EE 494 ENGINEERING DESIGN II

2018 – 2019 Spring Semester

FINAL REPORT

DEVICES TRYING TO SCORE IN EACH OTHER'S GOALS

DESIGN STUDIO SECTION	7
DESIGN STUDIO COORDINATOR	Mustafa Mert ANKARALI mertan@metu.edu.tr

SHAREHOLDERS	E-mail	Phone
Ali Birkan DÖNMEZ -2030484	a.birkandonmez@gmail.com	+90 531 705 33 89
Ayça YILDIRIM - 2094670	aycayldrmm@gmail.com	+90 545 854 35 27
Gökhan SOYLUKAN - 2031342	gokhansoylukan95@gmail.com	+90 505 610 11 68
Muttalip Caner TOL - 2031466	m.canertol@gmail.com	+90 507 334 38 95
Yunus DEMİRÖREN - 2030435	yunus.demiroren@metu.edu.tr	+90 537 788 39 95

DATE OF SUBMISSION	10.05.2019
STARTING DATE OF THE PROJECT	10.11.2018
THE PROPOSED PROJECT DURATION	175 Days
COMPLETION DATE OF THE PROJECT	17.05.2019
THE COST OF THE PROJECT	200 $

Table of Contents

EXECUTIVE SUMMARY 2

INTRODUCTION 3

BASIC COMPANY DESCRIPTION 4

Academic Background 4

Areas of Specialization 5

Roles in the Company 5

DESIGN REQUIREMENTS 6

Problem Statement 6

Constraints of the Project 6

Project Standards 7

Robot specifications 7

Wall specifications 7

Ball specifications 7

The goal and the playfield specifications 7

Game rules 8

OVERALL SYSTEM DESCRIPTION 9

3D drawings of the Robot 9

Tasks of the Project 11

Subsystems of the Project 12

Functional Diagram of the System 13

SUBSYSTEM ANALYSIS 13

Dribbling and Kicker Module 13

Driving Module 15

Vision Controller Module 19

Communication Module 20

Command Module 24

SUBSYSTEM COMPATIBILITY 26

TESTS 27

Experiment 1: Optimizing the resolution of the video stream 28

Experiment 2: Accuracy of ultrasonic sensor HC-SR04 for distance measurement 30

Experiment 3: Measuring the visual angle of the Raspberry Pi cam 33

Experiment 4: Measuring the stream delay 34

Experiment 5: Measuring the kicking range 34

Experiment 6: Directionality and Speed Measurement of Robot Mobility 35

FINANCIAL UPDATES 35

POWER AND WEIGHT ANALYSIS 36

ORGANIZATIONAL PLAN 38

SAFETY CONCERNS AND PRECAUTIONS 39

FEASIBILITY AND ROBUSTNESS ... 40
Economic Feasibility ... 40
Technical Feasibility ... 40
Operational Feasibility ... 41
Schedule Feasibility ... 41
POSSIBLE APPLICATION AND ENVIRONMENTAL EFFECTS ... 41
DELIVERABLES ... 42
CONCLUSION ... 43
APPENDIX ... 45
Appendix A – Robot Images ... 45
Appendix B – User Manual ... 46
REFERENCES ... 48

EXECUTIVE SUMMARY

Teleoperation or remote operation is a highly used concept which has lots of applications in many critical fields. Most of the projects were accomplished using tele robotics and these were the ones that took the humanity further in space, medical, military, mining, deep-sea exploration or search-rescue operations. With such a motivation, Poro Inc. was founded to accomplish a project which includes building a tele operated robot. This report is a detailed explanation of our proposal on this project.

The aim of the project is providing and implementing a solution for creating devices trying to score in each other's goals. There will be a hexagon-shaped playfield, two goals in the opposite corners, one ball and two robots on each half-field. Before the start of the game, two robots will stand on their own half-fields on the lines of their own goals and the ball will be on the center. Upon command, game starts, and the robots try to score in opponent's goal without keeping the ball in their own half-field for more than 20 seconds. Both of the robots will be able to be controlled by a human operator who stands at least 30 meters far from the playfield.

Although this will be the first project of Poro Inc., all our team members took part on many different projects separately. Therefore, each member has a background and experience

on various fields. The fact that our team was founded by the specialists from different areas, guarantees that our company, Poro Inc. is the best fit for conducting this project.

The system that we have designed has four main modules. Each module can be considered as a black box and they take inputs from previous modules and give output to the next modules. First, we have a Vision Control Module which includes a camera and several distance sensors. Our module takes video from the camera and distance information from the sensors. After processing the data, this module sends its outputs to the second module, Communication Module. This module basically connects the local and remote computer via the Internet. The robot is connected to the Internet using GSM protocol. The data is transferred through the Internet and comes to the remote computer. On the remote computer we have our third module, Command module and a Graphical User Interface. The GUI shows the data it takes from the Communication Module and shows to the human operator. Then, it takes commands from a human operator about the motion of the robot and sends these commands back into the Communication Module again. Our last module, Motor Control Module takes these commands and controls the speed of the wheels, activates the dribbling or shooting mechanisms. These process goes on in a loop until the game ends.

Poro Inc. does not only provide a teleoperated robot in the package but also offers you two years of warranty service. Also, we are providing a game ball, playfield, rechargeable battery, charger, the GUI software and a user manual for a better experience of our customers. All these package costs only for $198,4. The project duration is estimated as 175 days.

INTRODUCTION

Our company, Poro Inc., is composed of five candidate electrical and electronics engineering students who are interested in robotics. In this project, we aim to create a teleoperated robot which will try to score in opposing team's goal.

This project is composed of mainly two parts which are remote controlling unit and a tele operated robot controlled by an operator. We need to make our robot try to shoot and score in opposing team's goal by controlling the robot remotely and monitoring the play-field from the screen that is placed far away from the play-field. Main problem of this project is the communication between these two parts. The robot should be able to react to the commands of operators in minimum time interval. In other words, communication between these two units should occur in an acceptable time interval. We need to develop a communication module with minimum delay. Other than this problem, transmitting live video from the camera that is placed on robot to the operator's screen also is a very important issue for a tele operating robot. We need to spend time on this issue to be able to see clearly what is going on in the play field.

After we have completed this project, we will have obtained valuable experiences about the teleoperation. Tele operated systems are very important for a society because they are used in many areas such as space and military area for exploring and searching purposes, medical area for endoscopic surgeries and tele surgeries, mining area for unsafe areas etc.

This report consists of team organization section which explains the academic backgrounds, areas of specialization and roles of each company member, design requirements section which explains the requirements and objectives of the project, overall system description, sub-system analysis which gives detailed information about the sub-systems, test results, financial updates, weight and power analysis, feasibility and robustness section, and expected deliverables section which shows the services which will be provided with the project, conclusion section which briefly remarks the key points, references and appendices sections.

BASIC COMPANY DESCRIPTION

Academic Background

Members of Poro Inc. are all candidate electrical and electronics engineer but each member has different experiences in various areas. Therefore, each member is responsible from different subjects while working on the project.

- Ali Birkan Dönmez is studying in the electronics area and interested in programming, simulating optoelectronic and semiconductor devices.
- Ayça Yıldırım is studying in the electronics area and is interested in embedded systems and programming.
- Gökhan Soylukan is studying in the electronics area and is interested in software development, mobile application development, and mechanical designs.
- Muttalip Caner Tol is studying in the computer area and his research interests are deep learning, computer vision and cybersecurity. Therefore, during the project, he will be responsible for conducting the image processing implementations.

- Yunus Demirören is studying in the control area and he will be responsible for controllers, automation and robustness with respect to environmental changes of our robot.

Areas of Specialization

Areas of specialization of each member of the company have been determined by considering their academic backgrounds which is stated in the previous section.

Table 1. Qualification Areas of Group Members

	Signal Processing	Embedded Systems	Programming	Control	Motor Drive	CAD	Power Calculations
AyçaYıldırım	X	X	X				
BirkanDönmez		X		X	X		
GökhanSoylukan					X	X	X
M. Caner Tol		X	X	X			
YunusDemirören	X	X		X	X		

Roles in the Company

The organizational structure of Poro Inc. is shown in Figure 1.

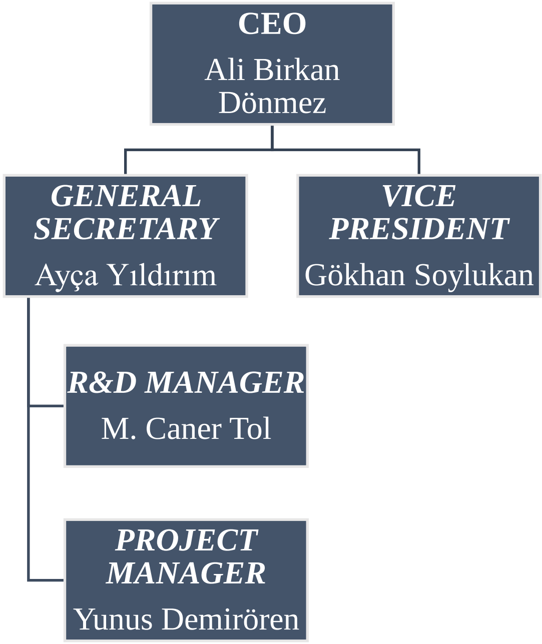


Figure 1: Organizational Structure of the Company

DESIGN REQUIREMENTS

Problem Statement

The current project of PORO Incorporation is designing and constructing a tele operated robot which tries to shoot and score in opponent's goal. One of the members will control the teleoperated robot as operator. The operator will control the robot remotely from a distance up to at least 30 meters without actually monitoring the play-field with naked eye. The playfield is hexagonal structure which is constructed by six 75 cm long wall. The playfield will be monitored by a camera. The width of the goal will be twice of the longest dimension of the robot. The robot must have mobility in order to take action to the ball, to defend its own goal, to drive the ball. A shooting mechanism must be designed on robot to score in opponent's goal. The communication module must be constructed to allow two-way communication between operator and robot. The communication module must transmit the camera view to the operator and must transmit the commands of operator. The commands will include the movement of robot, the shooting action, controlling the ball and defending actions. The robot is free to move in its own half-field only. The sign for the game start will be given externally.

Constraints of the Project

- The robot must fit in a circle with 64 cm diameter.
- The ball must be transferred to opponent's side in 20 seconds.

- The robot cannot grasp, scoop or carry the ball.
- The robot cannot cross the opponent's side.
- The robot cannot touch the ball before starting command.
- The operator cannot monitor the play-field with naked eye.
- Wi-Fi protocol or wire cannot be used for communication between operator and robot.
- The robot must be controlled from a distance up to at least 30 meters.

Project Standards

Robot specifications

a. The width and length of the robot cannot be longer than 75*sqrt(3)/2. This constraint is determined according to widest wall distance.

b. The size of the robot cannot change during the game.

c. The color of the robot must be different from lines, walls and the ball.

d. There must be an onboard camera on the robot.

e. The robot cannot fly.

f. The robot can have autonomy for not crossing the opposite field and keeping a certain distance to the side walls.

Wall specifications

a. The material of the wall should be MDF.

b. Wall should have 30 cm height, 75 cm width and 8 mm thickness.

c. Wall surface should be covered by black cardboard.

d. If black cardboard does not give a proper result, companies have a chance to cover the wall by using black paint.

Ball specifications

The selected ball based on the constraints mentioned below is Artengo (Beach Tennis Ball), it has a diameter of 36 mm and it will be painted to green color.

a. The ball should be lightweight and should not bounce a lot from the ground (after first bounce it should reach approximately 30-40% of its initial height).

b. If it has a color that is undesired, it is important that it should be repainted easily.

c. Its reflection from the wall should have almost the same angle with its incident angle.

The goal and the playfield specifications

a. Both goal and play field lines are represented by using red electrical tape.

b. The width of electrical tape should be 2 cm on the play field.

c. The width of electrical tape should be 4 cm on the wall to represent goal post.

d. Since goal post is wider, there should be 1 cm indentation from each side of goal post at the intersection points of the goal post and play field lines.

e. The floor of the game field is the floor of the KKM Building and E-Block of the Electrical and Electronics Engineering Department.

