Violet (Team SHP-2)

A robot designed to solve the WRO FutureEngineers 2021 task (Version Mk. IV)

Our video review of the robot on Youtube

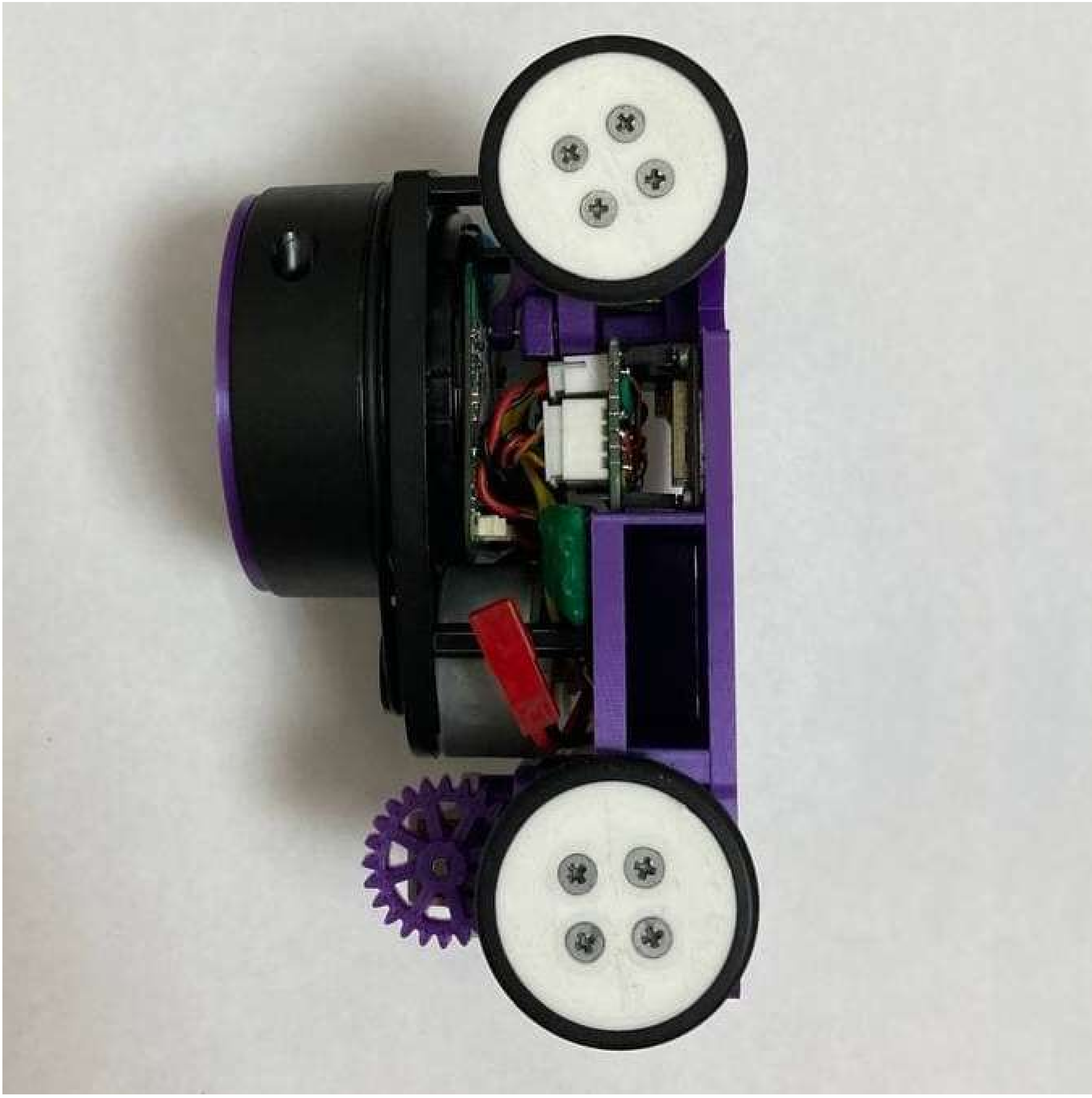
Introduction

We were tasked with designing and assembling an unmanned vehicle that could correctly and accurately accomplish the WRO FutureEngineers 2021 mission. To achieve our goals, we built a four-wheeled robot based on the ESP32. The main sensor of this robot was a lidar system, which gives information about the distances to points around the robot in 360 degrees. During development, geometric features of vehicles, such as Ackermann angles, were taken into account.

Repository content

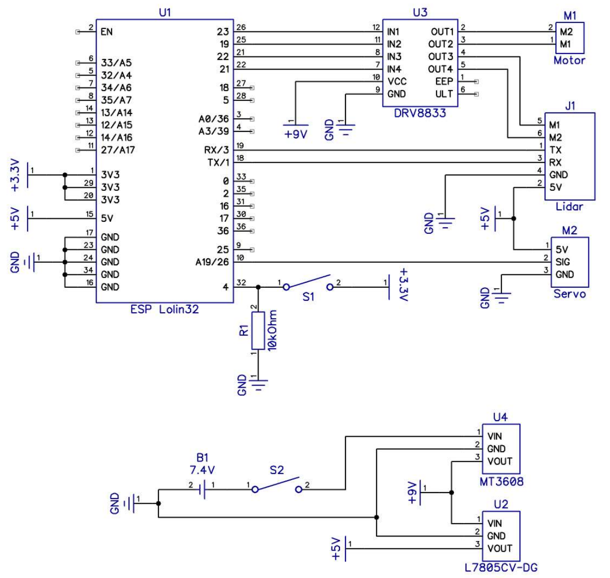
- The "3D-models" folder contains all versions of 3D models for printing the robot
- The "final/main" folder contains the final code for the final stage
- The "qual/main" folder contains the final code for the qualitication stage
- The "electromechanical component schematic" folder contains a schematic of the electromechanical representation of the robot
- The "Team photo" folder contains the team photos required by the regulations
- The "Views" folder contains overview photos of the robot
- The "readme_photo" folder contain photo for readme.md

Photo of the robot

*If it does not open, check the "views" folder, please

Schematic diagram of an electromechanical device



#	Prefix	Name	Description	Quantity
1	B1	Battery	Li-Po 7.4V S2	1
2	J1	Lidar	RPLidar A1 360 deg	1
3	M1	Motor	Pololu Micro Metal Gearmotor	1
4	M2	Servo	MG90S metal gear servo	1
5	R1	Resistor	10kOhm 6.8x2.8	1
6	S1	Button	Tactile switch 12x12x7.3	1
7	S2	Switch	Two-position switch	1
8	U1	ESP Lolin32	ESP-WROOM-32 BT/BLE/WiFi	1
9	U2	L7805CV-DG	Positive voltage regulator 5V	1
10	U3	DRV8833	Dual H-Bridge Motor Driver	1
11	U4	MT3608	DC-DC 5-28V Step Up Converter	1

Circuit diagram 'Violet'

