MIDDLE EAST TECHNICAL UNIVERSITY

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

EE 494-DESIGN STUDIO 2

FINAL REPORT OF GIMME FAST PROJECT

Company Name: Revolutionary Systems Inc.

Date : 31.05.2020

Group Members: **Onur Akdeniz**

Doğukan Atik

Ozan Berk Boyraz

Ahmet Demirdaş

Mert Eyüboğlu

Table of Contents

1. Introduction 3
2. Crisis Management Plans 3
2.1. Current Status 3
2.1.1. Current Status and Technical Capability 3
2.1.2. Top-Down System Description of the Overall Project 4
2.1.3. Logistics About Subsystems 5
2.1.4. Status of Subsystems 5
2.2. Change of Plans 6
2.2.1. The Tasks Completed Under Isolation Conditions 6
2.2.2. The Uncompleted Subsystems due to Emergency Situation 7
2.2.3. The Creative Solutions Regarding Uncompleted Subsystems 8
2.2.4. Subsystems Integrated Remotely 9
2.2.5. Detailed Breakdown of the Responsibilities 10
2.2.6. Demonstration Method 10
3. Termination of the Project 11
3.1. Obtained Results 11
3.2. Deliverables 14
3.2.1. Gimme Fast Simulator Graphical User Interface 14
3.2.2. Deliverables in Communication Subsystem 15
3.2.3. Deliverables in Transportation Subsystem 21
3.3. Cost Breakdown 22
4. Conclusion 23
5. Appendices 25
Appendix 1 : State Diagram of the Overall System 25
Appendix 2: The Flows of the Simulations 26
Appendix 3: The Objectives of Solution and Their Measure of Success 39

1. Introduction

As it is the nature of the engineering profession, engineers should assess risks and cope with extraordinary situations. In the spring term of 2019-2020 academic year, because of Covid-19 pandemic, Revolusys Inc. had to adapt to the new conditions in which group members have to realize their responsibilities in “Gimme Fast” project by fully isolating themselves from each other physically.

After serious considerations and reviews in several group meetings, Revolusys Inc. made many changes on the plans and the deliverables of the Gimme Fast project. Before the pandemic changed the normal life of the group members, the subsystems of the project had been completed and these were to be integrated. In addition, for all the subsystems, test plans were applied. In the final stage, group members had to get together and fully integrate the subsystems in their responsibilities. However, integration part couldn’t be achieved because of extraordinary situations.

In this report, our purpose is to present our crisis management plan and to show the results we have taken with the termination of the project. Firstly the current status of project components will be clearly stated. Following that, the changes in the plans will be specified with the required explanations. Finally the results and the deliverables with a guideline will be given with a detailed cost breakdown.

2. Crisis Management Plans

2.1. Current Status

2.1.1. Current Status and Technical Capability

Due to the Covid-19 pandemic, Middle East Technical University decided to continue its education via distance education. Naturally, this unexpected situation also affected EE493-494 Capstone Design project. During the pandemic, location of the team members is listed below:

Name of the Member	**Location**
Ahmet Demirdaş	Kayseri
Doğukan Atik	Ankara
Onur Akdeniz	Trabzon
Mert Eyüboğlu	Ankara
Ozan Berk Boyraz	Tokat

All the group members had their own smartphones and computers with internet connection. Although two of the members’ internet connection in their home are weakly supplied, it can be said that the group had the video/audio conferencing capability. The Zoom environment is tested and found suitable for the purpose. Also, weekly meetings with the DS Coordinator, Dr. Emre Özkan conducted via Zoom.

But, some of the equipments and items required for further studies were not reachable (oscilloscope, physically guided track etc.). Handling the hardware related tasks individually seemed to be challenging for Revolusys due to lack of equipment. Also, going out and buying some hardware was required which not possible in the outbreak conditions. In addition to all these, the integration of the subsystems without any group meetings was even harder.

After the decision of the DS Coordinators about the course, Revolusys Inc. decided to fully complete the software related parts of the project and utilize the previous experimental results for the simulation of the completed part under different conditions. And Revolusys aimed to "transfer the image successfully with the UART even under artificially-created error environment''.

For this purpose, required steps for achieving the aim were :

- UART Protocol implementation and optimization for efficient communication (assigned to Doğukan)
- Checksum Error Checking method implementation (assigned to Ahmet)
- Tests to be conducted for capturing and transferring a whole photo (assigned to Doğukan and Mert)
- Graphical User Interface for the screen at the transmitter terminal (assigned to Ozan)
- Test for vehicle operation and performance results (assigned to Mert)
- Transferring the previous experimental results and extracting the error for the implementation of the project on the virtual environment (Onur)

At the end of the given 7-week period, Revolusys proudly declares the full achievement on its goals. Revolusys is ready to demonstrate the image transfer on the virtual environment with proper error correction mechanism. Also, Revolusys presents a quick vehicle which can be usable for further implementation conditions.

2.1.2. Top-Down System Description of the Overall Project

State diagram of the overall system is given at the Appendix I. In the idle state of the system, the transmitter terminal is waiting for the start signal which is an on-off button. After taking the start signal the system waits for the camera taking photo. Next state is the compression of the image and its division into data chunks. The length of the chunks' is 10 kB.

Handshaking between transmitter and vehicle is waited to be completed. Here UART protocol is used with proper handshaking signals for the data transmission. If handshaking is done, data transmission starts. The data packet to transmit is also sent to vehicle in packets such that each packet is checked for error via Hamming method. If there is any error, the communication for the related chunk of data is retransmitted. After data transmission is done, vehicle is initiated for transportation to the receiver terminal. Here, the critical part of the transmission is the ultrasound sensor which will stop smoothly at a distance of 5-10 cm to the terminals on a distance- convertible rail. After reaching the receiver terminal, handshaking protocol is started via sending preamble signal to the receiver terminal.

Following the preamble signal, data transmission is initiated. Again, the data is sent by portion by portion, checking each portion via Hamming method. If any error is met, the transmission of that portion is canceled and the related portion is resent. The acknowledgement of the fallacies in data transmission can be solved by using a led at the receiver side so as to acknowledge the transmitter. After data transmission is done the packet end signal is generated by the transmitter. If unique end signal is generated, it means that all the data packets are sent and the image data is ready for the image reconstruction submodule. If unique end signal is not generated, then vehicle is transported back to the transmitter terminal in order to get the new data packet. Meanwhile, the data chunk on the vehicle is deleted in this phase.

After taking the unique end signal by the receiver terminal, the image file in binary form is merged into a file and then JPEG decompression is operated on the file. Finally, the image is sent to the display part and the photo is displayed on the screen.

2.1.3. Logistics About Subsystems

During the distance education conditions, Revolusys had some of the components to conduct the word, on the other hand, some of them were missing/non-reachable/ needs to be manufactured.

<u>Reachable Hardware Components</u>

- Raspberry Pi 3 and Zero for implementation of transmitter and receiver modules
- Raspberry Pi Camera Module for image capture
- A vehicle with Arduino Uno and DC Motor Set

<u>Hardware Components which is not Reachable</u>

- Transmitter Circuit
- Receiver Circuit

<u>Components Need to be Purchased or Manufactured</u>

- Physically Guided Track (U- Type Aluminium Profiles)
- Physical Construction of Transmitter and Receiver Terminals

2.1.4. Status of Subsystems

Detailed information about the status of the subsystems is given in the following sections. In below, status of the subsystems are presented summarily.

- **Communication Subsystem:** This subsystems consists of hardware and software parts. Software part of the communication subsystem is completed successfully. For the hardware part, Revolusys has a working transmitter and receiver circuits. But due to aforementioned reasons, they are not integrated to each other.

- **Image Acquisition & Reconstruction Subsystems:** This systems are completed successfully. Capturing image, storing,resizing and division of captured image realized without any problem for image acquisition. Also, image reconstruction system collects and merges the data and finally displays the reconstructed image.

- **Transportation Subsystem:** For the transportation subsystem, vehicle is constructed to accomplish back and forth movement as fast as possible with its sensor data.But due to aforementioned reasons, integration with the physically guided track and vehicle could not completed.

2.2. Change of Plans

2.2.1. The Tasks Completed Under Isolation Conditions

The status of the project at the beginning of the isolation period is briefly explained in the following paragraphs.

- In the communication subsystem, the designed circuit was successful at transmitting one packet of data between two Arduino devices. Note that full photo transfer was not tested and the error correction code was not completed yet. Also, no test was conducted using Raspberry Pi devices regarding communication subsystem.