Game rules

a. The robots are placed on their own goal lines and the ball is at the center of the field at the beginning of the game.

b. Start signal of the game will be given by instructors.

c. The ball cannot be lifted intentionally by the robot during the game.

d. Throwing the ball out of the game field is foul.

e. Sending the ball to the opposite more than 20 seconds is foul.

f. Crossing to the opposite field is foul.

g. After foul, the ball is given to the opponent's side and the ball can be placed anywhere on the other side of the field. Also, the opponent can also be anywhere on its own field after a foul.

h. After scoring goal, the ball is given to conceding side and the rule for placement of ball is the same as after foul.

i. The robot scoring 2 goals more than the opponent wins the game.

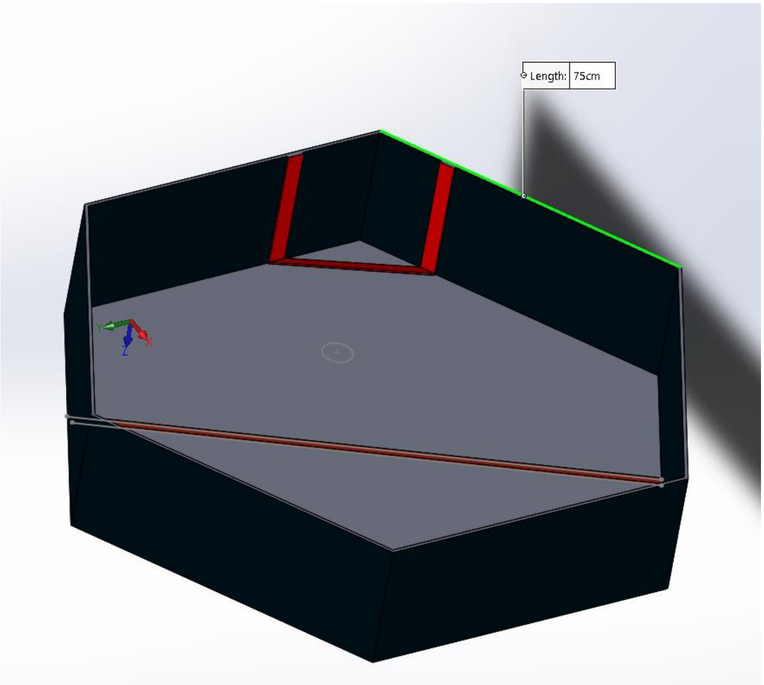


Figure 2: Game field

OVERALL SYSTEM DESCRIPTION

3D drawings of the Robot

As the shareholders of PORO Inc., we all know for sure that the mechanical design of the overall system is one of the most crucial points of a successful project. By designing a strong and durable chassis with light-weight, we can make sure that the components inside of the robot would not get harmed and the overall load on the motors could be decreased. To do so, first we designed our mechanical system with caution. A 3D representation of the final product, designed by using the SolidWorks CAD software, is provided in Figure 3 below. Each component has been drawn according to their real dimensions.

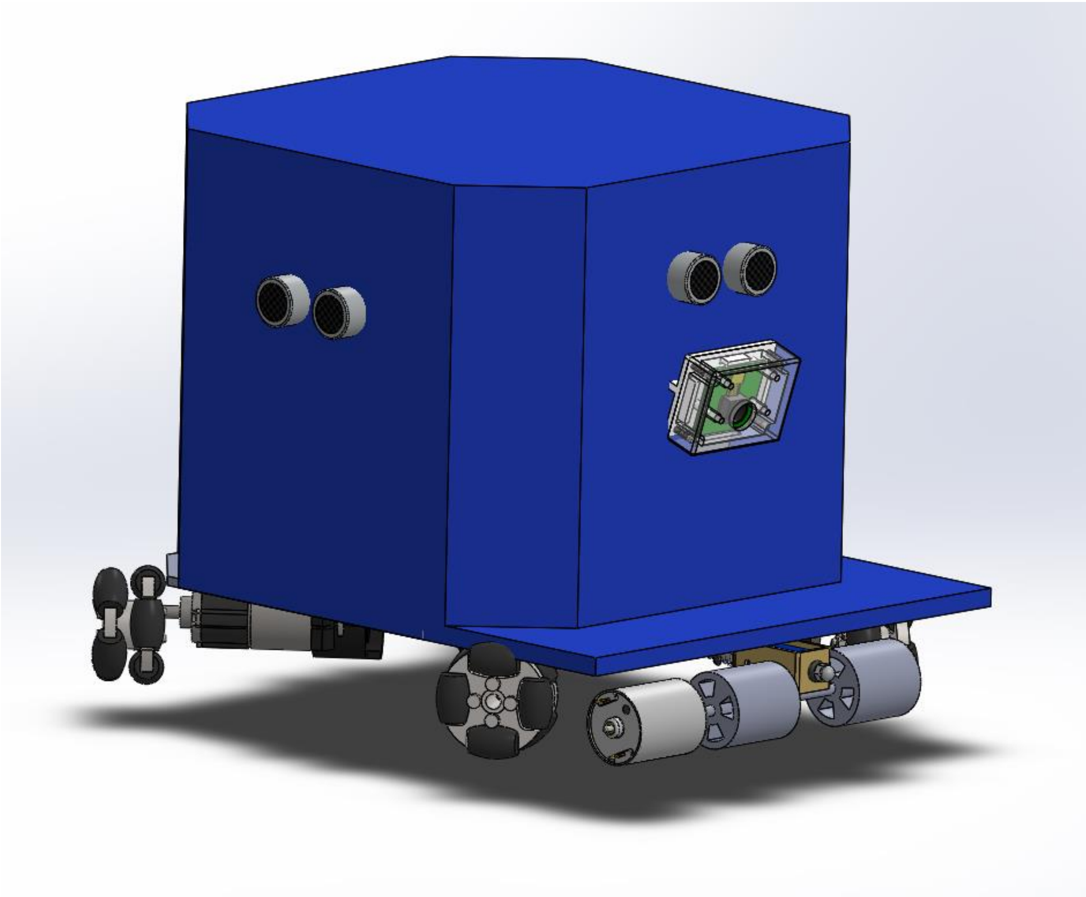


Figure 3: Isometric view of the robot

Top, front, right and isometric views of the drawing of the robot are included in Figure 4 below.

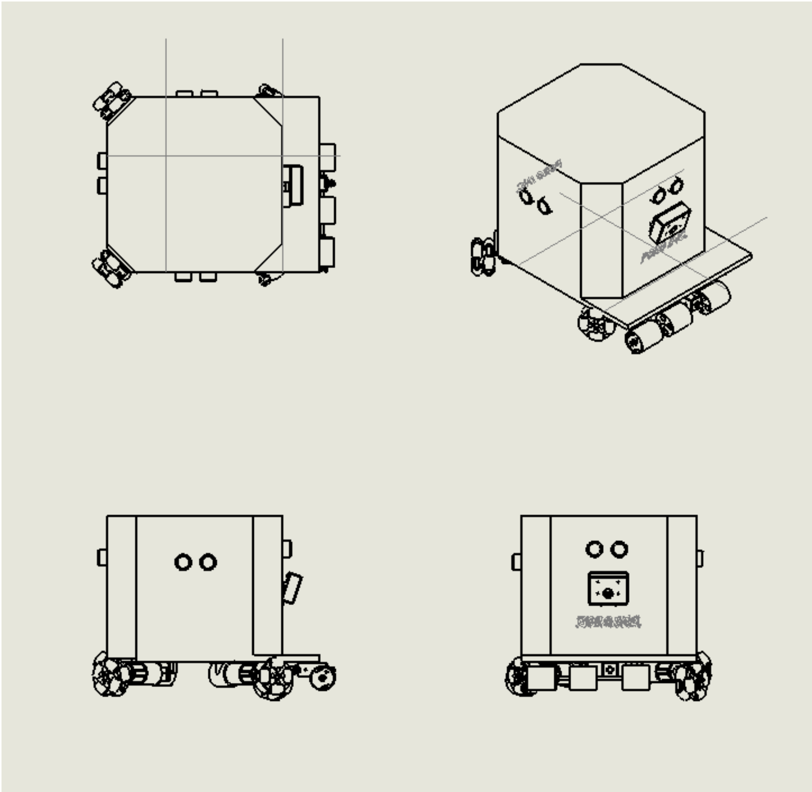


Figure 4: Technical drawing of the robot

We tried to make our robot as small as possible. Real dimensions of the game field, the ball and our robot can be seen in Figure 5.

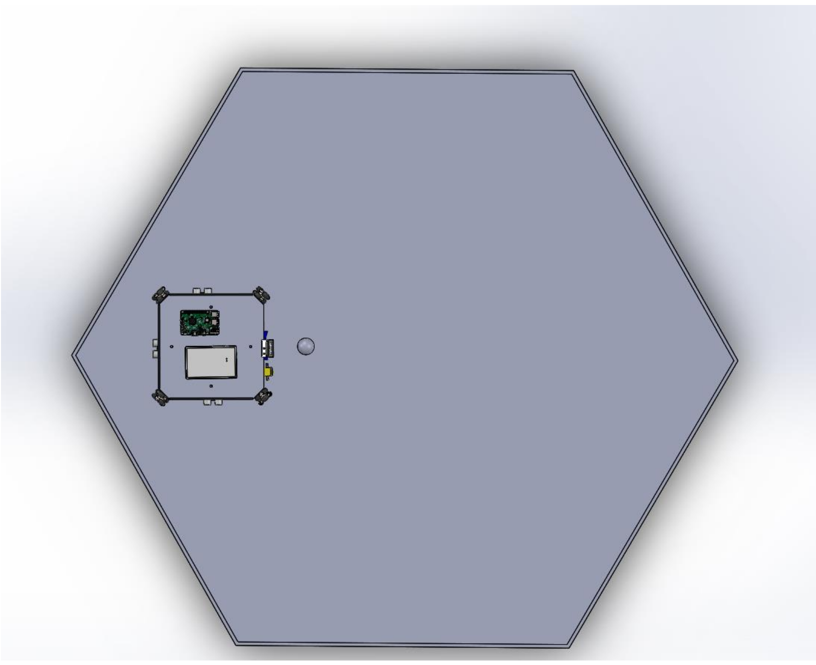


Figure 5: Our robot, ball and game field with real dimensions

Tasks of the Project

- Task 1: Video Streaming

The robot has an on-board camera. The user can't see the playfield, in order to allow user to observe game field and know the position of the robot, a video streaming on the user's laptop is provided. The communication module of the robot, which consists of 3G dongle and Raspberry Pi 3, provides the on-board camera view to the user via internet connection. The camera view is followed on the laptop by the user.

- Task 2: Command Transfer

The commands given by the user is transferred to the robot via internet connection. The user is connected to internet via Wi-Fi or Ethernet cable or 3G dongle. The robot is connected to the internet via a 3G dongle. The commands will be given via a joystick, the joystick command is sent to the Raspberry Pi and terminal of the robot sends the command to the Arduino. The onboard motors are connected to the Arduino and the corresponding action is taken according to command.

- Task 3: Moving Body

The robot is standing on 4 omni-directional wheels, each wheel is driven via encoder DC brushed motors. The motors are controlled by Arduino Mega. Each motor is controlled by PWM signals and direction pins. The encoder signals are used as feedback to smoothen the move of the robot.

- Task 4: Kicking

The kicking mechanism of the robot consists of a solenoid. The solenoid is triggered by user command and after kicking it restores its position.

- Task 5: Dribbling

The dribbling mechanism consists of a cylinder with varying radius. The dribbling mechanism is triggered by the user. The cylinder spins the ball to the robot and the robot can move with the ball smoothly.

- Task 6: Not Crashing with the Walls

The 3 ultrasonic sensors are employed to check distance to the walls. The distance to the walls will be streamed to the user screen. The robot cannot be closer than 5 cm to the walls in the direction of the sensors.

- Task 7: Defending the Goal

The robot will defend its goal while the ball is on the opposite side. The commands will be given by the user.

