MIDDLE EAST TECHNICAL UNIVERSITY

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

EE 494-DESIGN STUDIO 1

CONCEPTUAL DESIGN REPORT

COMPANY NAME: REVOLUSYS

PROJECT NAME: GIMME FAST

PROPOSED STARTING DATE: 02.11.2019

PROPOSED DURATION & COMPLETION DATE: 6.5 MONTHS/ 15.05.2020

COST OF THE PROJECT: 200 USD

REPORT SUBMITTED TO: DR. EMRE ÖZKAN

DATE OF SUBMISSION: 28.02.2020

revolusys

Doğukan Atik (*Telecommunication Engineer*)	**+905464070498**
Mert Eyüboğlu (*Control Systems Engineer*)	**+905398663601**
Ozan Berk Boyraz (*System Engineer*)	**+905314598425**
Ahmet Demirdaş (*Embedded Systems & Software Engineer*)	**+905423693903**
Onur Akdeniz (*Electronics Engineer*)	**+905389867561**

TABLE OF CONTENTS

A) EXECUTIVE SUMMARY

In this era, we are all aware of that accessing to information has a crucial importance, but the more important and challenging part is how to transfer that information. Although RF transmission has been the conventional mean of data transmission up to these days, due to the increasing mobile data transmission RF technologies started to struggle on resolving our needs. In addition to the highly occupied bandwidth RF technologies suffer from speed, security and power efficiency problems. The best way to accomplish these problems is using a revolutionary method: visible light communication.

As Revolutionary Systems we will design an image transmission platform where VLC is combined with data transportation by a vehicle. The system will consist of two end terminals that can receive or transmit an image as data packets while a vehicle rushes between these terminals to handle the long-distance transmission of the data. While combining communication and transportation the main purpose is to transfer an image as accurately as possible in the shortest time. For achieving this goal, the main tasks and our solutions approaches can be listed as:

- Data Compression and Division into Data Packets
  - Compression of image data to enable faster transmission
  - Division of the image data into smaller matrices to get smaller data packets
- Visible Light Communication
  - Sending and receiving light signals by LEDs and photodiodes
  - Modulation of the original signal for communication
  - Filtering in order to cancel noise
  - Use of preamble signals for handshaking between receivers and transmitters
- Data Transportation by Vehicle
  - Control of the vehicle on physically guided tracks
  - Distance detection by ultrasound sensor
  - Transceiver unit placed and memory chunk on the vehicle

Even though handling all these tasks require knowledge from a diversity of electrical engineering areas, our team contains engineers specialized in the areas of communication, computer, electronics and control. Thus, a correct division of labor between us enables us to handle this big task as smaller and manageable problems and come up with a competitive product build upon the skills of each member.

Our company aims to build up a fast and accurate system for the minimal cost while taking physical robustness and the immunity of the system to variable conditions into account. The project is planned to be finished in 4.5 months with a total budget of about 130$.

As the end product, a vehicle that is moving on a physically guided track with a transceiver on it and two end terminals which are a transmitter and a receiver will be delivered. In addition to these, a camera to take the photo and a display will be supplied to the customer with a user manual and two years of warranty.

This document is a critical design report of the described product which contains detailed technical information on how the product will be developed.

A) INTRODUCTION

Radio spectrum is used to transmit data wirelessly for an enormous amount of daily services including but not limited to TV and radio broadcasting, mobile phones, Wi-Fi communications, GPS and radar. The global mobile data traffic has increased by 71 percent in 2017 according to the yearly report of CISCO[1] .In addition, the increasing device connectivity because of IoT also puts more load on RF bandwidth. Hence, the ever-increasing demand for huge amount of information, faster communication and higher quality data, it is crucial to note that the usable radio spectrum is a scarce source where exponentially growing demand surpasses the supply. Apart from the narrow and already highly occupied bandwidth problem, some other issues with the convenient communication systems can be explained as security problems, power inefficiency and interference.

A recently developed communication method which is known as VLC (visible light communication) has a potential to solve these problems. Since VLC uses visible light, the bandwidth is increased tremendously, it is in between 430 THz to 790 THz[2] . Also, since the VLC receiver receives the signal only if the transmitter and the receiver are in the same room, it is more secure than RF communication. What is more, since a visible light source can be used for both illumination and communication, it saves extra power when compared to RF communication.

Although, there are existing communication architectures, they suffer from the aforementioned inefficiencies. A need exists for new communication methods. The design possesses a physically guided vehicle and a VLC system. The goal is to transport a picture from one terminal to the other terminal as fast and accurate as possible while keeping the cost minimal.

The professors of Electrical and Electronics Engineering in METU has requested from us to design a system which can transfer data via two complementing technologies, transportation and communication. From then on, the company have worked solidifying the technical approach on the project. By means of weekly meetings and R&D work conducted at Design Laboratory, currently, Revolusys can modulate the communication signal via OOK (On-Off Keying) and receiving the data packets with the rate of 20 kbps at 5-10 cm distance. Tests for communication, image acquisition and reconstruction and transportation submodules were also done by company.

This report contains the technical details of subsystems, critical algorithms for the flow of the project, requirement analysis of the subsystems, test plans, signal interfacing and integration of the overall system. The interaction of the subsystems and project parts like cost analysis, 3D drawing and power analysis will also be given.

By accomplishing this project, we will prove that a VLC system integrated with a transportation vehicle is implementable and this, in turn, will lead to a widespread use of the visible light spectrum. This will allow us to operate on a completely empty frequency band which will increase the amount of data transmission to a greater extent. Therefore, the entire society will reap the benefit of increased and more reliable data transmission and live in a more connected world.

C) REQUIREMENTS

The system level requirements of the solution are classified as functional requirements, performance requirements and physical requirements. These classified requirements are given below.

- *Functional Requirements*
    - The system must be able to take a photo.
    - Some portion of the photo must be transmitted to the vehicle by VLC (Visible Light Communication).
    - The vehicle should go to the receiver terminal on a physically guided track.
    - The data packets carried by the vehicle needs to be delivered to the receiver terminal.
    - The vehicle must go back and forth until the transfer of the full photo is done.
    - As the full photo is delivered, the photo must be displayed at the receiver terminal.

- *Performance Requirements*
    - A minimum DTR (data transfer rate) of 0.013 Mbps will be achieved.
    - The average velocity of the vehicle shouldn’t be lower than 25 cm/sec for the maximum distance case (1.5 meters).
    - The minimum accuracy rate of 90% should be achieved for the reconstructed image.

- *Physical Requirements*
    - The vehicle should be able to move on a physically guided track.
    - The receiver terminal will also be able to move on the track.
    - The distance between two terminals should vary between 0.4 meters to 1.5 meters.

Revolusys Inc. defined the design objectives as it is shown on the objective tree, Figure 1. The determined objectives regarding the solution of “Gimme Fast” project are scaled so that the company can assign points to the alternative designs and find the best-matching design considering the company’s milestones.

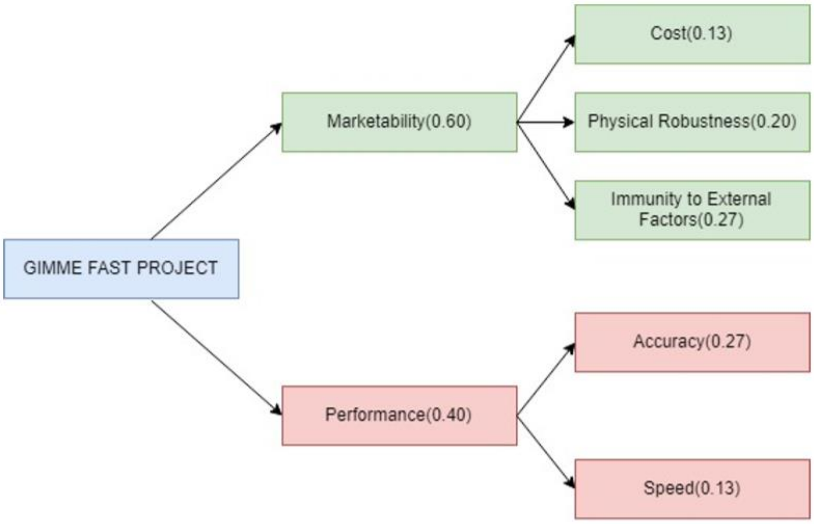


Figure 1: Objective tree for the evaluation of the overall system.

The constraints on the solutions to “Gimme Fast” project are listed below.

- There must be 5 cm between transmitter and receiver during the light communication.
- Maximum time for the total data transfer is 2 minutes.
- Microcontroller’s memory shouldn’t exceed 10 kB.
- Data transfer must be handled with 5 full round.
- Up to 8 LEDs and 8 photodiodes/LDRs can be used in the whole system.
- The distance between two terminals should be convertible up to 1.5 meters.
- The size of the vehicle shouldn’t exceed 20 cm.
- Motors of the vehicle must be on-board.

D) SOLUTION PROCEDURE

D.1) OVERALL SYSTEM BLOCK DIAGRAM

The solution of Revolutionary Systems Inc. to “Gimme Fast” project is a system that consists of four subsystems. These subsystems are image acquisition subsystem, communication subsystem, transportation subsystem and image reconstruction subsystem. Figure 2 shows the main block diagram of the system.

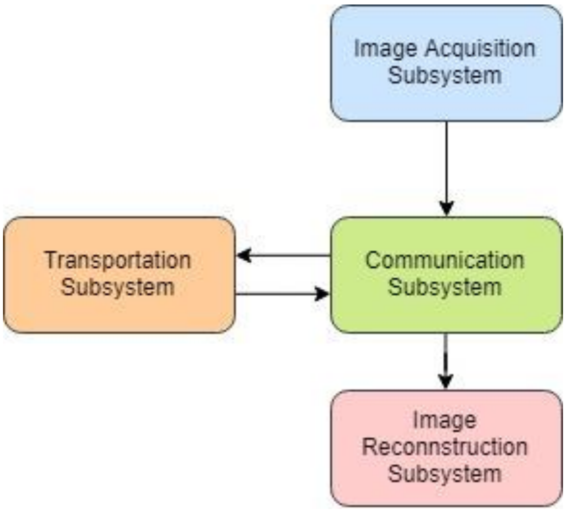


Figure 2: The main block diagram that shows the subsystems of the solution.