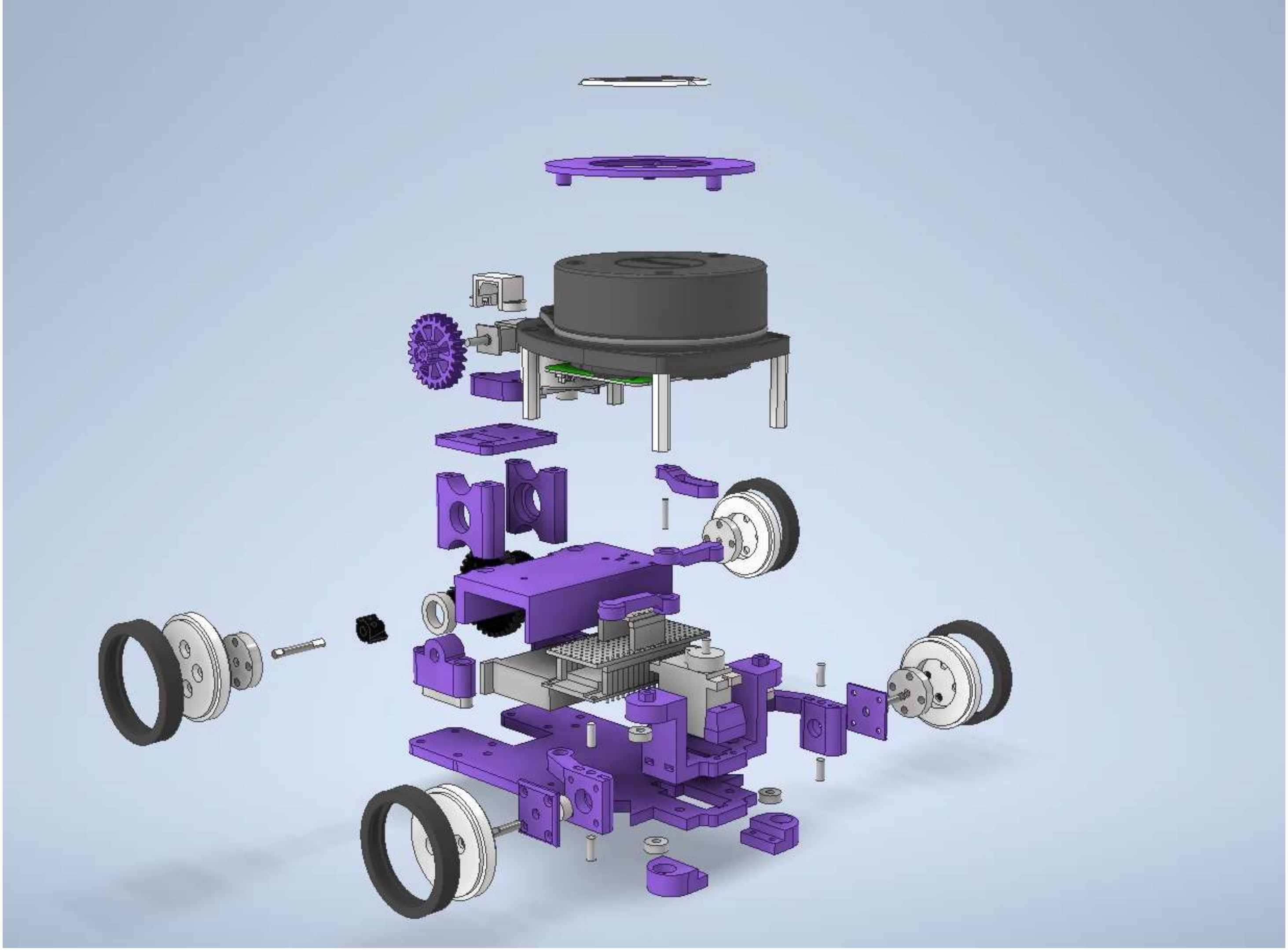
Made by: Ivanchenko Anton, Shirokovskikh Alexander

Date: 06.10.2021 | Ver: 3.0

Format: A4 | Sheet 1 of 1

*If it does not open or you need a better quality picture, check the "\electromechanical component schematic" folder, please

Assembling the robot

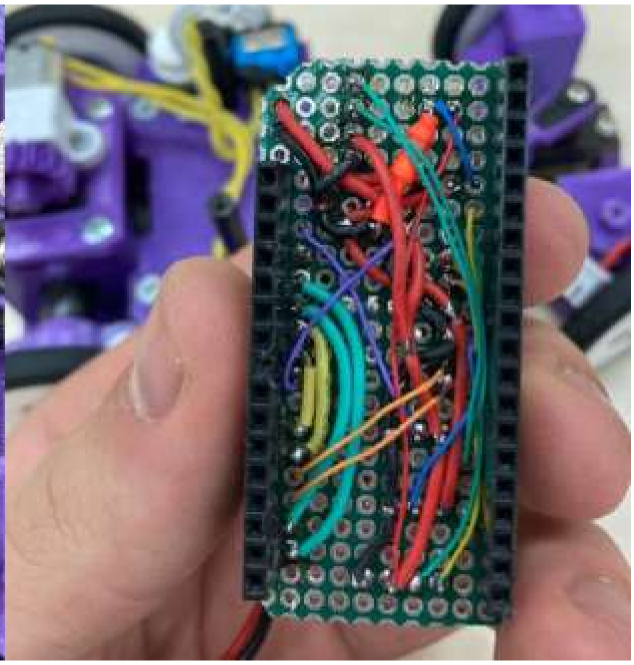


General Assembly

- First, you need to 3D print all the necessary parts of the robot. To do this, go to the "3D-models/RRO 2021 Mk.IV/Robot RRO 2021 Mk. IV/STL" folder and print on 3D-printer all the models in it.
- Second, you need to buy all the necessary components: lidar, motor, servo, motor driver and ESP Lolin32 board, as well as the components involved in the robot's circuit board. The robot also requires 4 3x8x3 bearings, 2 3x8x4 bearings, 3x5x4 brass bushings, differential and two silicon pololu tires. To power the robot you can use any 7.4V 2S Li-Po battery included in the dimensions 53x30x11.5mm, for example we use this battery. Two nylon ties and any glue will be needed to attach the button and the switch.
- Third, to assemble all the components you will need 12 M3 nuts, 12 M2 nuts, and the following screws (all lengths are from the head of the screw to the end of the thread):
  - 2xM3 7.6mm (countersunk)
  - 2xM3 20.8mm (countersunk)
  - 2xM3 16.25mm (countersunk)
  - 4xM3 16.25mm (countersunk)
  - 16xM3 6.5mm (countersunk)
  - 4xM3 33.8mm
  - 2xM2 7mm
  - 2xM2 8.8mm