Subsystems of the Project

The subsystems of the project are listed below.

- Dribbling and Kicker Module
- Driving Module
- Vision Controller Module
- Communication Module
- Command Module

Further technical details for each module are provided in Section 5.

All subsystems were designed to provide solutions that are efficient in all aspects while meeting the design requirements specified in the project definition and by the Standards Committee. One other design criterion for our company is our objectives which are summarized below.

Performance Objectives -Fast and accurate data exchange, Light weight, Fast and multi-directional moving ability, Ball control and accuracy, and Robustness

Marketing Objectives -User-friendly (Easy to use), Inexpensive, Low Power Consumption, and Presentable

Functional Diagram of the System

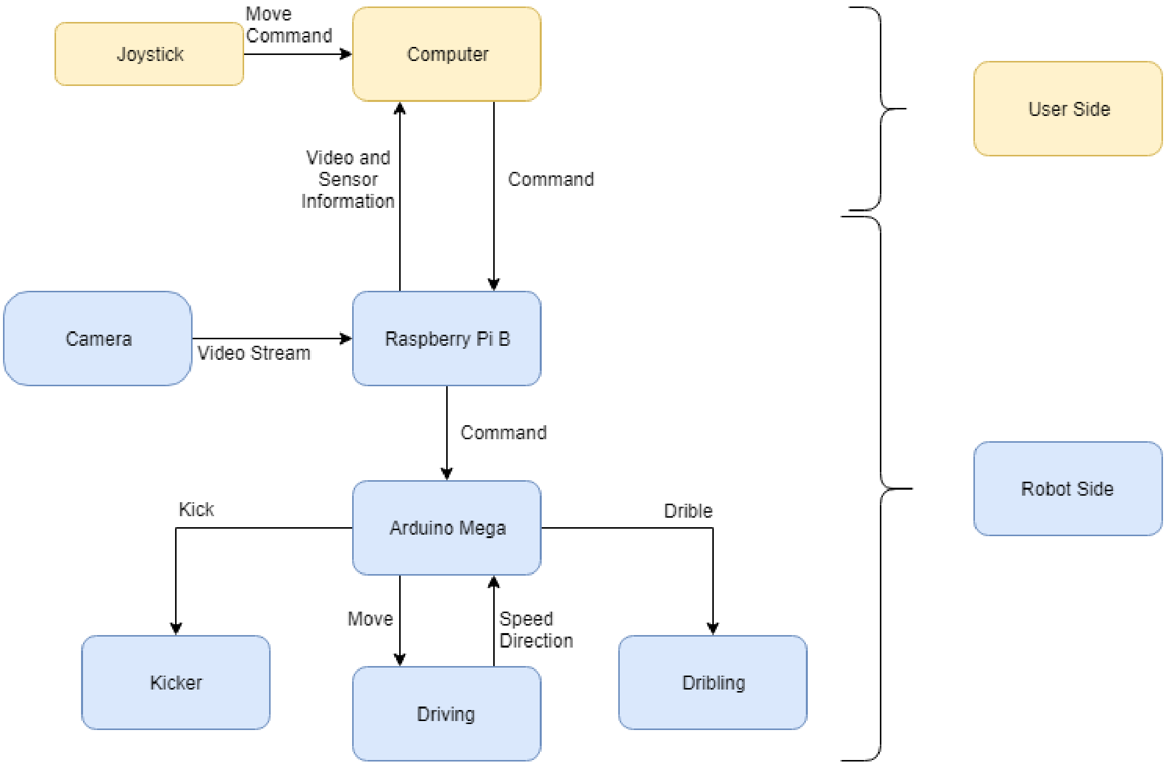


Figure 6. The Operation Diagram of the Overall System

SUBSYSTEM ANALYSIS

As mentioned before, the overall system is composed of five subsystems, all of which are explained in detail below.

Dribbling and Kicker Module

The very first thought that came into our minds about the kicking mechanism was to use the kinetic energy of the robot. Without any other external mechanisms, we thought that we could kick the ball inside of the opponent's side by hitting the ball with our chassis. Although this idea was realizable and easy, it was not the best way. This method had some drawbacks such as bad targeting, not having enough energy for kicking etc. As PORO Inc., we never run away from a challenge, if it will make our design better. Therefore, we have decided to use another mechanism for this purpose: A Solenoid Kicker (Figure 7).

Figure 7: Solenoid Kicker of our robot

We have chosen to use a solenoid kicker due to various reasons including:

- Shooting will be very fast as compared to using the momentum of the robot
- More durable and robust mechanism
- Accurate targeting

Electromechanical solenoid is the most optimal solution since it is quite light and powerful. Referring to Figure 8, plunger is moving in and out of the center and it is used to provide force to the ball.

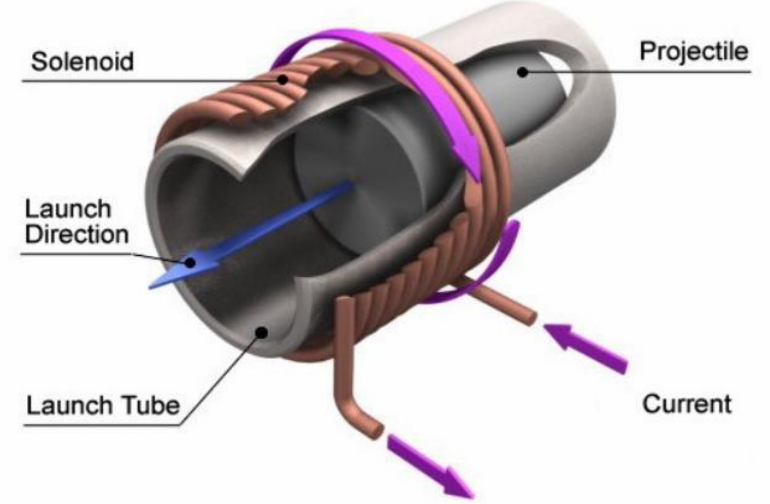


Figure 8: A schematic representation of a solenoid

At the beginning, the purpose of the dribbling/kicker mechanism was to control the ball and shoot it in the desired place. The dribbling mechanism is designed as a rotating cylinder which spins the ball back, so that the ball is drawn constantly to the robot. In that way, the robot can control the ball effectively and decrease the possibility of the ball bouncing from the robot. The designed dribbling mechanism consists of two completely identical cylinders. From the center of the radius, the dribbling mechanism is connected to a DC motor which is operator controlled. The DC motor starts to rotate when the operator gives the rotating command and it stops when the stopping command arrives. However, this implementation has not worked properly. We tried our best efforts to control the ball but it failed. On the other hand, we met another kicker mechanism besides of the solenoid kicker. This dribbling mechanism works well as a kicker mechanism. Therefore, at the end we have had two kicker mechanisms which is fine. The solenoid mechanism is placed between the dribbling cylinders which are printed in 3D printer. In Figure x9, the overall block diagram for the kicking mechanism is shown in detail.

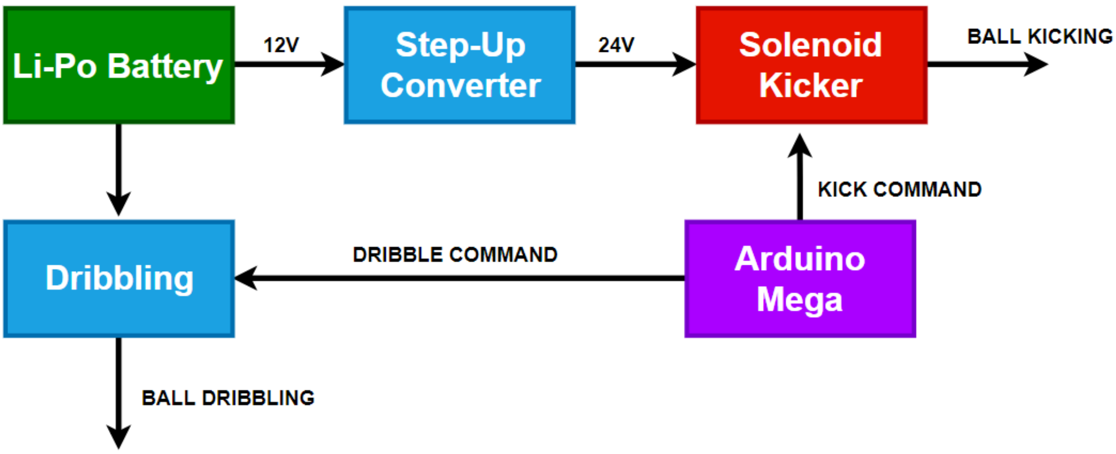


Figure 9: Block diagram of the Kicker & Dribbling subsystem

Driving Module

The robot must move to get the ball, correct its position and defend the goal. The movement is controlled by the user. The commands are transferred via 3G dongle. Raspberry Pi transfers the commands to the Arduino. Arduino drives the motors and the mobility of the robot is provided.

The Driving Mechanism of the robot consists of 4-Omnidirectional wheels, 4-encoder motors, 2 motor drivers and Arduino Mega.

Figure 10. Omni Directional Wheel of the Robot

The omni-directional wheels are chosen according to weight of the robot. Each wheel can carry 2 kg. The wheels are plastic but provides enough torque and speed for the robot.

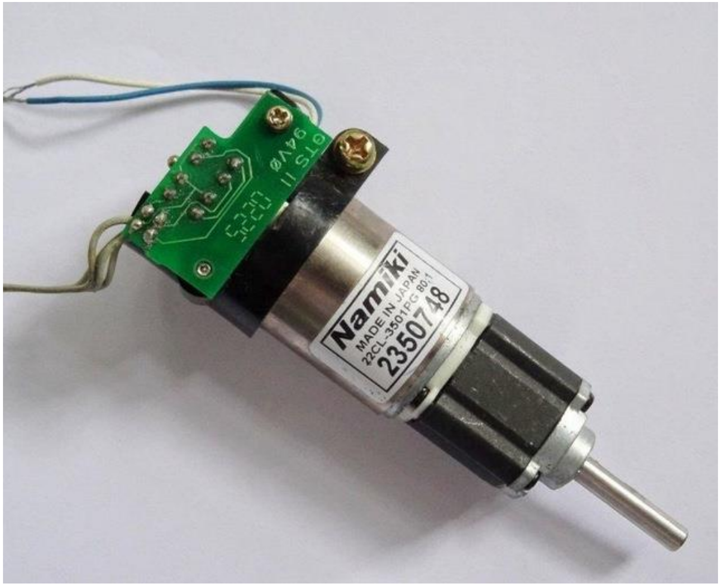


Figure 11. The Encoder Motor

The encoder motors are Namiki 22CL-3501PG. The motors have 16.5 kg*cm stall torque, 120 RPM output speed at 12 V supply, 5 kg*cm continuous torque, and 15 W rated power. These specifications are sufficient to provide movement of the robot.

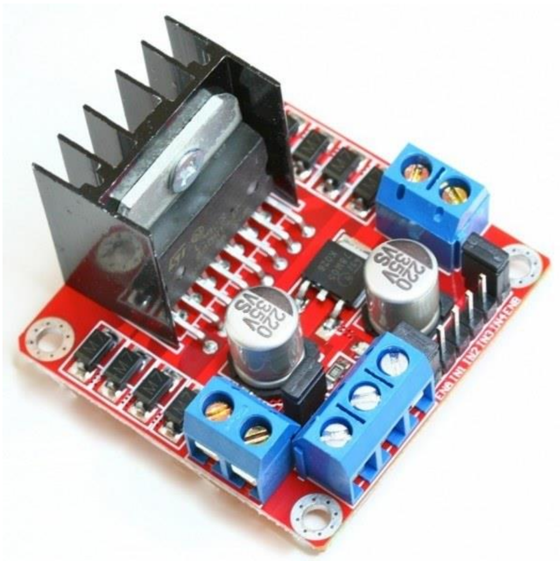


Figure 12. L298N Motor Driver Module

The motor driver is L298N Motor Driver Module. The module can control two motors separately. It can supply 2 A per motor. It enables PWM control.