The solution proposed and tested by Revolusys is briefly summarized in the following paragraph. This solution and its alternatives are explained in details throughout the entire “Solution” section of this report.

Input of the whole system is an image taken by a camera and inputted to image acquisition subsystem. Image acquisition subsystem compresses the data and sends to communication subsystem. The first part of the communication subsystem transfer data from first terminal to receiver placed on the vehicle. The vehicle moves to second terminal and via utilization of the communication subsystem, data is transferred from transmitter placed on the vehicle to receiver at the second terminal. Vehicle moves back to first terminal while data received by the second part of the communication system is sent to image reconstruction subsystem where image is reconstructed. The reconstructed image is displayed on a screen as the output of the whole system.

This process is visualized in Figure 3 which shows the inputs, outputs and interactions at subsystem level.

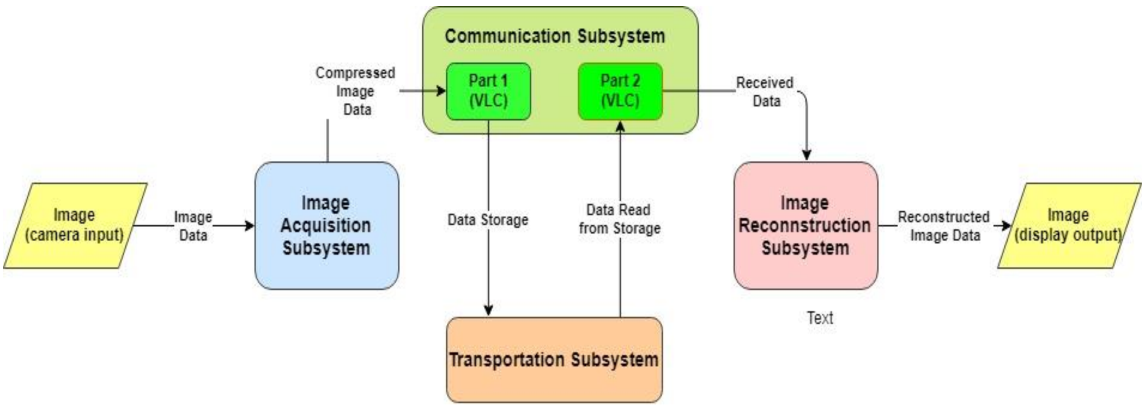


Figure 3: The diagram that shows the inputs, outputs and interactions at subsystem level.

The image compression method is selected as JPEG so that the size can be controllable by making a tradeoff between quality and resolution. It is feasible to have a data image size of 30-50 kB. Test plan was operated for JPEG compression, and by using PIL in Python, the image taken is downsized to **10 kB**.

D.2) SYSTEM LEVEL FLOWCHART

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.

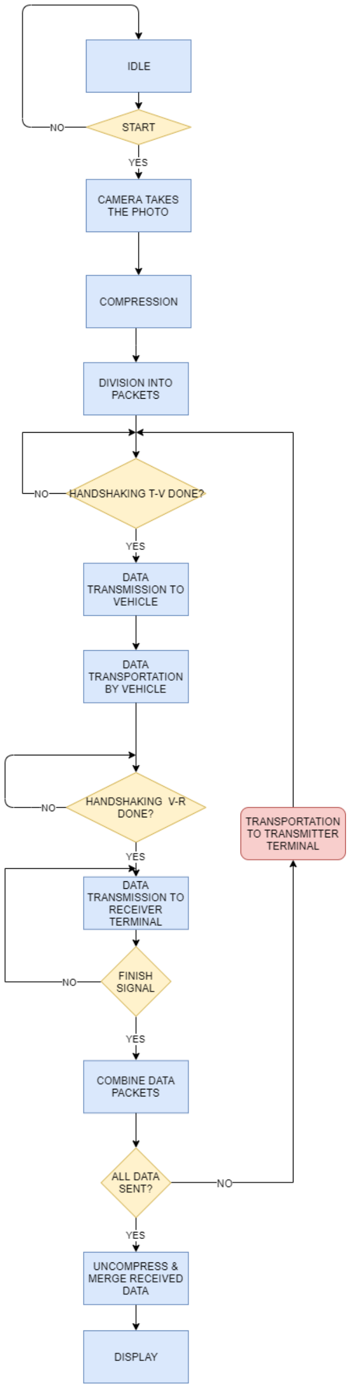


Figure 4: System Level Flowchart

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.

D.3) SUBSYSTEMS

D.3.1) Communication Subsystem

The communication subsystem is where the data transmission between the vehicle and the terminals take place. Based on our requirements analysis, it should reach to a data transmission rate of 13 kbps (kilobits per second). Furthermore, it should be able to work under different lighting conditions. Also, it should have a very low bit error rate since we are transmitting the bytes of a compressed image and even a single bit error can be problematic while decompressing the image.

Considering these issues, the communication subsystem is designed. It consists of two parts, the software part and the hardware part.

Software Part of the Communication Subsystem

In order to successfully transmit the bits, the software design should include,

- Protocol establishment and handshaking
- Error checking
- Modulation

These three parts are investigated seperately below under their own headings.

1) Protocol Establishment and Handshaking

During the transmission, the receiver and the transmitter should agree upon structure of the transmission so that they can understand each other. For example, they should know when the transmission starts or ends and they should understand how to infer the incoming waveforms. For this purpose, in the conceptual design report, the following method was proposed:

$$r(t) = rand(t) + sync(t) + m(t) + end(t) \quad (Equation\ 1)$$

where r(t) is the received signal, rand(t) is some random meaningless bit stream before the actual content of the message, synch(t) is the synchronization signal, m(t) is the data signal and end(t) is the ending indicator. This structure is illustrated in the graph below.

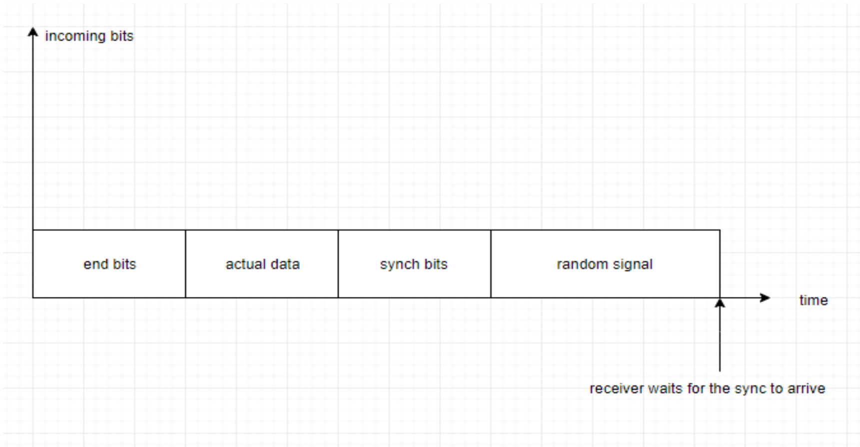


Figure 5: The arriving bit stream as the time passes

In Figure 5, structure of the message signal can be observed. As the time passes, the bit stream moves to the right and the receiver first bypasses the random signal and then realizes that the actual data is coming after fully detecting the synch bits. By this, we mean that after the receiver detects the synch bits, it starts a timer in its microcontroller and samples the incoming bits accordingly. Lastly, it understands that the transmission of one data packet ended whenever it detects the end bits. In this configuration, the length of the synch bits were 6 bits long and the end bits were also 6 bits long. The lenght of the message signal was 150 bits. In its essence, we proposed a serial asynchronous communication method.

However, we have decided to modify this approach due to the fact that the timer inside our microcontroller is not counting the time passed correctly. As a result, after some time we were losing some bits or sampling the wrong bit at the wrong time. After realizing this, there were two options in front of us. We could either change the microcontroller or come up with a more suitable transmission scheme. We are using Arduino as a microcontroller we didn't want to change it because it would create unnecessary cost. So, instead we have decied to modify our transmission scheme.

If investigated carefully, it can be understood that we were trying to develop a serial asynchronous communication method. We still didn't give up on this idea in our modified solution, we have just changed its implementation. Because of these issues discussed above, we decided to switch to the UART protocol. UART stands for Universal Asynchronous Receiver-Transmitter. In this protocol, no clock signal is used to synchronize the receiver and transmitter. Instead, we control the flow of data by using start and stop bits. The data packet lies in between those bits and it is 7-bits long. If we refer to the Equation 1, we can see that sync(t) becomes the start bit, end(t) becomes the stop bit and m(t) becomes the data bits. Therefore, by reducing the packet size and utilizing the UART chip inside the microcontrollers, we obtain stability and consistency during the transmission.

The data packet is used as 8-bits in standard so one can ask the reason we are using 7-bits instead of eight bits. The reason for that is to utilize error correction. Normally, without any error correction, to transmit 1 byte of data, we transmit one packet in UART. However, once we add error correction, we transmit two packets in order to transmit 1 byte of data. That is because in one packet, we insert four bits of data and three bits of error correction bits. Because of this, in order to transmit one byte, we send two bytes. This is a trade-off which we are willing to make to ensure the secure and correct transmission of data. The mechanism of error correction will be discussed in its heading in detail.

In UART communications, the receiver knows that the bit stream is coming when it detects the start bit and knows that one packet ended when it detects the stop bit. However, in this case, although the receiver knows when to detect data, the transmitter doesn't know when to start sending it. This is where the handshaking comes in. When the car reaches to a terminal and is ready to take in data, it sends a special signal via UART and then the terminal starts sending data.