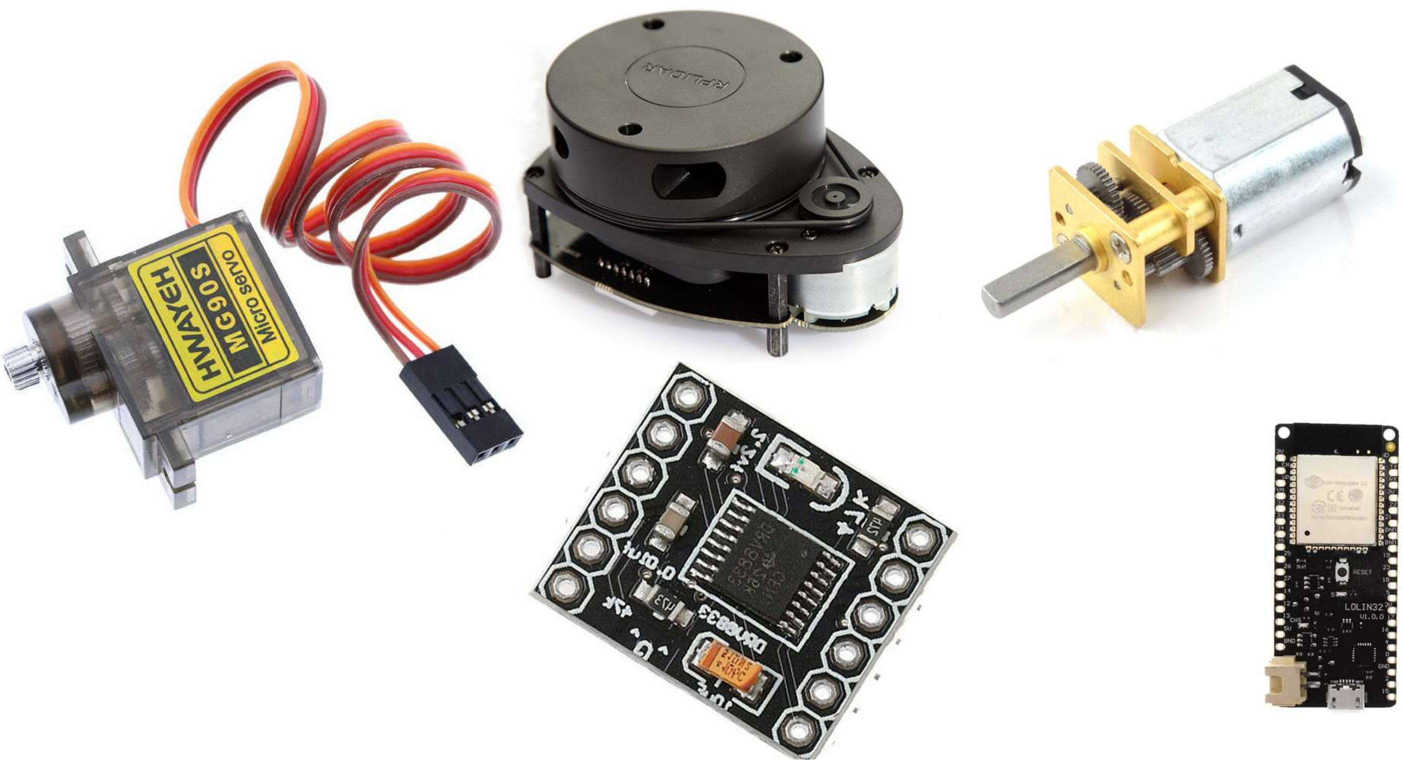
- 8xM2 6,5mm (countersunk)
- 6xM2 10mm (countersunk)

- Fourth, according to the schematic, it is necessary to have a slide board not larger than 55.8x25.4 mm. It is recommended to use male and female threaded slats for easy replacement of burned components. The control board should be assembled in the following order: ESP Lolin32, slide board and motor driver.

- Fifth, by opening the Inventor assembly file "3D-models/RRO 2021 Mk.IV/Robot RRO 2021 Mk. IV/Robot RRO 2021 Mk. IV.iam" assemble the robot as shown on the model.

About our electrical and mechanical components



Lidar

We use the RPLIDAR A1 in our robot. It is based on the principle of laser distance triangulation and uses high-speed data acquisition and processing equipment developed by Slamtec. The system measures distance data more than 8,000 times per second. The RPLIDAR A1 core rotates clockwise and performs an omnidirectional 360-degree laser scan of the surrounding space. The resulting data is converted by the MCU to build virtual walls.

ESP32

The Lolin D32 development board is based on the Espressif systems ESP32 low power system on a chip microcontroller. It has many powerful features including a dual core Arm based processor, Wi-Fi, Bluetooth, I2C, I2S, SPI, ADC, DAC, and 4MB flash. The D32 module provides these features in a convenient DIP format and can be programmed directly from a USB interface - no additional programming hardware is required. Additionally, the module also includes a charging circuit for a single cell 3.7V lithium battery, meaning that this module can easily be used in remote applications.

DRV8833

The DRV8833 driver is used to control the motors, which allows to control two collector motors simultaneously. Internally the driver chip contains two independent H-bridges designed for voltages from 2.7 to 10.8 V, with operating current of each channel up to 0.5 A without heat sink, or up to 1.5 A with heat sink mounted on the chip.

Polulu metal gearmotor

This gearmotor is a miniature medium-power, 6 V brushed DC motor with a metal gearbox. It has a cross section of 10 × 12 mm, and the D-shaped gearbox output shaft is 9 mm long and 3 mm in diameter. These tiny brushed DC gearmotors are available in a wide range of gear ratios—from 5:1 up to 1000:1—and with five different motors: high-power 6 V and 12 V motors with long-life carbon brushes (HPCB), and high-power (HP), medium power (MP), and low power (LP) 6 V motors with shorter-life precious metal brushes. The 6 V and 12 V HPCB motors offer the same performance at their respective nominal voltages, just with the 12 V motor drawing half the current of the 6 V motor. The 6 V HPCB and 6 V HP motors are identical except for their brushes, which only affect the lifetime of the motor.

