WRO FUTURE ENGINEERS-2023

JAWATHA ROBOT CHAMPIONS TEAM (J.R.S)

ENGINEERING NOTEBOOK

GITHUB REPOSITORY

This report offers data and knowledge on the ongoing advancement, growth, and development of our self-driving robot car, which we designed and programmed as a competitor in the World Robot Olympiad's (WRO) Future Engineers division.

Contents:

1-Team Info:

We are :

1- Ali albander

2- Abdullah Alruwaished

3- Yusuf Almuhanna

From Jawatha Private Schools.

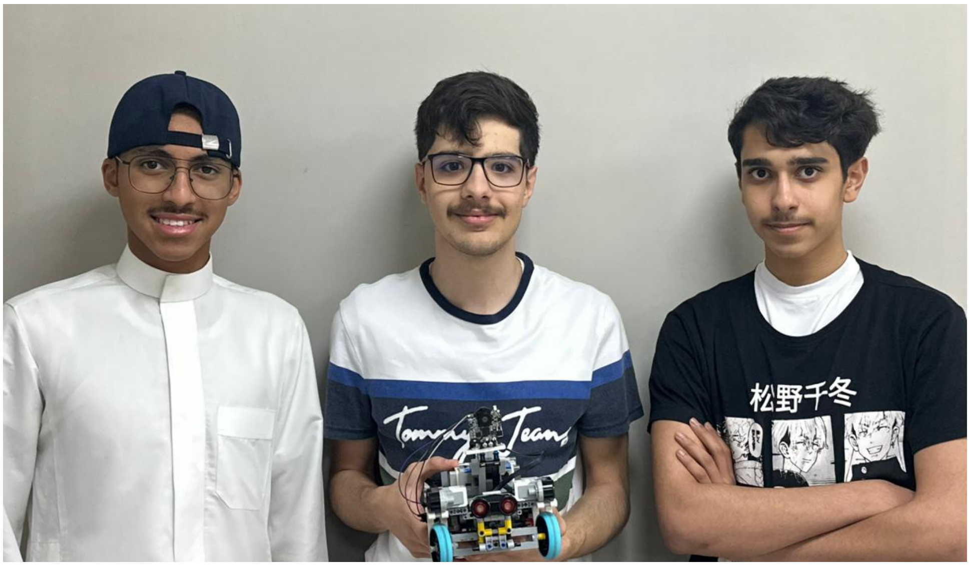

Figure .0 Team members photo

2-Robot Design:

Building an autonomous car for this challenge involves a careful process of reimagining and redesigning various components to achieve precision, safety, and efficiency. In this project, we use the Lego Mindstorm EV3 kit, which gives us powerful compatibility between motors, sensors, and EV3 Intelligent Brick.

to build our robot we used Lego digital designer(LDD) software to put the design ideas, after that we used Lego technic parts to build the robot.

Our robot dimensions is : (260 x 160 x 190) mm.