Lastly, the structure of one packet of data can be seen in Figure 6**.**

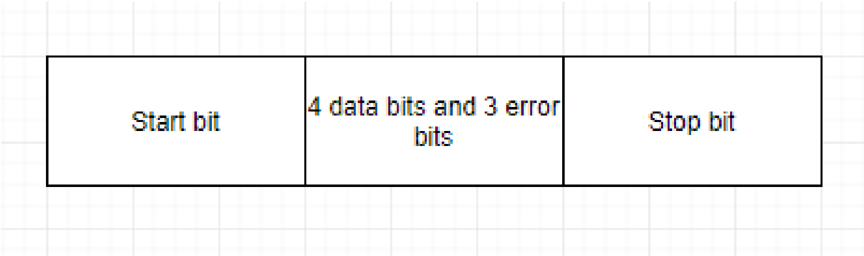


Figure 6: Structure of one data packet in UART

One final note to make here is that while the transmitter and the receiver are communicating with each other, they need some special signals to indicate that a special event is triggered. These signals are end of the photo signal, end of the packet stream signal and end of a packet chunk signal.

End of the photo signal tells that there will be no more data anymore so that the image reconstruction can begin. End of the packet stream signal informs the car that it is fully loaded and good to go. End of the 1 kB signal gives the receiver a chance to detect if there is an error in the packet stream that it received. If there is not an error, then it acknowledges that to the transmitter so that it can send a new packet. If there is no acknowledgement, then the transmitter concludes that there is something wrong and resends the data.

In order to perform handshaking, the transmitter first sends a request to send signal to the receiver. The receiver replies back by sending a vehicle request. Once this occurs, the transmission begins. After the transmission finishes, the transmitter sends one of the special signals indicated above to trigger any special event. The detailed flowchart of the transmitter side can be seen in Figure 6**.** Also, the detailed flowchart of the receiver can be seen in Figure 7**.**

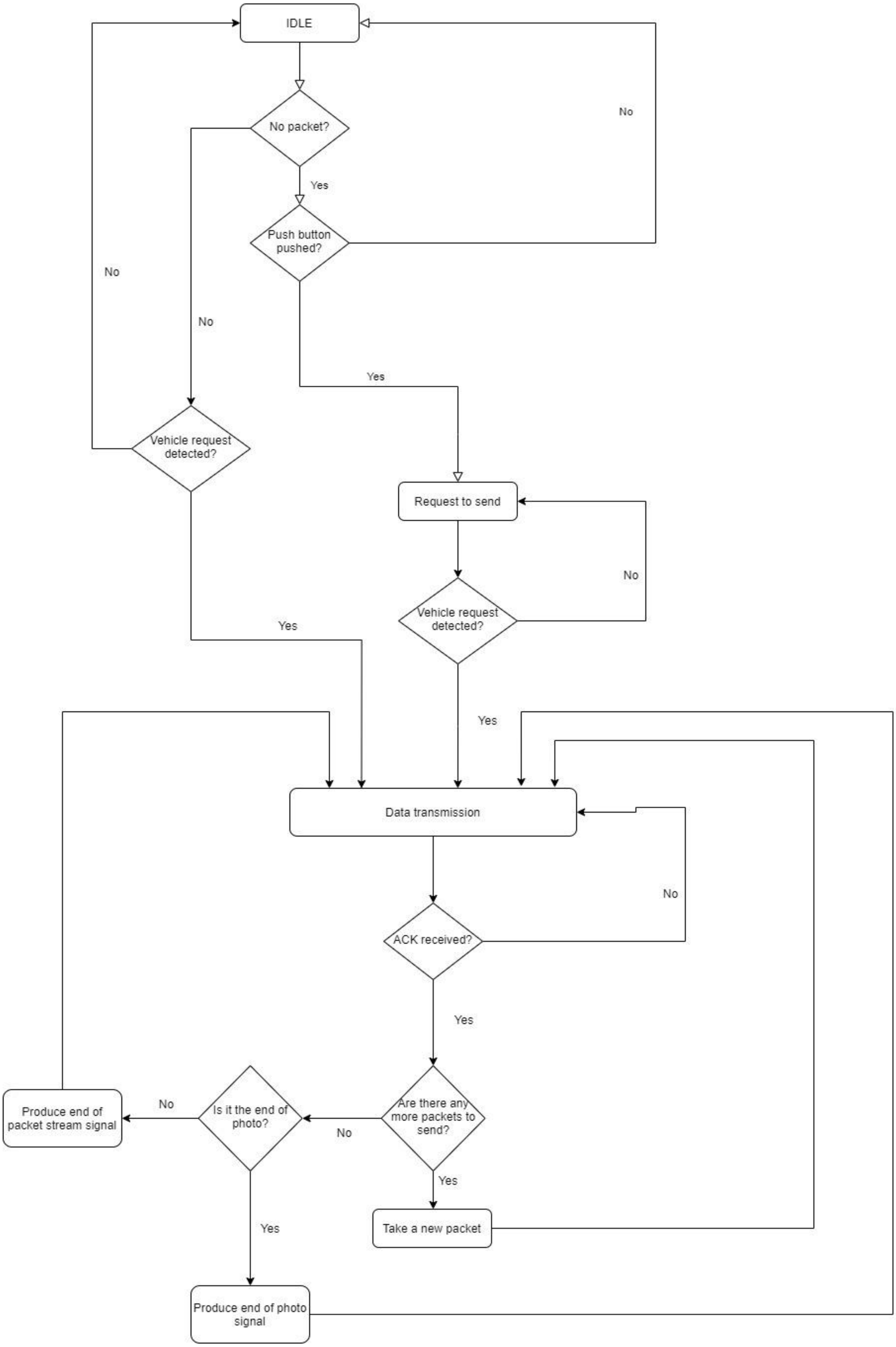


Figure 6: The flowchart of the transmitter

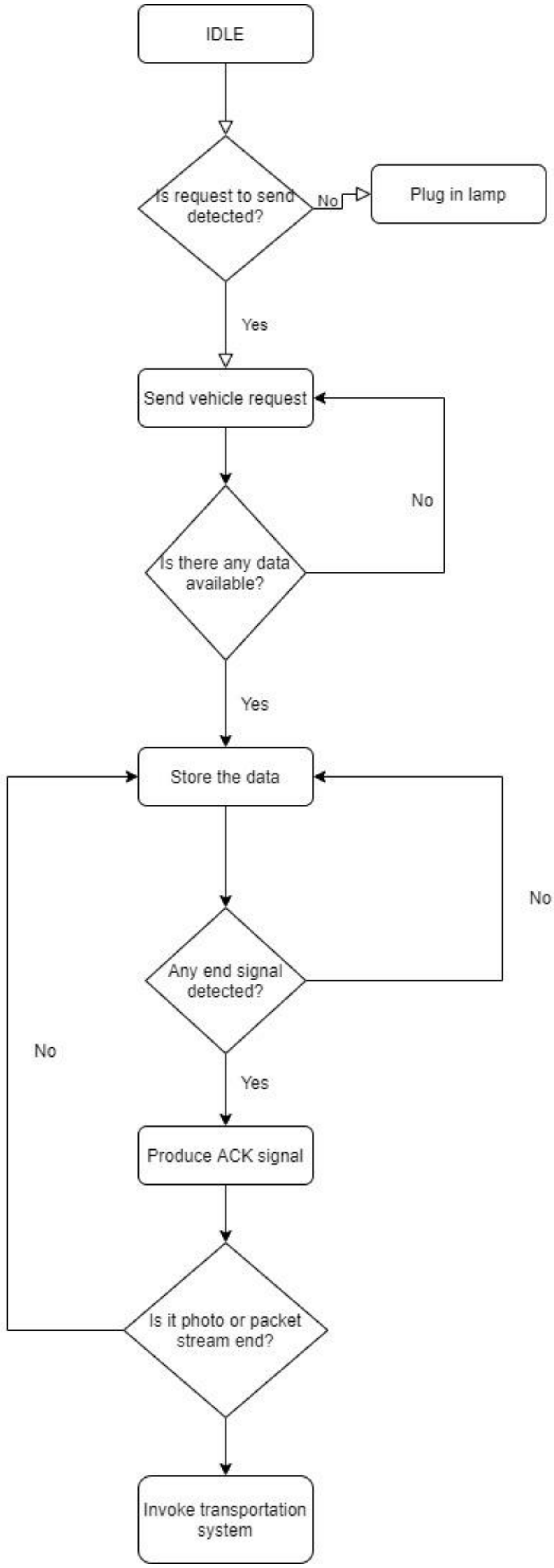


*Figure **7:** The flowchart of the receiver*

2) ***Error Checking***

In order to perform error correction Hamming coding is used. Hamming code is a mechanism which allow us to correct single bit errors in a given bit stream. It achieves that by adding some redundant bits to the actual data bits. In order to find the number of redundant data bits, the following formula is used.

$$2^r \geq m + r + 1 \quad (Equation\ 2)$$

where m is the number of data bits and r is the number of redundant bits. In our case, the number of data bits is equal to 4. Therefore, if we plug m = 4 into the Equation 2**,** we find out that the minimum r value is equal to 3. Therefore, we need 3 extra bits to protect 4 bits. The extra bits are placed to the locations which correspond to the powers of two. It is illustrated in Figure 8 below.

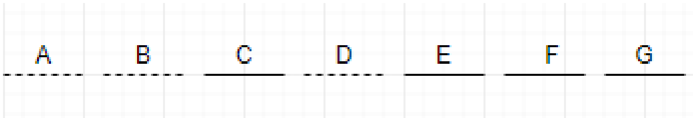


Figure 8: The placement of bits in one data packet

In Figure 8**,** each capital letter represents one bit. The bits A, B and D are the redundant bits and they are placed in the first, second and the fourth places. This placement is consistent with the idea of inserting them to the places which correspond to the powers of two. The rest of the letters C, E, F and G are data bits.

