FINAL YEAR PROJECT

COLLISION DETECTION SYSTEM USING LIFI TECHNOLOGY

DONE BY

M.UJWAL KUMAR(15EUCS150)

R.VISHNU(16EUCS523)

K.SREE VISHNU(16EUCS529)

GUIDED BY

MAJIDHA FATHIMA K.M

ASSISTANT PROFESSOR

COMPUTER SCIENCE DEPARTMENT

AT

SRI KRISHNA COLLEGE OF ENGINEERING AND TECHNOLOGY

COIMBATORE

TABLE OF CONTENTS

ABSTRACT

As the result of development of technology day after day people are mostly depending on gadgets. These gadgets are getting smarter enough to full fill the needs of human beings. They are improvised in such a way that they are accurate in their work even without intervention of human beings. As they are the requirement of the hour. These smart gadgets are functioned in a they are interconnected with all other objects around them. They use a technology IOT. Also known as internet of things. Nowadays they are spread across the fields like healthcare, security, defence etc. One such establishing field is Auto mobile communication. This helps in establishing a smart network communication between commuting vehicles. As the technology is evolving a new era of smart auto motives are a head in future. In order to enforce a smart communication which helps in keeping Sync in between auto mobiles with in a range and to message immediately by plotting the exact location of the victim. In terms of emergency this system is used. This system functions by communicating via light. Also known as light fidelity which is cost effective and can provide high band width compared to radio waves. With a simple set up requirement of LED which is already fixed to an auto mobile we can establish communication via light as a medium to transmit data to receiver which can process data using ARDUINO microcontroller on both ends of the receiver and transmitter can regulate the speed of the auto mobile come back with a high speed than required. And it can notify a message via light communication. This could intern reduce the probability of accident occur. LIFI is the budding technology which can be the future technology for data transmission at faster rates compared to radio waves and can help to reduce the consumption of electricity used by base stations which generate radio signals.

CHAPTER-1

INTRODUCTION:

1.1 OVERVIEW

As day by day increasing usage of auto mobiles and the improper measures for controlling. Traffic in our country is going on increasing. We need to have a good control and command on our vehicle as there are few in number. Accidents rate is increasing and even after any such incident no immediate precaution can't be provided by government to save people life because of a poor communication which leads to death of the people. In order to solve this issue with technological advancements prevailing today we can create smart systems which can communicate among themselves. So, through communications we avoid dangers while commuting. This communication is established by LIFI also known LIGHT FIDELITY which can be easily implemented without the necessity of any extra components. This can be implemented via LED which is already attached to an auto mobile. Communication takes place in a line of sight model. This circuit has both transmitter and receiver attached to it. Transmitter transmits signals in the form of light signals and it gets received at the end of Receiver. Paragraph below describes about the LIFI technology in detail.

1.2 LIFI TECHNOLOGY

LIFI means light fidelity as the name suggest that this communication is related to light it Uses light as a medium for the data communication. it uses visual light which is one type of Radiation signal in the electromagnetic spectrum. Its frequency of operation is higher Compared to radio waves and has less inability towards wavelength compared to radio Waves. This LIFI technology is advanced version of the VLC visual light communication. We see how the VLC works and then differentiate it with LIFI. VLC uses led to transmit Data wirelessly by using intensity modulation (IM). at the receiver the signal is detected by Photodiode (PD) and by using the principle of direct detection (DD). VLC is conceived as the Point to point data communication technique -essentially as a cable replacement. But LIFI Contrast describes a complete wireless networking system. this includes bi-directional Multiuser communication. Which enables point-to-multipoint and multipoint-to-point Communication. LIFI also involves multiple access points forming a wireless network of Very small optical attocells with seamless handover. this means LIFI enables full user Mobility. It's natural beam forming technique enables local containment of LIFI signals. And Because of blockage of the signals by opaque walls physical layer security can be enhanced. LIFI has its own layer of protocols which are required for establishing communication via Light. Precise channel models are required which take the spectral composition of the signal In to account. link level algorithms are required to optimally shape the signal to maximize the data output. In the context due to the positivity of the power signals in IM a new Theoretical framework is needed to establish the channel capacity. a new medium access Layer is required. Following below illustrates a layered protocol diagram of LIFI protocol Stack.