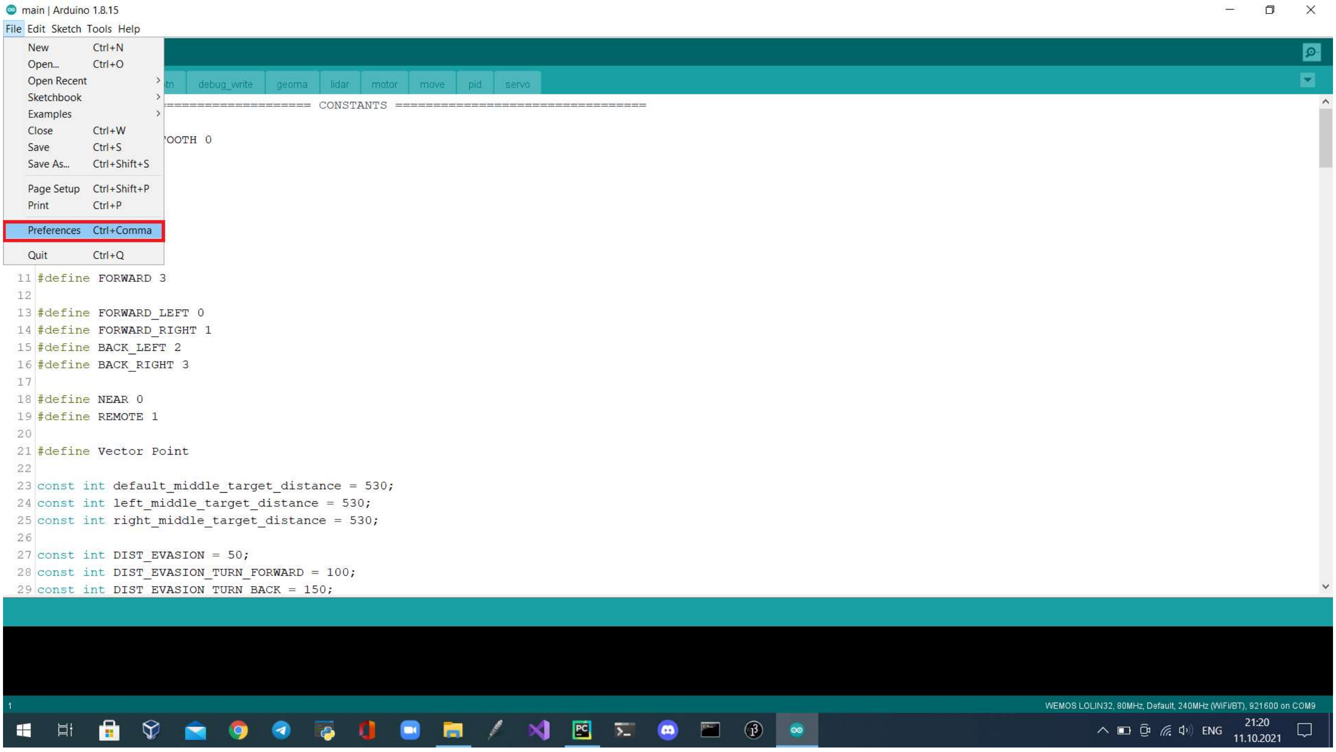
Servo MG90S

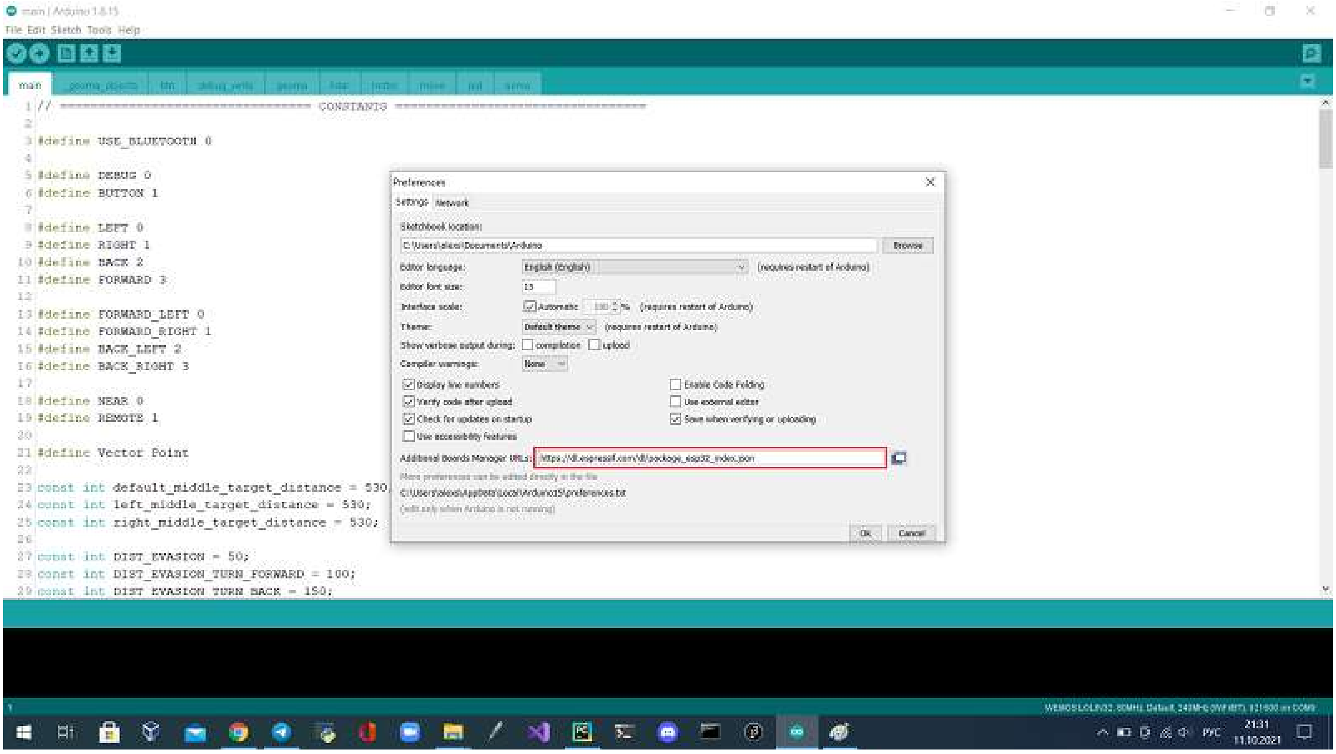
The MG90S is a small micro-format servo. So why not use its blue counterparts MG90? The answer is simple - metal gears. Thanks to them, the servo can allow you to exert a lot of force to turn the wheels without any consequences. With the servo drive, the robot can set the angle of the wheels quite precisely.

Installing the necessary programs and flashing the board

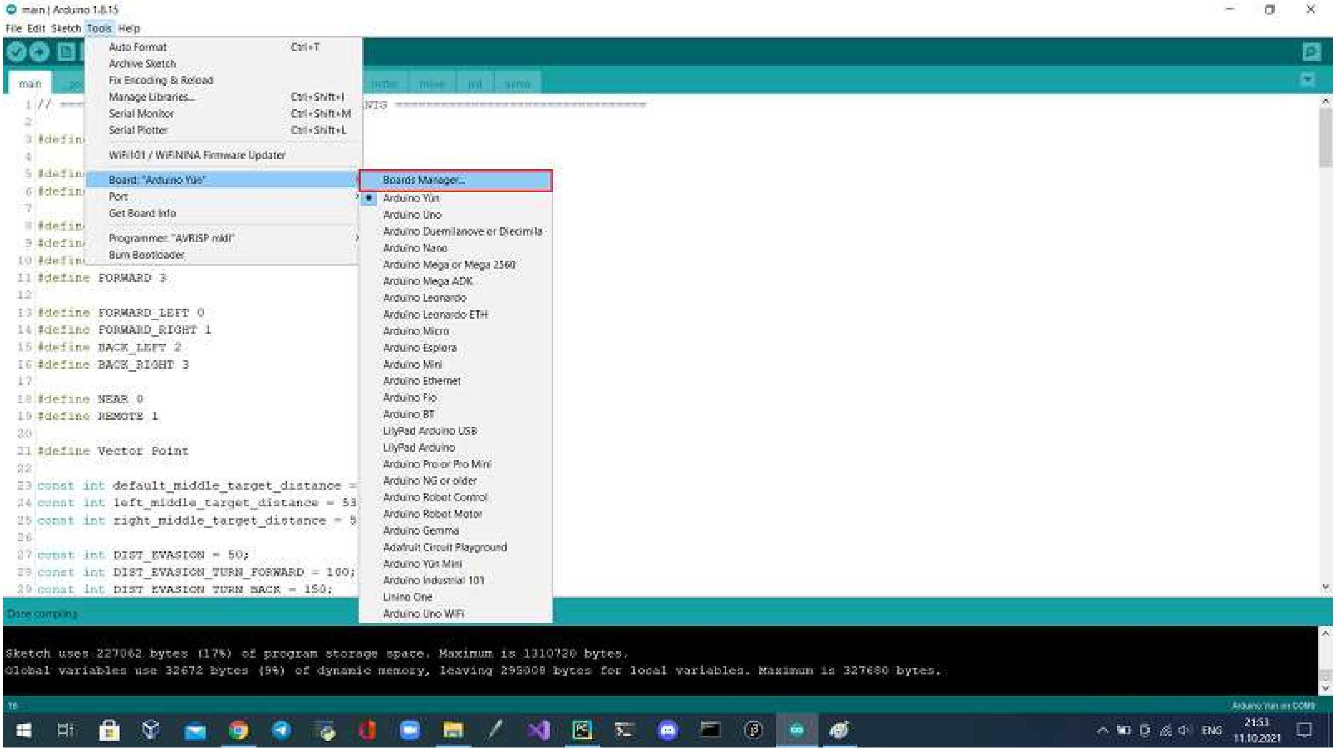
- To program the robot you need an Arduino IDE (https://www.arduino.cc/en/software). This open source software allows you to easily write code and upload it to the board. Open exe-file and follow the installer instructions.