The values of A, B and D are found by using even parity. In our case, A covers the bits C, E and G. Therefore, if the total number of 1's in C, E and G are odd, then A is 1, if not, then A is zero. B covers the bits C, F and G. D covers the bits E, F and G. Even parity is used for B and D, too.

Whenever, the receiver obtains a packet like in Figure 8**,** it performs the logic operation XOR on them in a predefined manner. The exact operations that it performs are as follows.

$$A \oplus C \oplus E \oplus G = P$$

$$B \oplus C \oplus F \oplus G = Q$$

$$D \oplus E \oplus F \oplus G = R$$

After performing these logic operations, it concenates the results as RQP and it treats it as a binary number. If that number is something other than zero, for example 3, then it finds the bit at the third location and toggles it.

3) ***Modulation***

In order to represent the logic 1 and 0, we proposed to use on-off keying. In this modulation method, we assign LED state as ON for binary 1 and OFF for binary zero. The receiver detects these on and off states and interprets them as 1 or 0. For this purpose, we need the UART ports of the Raspberry Pi and Arduino to produce 5V for logic 1 and 0V for logic 0. They achieve this function by the use of serial.write() command in their libraries. In the hardware section, the circuitry for the LED driver will be discussed.

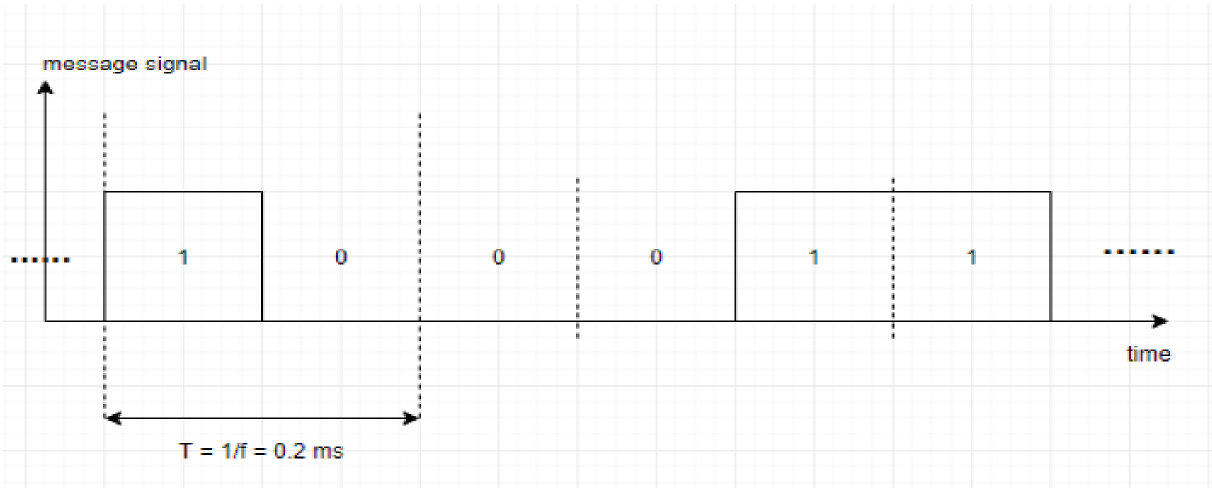


Figure 9: An example message waveform with OOK

In Figure 9 it is possible to see an example message waveform. In this case, the frequency of the waveform is 5 kHz which corresponds to 10 kbps. By utilizing a better circuitry for the receiver terminal, we managed to increase its speed to 30 kbps. Considering the fact that our speed requirement is 13 kbps, we have met that requirement quite well.

<u>Hardware Part of the Communication Subsystem</u>

The previous receiver circuitry was constructed with a transresistance amplifier with LM 741 operational amplifier. The output of LM 741 was coming with 2 V DC bias due to fact that negative bias of the LM 741 opamp was connected to ground. The company planned to use one power source which was used as Vs as seen in Figure 10. The serial connected diodes were used to provide voltage drop ≈ 2V so that voltage output of receiver circuit can have high and low level voltages which can be read by the RX pins of the Arduino or Raspberry Pi for serial communication purposes.

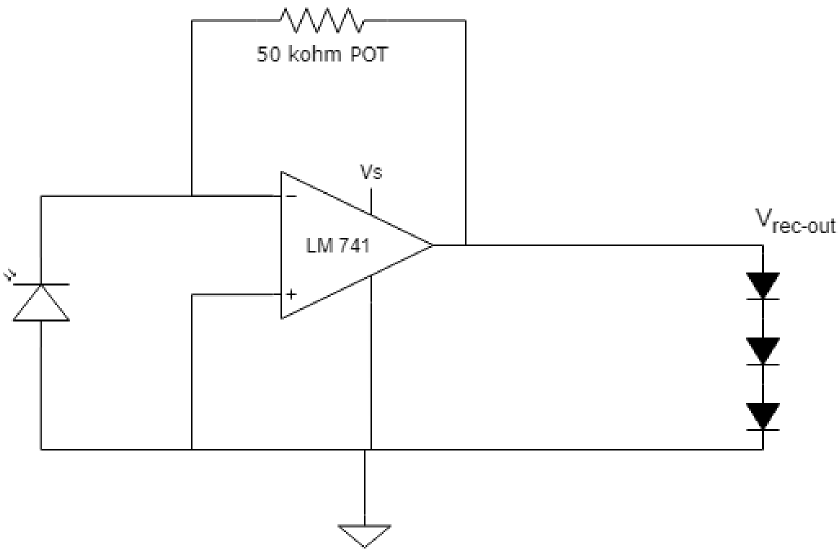


Figure 10: Previous receiver circuit

The circuit was operating in 5-10 cm range and output was very sensitive to external intense light sources. These problems are solved in the improved version of the receiver circuit which is seen in Figure 11.

A high pass filter is connected next to transresistance amplifier. It filtrates the DC part of the output of the transresistance amplifier and an AC signal which oscillates around zero is obtained. This signal is then inputted to a comparator where the reference voltage is 0V.

The resultant circuitry has much better distance and external light insensitivity. Additionaly, it operates at 6V supply which results in less power consumption. In our previous design, we used to operate at 9V level.

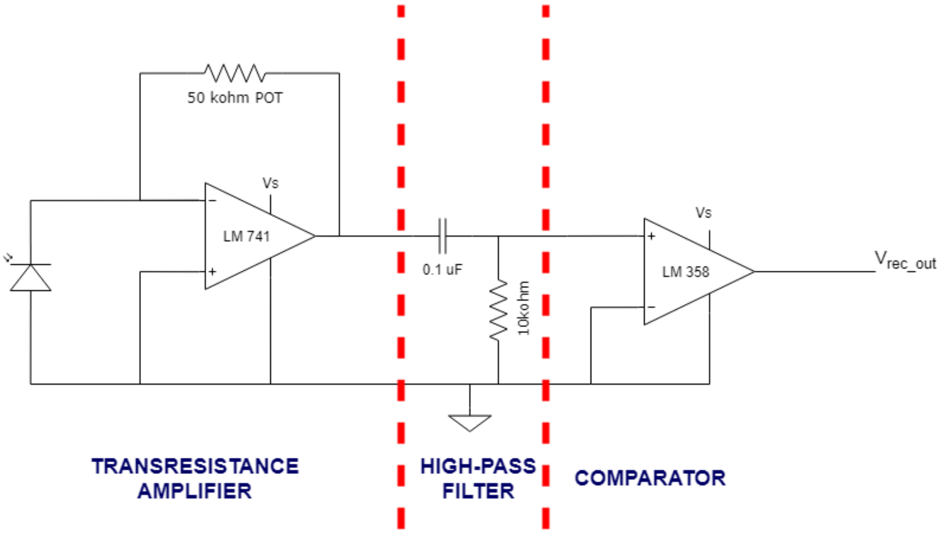


Figure 11: Improved receiver circuit.

An example output of the receiver circuit where receiver and transmitter distance is 30 cm, input is 10 kHz square and the photodiode is under intense light is seen in Figure 12.

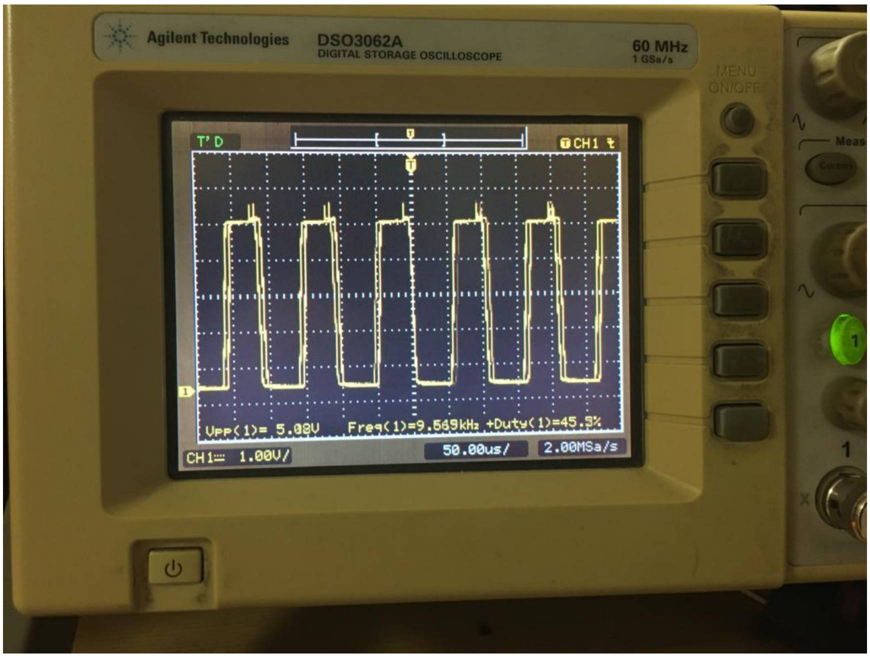


Figure 12: An example output of receiver circuit.