- In the transportation subsystem, the several tests were conducted on the ultrasound sensors and vehicle. The vehicle was able to move back and forth between two obstacles. The power unit of the vehicle and the rail which is the guided track of the vehicle was not implemented yet.

- In image acquisition and image reconstruction subsystems, an image was successfully divided and merged again using Raspberry Pi camera module and Raspberry Pi devices. The graphical user interface was not implemented yet.

- The terminals of VLC system was designed and technical drawings of them were prepared but physical implementation was not completed yet.

In the following paragraphs, the tasks completed under isolation conditions are given and explained briefly. The detailed information on the completed and uncompleted parts of the subsystems are given in the subsection 2.1.3 (Status of the Subsystems).

The Communication Subsystem

- **The implementation of UART communication protocol and optimizations on it:** A serial communication protocol is implemented. This protocole organizes the states of the terminals and vehicles and data transfer between them.

- **The implementation of checksum error checking algorithm:** An error correction algorithm is implemented so that the errors happens in VLC can be corrected.

- **The search on improvements on the circuitry:** A market search is conducted on the voltage regulators and comparators to improve the performance of the VLC circuitry. The details of these searches can be found in the Github (EE493-Revolusys). Note that it was not possible to add these components to circuitry due to reasons explained in the subsection 2.2.2 (The Uncompleted Subsystems due to Emergency Situation).

The Transportation Subsystem

- The test of the vehicle are conducted to ensure it meets the speed requirements. Also, the average braking distance of the car is checked in order to fine to the distance to start braking so that the vehicle stops 5 to 10 cm away from the terminals.

The Image Acquisition & The Image Reconstruction Subsystems

- **The implementation of graphical user interface:** A graphical user interface that will guide the user while taking photo in the transmitter terminal is implemented.

2.2.2. The Uncompleted Subsystems due to Emergency Situation

In the following paragraphs, the uncompleted tasks regarding subsystems and integration are given. The reasons behind the failure of completion of each uncompleted task are explained in detail.

The Communication Subsystem

- **Test of the circuit with its final design:** To change any components in the circuitry or perform any test on it, we had to acquire an oscilloscope. Any test or change on circuit without using an oscilloscope is likely to ruin the sensitive tunings of the circuitry. Also, to detect the source of any error in a circuitry, oscilloscope is a great help. The Revolusys team does not own any oscilloscope. Throughout the year, the team was using the oscilloscopes placed in the E block lab which is locked. However, even if there was an oscilloscope we can use, we would need to gather up as a group to analyze and discuss the circuitry. Otherwise, under isolation conditions, only one group member needed to take the entire hardware implementation and therefore, it was not feasible.

- **Duplication of the circuit:** At the beginning of the isolation period, we had one transmitter and one emitter circuitry, i.e. one pair of communication circuitry. This circuitry was built on breadboard. The next item in the agenda of the Revolusys team regarding circuitry was duplicating the circuitry build on breadboard to perforated pertinax boards to produce 8 pairs of communication circuitry. Each duplicated circuitry needs to be tested to check whether it is operating correctly or not. If there is an error in the circuitry, it is required to perform further detailed tests on it and correct it if possible. All these processes requires an oscilloscope to be used. Due to reasons explained in the previous article, Revolusys team do not have access to an oscilloscope. Therefore, completing the duplication of circuitry was not possible.

The Transportation Subsystem

- **Construction of the rail:** The construction of the rail requires shopping. At the beginning of the isolation period, Revolusys team was about to complete its market search regarding material and price on the rails and the wheels of the vehicle. As the as department stores are closed due to the ongoing coronavirus pandemic. Due to this, we couldn't search for and purchase the appropriate hardware for the construction of the rails, thus couldn't complete the construction of the rail.

- **Implementation of a power supply unit for the vehicle:** We haven't purchased a battery to power up the vehicle before the pandemic. We weren't able to buy one due the lockdown and had to leave that task to the next team. However, the requirements for the battery are presented in the Deliverables of the Transportation Subsystem part.

The Terminals of the VLC

The terminals are structures which contains circuitries, camera or display, power supply units and housing. Its design was completed and its technical drawings were prepared when the isolation period began. The construction process of the terminals requires equipments for drilling, cutting etc. which Revolusys team does not own. Also, power supply unit was not possible to implement due to same reasons in the transportation subsystem. Therefore, the construction of the terminals is not completed.

The image acquisition and the image reconstruction subsystems are fully completed therefore there is no task to mention regarding them in this subsection of the report.

In this section, the integration process is not examined. The integration process and the justifications regarding the uncompleted integration tasks are addressed in the subsection 2.2.4 (Subsystems Integrated Remotely).

2.2.3. The Creative Solutions Regarding Uncompleted Subsystems

As given in the previous subsection with justifications, several parts of subsystems are not completed. Therefore, Revolusys team is not able to present a complete product to its customers at this stage. Yet it is necessary to demonstrate the value of the completed work. For this purpose, a simulation on the computer environment is designed and implemented.

This simulation will be constructed using the following data and tools:

- The experimental results of transmitter and receiver circuit as the environmental light condition and the distance between receiver and transmitter vary at specific communication speed i.e. baud rate
- The experimental results of transportation subsystem; the speed of the car and the variation of the stopping distance of the moving vehicle (using sensor test data)

The simulation process is as follows: Upon capturing picture via image acquisition subsystem, it will be transferred to the communication subsystem. At this stage, a random error will be created depending on the selected environmental light condition and the distance between transmitting terminal and the vehicle (note that this distance is variable since there is small amount of variation at the stopping distance of moving vehicle). The communication between two terminals will be done directly from Raspberry to Raspberry. At the receiving end, error correction algorithm will work. After all data is successfully transferred and displayed, the communication time will be added to the transportation time calculated using speed of vehicle and the distance that vehicle should pass along the way therefore we will be able to see whether the process can be completed under 2 minutes or not.

Therefore, we replace the physical part of the communication subsystem (since physical communication subsystem is not completed yet) via adding an error rate to the VLC and we replace the transportation subsystem (since transportation subsystem is not physically integrated to other parts of the project yet) simple by adding it as a time consuming element in the simulation.

The results of one run of the simulation can be seen in subsection 3.1 (Obtained Results).

2.2.4. Subsystems Integrated Remotely

If all the subsystems could've been completed properly, there would be 3 main bodies to be integrated. Namely; transmitter terminal, vehicle and the receiver terminal. For the integration, alignment of the LED's on the transmitter parts with the photodiodes in the receiver parts of platform was very critical for the accuracy and the speed of the communication subsystem. To achieve the proper alignment, all the physical bodies must've been produced appropriately. It was experienced that in some cases we did not have enough equipment for cutting, drilling etc.

To test the overall system and for the optimization of it, physical integration is one of the musts. And it is proved by experience that this process needs at least 2 group members.

However, instead of physical integration, the group could work on software integration. In software integration, the image acquisition and reconstruction subsystems are integrated with the communication subsystem.

i) Software part of the Communication subsystem is successfully implemented. At each terminal, Raspberry Pi device is utilized as controller. Since the group didn't have an oscilloscope to optimize the photodiode and LED driver circuitries and to get necessary measurements, simply cable is utilized between terminals. However the group took results from the tests of the related circuitries by sending and receiving meaningful dataset before the pandemic started. So in software part, we benefited from the results taken such that the error rate in communication calculated is utilized in the code by generating error with the same rate. The error detecting method is internet checksum algorithm. It is also implemented in communication subsystem.

ii) In the image acquisition subsystem, the photo is taken via the camera and the taken photo is sent to the compression method implemented on the software. After this stage, photo is divided into the chunks, and then to packets (at most 1 kB length). In image reconstruction subsystem, the packets past through the error detection process successfully are concatenated one another. After the communication process is done, the image file is shown on the monitor.

The software written for both subsystems mentioned above are implemented in the same main loop. Namely, communication protocol for transmitter is integrated with image acquisition software in the same main loop and communication protocol for receiver is integrated with image reconstruction software in the same main loop.

2.2.5. Detailed Breakdown of the Responsibilities

Table 1: The Duties and the Responsible Members

Main Duties	Responsible Member
Integration of the vehicle and performing transportation related tests	Mert Eyüboğlu
Image Compression and Reconstruction Unit Optimizations	Onur Akdeniz
Establishment of the Communication Protocol between Terminals	Doğukan Atik
Error Checking Algorithm Implementation	Ahmet Demirdaş
Graphical User Interface Implementation	Ozan Berk Boyraz

The main duties and the responsible members are as written in the table above. In addition to them, for simulation, error generating code is implemented by Onur Akdeniz. It should be taken into the consideration that group members have always been in touch due to Zoom meetings. By doing small scale know-hows, group members share their experiences and knowledges with each other to complete the project steps successfully.