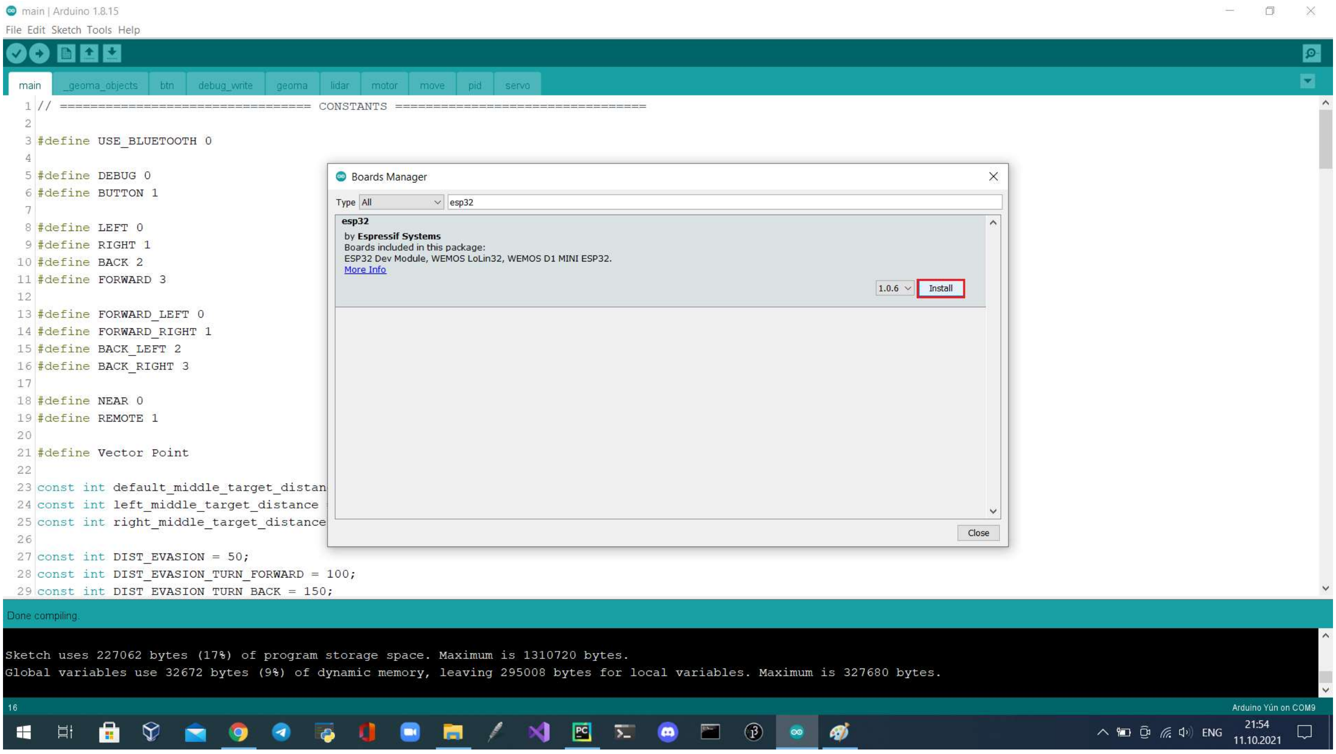
- Install ESP32 in the Arduino IDE board manager by adding the link (https://dl.espressif.com/dl/package_esp32_index.json) to the IDE settings.
  - First, add additional board manager links