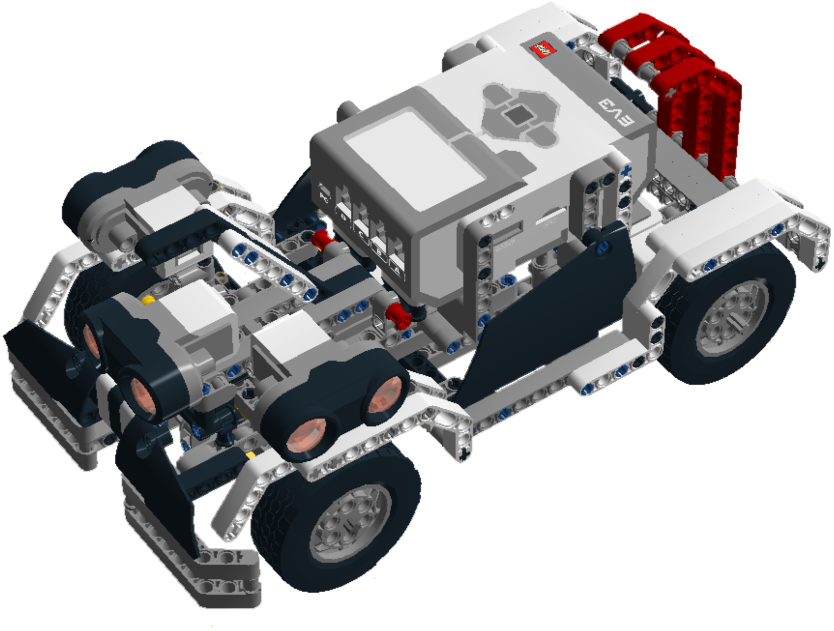

Figure .1 Robot design

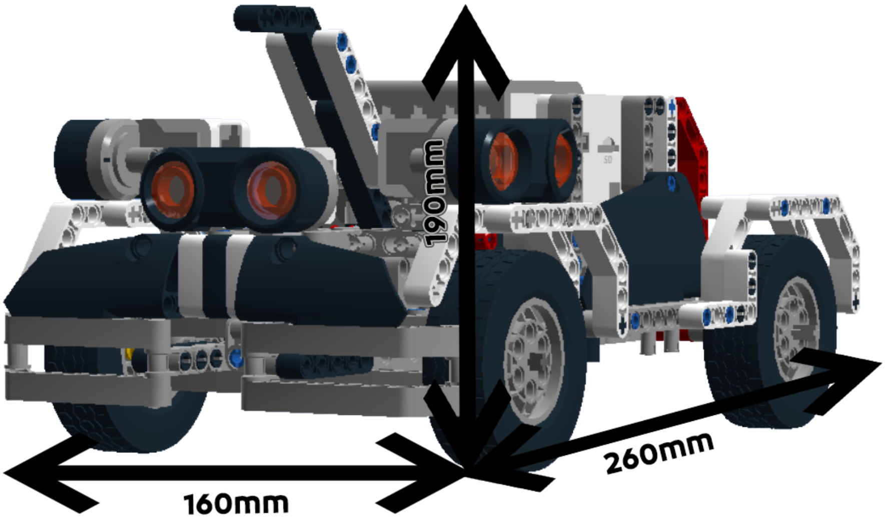

Figure .2 Robot dimensions

In our robot, we mainly used 2 systems:

A- Driving system.

B- Steering system.

A- Driving system:

For the driving system, we use one medium motor connected with custom gearbox to move the movement from one point to another, without changing the set speed or torque.

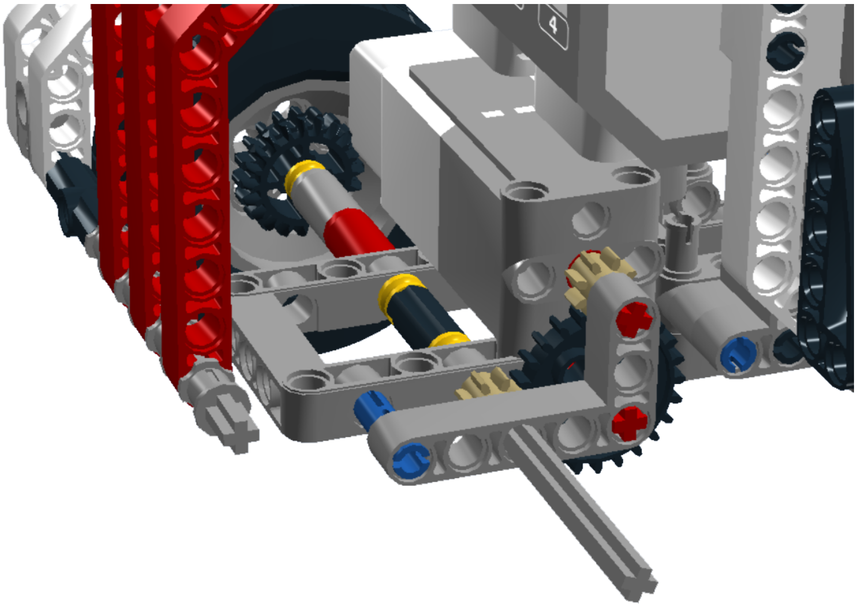

Figure .3 The driving system

B- Steering system.

