Table of Contents

[1 Description of Individual Subsystems 2](#_Toc533350540)

[2 Restatement of the Problem and Requirement Analysis 2](#_Toc533350541)

[2.1 Functional requirements: 2](#_Toc533350542)

[2.2 Physical requirements 3](#_Toc533350543)

[2.3 Performance requirements 3](#_Toc533350544)

[2.4 Metrics 4](#_Toc533350545)

[3 Main Processor Subsystem 7](#_Toc533350546)

[3.1 Solution and Relevant Algorithms 7](#_Toc533350547)

[3.1.1 Plan A 7](#_Toc533350548)

[3.1.2 Plan B 10](#_Toc533350549)

[3.2 Level Risks Assessment 10](#_Toc533350550)

[3.3 Error Sources 10](#_Toc533350551)

[3.4 Test Results 11](#_Toc533350552)

[3.5 Comparative Analysis 11](#_Toc533350553)

[4 References 12](#_Toc533350554)

# Description of Individual Subsystems

* **Power Supply Subsystem:** This subsystem is for distribute power towards other subsystems at required voltage level and with specified current limitations.
* **Communication and Telecontroller Subsystem:** This subsystem provides communication between the robot and controller subsystem. This system is used for sending commands and sending the data received by robot and detection subsystem.
* **Motion Subsystem:** This subsystem consists of wheels and motors driving these wheels by which we provide capability of motion to our robot in the game field.
* **Detection Subsystem:** This subsystem detects the boundaries of the field, the position of the opponent robot & the ball. This subsystem also locates itself in the field.
* **Main Processor Subsystem:** This subsystem is the main robot computer onboard which controls other subsystem with respect to feedbacks sent from these subsystems.
* **Shooting Subsystem:** This subsystem is responsible for sending the ball to the opponent’s goal with high precision to score a goal.

# Restatement of the Problem and Requirement Analysis

The goal of this project is to design and construct a teleoperated robot (controlled from a distance up to at least 30 meters) which can compete with a similar robot in shooting and scoring to opponent’s goal. This part is restatement of the problem and requirement analysis. We already state the problem for proposal report. This is the updated version of the previous statement and requirement analysis. Functional, physical and performance requirements of the project are as follows:

## Functional requirements:

* Detect the start signal
* Monitor the surrounding
* Process the monitored data
* Encode the processed data for communication
* Transfer the encoded data to the teleoperator

*If the ball is at players half-field and far away from the robot:*

* Transfer the movement direction command given by the teleoperator, to move toward the ball
* Perform the move operation respect to the command transferred from teleoperator
* Move robot to the ball until ball is in the shooting range
* Transfer the hit the ball command given by the teleoperator
* Perform the hit the ball operation given by teleoperator

*If the ball is at opponent’s half-field:*

* Transfer the movement direction command given by the teleoperator, to cover the goal, given by the teleoperator
* Perform the move operation respect to the command transferred from teleoperator
* Move robot to the own goal to protect it from the incoming shoot
* Protect the goal respect to the commands from teleoperator

## Physical requirements:

* Goals must be at least twice as wide as their defenders’ lateral dimensions.
* Robots can hit, push or otherwise drive the ball but not grasp, scoop or otherwise carry it. So, robot cannot have a grasping or scooping part.
* The playfield also has some physical requirements:
* The playfield should be regular hexagon on a bare floor, with center-line and goal lines marked by “masking tape”.
* The playfield constructed from 6 sidewalls of 70-75 cm length each and two goals snugly fit at the opposite corners, while preserving symmetry.
* The robot should fit in a cylinder to measure the maximum dimension.
* In order to move fast enough our motors should carry the weight of the remaining parts.
* Weight should be aligned in center for steady and controllable movement.

## Performance requirements:

* Ball should be transferred to opponent’s half-field in no more than 20 seconds from the project description. For our team for successful operation it should be less than 15 in order not to have any unforeseen violations. So, consistency of operation in this case is important. It should transfer the ball to the opponent field less than 15 seconds at least 55% of the time and other times it should be less than 20 seconds because of project description.
* The operator remotely controls the robot from a distance up to at least 30 meters.
* The transmission between the robot and receiver should be less than 1 second for successful operation. So, less than 4 points for transmission delay objective is unacceptable.
* In order to have successful operation aiming precision of the robot should not be less than 45%. So, less than 4 points for precision objective is unacceptable for our company.
* The robot’s throwing mechanism should be strong enough to throw the ball to opposing goal.
* The robot’s structure should not be affected by hitting the ball fast to the robot, so, durability of the robot cannot take less than 4 points.