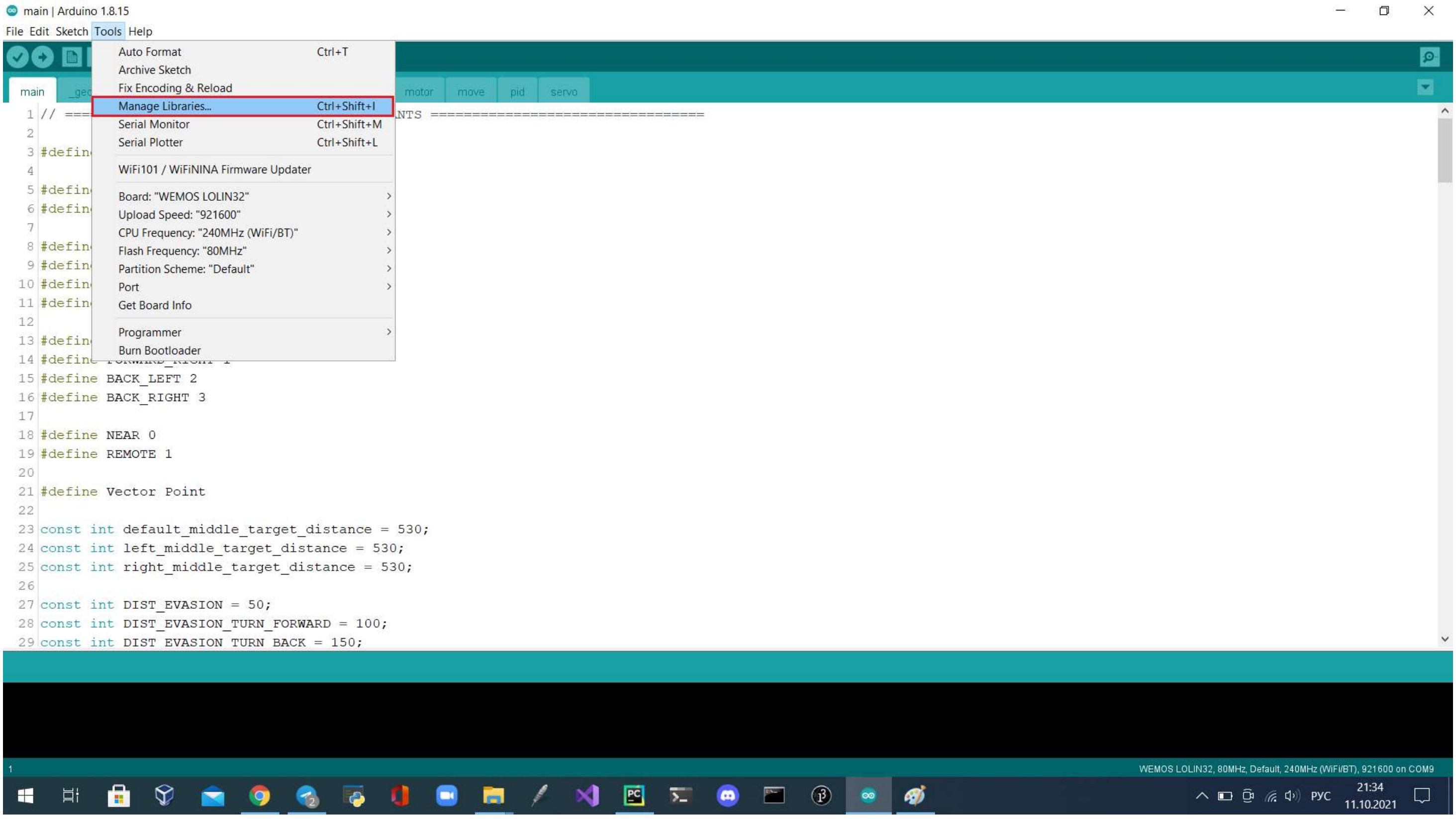


- Second, set the board model

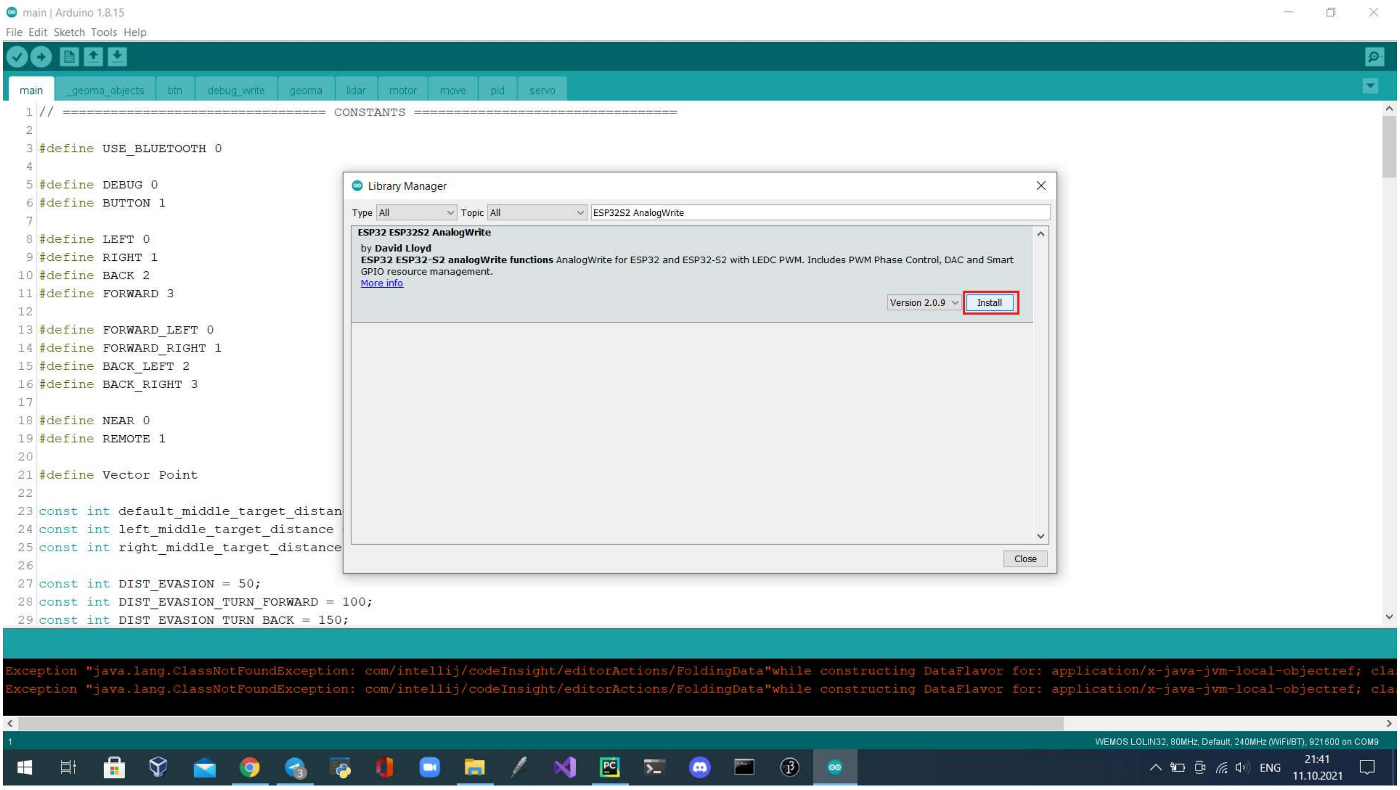




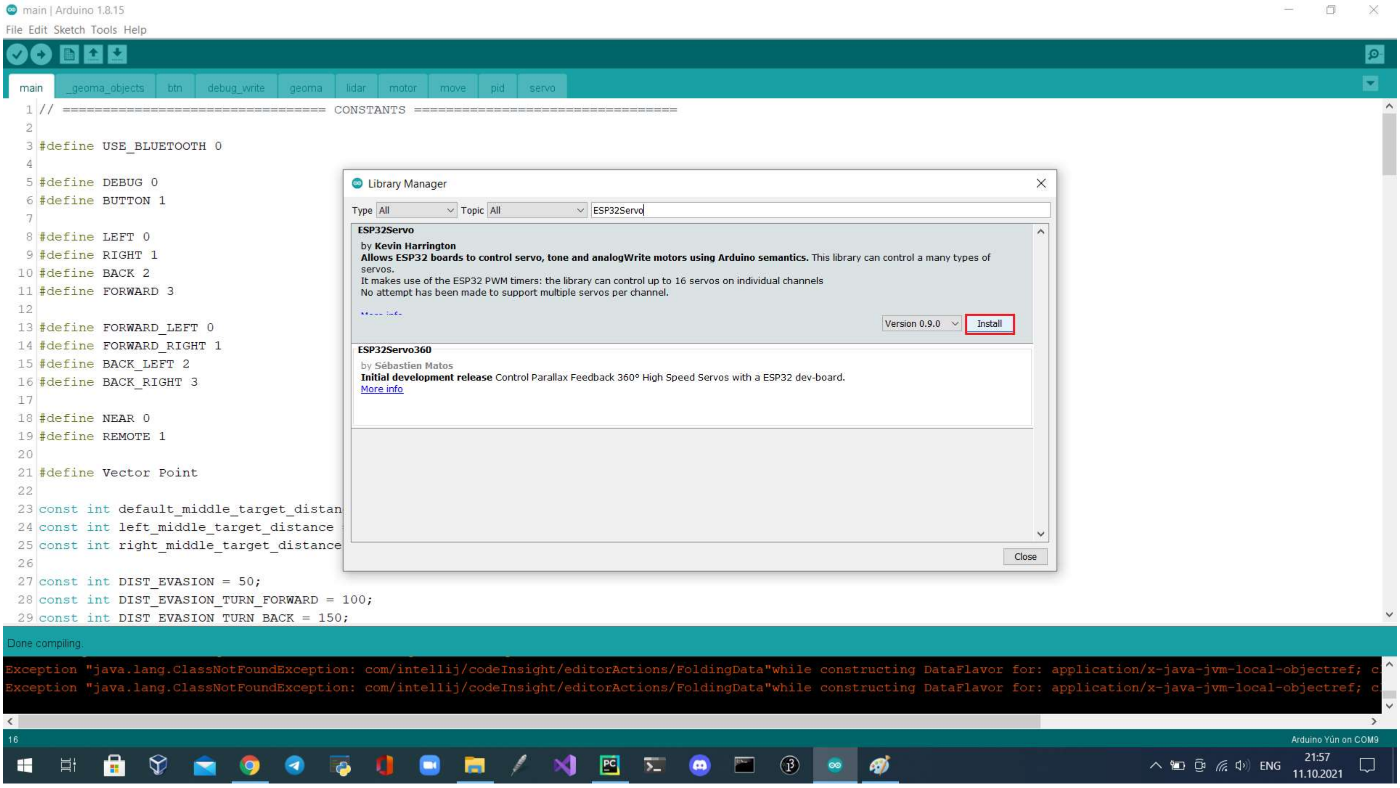
- Use the library manager to install the ESP32 libraries .



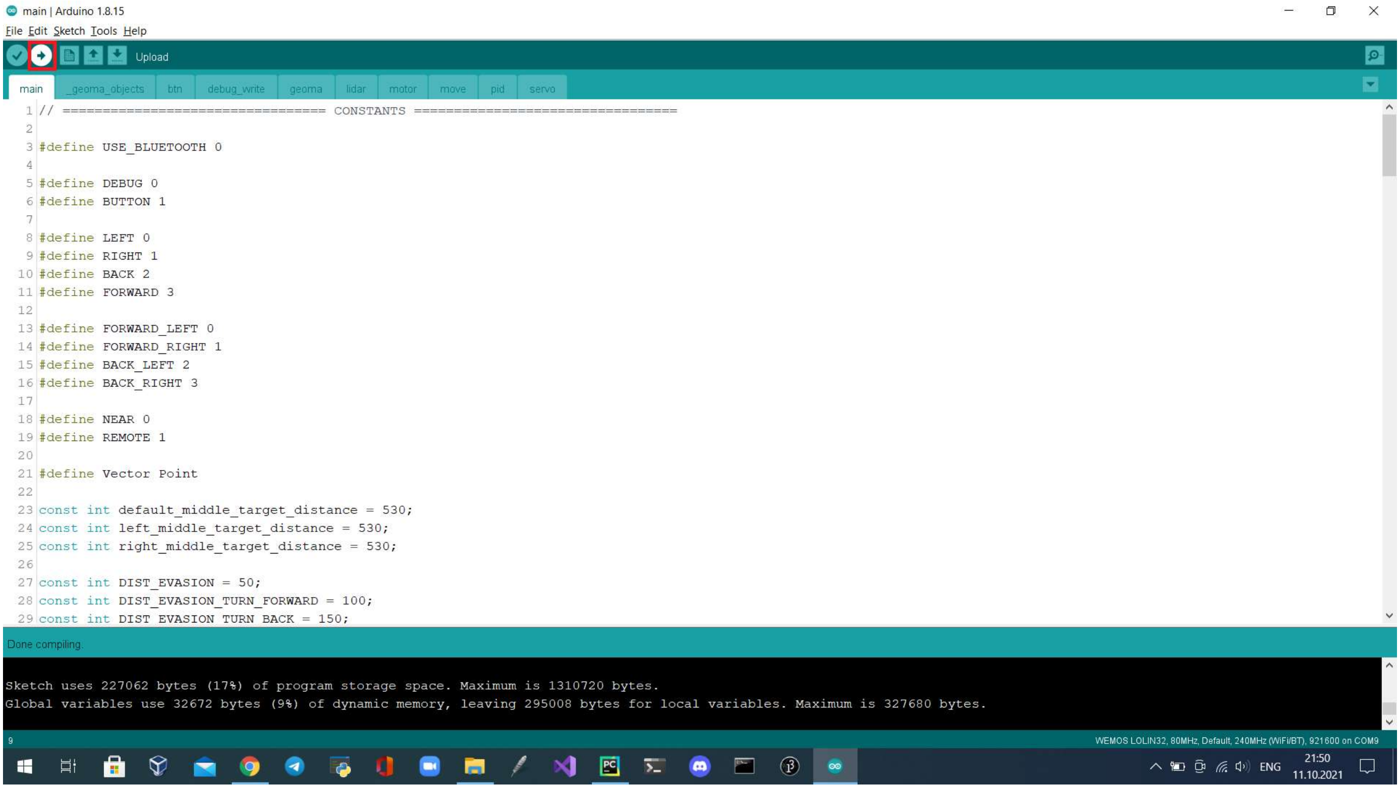
- ESP32S2 AnalogWrite



- And ESP32Servo library.



- Select the board "WEMOS LOLIN32". Connect the ESP32 board with the microUSB cable to the computer and select the corresponding port in the arduino IDE. Click the "Upload" button.