2.2.6. Demonstration Method

To demonstrate the final product, it is decided to prepare a video to show the communication process between two terminals. In the demo video, a chronological flow just like the simulation results in Appendix 2 will be obtained.

The audience will first meet the GUI named as Gimme Fast Simulator. In the interface, there are tabs designed for user to change the parameters like the environmental condition (light or dark conditions), distance between terminals (50-150 cm), desired distance for the vehicle to stop (5-10 cm). The communication process will begin following the group member's clicking on the "Send Image" button. The flow scheme will be generated during the process and the audience will be able to watch it. After the completion of the process, the received image will displayed.

To demonstrate the transportation subsystem, the vehicle's motion and its stopping at desired distances will be recorded and shared in the video.

3. Termination of the Project

3.1. Obtained Results

Test of Proximity Sensor

For testing the HC-SR04 ultrasound proximity sensor, distance isfixed and for that fixed distance, 100 measurement samples are taken (there is 250 ms time delay between each sample.) After measuring for that particular distance, mean and standard deviation of the measurements are calculated. Then, distance is changed, and same procedure is repeated. Distance values are indicated as follows: 3 cm, 5 cm, 7 cm, 10 cm, 15 cm, 20 cm, 30 cm, 50 cm and 150 cm.

This test investigates the precision of the measurement for various distances and high precision is expected for 5-15 cm range.

Table 2: Mean and Standard Deviations of the Measurements w.r.t. different distances

	3cm	5cm	7cm	10cm	15cm	20cm	30cm	50cm	100 cm	150 cm
Mean of the Measurements (cm)	3.80	4.87	6.82	10.33	16.60	20.39	29.56	49.08	97.68	147.49
Standard Deviation of the Measurements (cm)	0.04	0.05	0.05	0.22	0.12	0.07	0.13	0.11	0.41	0.41

As it can be seen from Table 2, sensor does not give reliable results below 5 cm and above 100 cm, yet, standard deviations are not too high. Also, as stated before in the Test Procedures section, high precision is expected for 5-15 cm range and sensor gives good result at 7 cm. Therefore, using ultrasonic sensor is good choice for distance measurement.

- **Test of Vehicle's Movement**

For the movement of the vehicle we had two specifications which are to stop at least 5 cm away from the terminal and to have a minimum average speed of 25 cm/sec for the case where the terminals are 1.5 meters apart. We have ensured that the first specification is satisfied by putting sticks 5 cm away from each terminal and checking if the vehicle hits them by running the car back and forth between the terminals for various distances between the terminals. In order to check the average speed requirement, we have recorded the elapsed time for the vehicle to complete five laps between the terminals for the terminal distances of 150 cm , 100 cm and 50 cm. This is repeated ten times at each distance and the average of the total durations to complete five laps are used to compute the average speed. It should also be noted that the distance covered by the vehicle is shorter than the distance between the terminals due to the length of the car and the fact that it starts and stops 5 cm away from each terminal. We have also took this into account in our calculations and used the actual distance travelled by the vehicle. In addition to this, of course the vehicle stops some time upon its arrival to any terminal, but we kept that time minimal and knew that the actual

average speed might be slightly higher than what we have calculated. The obtained results of the speed test are presented in Table 3.

Table 3: Results of the average speed test of the vehicle.

Distance Between Terminals	One Way Distance to be Covered by the Vehicle	Total Distance covered in 5 Laps	Total Duration of 5 Laps (Average of 10 trials)	Average Speed
150 cm	120 cm	1200 cm	19.52 sn	61.5 cm/sec
100 cm	70 cm	700 cm	13.95 sn	50 cm/sec
50 cm	20 cm	200 cm	9.18 sn	22 cm/sec

According to the results in Table 3 the average speed of the vehicle when the terminals are 150 cm apart from each other is 61.48 cm/sec which is much greater than our requirement. It should be noted that although the average speed of the vehicle drops as the terminals gets closer, the total time is actually getting shorter as a less ground needs to be covered. Thus, we can confidently claim that our vehicle meets the requirements leaving more than enough time for the visible light communication so that that the whole transmission is completed under 2 minutes.

Simulation Results

A simulation is prepared to demonstrate the work completed by the Revolusys team. As described in the subsection 2.2.3, experimental results of the completed and partially completed parts are used in the modelling of the VLC in computer environment. These are briefly listed below.

- The sensor experiment data is used to estimate real stopping distance for the intended stopping distance.
- The communication circuitry experiment data is used to estimate the error rate under different environment conditions for real stopping distance.
- The experimentally obtained lap times of the vehicle for different distances between terminals are used to estimate the time consumed for the transportation in the whole process.
- The communication speed is 19200 baud per second (equivalent to 1920 byte per second). This fact is used to estimate the time consumed to transmit one packet. The calculation is given below.

$$For\ each\ packet, 1024\ byte\ (for\ photo)\ +\ 5\ byte\ (for\ checksum)\ =\ 1029\ bytes.$$

$$Ignoring\ other\ communication\ signals\ (used\ for\ protocol):$$

$$1\ packet\ time\ = \frac{packet\ size(bytes)}{communication\ speed} = \frac{1029\ bytes}{1920\ bytes\ per\ second} = 0.536\ second$$

The simulation is run for different conditions to observe the performance of the system under different conditions and photo is successfully reconstructed in the receiver part. The flows of these runs are given in the Appendix 3: The Flows of the Simulations" subsection. The results of the experiments are given in the Table 4.

Table 4: The simulation results for different conditions

Run No	Conditions	Total Time
1	environment = dark stopping distance = 10 cm distance between two terminals = 150 cm	65.98 s
2	environment = light stopping distance = 10 cm distance between two terminals = 150 cm	65.98 s
3	environment = light stopping distance = 10 cm distance between two terminals = 150 cm * Error rate is multiplied by 10, to observe the effects of high error rate.	83.14 s
4	environment = light stopping distance = 10 cm distance between two terminals = 100 cm	61.50 s
5	environment = light stopping distance = 10 cm distance between two terminals = 50 cm	57.74 s
6	environment = light stopping distance = 5 cm distance between two terminals = 150 cm	63.84 s

As seen in the Table 4, the Revolusys team is able to transfer a photo between two terminals using VLC. Using rubric given in Appendix 3: The Objectives of Solution and Their Measure of Success, the results can be evaluated as follows:

Performance Objectives:

Speed:

• 5 points – Operation time less than 60 seconds
• 4 points – Operation time between 60 and 75 seconds
• 3 points – Operation time between 75 and 90 seconds

In the normal cases the operation time varied between 57.74 s and 65.98 s. Therefore, product takes 4-5 points from speed objective depending on the conditions. In an extreme case as in Run 3, this point drops to 3.

Accuracy:

• 5 points – The displayed image has 100-85% accuracy

The photo is transmitted with 100% accuracy, therefore the product takes 5 points from accuracy objective.

Marketability Objectives:

Cost: No relevance with the simulation.

Immunity to Variable Light Conditions:

• 5 points – Works under sunlight.
• 4 points – Works under indoor lightening.

The test results are obtained for dark and indoor condition and simulation is prepared according to these test results. The product take 4 points from immunity to variable light condition objective. No test is conducted under sunlight yet.

Robustness: No relevance with the simulation, since these results are not obtained for the physically integrated project.

3.2. Deliverables

3.2.1 Gimme Fast Simulator Graphical User Interface

Gimme Fast Simulator GUI provides access to the user for controlling image capturing features. User can see the real-time camera output on the window. User can capture image and re-take the image if necessary.