For the steering system, we use one medium motor connected with a gearbox that uses a bevel gear with rack gear to Convert the circular motion into straight motion.

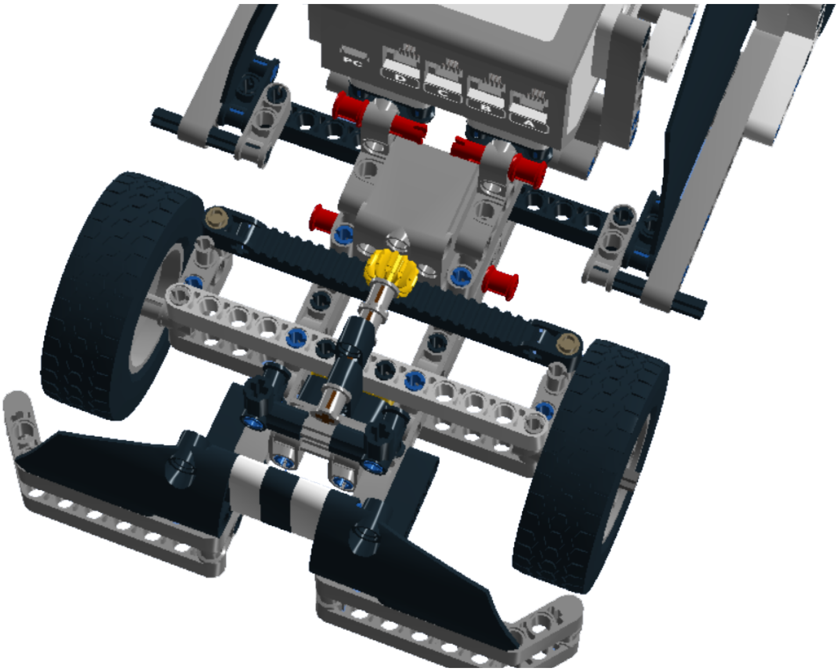

Figure .4 The steering system

3-Electronics and Circuit:

The robot contains the following parts:

1x EV3 intelligent brick with battery

2x medium motors

3x ultrasonic sensor

1x gyro sensor

1x pixy2 camera

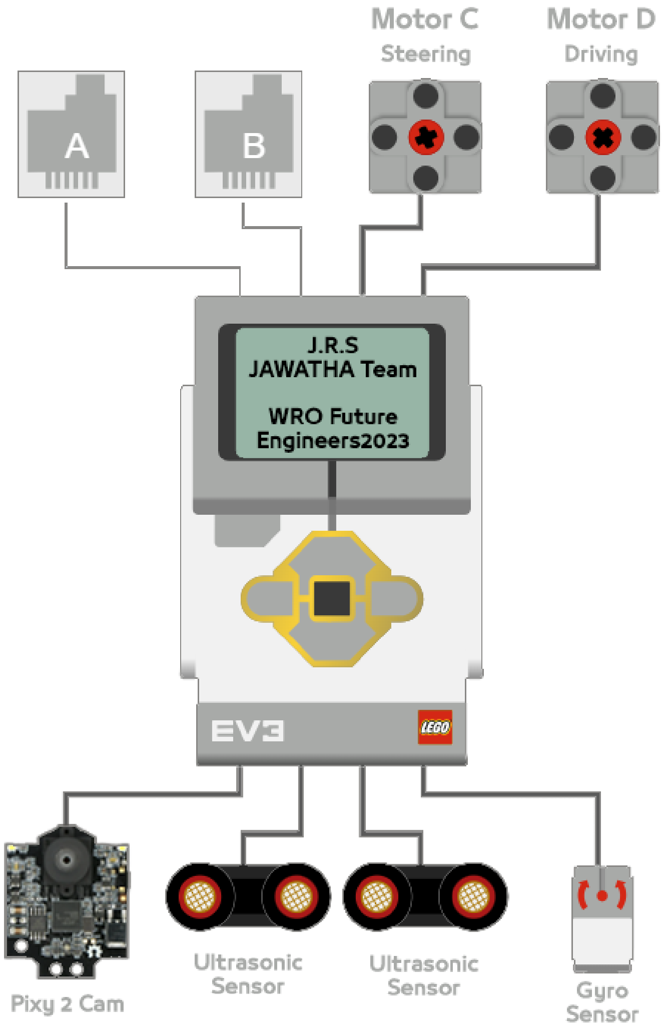

Figure .5 Electronics diagram

We connect the sensors and motors using flat black Connector Cables, linked sensors to the EV3 Brick using Input ports 1, 2, 3, and 4, and motors using output ports A, B, C, and D.

to connect the pixy2 camera to EV3 intelligent brick, we crafted ourselves LEGO cable using a motor\sensor cable and jumper wires, after that we connected the wire to the input port in EV3 intelligent brick, and from the pixy2 cam with I2C pins.

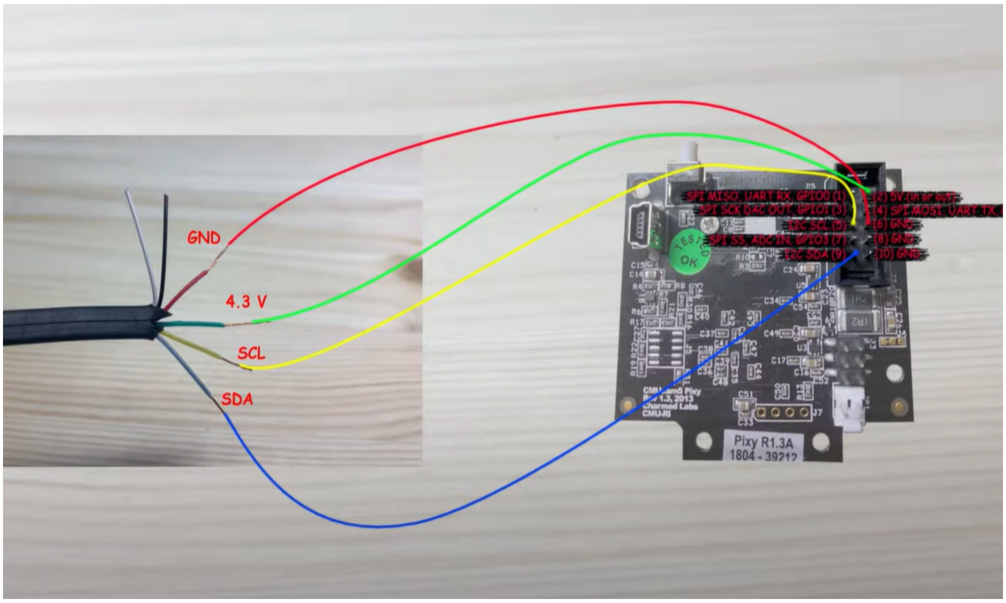

Figure .6 connection between ev3 wire and pixy2 camera

4-Programming:

Our strategy is to make the car able to define a path for it to follow depending on the color blocks it detects. We divide the playfield into eight sections, and the car stores whether it has to travel to the right or to the left in each section.

To program a robot, we use Lego Mindstorm EV3 software. The LEGO Mindstorms EV3 software provides a user-friendly programming environment for creating and controlling robots. It uses a visual programming language that is based on a drag-and-drop interface. with some key aspects of the programming languages, like Blocks-Based Programming, Sensor and Motor Integration, Variables and Data Handling, and capabilities for importing external and open-source blocks.

mainly we have two programs:

A-Qualification rounds program.

B-Final rounds program.

A-Qualification rounds program:

In the qualification rounds, we programmed the robot to be able to measure the distance of the robot from the sides walls and from the front wall, we also used a gyro sensor to measure the angle when rotating accurately and finally read the motor motion sensor to measure the distances traveled by the robot.