Another part to discuss is the transmitter circuitry. The transmitter circuitry basically consists of an LED driver. The circuit can be seen in Figure 13.

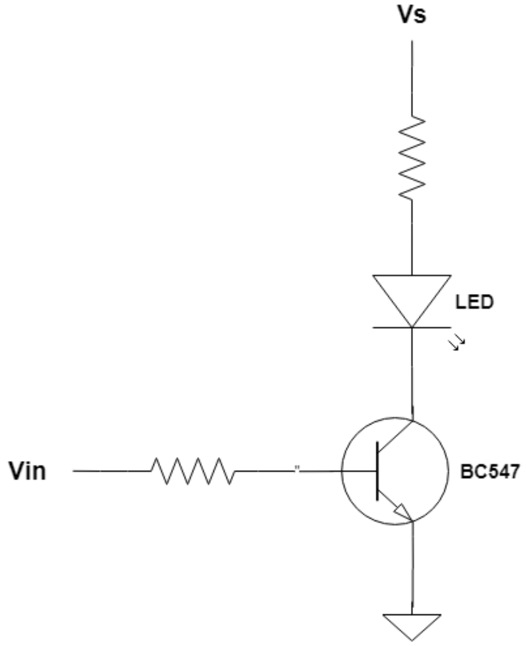


Figure 13: The transmitter circuit

The Vin in Figure 13 is connected to the TX pin of the Raspberry Pi. Whenever the TX pin is HIGH, the LED is ON and whenever it is LOW, the LED is OFF. This way, we can physically realize the OOK modulation.

D.3.2) Image Acquisition Subsystem

Image acquisition subsystem consists of a camera and a Raspberry Pi. The Python will be utilized for the software.

Capturing An Image:

An image will be captured via Raspberry Pi camera module. This camera has 2592x1944 pixel resolution. The command to take the picture will be given by user via a pushbutton. After command of user, 2 seconds will be spent before taking the photo to give the camera's sensor enough time to sense the light levels. [1]

Storing, Resizing and Division of the Captured Image:

After image is captured, it will be stored and processed using the Raspberry Pi placed in the first terminal.

The maximum resolution of camera module is 2592x1944(5 MP) and the minimum resolution is 64x64. The image will be obtained in the maximum resolution and it will be resized using the tools of Python Image Library (PIL). By doing optimizations on the tradeoff between data amount (resolution) and photograph quality (compression), the image file will be saved in JPEG format.

To transfer an image, it is required to convert an image to a bit arrays in such a way that these bit arrays represents the image. The binary data will be split into 5 packets of data chunk by simply splitting the array of data into five blocks of arrays. These packets will be delivered to communication subsystem to be sent via visible light communication. Throughout these steps, tools such as “numpy”, “math”, “os” and “image-slicer” are utilized.

The algorithm followed by the image acquisition subsystem is given in Figure 14 as a flowchart diagram.

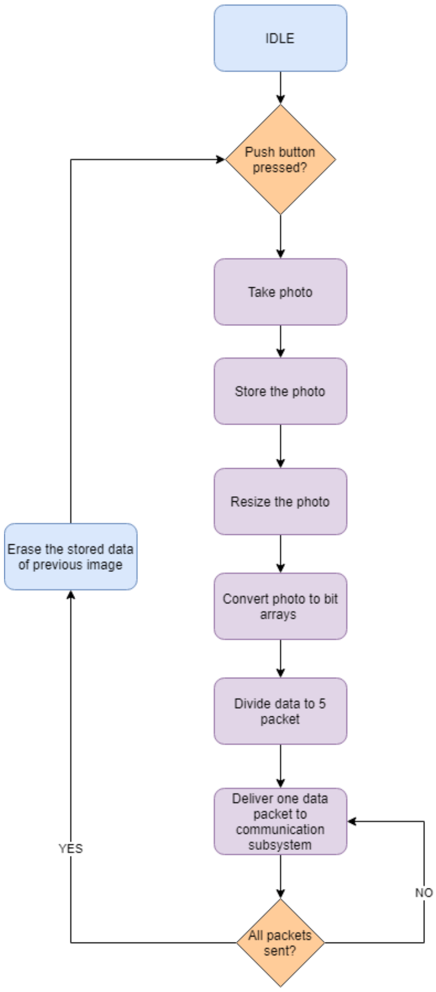


Figure 14: Flowchart of the Image Acquisition Subsystem

D.3.3) Image Reconstruction Subsystem

Image reconstruction subsystem consists of a display and a Raspberry Pi. The Python will be utilized for the software.

Collecting Data:

At the second terminal, communication subsystem will deliver the data that is received by the receiver to the image reconstruction subsystem and data is stored in the Raspberry Pi.

Merging the Data:

Upon receiving unique end signal, the process of collecting data is finished.

The bin files are merged on the general purpose computer as a JPEG file via PIL of Python. As image file is merged the proper signal is generated to initiate the display process.

Displaying the Reconstructed Image:

A Raspberry Pi compatible LCD screen, having a resolution of 128x128, is utilized by using SPI (Serial Peripheral Interface) via its drivers. The screen operates with 3.3 V. [2]

The algorithm followed by the image acquisition subsystem is given in Figure 15 as a flowchart diagram.

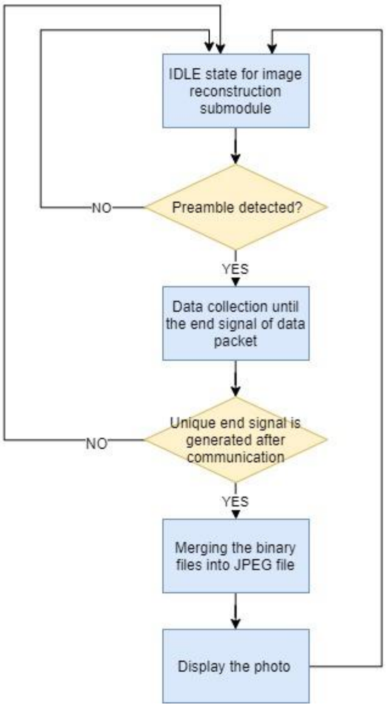


Figure 15: Flowchart of the Image Reconstruction Subsystem

D.3.4) Transportation Subsystem

The transportation subsystem is a complementary part to the Li-Fi communication in order to achieve image transmission between two end terminals where a vehicle transports the data between transmitter and receiver terminals. As the vehicle receives the data from the transmitting terminal with the methods explained in the communication subsystem description it transports the data towards the second terminal, while moving on physically guided tracks, to the location where again Li-Fi communication would take place. When the transported data is transferred to the receiving terminal the vehicle is required to go back to the transmitting terminal to receive the new packet.

The whole image transmission process is required to be done under 2 minutes in which both VLC communication and data transportation by the vehicle must be handled while the distance between the two end terminals are varying up to 1.5 meters. The maximum amount of data that the vehicle can carry is limited to 10kB while the vehicle is expected to do at least 5 laps, going back and forth between the terminals on physically guided tracks, for the transportation of the whole image. By considering the number of minimum laps and the time limitations we have stated in our conceptual design report that the average velocity of the vehicle must be at least 25cm/s for the maximum distance case when the two terminals are 1.5 meters away from each other. In addition to these, the vehicle must come to stop at least 5cm away from the terminals which requires a sensor fed controller considering that the distance between the terminals is variable.

To accomplish all these tasks the design of this subsystem can be divided into three parts which are the control loop design, physical structure of the vehicle and the design of the physically guided tracks that the vehicle will travel on.

Control of the Vehicle:

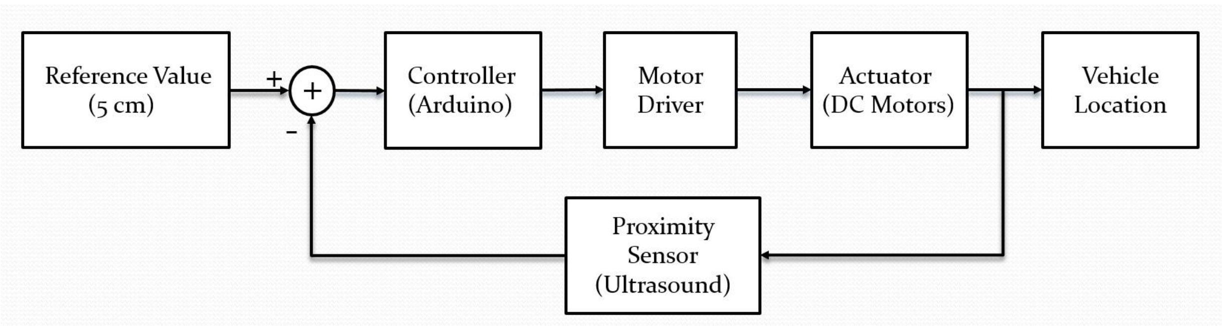


Figure 16: Block diagram of the vehicle transportation control loop

In order to control the vehicle, we designed a closed loop controller as seen in Figure 16 where the distance of the vehicle to the terminals measured by an ultrasound sensor is used as the feedback. We are using four DC motors as the actuators of our vehicle. We are using a L298N motor driver that is simply a double H-Bridge driver circuitry which enables turning the motors in both directions by changing the polarity of the applied voltage using switches as shown in Figurexxxx. We have connected the enable pins of the driver to 5V all the time, calculated the distance info using the sensor input that is fed to the Arduino UNO and fed the input pins of the L298N driver with either 0V or 5V from the digital pins of the Arduino board. We have tested the L298N motor driver with our vehicle and ensured that our vehicle can go back and forth in both ways under the control of an Arduino UNO microcontroller.