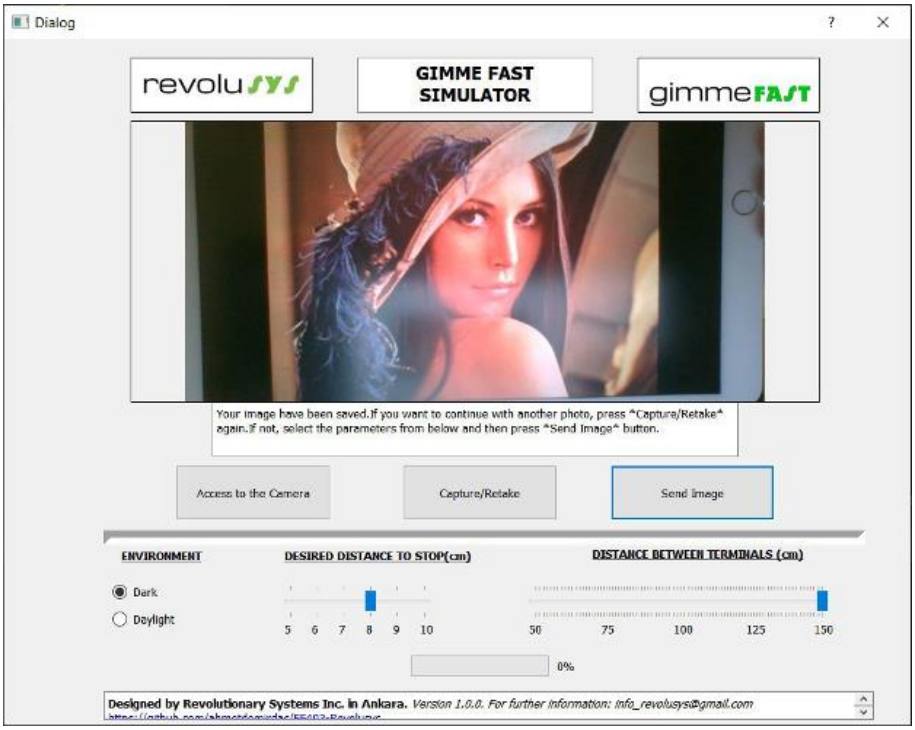
Some important parameters for the actual project were:

- Environment Conditions
- Stopping Distance of the Vehicle
- Distance Between Terminals

Based on the experimental results of the Revolutionary Systems Inc. , error with respect to the different parameters extracted. In this GUI, user can simulate the system with the help of previous experimental results. User can set:

- Environment : Dark/Light
- Desired Stopping Distance: 5 cm to 10 cm with 1 cm steps
- Distance Between Terminals: 50 cm to 150 cm with 2 cm steps

Also, GUI has designed with “Touch Screen Compatibility which can be very useful for further stages of the project like “integration with the touch screen’’. In other words, it does not need any keyboard to enter the input. There are “ slide-bars’’ and “ wide push buttons” which provide easy accessibility.



3.2.2. Deliverables in Communication Subsystem

3.2.2.1. Deliverables in the Circuitry

Some progress has been made regarding the communication circuit before the beginning of the pandemic. After the COVID-19 pandemic, unfortunately the team Revolusys could not work on the circuits anymore due to the lack of electrical equipment and the inability of the team members to meet up. However, there is still some work achieved and in this section, we will present the communications circuitry while pointing out its missing parts and weaknesses.

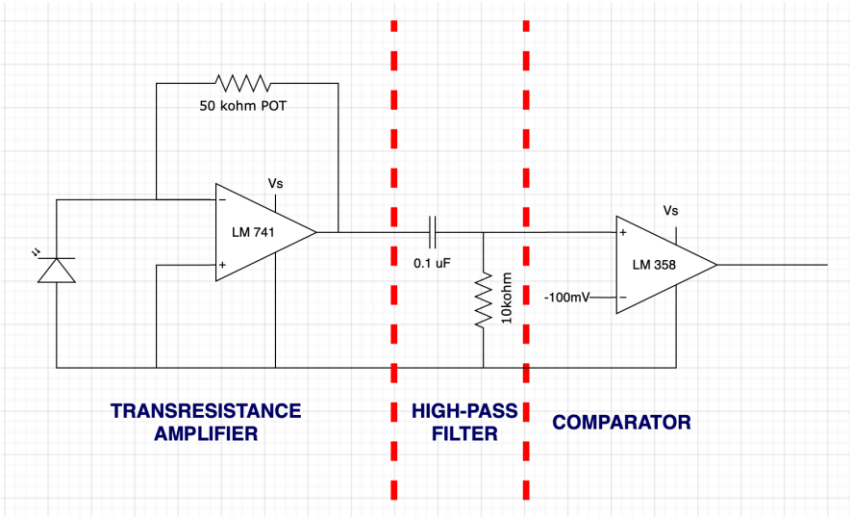


Figure 1: The receiver circuit

In Figure 1, it is possible to see the receiver circuit diagram. In this circuit, the first stage with the LM741 op amp is the photodiode receiver circuit. The photodiode receiver circuit converts the change in the received light intensity into desired voltage waveforms. However, the received voltage waveform comes with a DC offset. Therefore, the next stage is a high pass filter which gets rid of the DC offset. The final stage with the LM358 op amp is a voltage comparator. It compares the received voltage with -100 mV and produces the required square waves with a bottom value of 0V and a top value of 5V. The reason we used -100 mV but not 0V is to obtain a stable output when there is no light falling on the photodiode receiver. That is because when there is no light on the receiver, the output of the high pass filter is 0V and the comparator compares 0V with 0V. This causes an unstable output at the end of the comparator and this instability hurts the overall system performance.

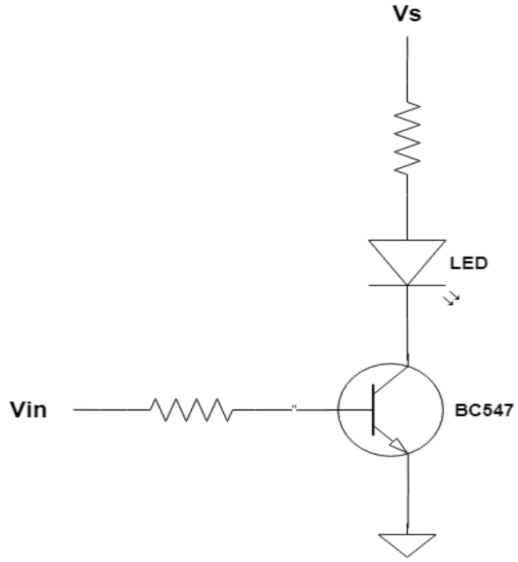


Figure 2: The transmitter circuit

In Figure 2, it is possible to see the transmitter circuit. It basically converts the variations in Vin into light intensity variations.

The last point about the circuits is about their usage guideline. In order to use these circuits, one should connect the Vin in Figure 2 to the UART transmission pin (TX) of the Arduino or the Raspberry Pi. Vs is recommended to be 5V or 6V. This way, the LED can blink correctly and transmit the UART data packets. In order to use the receiver circuit in Figure 1, one should connect Vs to 6V and the V_rec_out to the UART receiver pin (RX) of the Arduino. This way, the received waveform can be fed to the Arduino correctly. However, if this circuit is to be used with a Raspberry Pi, the supply voltage of the LM358 op amp should be equal to 3.3V since Raspberry Pi RX pin works in the range of 0-3.3V. However, it should be noted that although we are delivering a VLC circuit, further testing, adjustments and the duplication of the circuits will be required. The circuits in their presented form are not complete yet.

3.2.2.2. Deliverables in the Communication Software

During this time period, the Revolusys Group managed to develop the entire communication software. The software includes the fully established protocol and handshaking between the vehicle and the terminals, the error checking method and the image

compression method. The images are compressed in JPEG format and checksum is used for error detection. The software is delivered as two distinct python files, one for the transmitter and one for the receiver.

Image Compression and Error Detection

Due to the constraints of the Project, the size of the image has to be small. That is because in our design, the vehicle makes five turns and it can carry a maximum of 10 kB of data each time it makes a turn. Hence, the size of the image can not exceed 50 kB. This is achieved by coding a function which uses JPEG compression. This function takes any Picture and then compresses it until its size is reduced to 50 kB. This way, the size of the picture is reduced as desired.

As error detection method, IP (internet protocol) checksum algorithm is implemented. Checksum method produces a check value for the data packet to send. This value is sent to the receiver terminal following the related packet. At the receiver side, the check value taken is again put into the checksum algorithm with the packet and an error signal is generated as output in case the packet and the check value don't match.

Handshaking and the Communication Protocol

The data transmission should take place only when the vehicle is close enough to the transmitter and the vehicle should keep receiving data until all the necessary data is transmitted. Therefore, the vehicle should be able to inform the transmitter of its arrival and the transmitter should be able to tell the vehicle to leave.

In Gimme Fast project, the image should be divided into five pieces and each piece should be maximum 10 kB large so that the vehicle can complete its minimum of five turns. In order to prevent any confusions, in the remainder of the report, each one of these five pieces of the image will be referred to as a "chunk". For example, when the vehicle approaches to the transmitter to take the second piece of the image, we will refer to this as "The vehicle approaches to the terminal to obtain the second chunk". However, since VLC is prone to error, transmitting an entire chunk in one time is risky and it can cause erroneous transmission. Therefore, during transmission, each of these chunks will be further divided into smaller "packets" and after the transmission of a packet, error correction will be performed on it so that secure transmission can take place. The relation between a chunk, a packet and the entire image can be seen in Figure 3.