**Constraints:**

* The robot is not allowed to cross the center-line
* The ball must be transferred to the opponents half field no more than 20 seconds
* The communication between robot and teleoperator must happen from a distance at least 30 meters
* Carrying, grasping and scooping the ball is not allowed. Robot can only hit or push the ball.

Objectives, metrics and related objective trees are given in the next section of this report.

## Metrics

**Budget:** Money which is considered to be spent in order to develop and produce the final project.

In metrics, 10 points will be given for budget considered to be less than 200$, 8 points will be given for budget considered to be between 200$-250$, and 6 points for the budget between 250$-300$, 4 points for 300$-350$, 2 points for 350$-400$ and no points for budget 400$+.

**Fun:** Measure of how much each shareholder enjoyed from performing tasks required to complete the project.

In metrics, 2 points will be given for each shareholder who would enjoy.

**Performance:** Parameters which are considered as important. These can be investigated in three sub-categories:

**Durability:** Robots ability to preserve its structure and function against external impacts and wear down.

In metrics, 10 points for preserving its structure against the effects caused by collision to wall with a speed of 1m/s and no points for not preserving its structure against the effects of collision to the wall with a speed under 0.1m/s. The points between is distributed linearly with respect to speed.

**Consistency:** Robot’s ability to execute the same performance under different conditions (starting position, ambient lighting, ambient temperature) using the same line of commands.

In metrics, 10 points for similarity of behavior over 95%, 8 points for similarity of behavior between 85%-95%, 6 points for similarity of behavior between 75%-85%, 4 points for similarity of behavior between 65%-75%, 2 points for similarity of behavior between 55%-65% and no points for similarity of behavior under 55%.

**Power Consumption**: How long the robot would last on same battery capacity of 1750mAh.

In metrics, 10 points for operation over 2 hours, no points for operation under 20 minutes. The points between will be distributed linearly.

**Transmission Delay:** Measure of how many second delay of the transmission will the solution have.

In metrics, 10 points for operation less than 100ms delay, 8 points for 400ms delay, 6 points for 700ms delay, 4 points for 1s delay, and more than that will be failure.

**Creativity**: Measure of how many different solution approaches can be proposed.

In metrics, 2 points will be given for each proposed solution and 1 point will be given for each proposed sub-solution.

**Precision**: Measure of how accurate robot will aim the opponent’s goal.

In metrics, 10 points for more than 90% precision, 8 points for precision rate between 90%-75%, 6 points for precision rate between 75%-60%, 4 points for precision rate between 60%-45%, the other points will be given accordingly. Less than 4 points is a failure because of our performance requirement.

Table 1: Pairwise comparison chart for top objectives

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Budget** | **Performance** | **Fun** | **Creativity** | **Precision** | **Total** | **Weighted** |
| **Budget** | - | 0 | 0 | 1/2 | 0 | 0,5 | 0,05 |
| **Performance** | 1 | - | 1/2 | 1 | 1 | 3,5 | 0,35 |
| **Fun** | 1 | 1/2 | - | 1 | 1/2 | 3,0 | 0,30 |
| **Creativity** | 1/2 | 0 | 0 | - | 0 | 0,5 | 0,05 |
| **Precision** | 1 | 0 | 1/2 | 1 | - | 2,5 | 0,25 |

Table 2: Pairwise comparison chart for performance objective

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Durability** | **Consistency** | **Power Consumption** | **Transmission**  **Delay** | Total | Add 1 | Weigthed |
| **Durability** | - | 1/2 | 1 | 0 | 1,5 | 2,5 | 0,25 |
| **Consistency** | 1/2 | - | 1 | 1/2 | 2,0 | 3,0 | 0,30 |
| **Power Consumption** | 0 | 0 | - | 0 | 0 | 1 | 0,10 |
| **Transmission**  **Delay** | 1 | 1/2 | 1 | - | 2,5 | 3,5 | 0,35 |



Figure 1: Weighted objective tree for device trying to score in each other's goal

# Main Processor Subsystem

Processor subsystem consists all the processors in different subsystem.

## Solution and Relevant Algorithms:

A processor is needed to identify the command and give an output respectively. We have already tried different processors for this task. At first, we tried TIVA Board for identifying the frequency. It worked but with TIVA board we needed another module of PWM converter. So, we search for different and easier ways to identify a signal. For the demonstration we used Arduino MEGA. For a cheaper option we also tried Arduino UNO but for demonstration it failed to give enough current for our system.