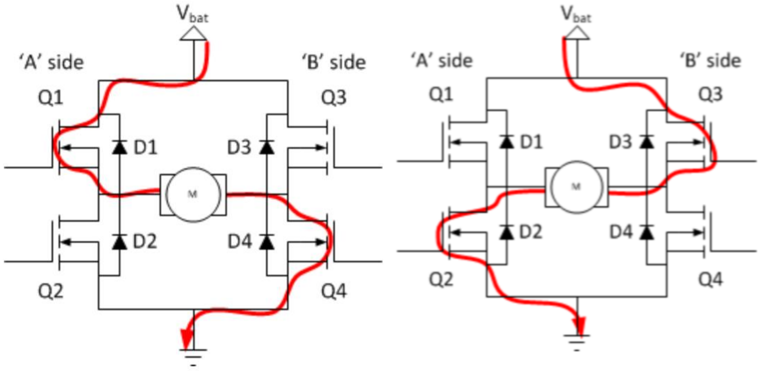


Figure 17: H-Bridge circuitry which enables the rotation of DC motors in both directions

The vehicle would understand that the data is transmitted to its receiver as the specific end signal is received and then it would rush towards the receiving station. At this point, we were planning to increase the speed of the motors gradually but in our tests we observed that applying full voltage of 5V from the beginning does not cause any problems. Doing so does not cause the motors to draw more current than the L298N can handle and mechanically the vehicle is able to withstand to that instant acceleration which makes this method better as we want to be as fast as possible. Throughout the transportation, the controller would get the distance data of the vehicle and as the distance gets smaller than a certain threshold the vehicle starts to brake as we have proposed earlier. The earlier proposed method of deceleration was again to decrease the duty cycle of the PWM signals that are feeding the enable pins of the motor driver. Instead, we learned a new method of braking that can be implemented without varying the duty cycles of the enable pins. By closing the switches labeled as Q1 and Q3 in Figure 17 and opening Q2 and Q4, the stored current inside the DC motor due to its inductive nature is dissipated as heat as it flows through the resistance of the inductor and the on-state resistance of the two FETs. This method enables the slow decay of the stored current which actually results in the quick stopping of the motors. As this method is still causing gradual deceleration rather than locking of the motors we observed in our test that it enables us a smooth enough operation.

By obtaining the distance measurements from the ultrasound sensors at both ends of the car we would apply such a closed loop control of the vehicle where are using kind of an ON/OFF control method where the driver puts the vehicle either in full-gas or in braking mode of operation.

As an alternative method of sensing the distance we have proposed to use an IMU for the case that we are not satisfied with the performance of the ultrasound sensors. However, we found the performance of the ultrasound sensors satisfactory inside range that we are interested in as presented in the test results section of this report.

<u>Physical Structure of the Vehicle and Physically Guided Track:</u>

Vehicle's CAD drawing is given in Appendix XX. Inside the vehicle there are 3 layers of plexiglass for placing the motors, motor driver, battery, Arduino Mega and driver circuitries. At front and back ends of the vehicle, there are ultrasonic distance sensors to calculate the distance between vehicle and the terminals. Wheels of the vehicle will be suitable for operating on the rail. Length of the vehicle is 19 cm and overall height (calculated from the bottom of the wheels to the top of the vehicle) is 20 cm.

For the physically guided track, two parallel u-aluminum profiles with 2 meters length will be used. Terminals will be located on the profile ends and their position will be adjustable on the profile.

D.4) TEST PROCEDURES

There are four main subsystems to be tested in solution proposed by Revolutionary Systems Inc. to "Gimme Fast" project. These are image acquisition subsystem, communication subsystem, transportation subsystem and image reconstruction subsystem. The tests are designed so that the requirements of the each subsystem can be checked whether the implemented parts are satisfying these requirements or not. The procedures that are followed in these tests along with the results are given the following subsections.

D.4.1) Test Procedure for Communication Subsystem

In tests of communication subsystem, Revolusys demonstrated the communication system that utilizes the on-off modulation at 19200 baud with UART pins of the microcontrollers.

The test data stream was upload to input as input and output of the system for transmitter led to receiver photodiode distances from 5 cm to 15 cm (with 1 cm steps) and additionaly 20 cm 30cm and 45 cm for different light conditions (dark, lab light) was observed. This process was repeated 25 times for each data point and the average error was taken into account.

In these, to illustrate the proper operation of the commination system, 1024 bit message signal was transmitted. The number of message bit was selected arbitrarily, the communication system will transmit more than 1024 bits at once on its operation. The transmitted stream utilizes UART chip of the microcontroller and therefore, its start and stop bits are already defined for each data frame.

The received data (with serial read command of Arduino), was read on the COM screen of Arduino and compared with the message signal in MATLAB. The number of error in the received 1024 bit message signal was determined and recorded for different trials.

The setup for lab light condition test is seen in Figure 18. The setup for dark condition test is seen in Figure 19 and Figure 20. In this test, a dark environment was reached via using wooden pieces seen in Figure 19. The light level at this test condition is seen in Figure 20. "1" in these figures shows the LED while "2" shows the photodiode.

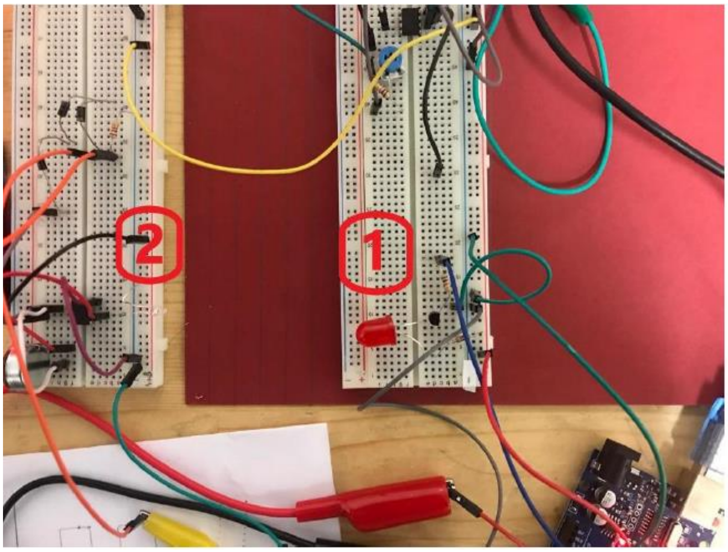


Figure 18: The test setup for lab light condition

Figure 19: The test setup for dark light condition

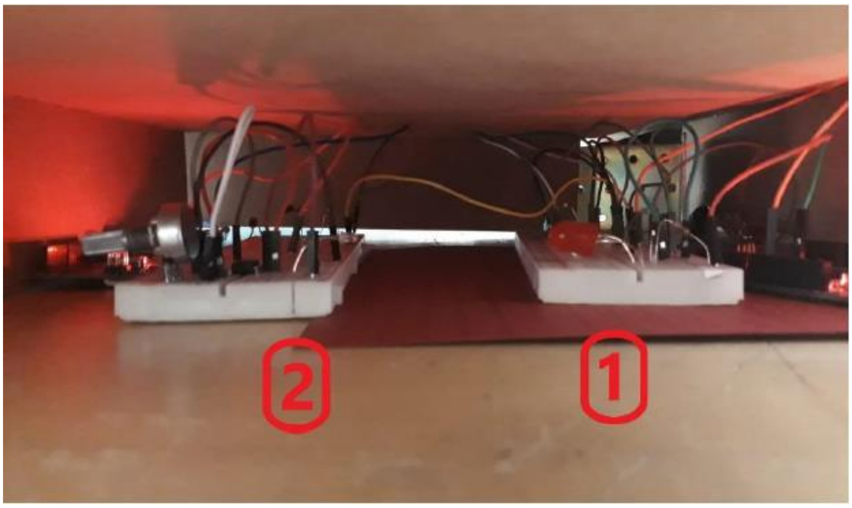


Figure 20: The test setup for dark light condition

D.4.2) Test Procedure for Image Acquisition Subsystem

In the test procedure followed in the image acquisition subsystem, a photo is captured using the Raspberry Pi camera module. Upon capturing the photo, this photo is processed.

In the first part of the test, the dimensions and the quality parameter of the photo are changed and the sizes of the output photos are observed. In the second part of the test, one of the resized images is divided to 5 pieces vertically.

D.4.3) Test Procedure for Image Reconstruction Subsystem

In the test procedure followed in the image reconstruction subsystem, a photo is divided to 5 pieces vertically; these pieces are transformed to bit arrays. The bit arrays represent the data delivered by the communication system. These arrays are then used to construct photos and 5 photos are merged to construct the whole photo back and this photo is displayed at an LCD screen.

D.4.4) Test Procedure for Transportation Subsystem

For testing the vehicle subsystem Revolusys proposed 3 different test procedures:

- Test of proximity sensor
- Test of vehicle's top speed and start-stop cases
- Weight capacity of the vehicle

-Test Procedure for Proximity Sensor

Vehicle uses ultrasonic proximity sensor (HC-SR04) for distance measurement. For testing the proximity sensor, distance will be fixed and for that fixed distance, 100 measurement samples will be taken (there will be 250 ms time delay between each sample.) After measuring for that particular distance, mean and standard deviation of the measurements will be calculated. Then, distance will be changed, and same procedure will be 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.

Since vehicle can approach to the terminal at most 5 cm, it is good to stop the vehicle with some margin which corresponds to stopping 5cm to 10 cm away from the terminal. Therefore, measurements should be precise enough for 5-15 cm range. As a summary, this test investigates the precision of the measurement for various distances and high precision is expected for 5-15 cm range.

-Test Procedure for Vehicle's Movement

Driving the vehicle at its top speed is aimed to meet the time requirement:

- *The average velocity of the vehicle shouldn't be lower than 25 cm/sec for the maximum distance case (1.5 meters).*

For testing the speed of the vehicle, it will be driven at full gas for 3 seconds and braking will be applied. The travelled distance will be measured from which the average speed will be calculated.