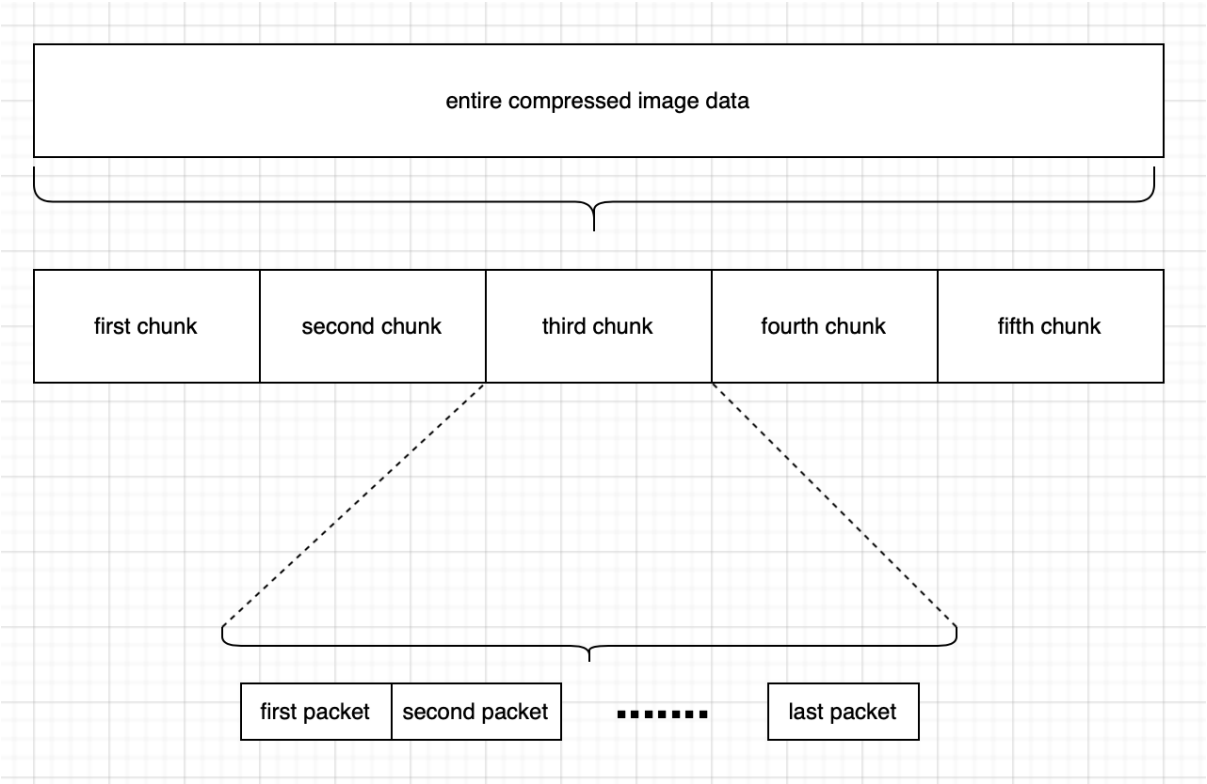


Figure 3: The relation between a chunk, a packet and the entire image bits

Under the light of the above paragraphs, now we can explain how the communication software works. In order to receive a chunk, the vehicle sends a request to the transmitter. This request informs the transmitter that the vehicle is ready to receive a chunk and it further tells which chunk the vehicle desires to receive. After receiving this request, the transmitter stars to send this chunk in packets of 1 kB. However, it doesn't directly send the packet. The packet is wrapped in a structure seen in Figure 4. In the structure of a packet, there is first a pre-indicator. This pre-indicator tells whether there will be more packets coming after the current packet or that it is the last packet and the vehicle can leave if there is no error detected. After receiving the pre-indicator, the corresponding image bits are received. The vehicle keeps reading the data until it receives the end-indicator. The only purpose of the end-indicator is to tell the receiver that the packet is fully transmitted. One might question the necessity of the end-indicator if the packet length is already known as 1 kB. However, the packet length does not have to be 1 kB. It is actually either equal to 1 kB or smaller than that. Therefore, the packet length and the checksum length can differ from packet to packet. Due to this issue, an end-indicator is necessary. The vehicle then uses the image bits and the checksum bits to perform error detection.

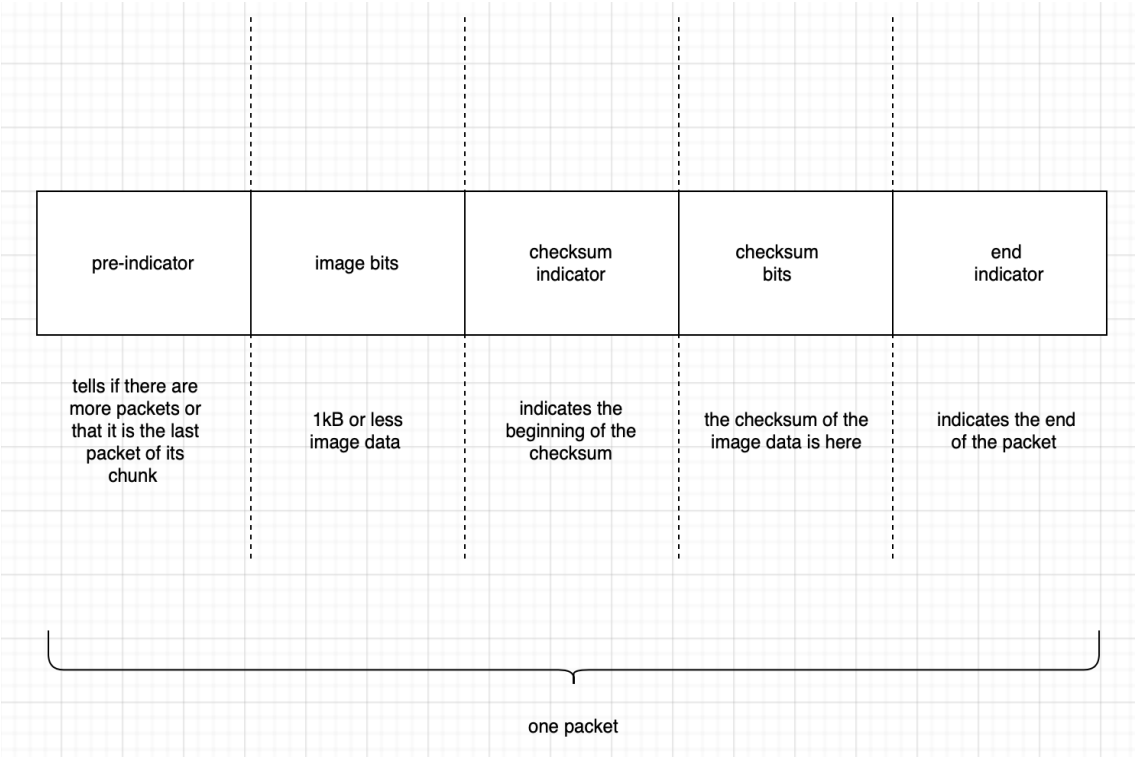


Figure 4: Structure of a packet

After sending a packet, the transmitter starts to wait the response of the vehicle. The vehicle performs error detection on the received data via checksum and if it finds no error, it sends a message to the transmitter that it is ready to receive the next packet. This is called as the "NEXT" message. After receiving the message "NEXT", the transmitter sends the next packet to the vehicle. However, if the vehicle detects an error in the packet, then it sends another message to the transmitter so that it sends the same packet one more time. This message is called as the "AGAIN" message. This process is repeated until the correct version of the packet is received.

Before moving on, one might think what happens if the transmitter fails to detect the NEXT or AGAIN messages and therefore fails to decide what to do. In this case, it should be noted that the transmitter does not do anything until it receives one of these messages. It simply keeps waiting. Therefore, after waiting for a predetermined amount of time, the receiver sends the same NEXT or AGAIN message again. The vehicle keeps doing this until the data it is waiting for arrives from the transmitter. We call this situation as a time out. Time out principle holds true for all the indicator messages. The indicator messages include the vehicle requests, the pre- and end indicator of the packets and the NEXT and AGAIN messages. This way, the loss of an entire packet is prevented since the entire system waits until the packet is received. Additionally, the errors in a single packet is detected by the checksum. Hence, we can conclude that both packet loss and the error in a received packet are detected correctly.