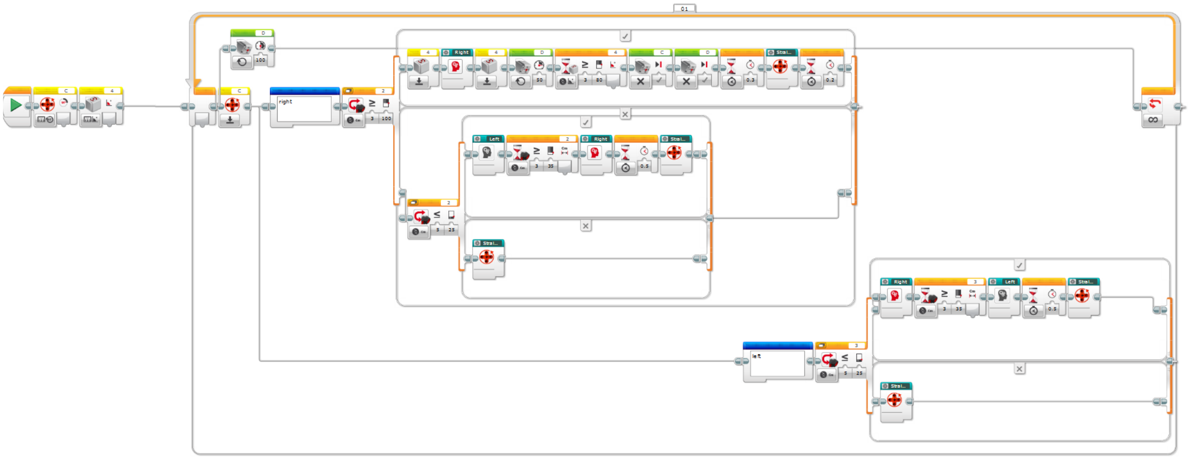

Figure .7 Qualification round code example

B-Final rounds program:

In the final rounds, in addition to the code for the qualification rounds, we used a camera to be able to recognize the colors of the obstacles in the road.

Our robot could revolve around the obstacles with the ability to recognize colors so that it changed the direction of its movement based on the color that was read.

The robot was able to do this in the beginning. By defining the colors that we want to read to the camera and giving each of them a unique signature, then the robot rotates using the distance sensor until it reaches the equipment. The robot reads the color and stores it in a variable so that our program can continue the journey based on reading the new variable.

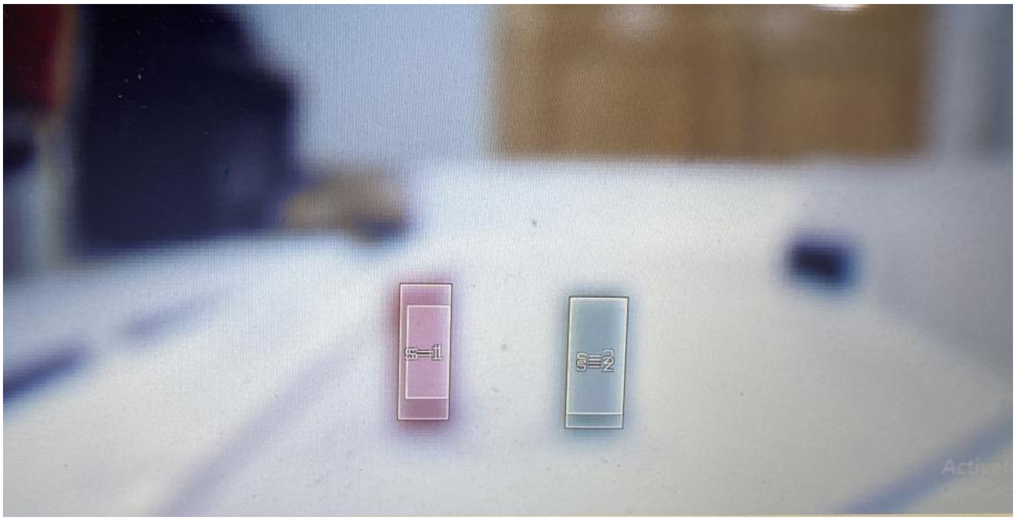

Figure .8 Example for giving a signature for the colors

😀 We will be delighted if you can check out all of our work by scanning the QR code at the top right of the first page and visiting our GitHub repository 😀