-Test Procedure for Vehicle's Weight Capacity

Vehicle's weight capacity has a crucial importance on the robustness of the system and it also affects the top speed of the vehicle. Vehicle has a multi-layered design to decrease the stress on it. Each layer is made from 3mm plexiglass. For testing the robustness of the vehicle, certain amount of load will be placed on each layer and twisting will be checked if there will be any.

D.5) POWER CONSUMPTION

Gimme Fast project is consisting of 4 submodules; communication, transportation, image acquisition and image reconstruction. All these submodules are consuming electrical power and Revolusys Inc. has been tried to find the most effective solution for the power consumption. The first attempt for the solution was to minimize the power required for the circuitries constructed for communication submodule.

The first circuitry constructed by the company was consuming 0.36 W (9 V, 0.04 A). The op-amps utilized at this stage of the project are LM741 op-amps. In the comparator circuitry, the op-amps are changed to LM358 op-amps and with help of other simple alterations, the power consumption is minimized to 0.108 W (6 V, 0.009 A).

For the transportation submodule, it is measured that the vehicle is consuming approximately 2.67 W (8.9 V,0.3 A).

In image acquisition and image reconstruction submodules, two Raspberry Pi general purpose computers are utilized. Each one is consuming 260 mA at 5V (which is about 1.3 W-1.4 W). Total power consumption for these submodules is 2.8 W.

The project is physically separated into three parts; transportation vehicle, receiver and transmitter terminals. Therefore 3 power supplies will be utilized at each physical part. For terminals, Lithium Ion power banks are considered to be used. Power banks will have 10000 mAh capacity at 5 V which is powering the PI's for 38 hours.

For the power supply at the vehicle part, two Li-ion cells will be utilized. Two Li-ion cells will have 2500 mAh at 9 V.

E) INTEGRATION

Transmitter terminal will contain following elements:

- Raspberry Pi 3: For image taking, division and feeding the LEDs
- 1 LED for Communication with Vehicle
- 1 Photodiode for receiving feedback from the Vehicle

- Driver Circuitry for LED and photodiode

Vehicle will contain following elements:

- Arduino Mega for controlling the Vehicle and carrying the received data packets
- 1 Photodiode for Communication with Transmitter
- 1 LED for giving feedback to the terminal (states that loading the data is completed)
- 1 LED for communication with the receiver
- 1 Photodiode for receiving feedback from the Receiver
- Driver Circuitry for LED and photodiode

Receiver terminal will contain following elements:

- Raspberry Pi Zero for image reconstruction
- LCD for displaying the image
- 1 Photodiode for communication with Vehicle
- 1 LED for giving feedback to the Vehicle (states that loading the data is completed)
- Driver Circuitry for LED and photodiode

At last the stage of the physical integration, there will be 3 main bodies to be integrated: Transmitter terminal, vehicle terminal and receiver terminal. All of them will be aligned on the same physically guided track (i.e. rail). At that point, alignment of the LED's on the transmitter with the photodiodes on the vehicle is very important for the accuracy of the communication. Same situation is valid for the other end of the vehicle and receiver terminal. To obtain full alignment, all of the bodies will be produced appropriately. Also, since all bodies have its own power source, there will be no power distribution line between terminals and vehicle. Overall system tests and optimizations will follow the physical integration.

F) TEST RESULTS

F.1) Test Results for Communication Subsystem

The data obtained in the measurements are plotted as in Figure 21. In the graph, it can be seen that the error rate for laboratory light and dark conditions are pretty close to each other until 15 cm. However, after that point, the difference between them grows largely. According to these test results, we are planning to operate the vehicle in the range of 5-15 cm. This way, we can obtain a system which is more robust to the changing light conditions.

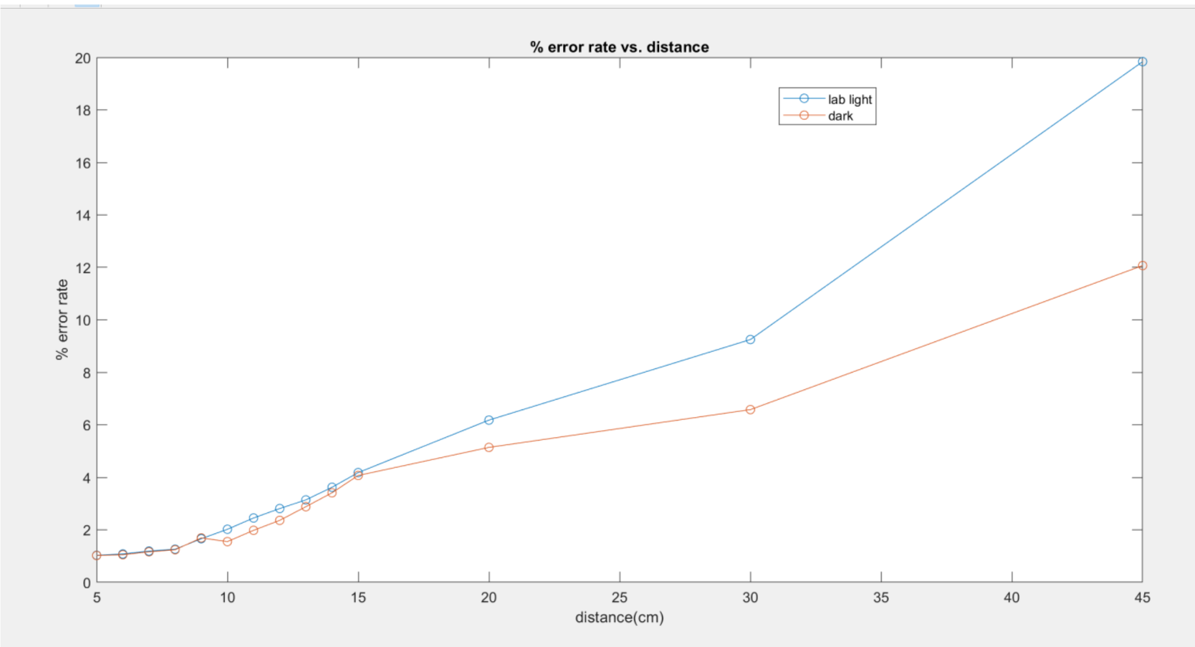


Figure 21: The bit error rate graph of the test results with respect to the distance

The requirements which are discussed in the beginning of this report regarding "Gimme Fast" project and related with the communication subsystem are listed and the results of the conducted tests are related with them in the following paragraphs.

a. Some portion of the photo must be transmitted to the vehicle by VLC (Visible Light Communication). (functional requirement)

b. The data packets carried by the vehicle needs to be delivered to the receiver terminal. (functional requirement)

In the conducted tests, a photo is not transmitted but a 1024 bit message signal is successfully transferred from transmitter part to receiver part.

c. A minimum DTR (data transfer rate) of 13 kbps will be achieved. (performance requirement)

In the conducted tests, data transfer rate was 19200 baud which is equal to 15 kbps. Therefore, we can safely say that the communication subsystem meets the speed requirement.

d. The minimum accuracy rate of 90% should be achieved for the reconstructed image. (performance requirement)

In the conducted tests, no picture was reconstructed but considering the percentage error at the received data, communication system was successful at all light conditions at a distance varying from 5 cm to 15 cm.

F.2) Test Results for Image Acquisition Subsystem

In the first part of the tests, an example photo captured via Raspberry Pi camera module is seen in Figure 22. The dimensions of this photo are 2592*1944 and its size is 2.7 MB. The dimensions of the photo are constant while its size is varying at different photos.

Figure 22: The photo captured by the Raspberry Pi camera module.

The dimensions of 256*512 and 256*256 are tested for quality values of 100%, 90%, 80% and 50%. The obtained photos are seen in Figure 23 and Figure 24. The sizes of the output photos are given in the Table 1.

(23.a) *(23.b)* *(23.c)* *(23.d)*

*Figure 23: The output photos where the dimensions are decreased to 256*512 and quality is decreased (100% in 23.a, 90% in 23.b, 80% in 23.c and 50% in 23.d).*

(24.a) *(24.b)* *(24.c)* *(24.d)*

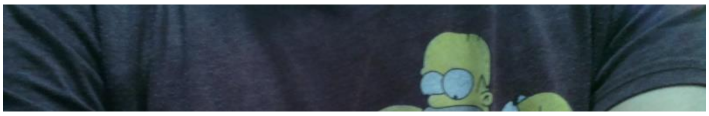
*Figure 24: The output photos where the dimensions are decreased to 256*256 and quality is decreased (100% in 24.a, 90% in 24.b, 80% in 24.c and 50% in 24.d).*

TABLE 1: Size of the Image vs Quality and Photo Dimensions

Quality	**Photo Dimensions**	
	256*256	**256*512**
100	50.2 kB	91.6 kB
90	17.2 kB	30.3 kB
80	11.6 kB	20.4 kB
50	6.7 kB	11.6 kB

The requirement related with the image acquisition subsystem is “The system must be able to take a photo.” As seen at the tests, image acquisition subsystem is able to take photo. Furthermore, this photo is successfully resized.

At the second part of the test, photo seen in Figure 1 is vertically divided to 5 pieces. These new photos are given in Figure 25. These photos are the converted to bit arrays.

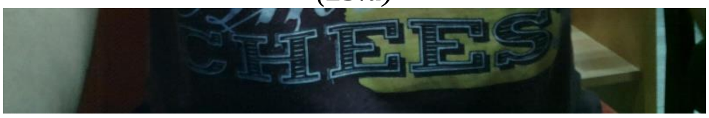


(25.a)

(25.b)

(25.c)

(25.d)



(25.e)

Figure 25: The vertical slices of the photo given in Figure 22, which are obtained using Python Image Library.

Therefore, the image acquisition subsystem is successfully divided to 5 pieces.

F.3) Test Results for Image Reconstruction Subsystem

The photos seen in the Figure 25 are converted to bit arrays. Then these bit arrays are concatenated and an image is successfully reconstructed using relevant Python tools. This photo is then displayed in an LCD screen. The requirement related with the image reconstruction subsystem is "As the full photo is delivered, the photo must be displayed at the receiver terminal." Although the data used to reconstruct a photo was not delivered by communication subsystem, it is seen that image reconstruction subsystem is able to construct an image from the inputted bit arrays and display it.