This way, the entire chunk is transmitted by the transmitter to the vehicle packet by packet. However, an important thing to note here is that in order to indicate that the last packet of the chunk is transmitted, the pre-indicator of the last packet is altered to a different message so that the vehicle can understand that it received the last packet and if there is no error, that it can leave.

Hence, the transmitter and the vehicle communicates with each other successfully. The vehicle sends a total of five different requests to the transmitter so that it can receive the five chunks of the picture in the correct order. The visualized flowchart of the processes described above can be seen in the flowchart in Figure 5.

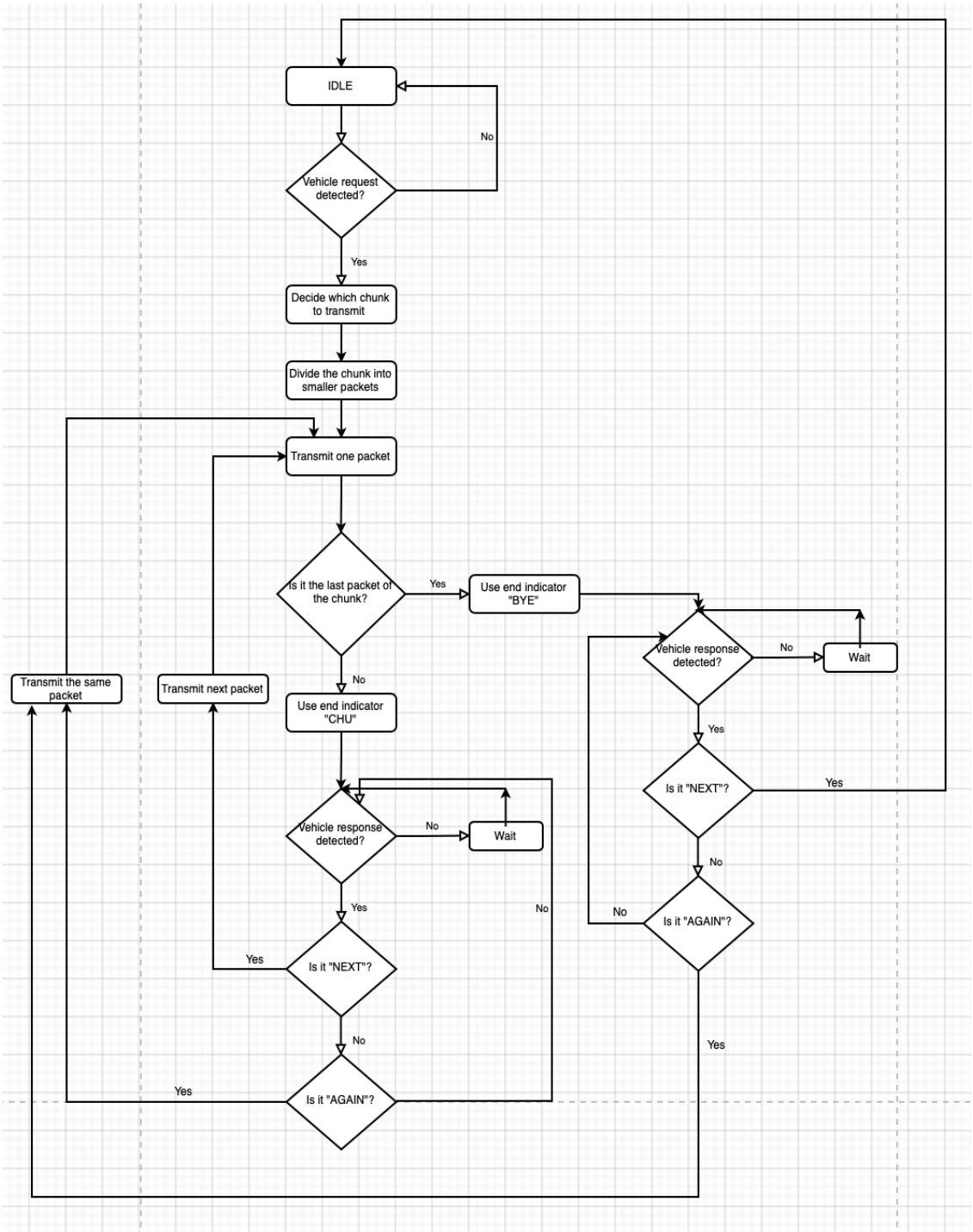


Figure 5: The flowchart of the transmitter

3.2.3. Deliverables in Transportation Subsystem

Even before the COVID-19 pandemic, Revolusys team have completed the construction of the vehicle and implemented the basic code to drive the vehicle back and forth between the terminals with an Arduino UNO controller. Unfortunately, as mentioned earlier the rail system to keep the vehicle in line with the terminals could not be developed as required materials were not purchased. However, the vehicle itself has reached its final state to be delivered and as explained in the results section it is capable of fully meeting our requirements.

We have developed a four wheel drive, Arduino UNO controlled car as the transportation vehicle of our project. The car is 20 cm in length and 15.5 cm in width. Other than the wheels and the frame of the car, the vehicle consists of four DC motors, a L298N motor driver, a small breadboard for the connections, and two HC-SR04 ultrasonic proximity sensors placed at each end of the car. As we were not able to purchase a battery, it is lack of one and currently being powered by a power supply.

The vehicle is being controlled by the Arduino UNO board through the L298N motor driver. As mentioned in the earlier reports an ON/OFF control method is used as the vehicle doesn't have to stop exactly at 5 cm away from the terminals but it is rather enough to stop between 5 to 10 cm away. The vehicle rushes to a terminal with its maximum speed while measuring the distance to the terminal using the proximity sensor on the corresponding side. As it gets closer than 15 cm to the terminal it starts to brake which enables the vehicle to come to a stop in the acceptable area.

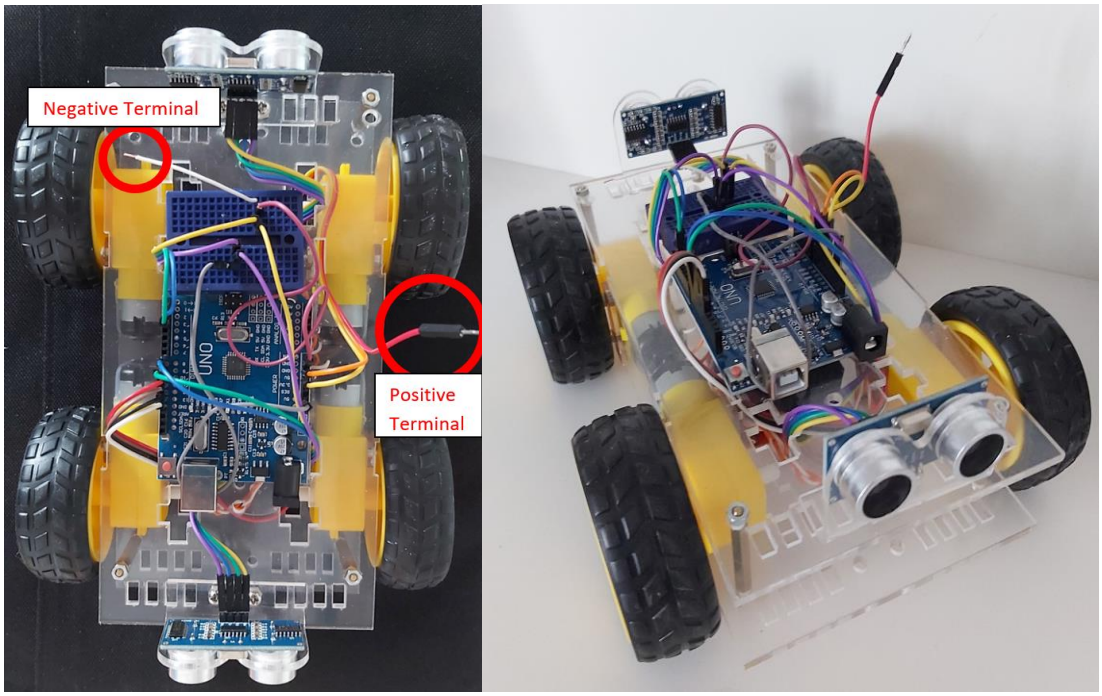


Figure 6: The vehicle to be delivered

In order to power up the vehicle, the red cable in Figure 6 must be connected to the positive terminal of the battery while the white cable must be connected to the negative terminal. The power applied directly goes to the motor driver while from the 5V output of the L298N driver the Arduino controller is being powered. A voltage from 5-35V can be applied according to the L298N datasheet, but we have used 9V and made our calculations

according to that. Thus, using a 9V battery is recommended. The battery must also be able supply at least 600mA of current according to the current test however, this might increase after the addition of the receiver and transmitters due to the increased weight..