The motors are driven by Arduino Mega. The PWM pins and Digital I/O pins are used. There are 6 interrupt pins, but the number will be increased via pin change interrupt. The interrupt pins are used as feedback to the driving system.

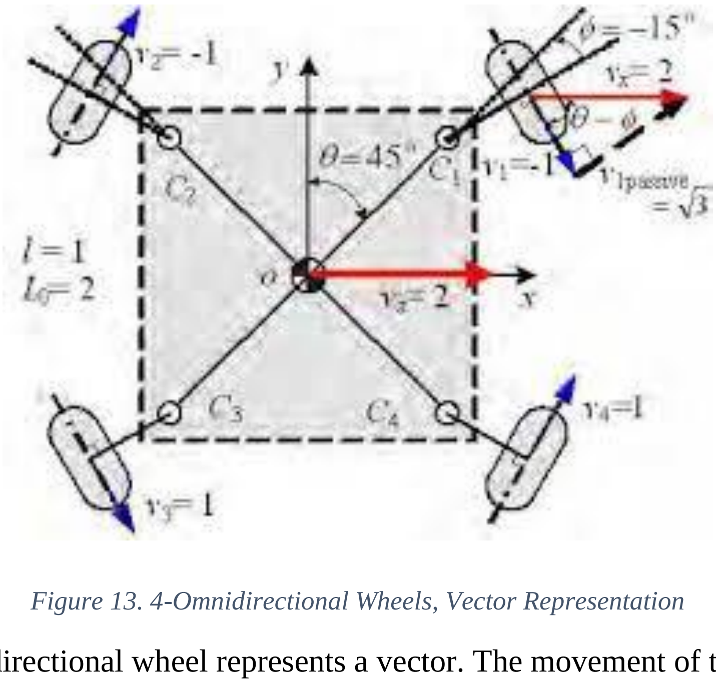


Figure 13. 4-Omnidirectional Wheels, Vector Representation

Each omnidirectional wheel represents a vector. The movement of the robot is the sum of all the vectors. The power transferred to each wheel is controlled via PWM signals and the wheel, motor and other non-idealities will be balanced by regulating PWM signals. The encoder signals will give information about rotation speed and rotation direction. This information will be fed into the Arduino code to regulate PWM signals.

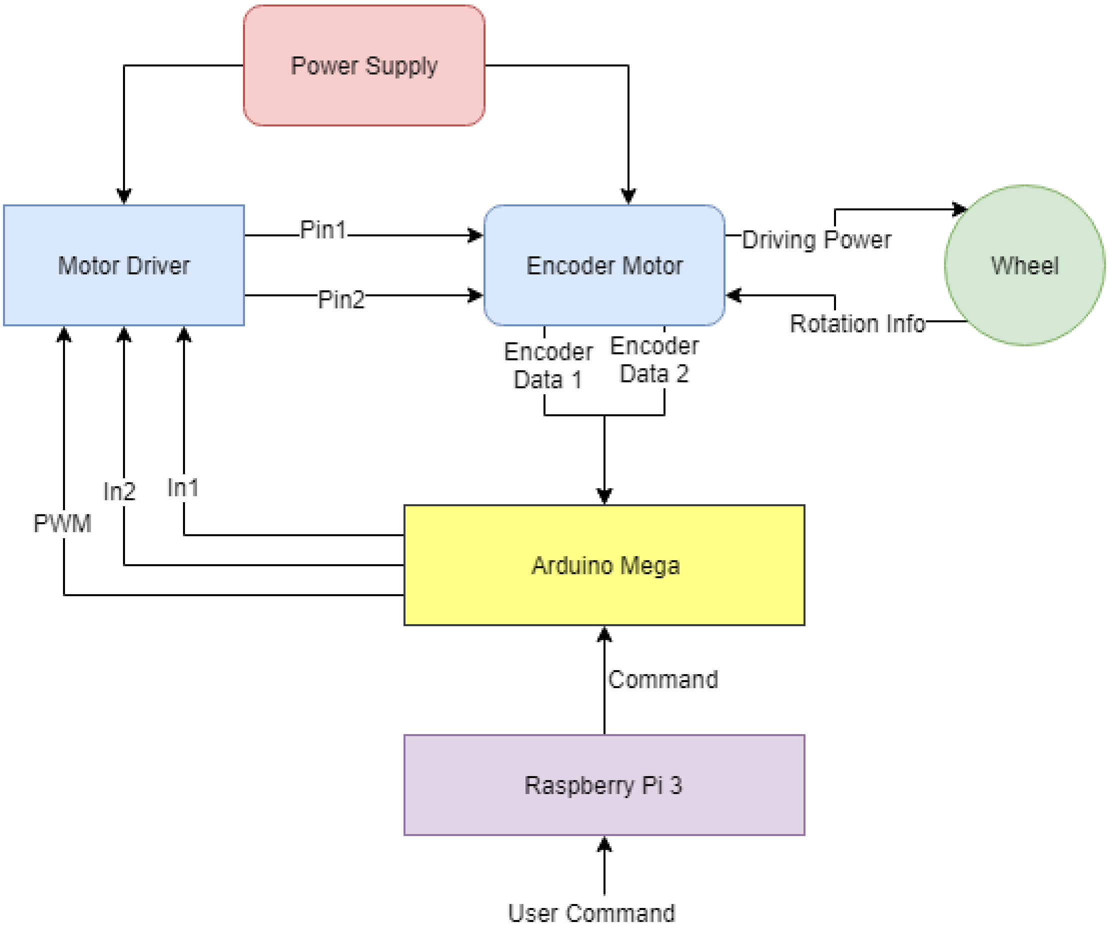


Figure 14. Operation Diagram of Driving System

Speed control of the robot movement is adjusted for two different levels which can be classified as slow and fast modes. The speed of the robot is controlled by the PWM signal provided by the Arduino Mega to motor drivers. For slow mode, PWM of the motor is adjusted to provide 50% of the maximum speed. For fast mode, full PWM signal which is 100% is applied to enable pins of motor drivers. The mode is decided by the position of the joystick. Position value of the joystick ranges from -1 to 1. For exact value of the position value, slow mode PWM is applied to motors for exact position values from 0 to 0.5. For values out of the slow mode range, fast mode PWM is applied to motors.

For bidirectional movement of the robot which is to move in x or y direction only, all four motors are employed. The rotation directions of the motors are adjusted according to Figure 13. For unidirectional movement of the robot, only two motors chosen according to direction are employed. Also, corresponding motors for bidirectional movement can be examined in Figure 13.

Encoder reading and feedback mechanism to smoothen robot movement weren't employed fully. Due to smaller field size and similar motors and robot design, movement of the robot is smoothened via adjusting PWM signal magnitudes. If there will be any need for feedback signals, encoder pins and encoder outputs of the motors are arranged and the process for smoothened movement will be conducted.

Vision Controller Module

Inputs: Video, Distance to Walls, Touch sense

Outputs: Processed Video, Relative Position Information, Signal of Touching the Ball

Since the operator cannot control the robot by seeing it with naked-eye, there should be such a module on the robot that we can control its position and movement properly. Moreover, we must be able to control the ball to a certain level and shoot when the right time comes.

For our project, we propose to solve these problems with a vision controller module mounted on the robot. This module includes both software and hardware parts in it. There is a standard determined by the standard committee which claims that there should be an onboard camera on the robot. Therefore, we use A Raspberry Pi Camera Module v1.3 as our onboard camera which can be seen in Figure 15. It continuously streams video from the play-field. Since the data transfer rate of the communication module is to be limited, we have limited range of resolution and fps (frames per second) values.

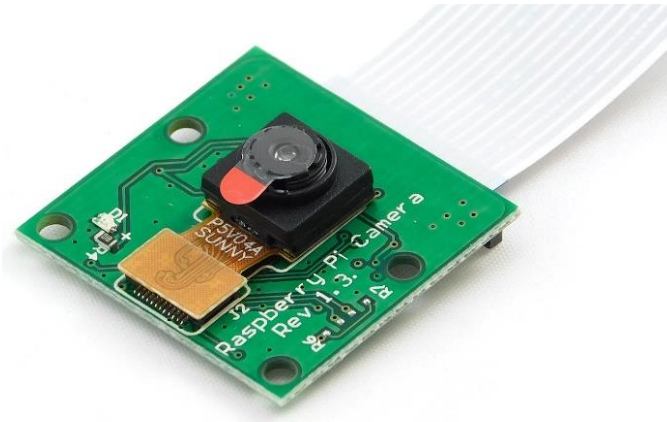


Figure 15: Raspberry Pi Camera v1.3

We have optimized these values by the experiments explained in the Section 4.6. To replace the information loss and increase the performance of the operator, a software unit in the Raspberry Pi, processes the video and makes some useful interference if the operator wants.

There is a command which will be implemented at the end of the project as an extra feature. The name of the command is “Find the ball”. In this mode, the streaming video will be processed to detect the ball on the playfield. When the ball appears in the center vertical line of the video frame cut the power from the motors of the wheels to pause the motion until a next command.

This command is added to help the human operator to align the direction of the robot with the ball. When the operator wants to find the ball and face towards to it, he/she sends the “Find the ball” command and the robot enters a different state. In this state, the operator can rotate the robot around itself.

In addition to visual information, this module also takes distance information of the robot to the walls from HC-SR04 ultrasonic sensors because they are cheap and have enough performance for our case. With the distance information, we can better understand the position of the robot in the play-field.

Figure 16: HX-SR04 ultrasonic sensor

Communication Module

Input from Raspberry Pi side: Video (compressed version)

Output from the computer side: Video (compressed version)

Input from the computer side: Commands (Movement, Kicking, Dribbling)

Output from the raspberry side: Commands (Movement, Kicking, Dribbling)

For this module, we have decided to use 3G communication for simplicity and applicability. By using 3G Dongle Modem and SIM Card, Raspberry Pi is connected to the Internet. Then we have installed Dataplicity to our Raspberry Pi. Dataplicity allows us to connect to our Raspberry Pi from anywhere in the world that has the Internet connection. We have done port forwarding and forwarded the two ports of our Raspberry Pi by using

Dataplicity. One of the two ports is Port: 22 which allows us to make an SSH connection between Raspberry Pi and our computer. The other port is Port: 5901 which is used for streaming (using the camera of Raspberry Pi) and using the GPIO (General Purpose Input/Output) pins of Raspberry Pi which allow us to send commands from our GUI. We have used MJPEG Streamer to start live video streaming in Port: 5901 and observe the live video stream in a browser via Port: 5901. After we have done with port forwarding in Raspberry Pi, we have installed Dataplicity Porthole application to our computer to make a connection between Raspberry Pi and our computer. We can now see the live video stream and GUI by entering the unique web address that is given from Dataplicity to our browser. Thanks to this connection, we have no limit such as 30 meters as mentioned in project description. We have a global connection that provides communication between Raspberry Pi and our computer easily.

