ru/uploads/entity/lghg2l.pdf
5. Gangbei-0630B: aliexpress.ru/item/32778544762.html
6. KNK SI0.5-60: www.khkgears.us/catalog/product/SI0.5-60
7. KNK SS0.5-20: www.khkgears.us/catalog/product/SS0.5-20B
8. TE 1473971-2: www.te.com/usa-en/product-1473971-2.html
9. TE 6376042-2: www.te.com/usa-en/product-6376042-2.html