Along with the vehicle that can be seen in Figure 6 the Arduino code to obtain the mentioned operation will be delivered. From this point, firstly a rail structure must be constructed for the vehicle. After the construction of the rail structure, the braking distance of the vehicle might change depending of the friction caused by the rail system, which might require retuning of the instant to apply the brakes. Next, by adding the receivers and transmitters on the car its communication with terminals must be enabled and the provided code must be updated such that the car decides what to do according to the information taken from each terminal.

3.3. Cost Breakdown

At the final version of the project, due to implementation and integration limitations, necessity of some items is not required. But these items is listed in another table in case of completion of the rest of the project by the Revolusys or another engineering group.

Expense Items	Cost
Raspberry Pi 3 (Representing the transmitter terminal)	205 TL
Rasppberry Pi Zero (Representing the receiver terminal)	130 TL
Arduino Uno (for controlling the vehicle) with DC Motor set	90 TL
Raspberry Pi Camera Module	16 TL
Others(Jumpers, breadboard, PCB Plexiglass,Wheels,3D printing costs etc.)	100 TL
Total Cost:	541 TL / 79.33 $

Expense Items (for completion of the project)	Cost
4 x LED (to be used for VLC at the transceiver terminal and on the vehicle)	0.5 TL
4 x Visible Light Sensitive Photodiode(to be used for VLC at the transceiver terminal and on the vehicle)	6 TL
1.4`` LCD Screen (to display the reconstructed image at the receiver terminal)	50 TL
Aluminium Profiles for Physically Guided Track	50 TL
Total Cost for the Rest of the Project:	156.5 TL / 22.94 $

As it can be seen from the tables above, current version of the project costs 79.33 $ and to complete the project, only 22.94 $ required.

Cost Objective (in Marketability Objectives)

• 5 points – Total cost is under 100$
• 4 points – Total cost is between 100-125$
• 3 points – Total cost is between 125-150$

The Revolusys Inc. has succeeded to get 5 points in cost objective with current version of the project. If physical integration had been done, the group would take 4 points in this objective.

4. Conclusion

In conclusion, we have presented the tasks we have and we have not completed up to now. Unfortunately, we were not able to deliver a fully completed and integrated project due to the COVID-19 pandemic, but we tried our best to be able to advance as much as possible. We have developed the core parts of each subsystem before the coronavirus lockdown as mentioned earlier. This enabled us to complete some tasks individually or by making online group meetings. We achieved to complete many tasks successfully, however we couldn't completed some parts such as the receiver and transmitter circuitries or the physically guided tracks due to lack of required equipments. In addition to these, we were not able work on the integration of the subsystem, which we believe, would be the most challenging and the most rewarding the part of this whole project.

Probably, the integration of the subsystems would be the part that we would learn the most and gain very valuable experiences. In addition to that, we missed the chance of observing the outcome of our many sleepless nights and the product of our studies in this department as a complete project. However, we still believe that witnessing such a worldwide pandemic was also an important experience for us. As mentioned to us many times, the important thing about this project was to learn the process of engineering design rather than obtaining the final product. In order to handle the current situation, we changed our plans, discussed how we can proceed in the changed conditions and experienced a case of crisis management. All these, enabled us to actually experience a quite interesting design process where our previous plans and predictions suddenly became not applicable.

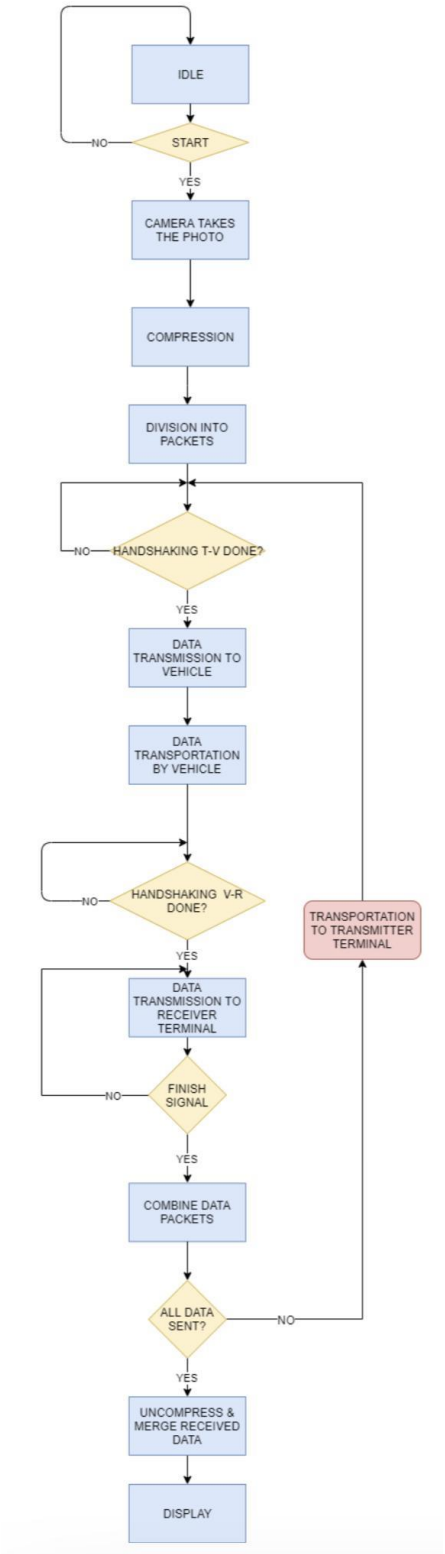
After updating our plans, we mostly focused on the software related parts of the project. We have successfully completed the communication protocols and conducted its tests by connected two Raspberry Pi's via cable instead of using visible light communication. Other than that, we have managed to integrate the image acquisition and image reconstructions subsystems with the communication protocol so that we could take a photo, transfer it according to the requirements of the visible light communication and the transportation subsystem and display on the other side. In addition to this, we implemented a GUI so that the user can shoot the photo to be transmitted easily. Also, by making final adjustments we developed the vehicle into its final form.

Although the explained tasks might seem a little bit disconnected like this, they would actually become parts of a complete project that work in harmony. As Revolusys, we aimed to advance every subsystem in a way that a following team other than us would keep up with the project and complete it. We believe that by reviewing all our previous reports, the aim and integrity of each subsystem could be understood by the following engineering team and especially in this report we explained how to use and for what to use each and every deliverable that we are transferring to the upcoming team.

We as the five engineers of team Revolysys, are hereby ending our part in the 'Gimme Fast' project. We believe that this project enabled us to gain priceless experiences. Being able to apply our theoretical background on a practical problem and seeing the products of our electrical and electronics engineering education has certainly strengthened our passion for engineering. Undoubtedly, we will always remember this as our first project throughout our professional lives and always benefit from the experiences that we have gained during the whole process of this project.

5. Appendices

Appendix 1 : State Diagram of the Overall System



Appendix 2: The Flows of the Simulations

Here, the flows of the simulation for different conditions are given. The user inputs are environment selection, stopping distance selection and selection of the distance between two terminals. The communication speed, the communication time consumption per packet, the average speed and travel time between two terminals (not back and forth) are printed at the beginning of the simulation. Each time vehicle reaches to transmitter terminal, an informative note is printed. At each stop of the vehicle, its stopping distance is calculated (using Gaussian distribution) and error rate is calculated (using stopping distance and Gaussian distribution) and printed. For each packet, checksum result and whether the packet is right or wrong is printed. At the end of the flow, total time consumption (estimated using travel time and communication time) is printed.