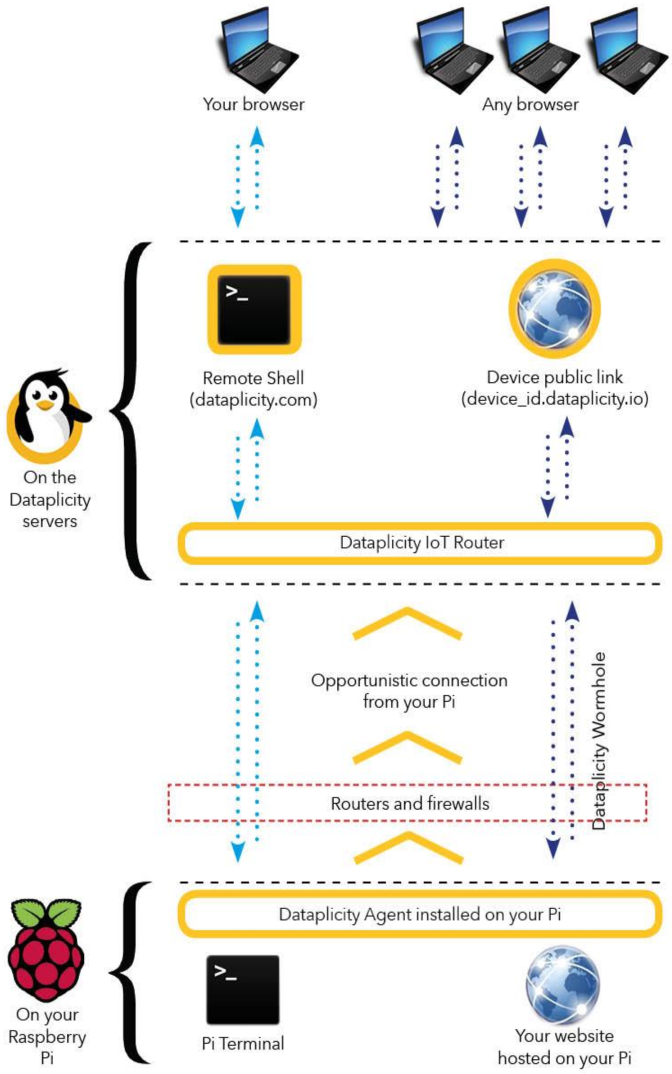


Figure 17: Overall view of the communication with Dataplicity

After all these steps, the person who is in a controller side simply can connect his/her computing unit to the Internet to be able to receive data which comes from the robot as a video and similarly to be able to send data to the robot as a simple command that come from the graphical user interface of computing unit. After successful communications, we can see the data transfers in both directions easily thanks to the well-designed GUI and sending small size video.

3G dongle that is decided to implement is ZTE MF667 3G USB Modem is shown in Figure 18. It has maximum transmission speeds with 21.6 Mbps Download / 5.76 Mbps Upload, which is quite enough for our application. This dongle makes connections with

Wideband Code Division Multiple Access special modulation method for 3G. It operates in a temperature interval with (-10 ^{o}C, +60 ^{o}C).

Figure 18: 3G USB Modem

We have less than half a second delay in live video streaming with this communication module. Detailed test results of the communication module can be seen in "Test Results" section of this report. Overall view of the communication module can be seen in Figure 19.

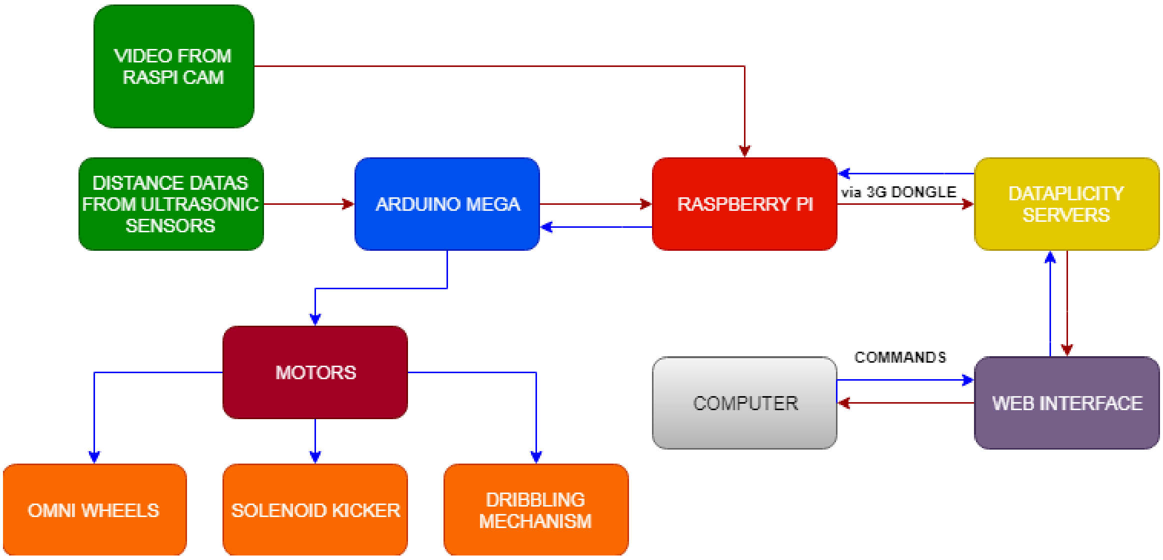


Figure 19: Block diagram of communication module

Command Module

Inputs: Commands for movements, kicking and dribbling from the gamepad

Outputs: Command messages that will be transmitted to Arduino via Raspberry Pi

Controlling the robot from a distance is the main part of the project. For this reason, command module is a crucial part of the robot and it unites all subsystems in a way that our project can reach its goal.

First, we want to explain how we obtain the environment for sending commands and receiving information back from the robot. For this purpose, we use WebSocket protocol which is a computer communication protocol. The feature that distinguishes WebSocket from other protocols is that it provides full-duplex communication channel over a single TCP connection, thus it enables bidirectional communication in real time over the web. In other words, it enables interaction between a web client (such as a browser, in our case the website that is provided by Dataplicity, as mentioned in section 6.4) and a web server that we started using Tornado Python module, facilitating real-time data transfer from and to the server. Also, since WebSocket provides bidirectional communication, we are able to get sensor data from the robot to the website without client request.

To use the website as our basis for sending commands and seeing the stream, the codes split into three parts, HTML, JavaScript and Python. HTML codes arranges the layout and the format of the website, also allows us to see the video stream. JavaScript codes enables WebSocket protocol and keeps the current states of the joystick and buttons of the gamepad and sends the messages when there is a change in these states by evoking functions that we wrote. Python codes starts the server and receives the messages from JS code.

We decided to use a gamepad as a controller, as in Figure 20. Since we are using omni-wheels, we did not want to lose the efficiency of the moving ability that omni-wheels give us. After trying different numbers of directions, we decided that moving in 8 distinct direction enables a smooth control of the robot over the game field and it is enough for our purposes.

Figure 20: Snopy Gamepad

In the current state of the controllers, the left stick is used for movement in 8 different directions and the right stick is used for turning the robot to right or left around its own axis, starting the dribbling and kicking mechanism. But it is possible to change the configuration of the controllers based on user preferences. Dribbling and kicking mechanism can be controlled with buttons, too, since we also sent button data. Figure 21 and 22 show the examples of the message sent to RPI from website and the message sent from RPI to Arduino for stick and for buttons, respectively.

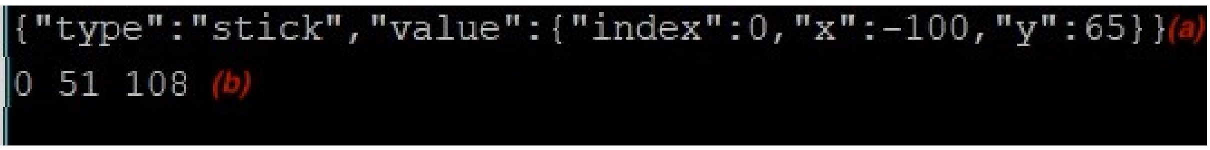

{"type":"stick","value":{"index":0,"x":-100,"y":65}}(a)
0 51 108 (b)

Figure 21: The message sent to RPI (a), The message sent to Arduino (b) for stick.

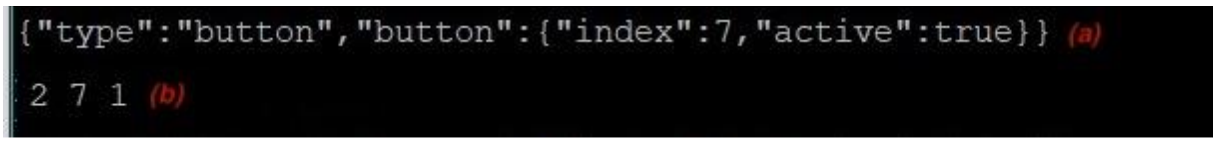

{"type":"button","button":{"index":7,"active":true}} (a)
2 7 1 (b)

Figure 22: The message sent to RPI (a), The message sent to Arduino (b) for buttons.

As it can be seen from Figure 21 and 22, the message for Arduino is always consists of three numbers. The first number is for differentiating the source of the data. The first number is equal to “0” for the right stick, “1” for the left stick, and “2” for the buttons. When either one of the sticks are used, the remaining two numbers are for the angle and for the magnitude, respectively. The angle information is mapped between 0 and 127, because of limitation in the message size. When any button is used, the remaining two numbers are the index and state of the button, respectively. When Arduino received these messages, it interprets these numbers and controls the pins accordingly.

The control of the robot is realized in this way: When the Raspberry Pi is connected to internet, we will open the webpage provided by Dataplicity, therefore WebSocket protocol will be started and the connection will be established. Then we will connect the gamepad. When we move the joystick or press buttons, a suitable command will be sent to Raspberry Pi. Raspberry Pi and Arduino Mega is connected and according to the incoming commands to Arduino from Raspberry Pi, Arduino Mega will change the states of its pins and perform the proper action. The block diagram in Figure 23 summarizes the control of the robot. The control of the robot is realized in this way: When the Raspberry Pi is connected to internet, we will open the webpage provided. As it can be seen from Figure 9 and 10, the message for Arduino is always consists of three numbers. The first number is for differentiating the source of the data. The first number is equal to “0” for the right stick, “1” for the left stick, and “2” for the buttons.

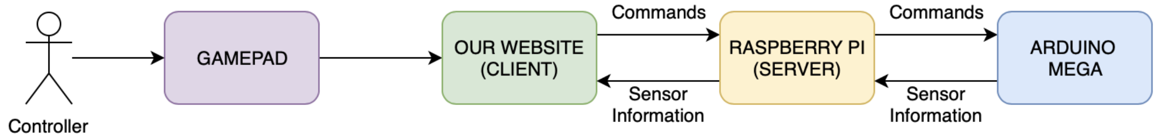


Figure 23: Block diagram of the command module

SUBSYSTEM COMPATIBILITY

The overall robot consists of five subsystems which are

- Command Module
- Vision Controller Module
- Communication Module
- Driving Module
- Dribbling & Kicker Module

Each subsystem is controlled and checked separately. The connections between subsystems have been made according to the plan.

The critical module is communication module. The communication module allows us to transfer data and command in both directions. Raspberry Pi is connected to internet via a 3G dongle. The data transfer will be handled by internet. The Raspberry Pi will be connected to internet during all operation process. The user will be connected to via its own device. User and Raspberry Pi will communicate on a website.

The vision controller module consists of Raspberry Pi and its camera. The view of the camera is transferred to the user via Raspberry Pi. The camera view is sent to a website and the user can reach the view from that website.

The driving module consists of wheels, motors, motor drivers, and Arduino Mega. The subsystem parts have been connected, and the tests have been conducted without any external command module. The external module has been added after correcting the mistakes. The external command module which consists of Raspberry Pi and user interface has been tested with Arduino Mega at the first stage. After first stage and separate stages, all driving module has been connected to the communication and command module.

The dribbling & kicker module consists of a motor, a motor driver, a solenoid, and two cylinders. The dribbling side of the module consists of a DC motor and cylinders. Cylinder spins in the opposite direction and the ball will stay with the robot. The motor driver has driven this part. The kicker mechanism consists of a solenoid and a kicker. The kicker allows us to increase shooting area. These systems are operating together. The kicker and dribbling mechanisms are triggered by user and are stopped by the user. These systems are connected to the Arduino Mega. The operation between Arduino Mega and both mechanisms are tested via basic commands and the communication and command module are also integrated.

The overall system diagram is also summarized in Functional Diagram of the System.

TESTS

The completed tests and their related subsystems are shown in Table 2. Related subsystems are labelled with numbers under the table.

Table 2. Completed Tests and Results

Completed Tests	**Related**	**Test Results**

	Subsystem	
Optimizing the resolution of the video stream	4	We decided the resolution of 200x200pixel is enough for observing the game field.
Accuracy of ultrasonic sensor HC-SR04 for distance measurement	3	After 5cm, the sensor measures the distance with at most 5 % error.
Measuring the visual angle of the Raspberry Pi cam	3	With using an external lens, the visual angle has become 91.2^{o}
Measuring the stream delay	4	When we have a stable connection, the worst delay is 590 ms, the best delay is 400 ms.