F.4) Test Results for Transportation Subsystem

- **Test of Proximity Sensor**

Test procedure is proposed at the relevant section. Following results are obtained which are given in Table 2.

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**

When vehicle is driven at full gas for 3 seconds and then braking is applied, it travels a distance of 1 meter. This corresponds to 33 cm/sec where the requirement is 25 cm/sec. This requirement is satisfied easily.

Also, when vehicle is driven at its top speed, no slippage appears at the start. And no problem appears when it stops, it brakes smooth enough without going extra distance.

- **Test of Vehicle's Weight Capacity**

Vehicle consists of 3 plexiglass (with 3 mm thickness) layers:

- First layer is reserved for motors and motor controller. (98 grams)
- Second layer is reserved for Arduino Mega and battery. (127 grams)
- Third layer is reserved for LEDs & photodiodes with their driving circuitries. (60 grams)

There is no twisting of the layers under that conditions which is good for proper operation.

G) PROJECT MANAGEMENT

G.1) GANNT CHART

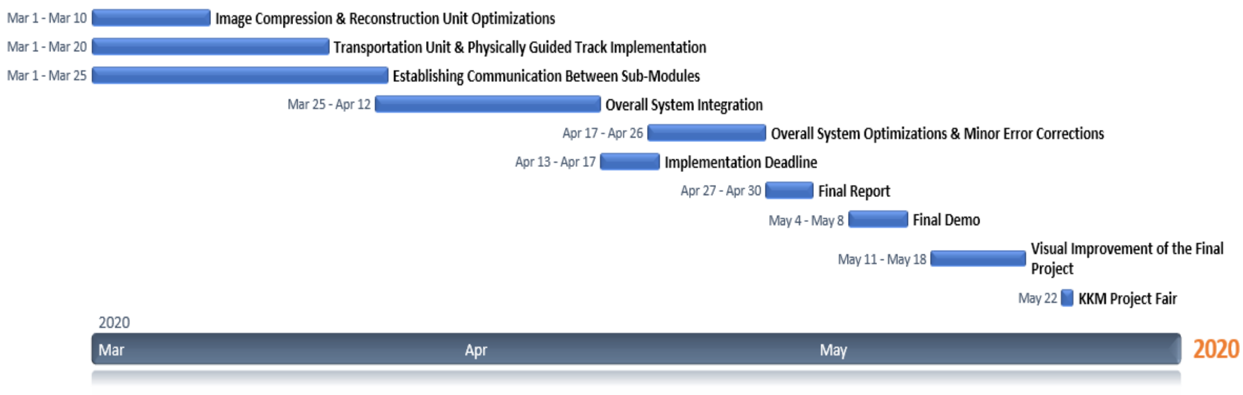


Figure 26: Gannt Chart of the Project Schedule

- Ahmet Demirdaş and Onur Akdeniz will be responsible from Image Compression and Reconstruction Unit optimizations.
- Mert Eyüboğlu and Ozan Berk Boyraz will be responsible from Transportation Unit and Physically Guided track implementations.
- Doğukan Atik ,Mert Eyüboğlu and Ahmet Demirdaş will be responsible from Establishing Communication Between Sub-Modules.
- All team members will participate to the Overall System Integration and Optimization.
- Final Report will be written with the contribution of all of the members. All members will represent the final product of the Revolusys together at Demonstrations and KKM Fair.

G.2) COST ANALYSIS

EXPENSE ITEMS	**COSTS**
Raspberry Pi 3 (To be used in transmitter terminal)	205 TL
Raspberry Pi Zero (To be used in transmitter terminal)	130 TL
Arduino Mega + Vehicle Kit with DC Motor Driver (to implement the vehicle with microcontroller embedded on it)	152.5 TL

Raspberry Pi 3 Camera Module (to take the photo at the transceiver terminal)	16 TL
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 receiver 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
Others(Jumpers, breadboard, PCB Plexiglass,Wheels,3D printing costs etc.)	100 TL
Batteries	160 TL
TOTAL COST	**870 TL (140 $)**

H) DELIVERABLES

Equipment

- Vehicle

The user will be provided with a vehicle which has a transceiver embedded on it. The transceiver unit includes 4 LEDs and 4 photodiodes. The vehicle is able to detect the terminal and accelerate or decelerate accordingly.

- Physically Guided Track

The user will be provided with a 1.5 meters long aluminum constructed rail on which the vehicle can move.

- Transmitting Terminal

The user will be provided with a rectangular prism shaped transmitting terminal which contains a camera, to take a photo, transmitter unit which consists of 4 LEDs and its own computational unit.

- Receiving Terminal

The user will be provided with a rectangular prism shaped receiving terminal which contains a receiver unit consisting 4 photodiodes, a 3.5 inch LCD screen to display the taken photo and its own computational unit.

Documents

- Warranty

Revolusys Inc. provides two (2) years warranty in both transportation and communication breakdowns of the system except the user faults.

- Manual

A manual will be provided to the users to get informed about the utilization and the maintenance of the system.

I) CONCLUSION

By the end of this report we, as Revolusys Inc., have decided on and presented our final solution methods. We have accomplished to test and verify that our solution procedures for the main subsystems are capable of meeting the requirements of the 'Gimme' Fast project. The next step would be to perfect the implementation of our solutions for the subsystems and then to enable the interaction of the subsystems with each other.

After the interaction of the subsystems is completed which means that we are meeting the technical requirements and objectives of the project, we would work the physical appearance and neatness of our overall system and would prepare the deliverables so that our product is ready to meet with the customers.

Revolusys Inc. is aware of the fact that a busy and probably a stressful process is awaiting for us through the completion of this project. We are aware of the fact that we may face many problems during the process of integration of the subsystems but in aware of this we are going to work intensely on this subject throughout the following two months. By the end of this process, we are sure about that we will complete the project by success and win the appreciation of our customers.

J) REFERENCES

[1] “Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2017–2022 White Paper,” Cisco, 19-Feb-2019.

[Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-738429.html. [Accessed: 01-Nov-2019].

[2]L. U. Khan, “Visible light communication: Applications, architecture, standardization and research challenges.”

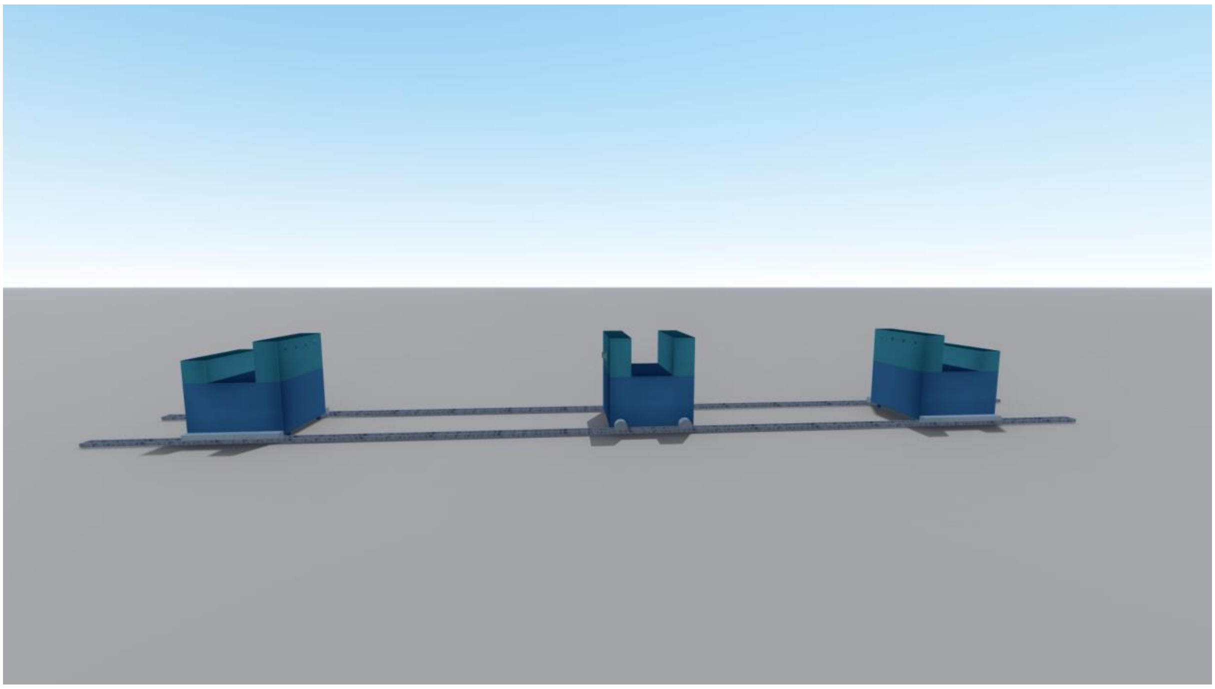
[Online]. Available: https://core.ac.uk/download/pdf/82741711.pdf

[3]“Camera Module,” Camera Module - Raspberry Pi Documentation. [Online]. Available: https://www.raspberrypi.org/documentation/hardware/camera/. [Accessed: 27-Feb-2020].

[4] “Raspberry Pi 1.4 inç 128x128 Çözünürlük LCD Ekran Uygun Fiyatıyla Satın Al - Direnc.net®,” https://www.direnc.net/. [Online]. Available: https://www.direnc.net/raspberry-pi-lcd-ekran-128x128-1-44-inch?language=tr&h=016cc0cd&gclid=EAIaIQobChMI443l0Nrx5wIVC7DtCh1Pqgj_EAQYASABEgKW2_D_BwE. [Accessed: 27-Feb-2020].

K) APPENDIX

K.1) Techical Drawing of the System



K.2) Technical Drawing of Vehicle