Flow 1

Conditions:

- environment = dark
- stopping distance = 10 cm
- distance between two terminals = 150 cm

```
Communication speed is 19200 bps
It takes 0.536 seconds to transmit one packet
Distance between the terminals is  150  cm
Average speed is  61.5  cm/sec
The time it takes to travel is  1.95  seconds

The vehicle arrives to the terminal for the first time
The vehicle is sending the request
>> FIR
The vehicle sent the request
Error rate is  0.008610882450455503
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the second time
The vehicle is sending the request
>> SEC
The vehicle sent the request
Error rate is  0.018449108838323788
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the third time
The vehicle is sending the request
>> THI
The vehicle sent the request
Error rate is  0.016831989391526143
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fourth time
The vehicle is sending the request
>> FOU
The vehicle sent the request
Error rate is  0.03516269673137887
Checksum result is  0
NO ERROR FOUND
Checksum result is  28032
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fifth time
The vehicle is sending the request
>> FIF
The vehicle sent the request
Error rate is  0.026158561669423402
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  29112
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Total time spent is:  65.98400000000007
DONE!
```

Flow 2

Conditions:

- environment = light
- stopping distance = 10 cm
- distance between two terminals = 150 cm

```
Communication speed is 19200 bps
It takes 0.536 seconds to transmit one packet
Distance between the terminals is  150  cm
Average speed is  61.5  cm/sec
The time it takes to travel is  1.95  seconds

The vehicle arrives to the terminal for the first time
The vehicle is sending the request
>> FIR
The vehicle sent the request
Error rate is  0.02378723432127151
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the second time
The vehicle is sending the request
>> SEC
The vehicle sent the request
Error rate is  0.02279623323476817
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the third time
The vehicle is sending the request
>> THI
The vehicle sent the request
Error rate is  0.0030599916761915383
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fourth time
The vehicle is sending the request
>> FOU
The vehicle sent the request
Error rate is  0.011748459119432536
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  33790
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fifth time
The vehicle is sending the request
>> FIF
The vehicle sent the request
Error rate is  0.032494948102635916
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  34736
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Total time spent is:  65.98400000000007
DONE!
```

Flow 3

Conditions:

- environment = light
- stopping distance = 10 cm
- distance between two terminals = 150 cm
- Error rate is multiplied by 10, to observe the effects of high error rate.

```
Communication speed is 19200 bps
It takes 0.536 seconds to transmit one packet
Distance between the terminals is  150  cm
Average speed is  61.5  cm/sec
The time it takes to travel is  1.95  seconds

The vehicle arrives to the terminal for the first time
The vehicle is sending the request
>> FIR
The vehicle sent the request
Error rate is  0.14976112134669534
Checksum result is  35904
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  18938
***ERROR DETECTED***
Checksum result is  61996
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  29239
***ERROR DETECTED***
```

```
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND

The vehicle arrives to the terminal for the second time
The vehicle is sending the request
>> SEC
The vehicle sent the request
Error rate is  0.2847345796822986
Checksum result is  0
NO ERROR FOUND
Checksum result is  12058
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  47427
***ERROR DETECTED***
Checksum result is  8418
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  1025
***ERROR DETECTED***
Checksum result is  33341
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  181
***ERROR DETECTED***
Checksum result is  4642
***ERROR DETECTED***
Checksum result is  35303
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  14758
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND

The vehicle arrives to the terminal for the third time
The vehicle is sending the request
>> THI
The vehicle sent the request
Error rate is  0.3339712756159727
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  37993
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  19021
***ERROR DETECTED***
Checksum result is  39639
***ERROR DETECTED***
Checksum result is  24425
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
Checksum result is  0
NO ERROR FOUND

The vehicle arrives to the terminal for the fourth time
The vehicle is sending the request
>> FOU
The vehicle sent the request
Error rate is  0.2461960288631042
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fifth time
The vehicle is sending the request
>> FIF
The vehicle sent the request
Error rate is  0.18056798777530364
Checksum result is  0
NO ERROR FOUND
Checksum result is  46183
***ERROR DETECTED***
Checksum result is  15674
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  33815
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
Total time spent is:  83.13600000000008
DONE!
```

Flow 4

Conditions:

- environment = light
- stopping distance = 10 cm
- distance between two terminals = 100 cm

```
Communication speed is 19200 bps
It takes 0.536 seconds to transmit one packet
Distance between the terminals is  100  cm
Average speed is  50.0  cm/sec
The time it takes to travel is  1.39  seconds
```

```
The vehicle arrives to the terminal for the first time
The vehicle is sending the request
>> FIR
The vehicle sent the request
Error rate is  0.0342715436069655
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the second time
The vehicle is sending the request
>> SEC
The vehicle sent the request
Error rate is  0.01539410086169617
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  53644
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the third time
The vehicle is sending the request
>> THI
The vehicle sent the request
Error rate is  0.03219114961920019
Checksum result is  0
NO ERROR FOUND
Checksum result is  11396
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fourth time
The vehicle is sending the request
>> FOU
The vehicle sent the request
Error rate is  0.031252202843084054
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fifth time
The vehicle is sending the request
>> FIF
The vehicle sent the request
Error rate is  0.04136110619734086
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Total time spent is:  61.504000000000055
DONE!
```

Flow 5

Conditions:

- environment = light
- stopping distance = 10 cm
- distance between two terminals = 50 cm

```
Communication speed is 19200 bps
It takes 0.536 seconds to transmit one packet
Distance between the terminals is  50  cm
Average speed is  22.0  cm/sec
The time it takes to travel is  0.92  seconds
```

```
The vehicle arrives to the terminal for the first time
The vehicle is sending the request
>> FIR
The vehicle sent the request
Error rate is  0.024741877713717626
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  1805
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND

The vehicle arrives to the terminal for the second time
The vehicle is sending the request
>> SEC
The vehicle sent the request
Error rate is  0.027603987824547174
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND

The vehicle arrives to the terminal for the third time
The vehicle is sending the request
>> THI
The vehicle sent the request
Error rate is  0.03391916101078936
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  25531
***ERROR DETECTED***
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fourth time
The vehicle is sending the request
>> FOU
The vehicle sent the request
Error rate is  0.025807347447123354
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND

The vehicle arrives to the terminal for the fifth time
The vehicle is sending the request
>> FIF
The vehicle sent the request
Error rate is  0.037201339787121764
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Total time spent is:  57.74400000000005
DONE!
```

Flow 6

Conditions:

- environment = light
- stopping distance = 5 cm
- distance between two terminals = 150 cm

```
Communication speed is 19200 bps
It takes 0.536 seconds to transmit one packet
Distance between the terminals is  150  cm
Average speed is  61.5  cm/sec
The time it takes to travel is  1.95  seconds
```

```
The vehicle arrives to the terminal for the first time
The vehicle is sending the request
>> FIR
The vehicle sent the request
Error rate is  0.012505601614272518
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND

The vehicle arrives to the terminal for the second time
The vehicle is sending the request
>> SEC
The vehicle sent the request
Error rate is  0.016711153516820083
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the third time
The vehicle is sending the request
>> THI
The vehicle sent the request
Error rate is  0.004249739438375318
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fourth time
The vehicle is sending the request
>> FOU
The vehicle sent the request
Error rate is  0.031507263933125876
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
```

```
The vehicle arrives to the terminal for the fifth time
The vehicle is sending the request
>> FIF
The vehicle sent the request
Error rate is  0.008237171311500244
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Checksum result is  0
NO ERROR FOUND
Total time spent is:  63.84000000000006
DONE!
```

Appendix 3: The Objectives of Solution and Their Measure of Success

To evaluate the obtained results, the following rubric is used.

Performance Objectives:

Speed:

• 5 points – Operation time less than 60 seconds
• 4 points – Operation time between 60 and 75 seconds
• 3 points – Operation time between 75 and 90 seconds
• 2 points – Operation time between 90 and 105 seconds
• 1 points – Operation time between 105 and 120 seconds
• 0 points – Operation time over 120 seconds

Accuracy:

• 5 points – The displayed image has %100-85 accuracy
• 4 points – The displayed image has %85-70 accuracy
• 3 points – The displayed image has %70-55 accuracy
• 2 points – The displayed image has %55-40 accuracy
• 1 points – The displayed image has %40-25 accuracy
• 0 points – The displayed image has under %25 accuracy

Marketability Objectives:

Cost:

• 5 points – Total cost is under 100$
• 4 points – Total cost is between 100-125$
• 3 points – Total cost is between 125-150$
• 2 points – Total cost is between 150-175$
• 1 points – Total cost is between 175-200$
• 0 points – Total cost is over 200$

Immunity to Variable Light Conditions:

• 5 points – Works under sunlight.
• 4 points – Works under indoor lightening.
• 3 points – Works at shadows at daytime.
• 2 points – Works under indoor shady lightening.
• 1 points – Works unsteadily under shady indoor lightening.
• 0 points – Works only under complete darkness.

Robustness:

• 5 points – System is robust for its operation and transportation of the system.
• 4 points – Risk of disintegration of several parts of the system while packing and transporting whole
system to another location.
• 3 points – Disintegration of several parts of the system while packing and transporting whole system to another location.
• 2 points – Risk of disintegration of one part of system in operation time.
• 1 points – Disintegration of one part of system in operation time.
• 0 points – Disintegration of several parts of system in operation time.