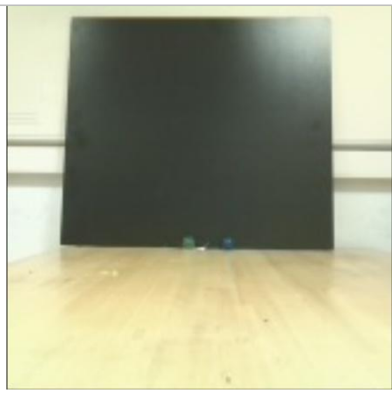
Experiment 1: Optimizing the resolution of the video stream

Objective: Since we have a limited bandwidth of the Internet connection, we should have to use the lowest possible resolution to obtain the lowest possible communication delay. However, there is a limit of resolution while lowering it since we must preserve the useful information on the image, such as, the ball, the lines of the goals, the playfield and the opponent robot. Therefore, the objective of this experiment is determining the lowest possible video resolution without losing the necessary information.

Setup: Since the object with the lowest dimensions in the playfield will be the ball, we will use it as our baseline for seeing the objects on the playfield. The playfield has a shape of equilateral hexagon with 75 cm edges. Since the distance between robot and the ball will be at most 150 cm, we have used an experiment setup like the Figure 24 shows.

Figure 24. Image resolution optimization test setup

Results: At the end of each trial with different resolutions we decided that 200x200 is the lowest resolution that the ball can be seen without much effort with human eye. Resolutions greater than 200x200 have no significant contribution to easiness of detection. Therefore, there is no need to use them.

40x40	80x80	160x160
		
200x200	240x240	320x320

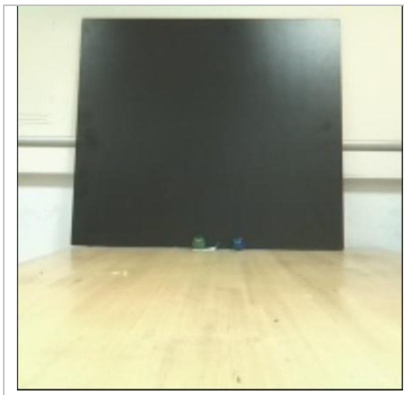
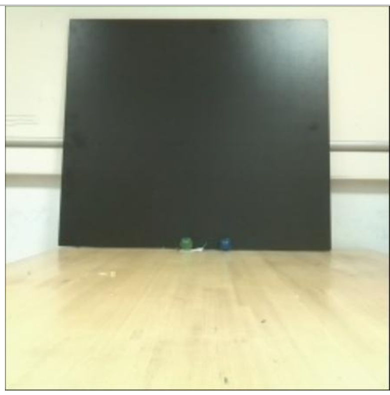
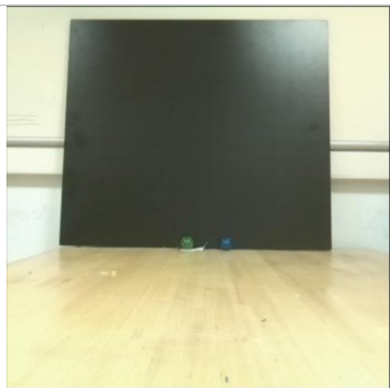
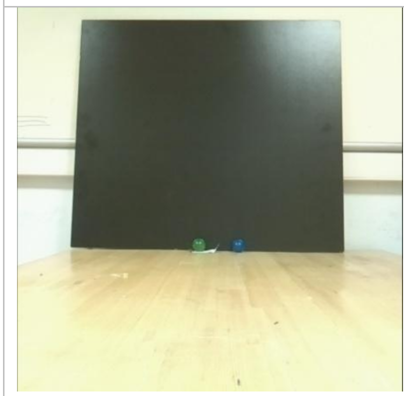
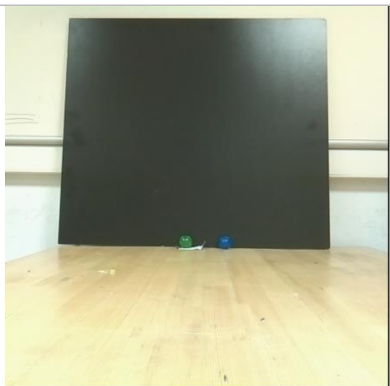
		
480x480	720x720	1080x1080
		

Figure 25 Images captured with different resolutions

Experiment 2: Accuracy of ultrasonic sensor HC-SR04 for distance measurement

Objective: We have a camera that shows the front of the robot. However, it is a difficult task to locate the robot on the playfield and determine the distance to the walls just by looking at to the video stream. Therefore, we use ultrasonic sensors to measure the distance. The objective in this experiment is to determine the accuracy of these measurements and decide whether they are reliable or not.

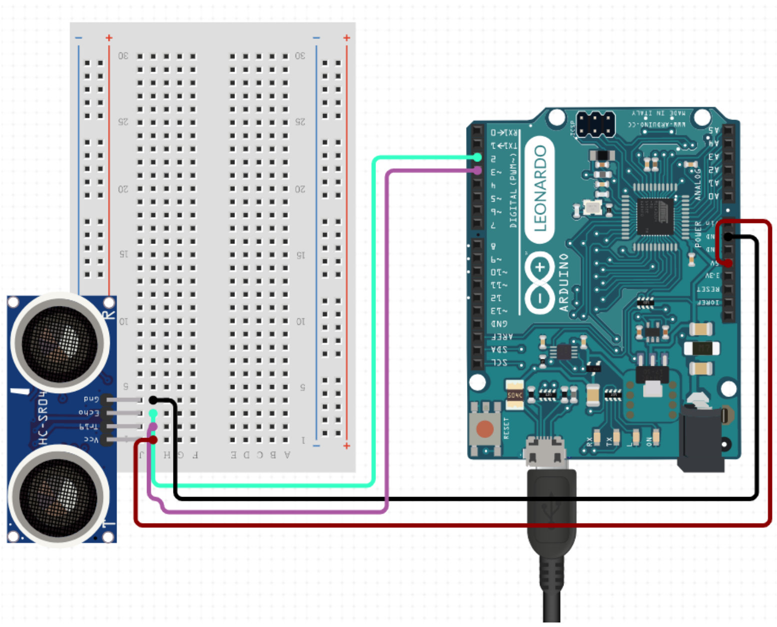


Figure 26. Connections of HC-SR04 and Arduino Leonardo

Figure 27. Distance measurement accuracy of HC-SR04 sensor test setup

Setup: To get the distance measurements from HC-SR04, we have used Arduino Leonardo. We placed the sensor to a distance from a wall of the playfield. By changing the distance, we recorded the measured values.

Results: In Table 4, the real distances and their corresponding measured values are shown. It can be understood that HC-SR04 cannot measure the distance correctly up to 5 cm. After 5cm, the sensor measures the distance with at most 5 % error, which is acceptable for our application. For the distances lower than 5 cm, it measures values between 3 and 5 cm. Therefore, we can understand that if the sensors show values lower than 5 cm, we should not drive more to that direction to avoid any collision.

Table 3. Real distances and Corresponding Measured Values

Distance (cm)	Measured Value (cm)	Error (%)
1	3,54	254
2	3,18	59
3	3,99	33
4	4,55	13,75
5	5,02	0,40
6	6,29	4,83
7	7,25	3,57
8	7,84	2,00
9	9,21	2,30
10	10,17	1,70
15	15,07	0,46
30	29,26	2,47
40	38,76	3,10
60	58,49	2,52
75	73,25	2,33
120	118,26	1,45
150	147,56	1,63

Experiment 3: Measuring the visual angle of the Raspberry Pi cam

Objective: We should have enough angle of view to see the walls, opponent's goal and the ball (if it is in front of the robot) at the same time. We did this experiment before and we saw that only RPi cam is not enough, this time we used additional lens. The objective of this experiment is checking if we have the enough angle with the Raspberry Pi cam with an external lens.

Set-up: We used mobile robot we built for the demonstration. We started video streaming and we used cups to mark the first points that cam can see. Then we measured the distances between the cups and robot. Figure 26 shows the set-up and measured distances.

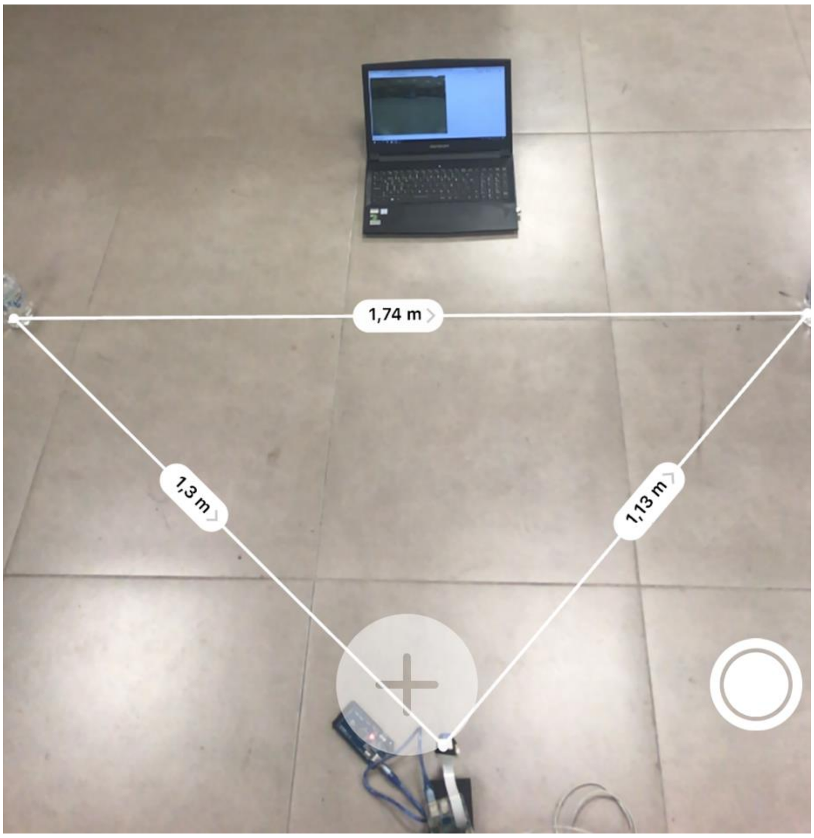
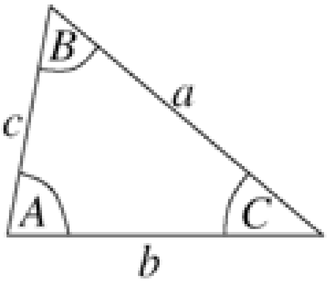


Figure 28. Measuring the visual angle of the Raspberry Pi cam test setup

Results: We calculated the angle by using the distances seen in Figure 28 and cosine rule in Figure 29. And the calculated angle is equal to 91.2°, before without lens it was 44.52°. Now we have enough vision angle.



$$\cos(A) = \frac{b^2 + c^2 - a^2}{2bc}$$

Figure 29. Cosine rule for calculating an angle

Experiment 4: Measuring the stream delay

Objective: Since we cannot monitor the play-field with naked eye while controlling the robot, we should have as small as possible delay in the stream to be successful and ideally the delay should not exceed 800 milliseconds according to our own constraints. This experiment's objective is to see if we have manageable delay.

Setup: We started the stream from RPI and a chronometer from a laptop, then we directed the cam into the chronometer and watched it from the same laptop. The difference between the real chronometer and the streamed chronometer gave us the delay.

Results: Figure 30 shows the screenshots of the best and worst case, respectively. As it can be seen from the figures, when we have a stable connection, the delays are in our margin. Of course, our goal is improving these values by finding the optimum fps rate.