### Plan A

Our main plan for main processor subsystem is to use Arduino Mega. Figure 1-2 below shows an Arduino Mega and its specifications.



Figure 1: Arduino Mega [1]



Figure 2: Arduino Mega specifications [1]

Our main plan is to use Arduino Mega for main processor subsystem it is chosen because it has better DC current than UNO. Also, Arduino has a variety of online sources that are easily reachable which makes it easier to use from alternatives like TIVA. The Figure below shows the code for code for receiving sine input and identify the frequency, then output from different GPIO pins.





Figure 3: Code for GPIO control of Arduino

From the Figure 3 above it can be seen that it is easy to program Arduino. This code both reads the input and give output from different ports. Output of this system will be connected to the motors of motion subsystem.

### Plan B

Our plan B for this subsystem is to use another processor for example TIVA board, Arduino Uno or Raspberry Pi can be used for this purpose. All of these have different advantages and disadvantages. These are going to be covered in 3.4 Comparative Analysis section. TIVA board has already been tried for frequency control and outputting. Arduino UNO works with the same principle as Arduino Mega there is no need to change the code above in Figure 3. In addition to that Raspberry Pi also has GPIO ports that can be used for this purpose but it is not as efficient as Arduino Mega.

## Level Risks Assessment

There are few risks of using the above-mentioned processors.

* The first risk is these systems heats a lot which can create a risk for other parts of the system.
* Processors are one of the heaviest part of our whole system, so they can create alignment problems.
* Processors have their built-in fault algorithms which causes them to shut down at unforeseen situations.

## Error Sources

Some internal and external sources can cause an error at this subsystem. The possible error sources are as follows:

* Environmental temperature conditions can be a source of error. To eliminate this effect heat sink or fan can be used.
* Overflow can be an error caused by increasing the current coming to the system. To eliminate this a voltage comparator can be used so that we should be sure that we are not feeding the systems a voltage higher than 3.3V.
* For higher frequencies sampling rate can cause discrepancies. To eliminate this kind of error we are going to work on comparatively lower frequencies.

## Test Results

We conducted a test to find error rate of the Arduino Mega frequency counter. We experimented the frequencies between 0-20kHz. The results in Figure below showed approximately 2% error for these frequencies which is acceptable in many cases.

Figure 4: Frequency vs Error Rate for Arduino Mega frequency counter

## Comparative Analysis

In order to choose our main solution, we compared possible solutions. Comparative analysis can be seen in Table 1 below.

Table 1: Comparative analysis for main processor

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Arduino Mega | TIVA Board | Arduino Uno | Raspberry Pi |
| Price ($) | 38.5 | 14 | 20 | 5-35 |
| Analog compatibility | Yes | No | Yes | No |
| Max DC current(mA)(per GPIO) | 50 | 40 | 40 | 40 |
| Weight(g) | 37 | 20 | 25 | 23 |

Even though, Arduino Mega has the worst values for price and weight it can has higher maximum DC current rate and analog compatibility. Analog compatibility and DC current are the most important specifications for selection, so we selected Arduino Mega even though it is the heaviest and priciest option.

# Test Procedure

Test procedure mentioned below consists all the subsystems rather than the ones in demonstration. It can be seen as a User Manual for end user. Test procedure is as follows:

1. All switched should be in off position.
2. Li-po connections for data receiver and robot should be made.
3. Powerbanks should be connected.
4. All switches and powerbanks should be on for now on.
5. At this part we should observe some fan actions on the robot and receiver which are located for cooling purposes.
6. Moreover, video or image data transmission should be visible at this stage if not check the previous steps.
7. From now on robot should be controllable by telecontroller.
8. Start the test procedure when robot and telecontroller at the same place. (Approximately 0 m distance)
9. The buttons on telecontroller should create forward, backward, left and right movements. Also, not decided yet if any, shoot button controls the shooting subsystem.
10. Different combination of pushes will be tried during the process.
11. The steps 9 and 10 will be tried for different distances until the data transmission and command transmission both end.
12. At the end of the process it is necessary to wait at least 30 seconds without pushing any button in order to be sure that all motor activity ended.
13. The robot lifted up and switches should be off from now on.
14. Li-po batteries will be disconnected from the system for safety.

# References

[1] Store.arduino.cc. (2018). *Arduino Mega 2560 Rev3*. [online] Available at: https://store.arduino.cc/usa/arduino-mega-2560-rev3 [Accessed 23 Dec. 2018].