Starting the robot

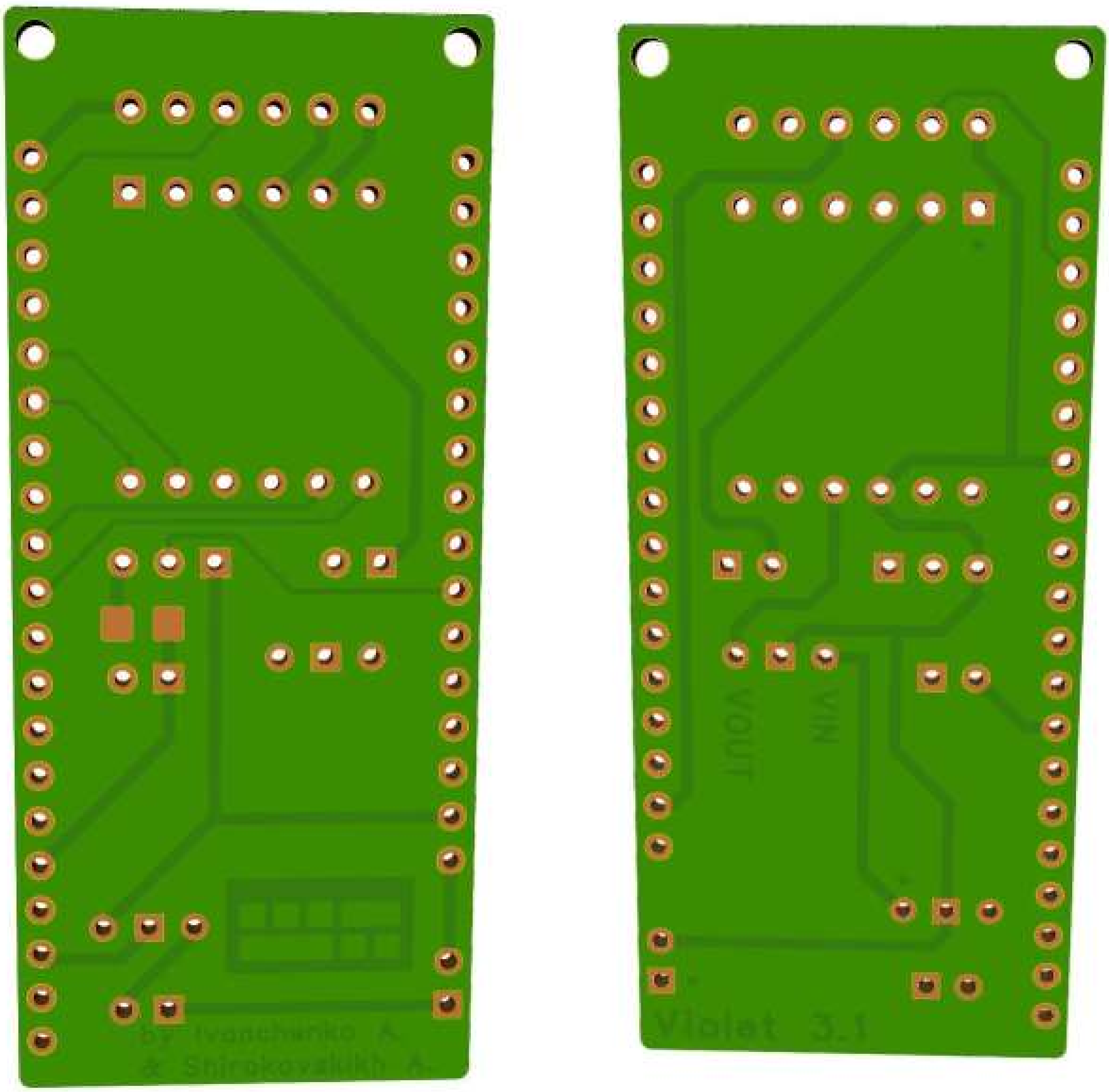
- Turn on the robot with the toggle switch
- Set up the card as specified in the rules
- Put the robot on the track
- Press the button to start

Our future plans

As always, there isn't enough time or money for everything. And ideas come up often. That's why we created this section with our plans. With the servo drive, the robot can set the angle of the wheels quite precisely

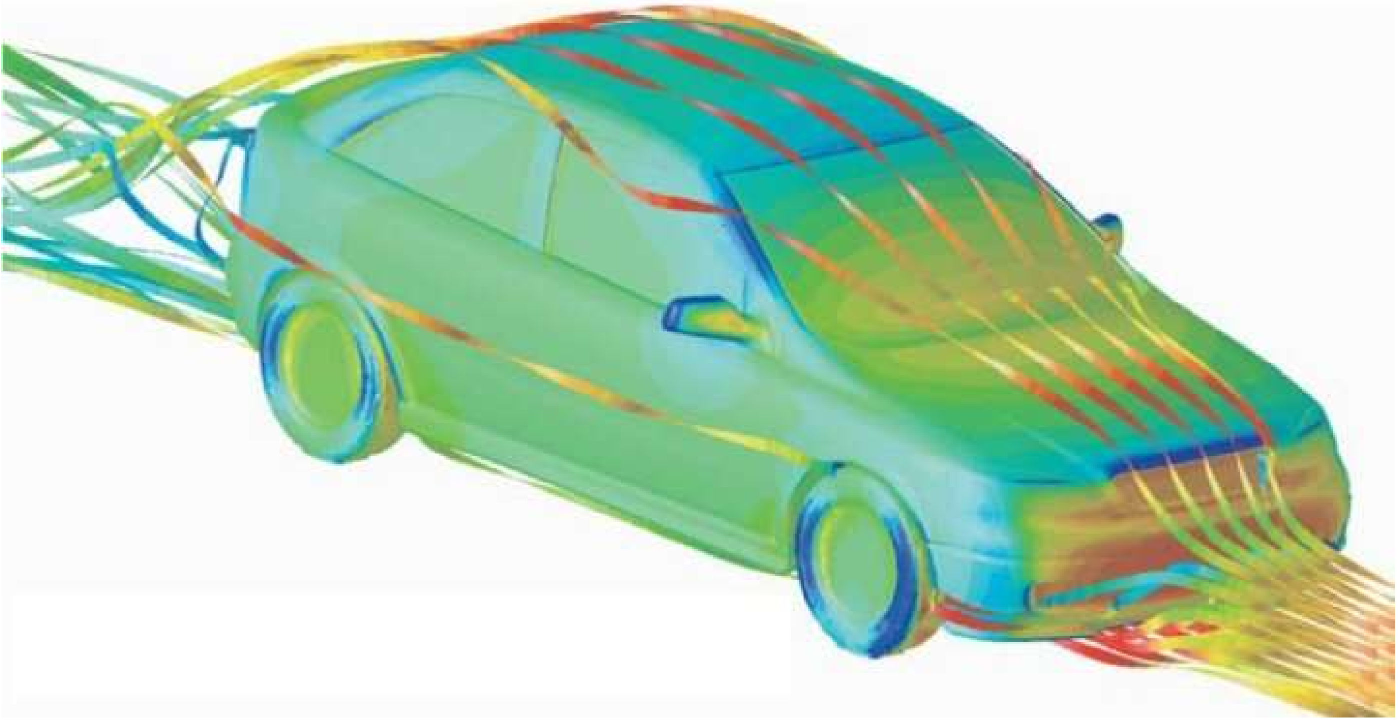
- ### Change the board created on the breadboard to the board created at the factory.

  For this idea, we slightly changed the circuit diagram of the robot, and also created the layout of the board itself. All that remains is to wait for the finished board from the factory and test it.



- ### Create a nose fairing and work on the aerodynamics of the robot

  This idea came when we needed to speed up the robot. In the end, we didn't take it as a very complicated option and left it for the future, but the streamlined shape of the robot gives pluses to stability, which is a priority.



- ## Make the robot even smaller

From the regional stage to the All-Russian stage, the size of the robot has decreased by almost three times. But we are sure that this is not the limit and it is possible to reduce the design. The small size of the robot allows us not to make unnecessary maneuvers on the track, which speeds up the passage time.