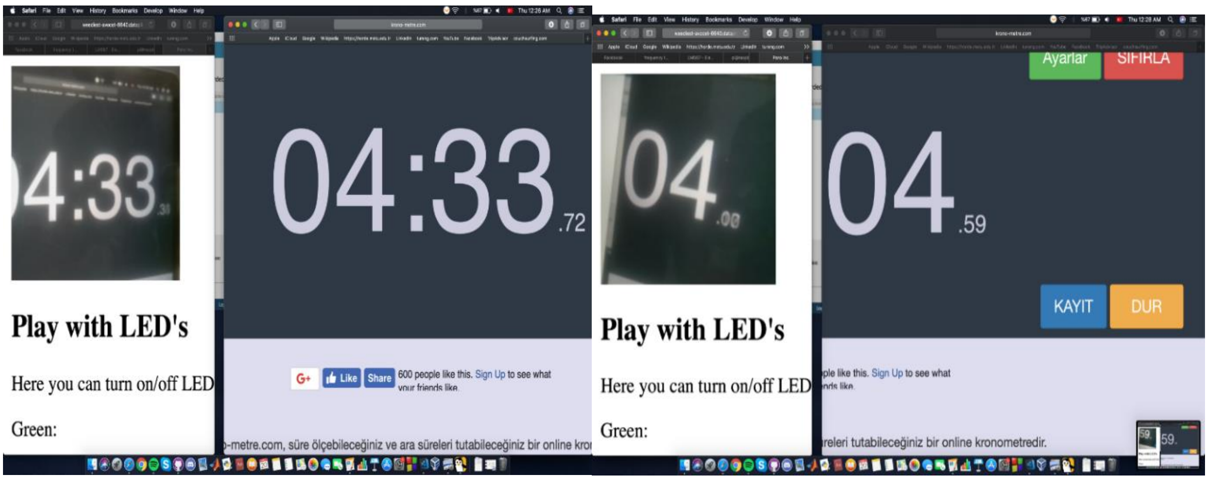


Figure 30. Best delay (around 400 ms) and Worst delay (around 590 ms)

Experiment 5: Measuring the kicking range

Objective: To make a score during the game, our robot should be able to kick the ball to the opponent's goal. Therefore, the objective of this experiment is to determine whether our robot can kick the ball to a distance which is enough to make a score.

Setup: We placed the ball in front of the robot with a certain distance between them. Varying the distances, we have measured the kicking range. The test setup can be seen on Figure 31.

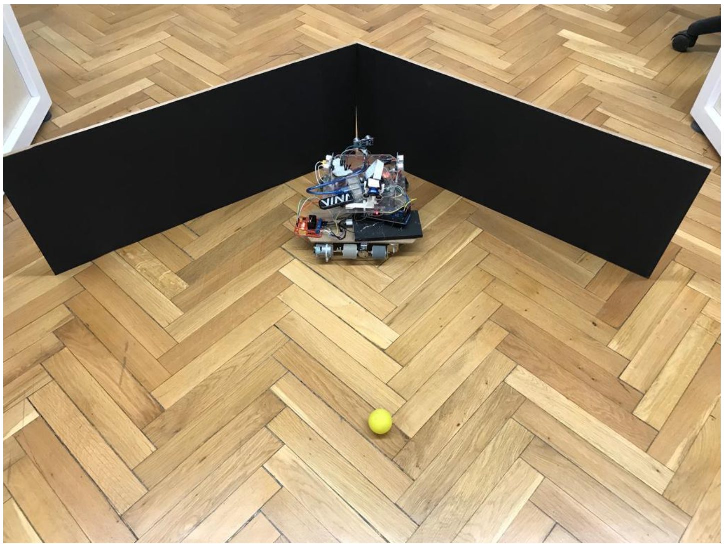


Figure 31. Image resolution optimization test setup

Results: Our robot can kick the ball to 2,5 meters when distance between them is initially 30 cm. When the distances are 20 cm and 10 cm, kicking range is 1,7 meters and 0.8 meters respectively. Since our robot should be able to kick the ball up to 1,2 meters' distance, we should keep our robot about 15 cm away before shooting.

Experiment 6: Directionality and Speed Measurement of Robot Mobility

Objective: To check the directionality of the movement and to check speed difference for fast and slow modes.

Setup: Commands of corresponding directions and PWM values are adjusted for Arduino Mega and the code is uploaded to Mega. The robot is fully loaded with all components. For each direction and mode, the directionality and speed are observed. Due to smaller field size, the speed and directionality are only observed with eye, no measurement is conducted.

Results: According to observations, the directionality of the robot is enough for the game field. For bidirectional moves, all motors with same PWM value provides a smooth movement. For unidirectional moves, only two motors with same PWM value are employed and the directionality of movement is enough for the field size. For two different mode with different PWM sizes, the speed is changed significantly.

FINANCIAL UPDATES

The project has a budget of $200. In Table 4, the components, quantity of them and the price of each component is listed.

Table 4.Cost Analysis of Project

Components	Quantity	Cost	Total Cost
Raspberry Pi	1	\$33	\$33
Arduino MEGA Clone	1	\$6	\$6
Raspberry Pi Camera	1	\$5	\$5
Ultrasonic Sensor HC-SR04	4	\$1	\$4
Brushed DC MOTOR	1	\$1	\$1
DC MOTOR with encoder	4	\$10	\$40
L298N Dual H-Bridge Motor Controller	3	\$2.5	\$7.5
3G Dongle Modem	1	\$7	\$7
SIM Card	1	\$11	\$11
Chassis	1	\$10	\$10
11.1V Li-Po Battery 1350mAh	1	\$13	\$13
10000mAh Power Bank	1	\$10	\$10
Solenoid Kicker	1	\$2	\$2
Omni Wheels	4	\$4.5	\$18
Coupler	4	\$2	\$8
Design Material	-	\$6	\$6
Gamepad	1	\$4	\$4
Overall			**\$195.5**

Poro Inc. can manufacture the product with a cost of \$195.5 which is in the range of product budget.

POWER AND WEIGHT ANALYSIS

The overall power consumption analysis is included in Table 5.

Table 5. Power Consumption Analysis

Equipment	**Quantity**	**Power**
Raspberry Pi 3B + RaspiCam v1.3 + MF667 3G Dongle	1	10 W
Arduino Mega	1	2500 mW
HC-SR04 Ultrasonic Sensors	4	100 mW
DC Motor (For Dribbling)	1	6 W
DC Motor with Encoder (For Driving)	4	40 W
Solenoid Kicker	1	16 W
	Total	**74.6 Watts**

The maximum current drawn by the modules and components are given in the Table 6.

Table 6. Current Draw of Components

Equipment	**Quantity**	**Current**
Raspberry Pi 3B + RaspiCam v1.3 + MF667 3G Dongle	1	2000 mA
Arduino Mega	1	500 mA
HC-SR04 Ultrasonic Sensors	4	20mA
DC Motors	5	7000mA
Solenoid Kicker	1	1400mA
	Total	**10920 mA**

One of the objectives of our project is accomplishing operation for at least 30 minutes without power failure. To supply power to our system it is decided to use a power bank with two 2 A USB ports and a ProFuse 11,1V Lipo 1350mAh 25C battery.

Table 7. Weight Analysis

Component	Weight (in Grams)	Number	Total Weight
Encoder Motor	140	4	560

Arduino Mega	37	1	37
Raspberry Pi 3	30	1	30
Raspberry Pi Camera	15	1	15
Plexi	250	2	500
Wheel	30	4	120
Li-Po Battery	115	2	230
Powerbank	223	1	223
Motor Driver	20	3	60
Raspberry Pi Case	85	1	85
Camera Case + Lens	80	1	80
3G Dongle	30	1	70
Cables	0.1	N.A.	-

ORGANIZATIONAL PLAN

One of the critical points in the project design stages is making plans and following them. Therefore, as PORO Inc., we prepared our time plan in a Gantt Chart format which is given in Figure 31. After the critical design review report, we will be mainly focusing on the testing and integration of subsystems. Before the integration, each subsystem will be tested one by one because we know that solving the possible problems in early stages will be beneficial for us. After the integration of the subsystems and mechanical design are done, we will keep doing overall tests to make sure that our system works well and fits in the requirements of the project. After all the system integrations are finalized and all the tests are completed, the hockey robot will be ready for the project fair. The following Gantt chart shows the project timeline.

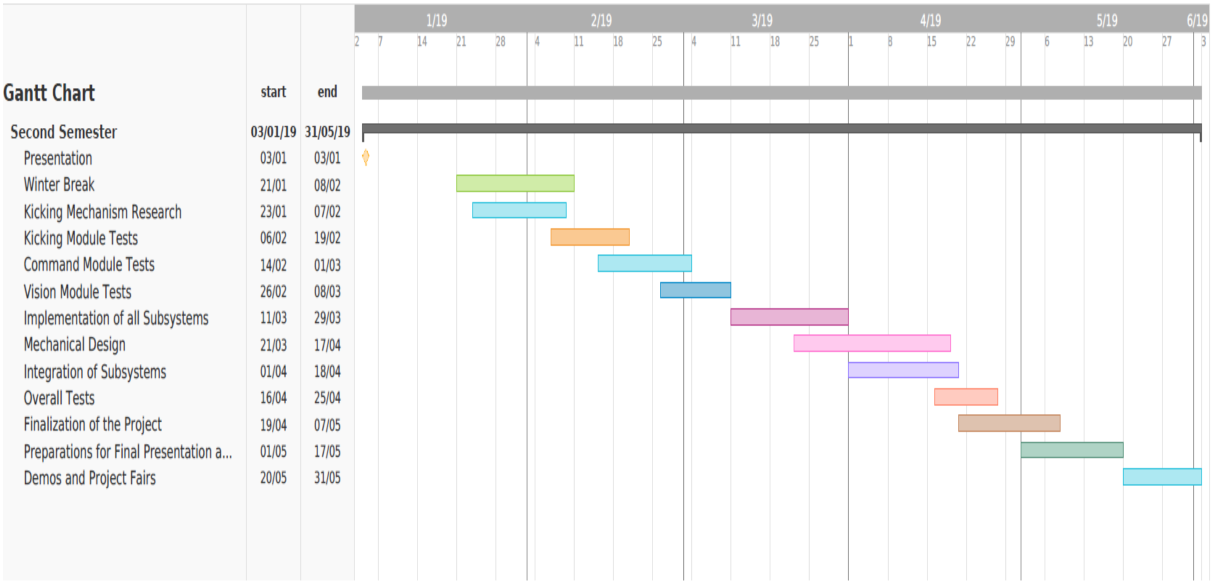


Figure 32. Gantt Chart of the Project

SAFETY CONCERNS AND PRECAUTIONS

Situations that can lead to safety problems and their respective safety precautions are provided in the Table 8 below.

Table 8. Safety Issues and Solutions

<u>**Safety Issue**</u>	<u>**Prevention Measure**</u>
<u>**Battery**</u> - The Li-Po battery voltage in any cell has gone below 3.7V. - There are deformations on the battery surface (the surface is bended). This means the battery is not working properly and there is risk of explosion.	- The regular battery check procedure is explained in the user manual.
<u>**Cabling**</u> - The wires connected to the battery and the other components can get short circuited, thus causing a fire.	- The cables have the proper covering such that close contact with the battery terminals is prevented.
<u>**Microcontrollers safety**</u> - Raspberry Pi can overheat due to high computational load and may be damaged.	- Several heat sinks and a cooling fan are added in order to keep the processor at a safe temperature level.

- When the battery is turned off before the microcontrollers, the circuits will try to draw current from the microcontrollers alone, causing them to burn out.	- Turn off the microcontrollers before the battery is turned off.

FEASIBILITY AND ROBUSTNESS

The solution procedure is explained explicitly; the subsystems are clearly described. Components of subsystems are determined. The tests for camera view transfer, command transfer, and robot mobility are conducted and reported in the report. The other subsystems are also tested, and test results are included in the report. All aspects of feasibility analysis are explained in below.

Economic Feasibility

The budget of the last product is limited to 200 dollars. There is no engineering cost for project is considered due to be an engineering design course. The cost analysis of the components is listed in detail in Table 4. The research and development costs are not included in the 200-dollar budget but due to accurate planning, the research and development costs are minimized.

Technical Feasibility

The technical aspect of the project includes wireless communication module, motor driving, mechanical design, sensors and microcontrollers.

DC motors with encoder are used in project. The group members are familiar with motor driving due to previous experiences in the Electrical and Electronics engineering department. Also, in any unexpected situation, the group members have the ability to solve the problem and implement it according to last product plan.

Wireless communication module is decided as 3G module. At the beginning of the project, all group members researched the module and got familiar with it. All group members mastered the communication module and the wireless communication module can be used conveniently. Also, the group members studied the other communication protocols, such as Zigbee protocol, for an unexpected situation.

The sensors which are used in project are familiar to all group members.

Raspberry Pi 3B and Arduino Mega are the microcontrollers which will be used in project. Raspberry Pi 3B is using Linux based an operating system which is familiar to group members. Arduino Uno is used at previous projects of group members and Arduino Mega is more advanced version of Arduino Uno. It is programmed in the same manner with Arduino Uno. Due to familiarity with Uno, the group members are easily adapted to Mega environment.

Mechanical design is the most unfamiliar part of the project for group members. All group members are joined the mechanical design process. Due to unfamiliarity and inexperience, the mechanical design is kept as simple as possible. The group members have done research during the design process to solve any unexpected situation.

Operational Feasibility

The project has a reliable design. The components of the projects are tested separately. The design can be produced as a mass product due to its simplicity and expense. The project can easily improvable. The software of the design can be developed in delay reduction, user interface. Also, virtual telepresence can be developed for project. The hardware of the project is designed according to keep it simple principle and the mechanical parts can easily be substituted. The mechanical design also can be easily adapted to any environment by small changes. The 3G module allows design to be used in any place which has access to mobile network and the robot can be controlled from any distance. Due to these reasons, the design has adequate operational feasibility.

Schedule Feasibility

The design and implementation duration of the project is nine months. The group members planned all the process and included all the stages of the project.

According to the progress shown, the plan and the efforts of group members, the project has been fully completed before the demonstration.

POSSIBLE APPLICATION AND ENVIRONMENTAL EFFECTS

The manufactured robot, which satisfies the required conditions specified in the project description, has remote control, moves in every direction and shoot with its kicker and dribbling mechanism.

The most outstanding feature of the robot is remote-control which allows user to control robot without any condition except available 3G connection. Connection range of the robot allows user distance-independent control. This feature can be utilized for such situations that any human can not go or reach. In a fire, the remote-control feature allows firemen to observe the environment and check whether any person is inside. In a ruined building due to earthquake, the needed tools, food and water can be delivered with the help of the remote-control. Also, remote control allows lifesavers to check any place of ruined building to check any sign of life.

Another prominent feature of the robot is the ability to move in each direction. The robot can be utilized to deliver anything that can be attached to the robot. Combining remote control and omni directional feature of the robot, the robot can be utilized previously mentioned situations.

Due to designing a hockey robot, a kicker and dribbling mechanisms are placed on the front face of the robot. These mechanisms can be changed with other mechanisms such as artificial arm which can be utilized for unreachable situations by humans.

The robot can be utilized for defense industry applications with the application-specific changes on the robot.

To reduce negative effects on nature, the batteries of the robot are chosen rechargeable. The design procedure of robot is conducted aiming to reduce plastic usage.

DELIVERABLES

As Poro Inc., when you buy our product, you will receive the following items:

- **Robot:** A tele operated robot that tries to score in opposing team's goal
- **A game ball:** Artengo (Beach Tennis Ball)
- **3 walls of the play field:** Each wall has 30 cm height, 75 cm width and 8 mm thickness.
- **Power Supplies:** A rechargeable battery and a charger
- **Application that includes GUI:** Application with an interface to view the camera of the robot and control the robot

- **Customers service for configuring communication between robot and computer of the client:** Poro Inc. is at your service to make necessary connections between your device and the robot

- **Two years' warranty:** Poro Inc. will be more than happy to fix your problems for free except user faults.

- **User manual:** A user manual that includes the instructions before starting to run the robot.

CONCLUSION

The aim of the project is to find a solution to the aforementioned problem which is building a robot that is controlled from at least 30 meters' distance to perform a play with an opponent robot. In this report we stated the problem with its constraints and requirements. We explained our final solution, in detail. Moreover, we provided test results related with the subsystems, detailed information about team organization, several analysis of the robot concerning its weight, power consumption and cost. We presented our predictions about the possible future application areas of the robot.

The design is composed of four main subsystems, namely, dribbling/kicker module, driving module, vision controller module, communication module, and command module. Each of them is crucial for the design of the robot.

The first subsystem performs the task of controlling the ball and sending it to the opponents' side. This is accomplished by using servo motor, two cylindrical structures that will turn in clockwise direction, and a solenoid-based kicker.

The second subsystem is responsible from the movement of the robot and it consists of four servo motors and four omni wheels. The robot can move in eight different directions. For each direction, the voltage values that the motors' positive and negative poles will receive are arranged.

The third subsystem allows us to see the game field with RPI cam, also it gives us a feeling about the position of the robot in the game field with the help of four ultrasonic distance sensors.

The fourth subsystem is a very fundamental system for the robot. It connects the robot to the outside world and it allows us to see the game field by streaming the video from the cam, get the sensor data from the robot, send the necessary commands for controlling the movements, dribbling and kicking. It consists of 3G modem, several software programs, and a website provided by Dataplicity.

The fifth subsystem allows us to control the robot with a gamepad. This is accomplished by reading the states of the sticks and buttons of the gamepad, i.e., the position of the sticks with respect to the origin and whether the buttons are pressed or not. The proper command data is composed according to these states and sent to the robot. Then the desired action is performed.

There were many things that needed to be done in order for this project to be finished successfully. We have achieved realizing a feasible solution for each problem and the project is successfully finished in time. We have been prepared cost effective and smart solutions for the problems we have faced. While coming up with a solution, our main priority was customer satisfaction, environmental and social concerns. The progress made by the company thus far shows that this group has a promising future and can successfully enter the IoT industry.

APPENDIX

Appendix A – Robot Images

Figure 33. Isometric View of the Robot

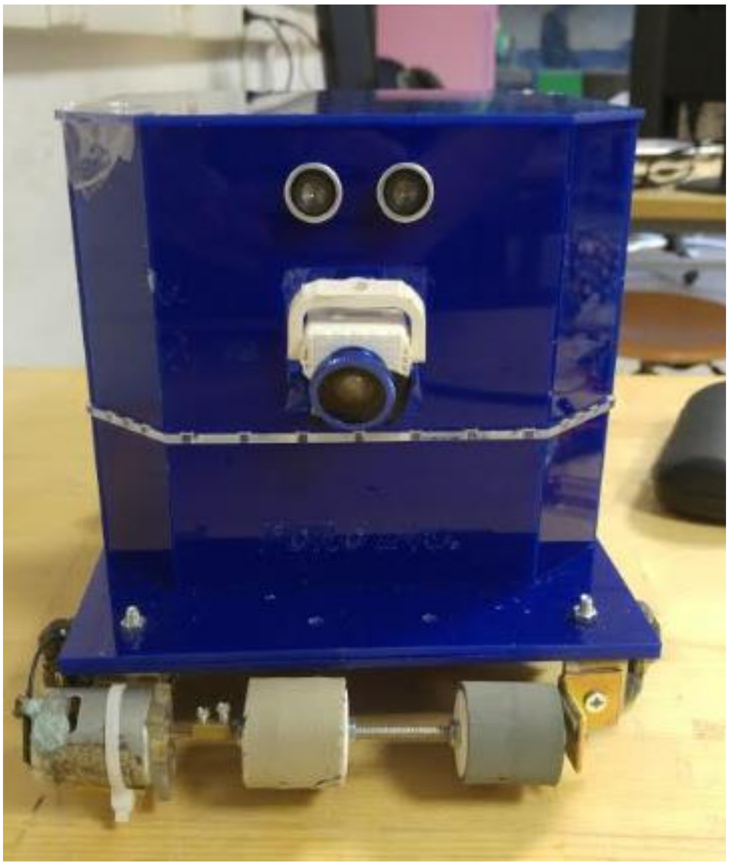


Figure 34. Front View of the Robot

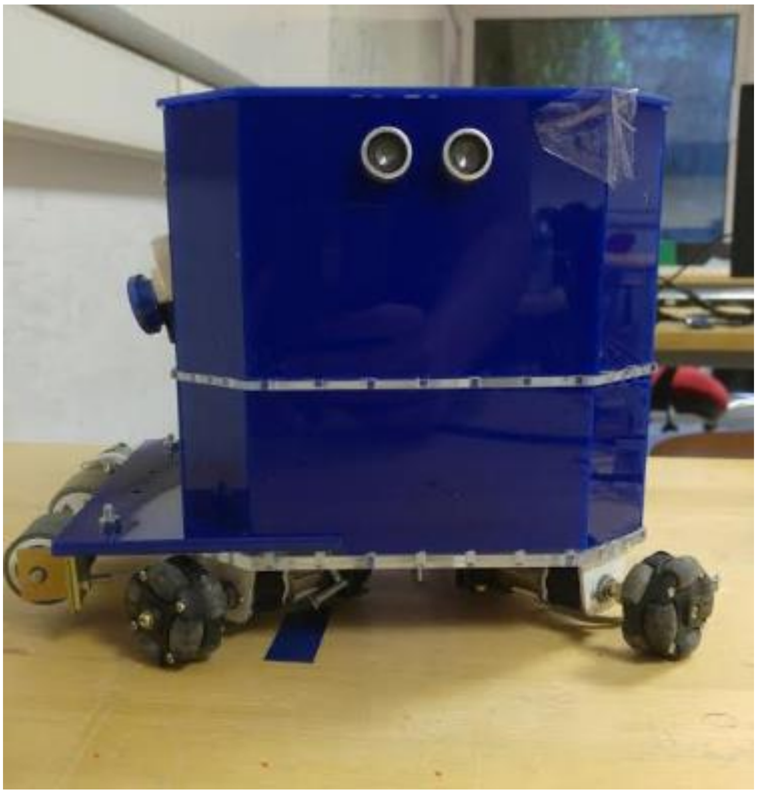


Figure 35. Right Side View of the Robot

Appendix B – User Manual

-Starting Robot

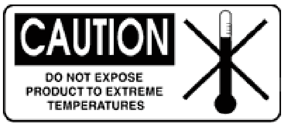
To be able to run the robot properly user should complete the following steps;

- Place the robot on the goal line that is placed in the game field and make sure that the robot does not move.
- Open the top cover of the robot and check 3G USB Modem. Make sure that it is connected Raspberry Pi's USB socket properly.
- Power up to whole system by plugging in the rechargeable Li-Po battery's cable to the system.
- After this step, system will start and connect to the internet automatically in a few seconds.
- Open the Dataplicity Porthole software from your computer.
- Go to our website to reach the interface of the robot.
- Connect your joystick (gamepad) to your computer.
- Now the system is ready. Good luck at hockey game, have fun!

-Charging the Batteries

To charge the Li-Po battery, use the charger that is provided in the box by attaching the Li-Po sockets to the charger carefully. To charge the power bank, use the android charger that is used for mobile phones or just plug it in to the USB socket of your computer.

-Maintenance and Safety

- 

This robot can be used in an environment that has temperature between 0°C and 40°C.
- 

Do not expose the robot to direct sunlight.
- Do not apply excessive force to the chassis of the robot.
- Keep your robot dry. Do not expose it to any fluid or rain water.
- Do not leave the robot and its accessories within reach of a pet or children.
- Disconnect the power cable if you are not using the robot.
-  Do not try to fix or repair the robot on your own. The product has warranty for 2 years.
-  Do not touch or try to put your fingers on the sensors or motor driver and especially on the batteries when the robot is working.
-  Beware of sharp edges of the chassis of robot.

REFERENCES

[1] SparkFun, "General Purpose Type Distance Measuring Sensors", GP2Y0A21YK/GP2Y0D21YK datasheet, Dec. 2009. Available at: https://www.sparkfun.com/datasheets/Components/GP2Y0A21YK.pdf

[2] Simple Electric Motors, "Calculations" [online]. Available at: http://simplemotor.com/calculations/

[3] Open Impulse, "TL1838 Infrared Receiver Datasheet", TL1838 datasheet. Available at: https://www.optimusdigital.ro/index.php?controller=attachment&id_attachment=4