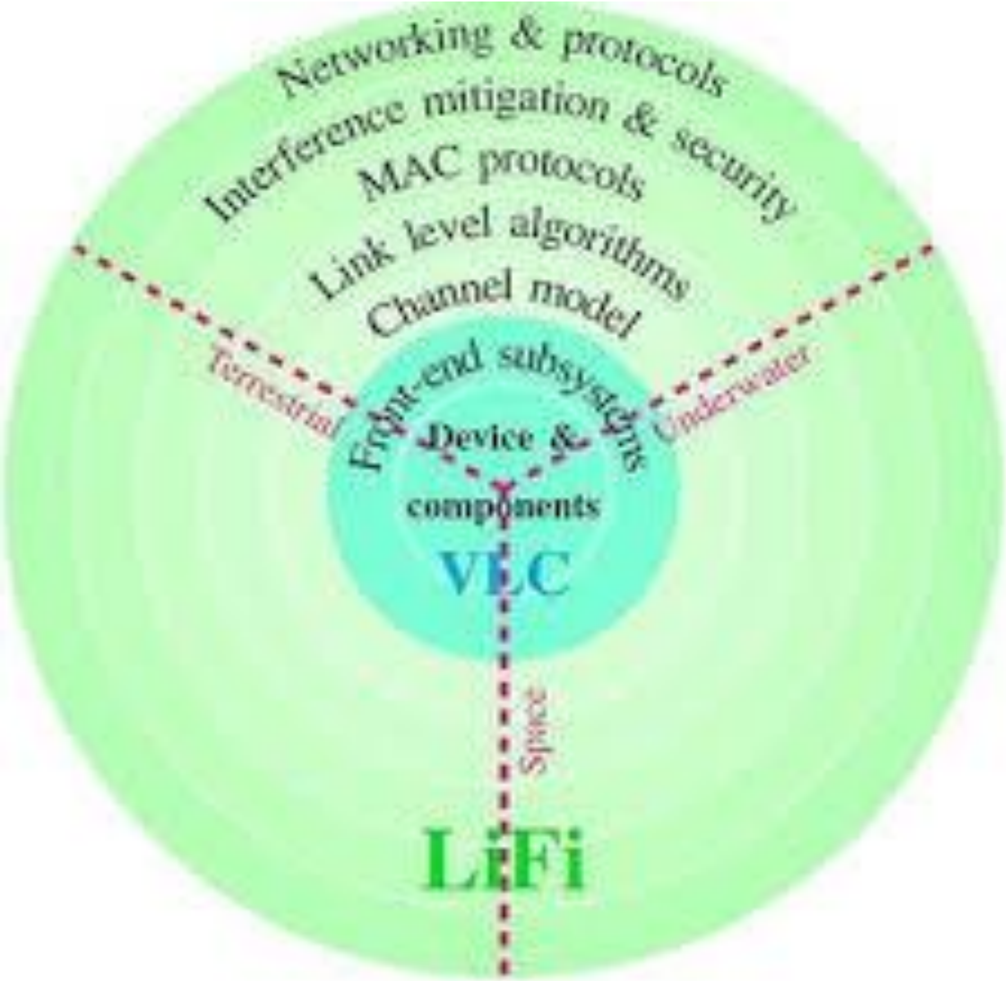


FIG -1.2.1 illustrates the lifi protocol stack with its usage in different environments

1.3LED TECHNOLOGY

The below diagram for a diode an LED will begin to emit light when the on-voltage is Exceeded. Typical on voltages are 2-3 Volt like a normal diode, the LED consists of a chip of semiconducting materials impregnant or doped, with impurities to create a p-n Junction. As in other diodes, current flows easily from the p-side or anode to the n-side or Cathode, but not in reverse direction.

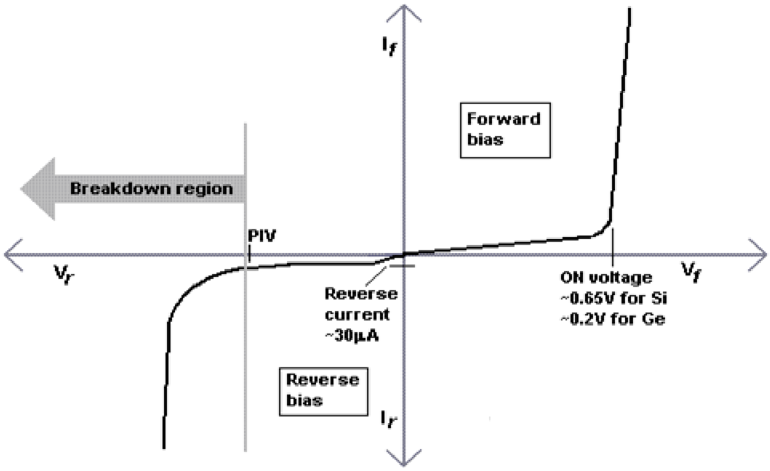


FIG 1.3.1 THE INNER WORKINGS OF AN LED

1.4 BLOCK-DIAGRAMS WITH DESCRIPTIONS:

1.4.1 SYSTEM-ARCHITECTURE:

Below diagram illustrates Two circuit's which are known as LIFI-TRANSMITTER and LIFI-RECEIVER controlled by microcontroller each for processing and receiving data Uses Led as a main source for transmitting and receiving data via light.

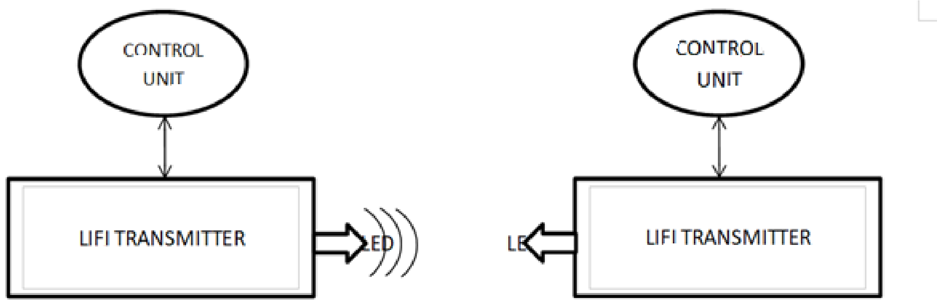


FIG 1.4.1 SYSTEM ARCHITECTURE

1.4.2 LIFI-TRANSMITTER:

LIFI-TRANSMITTER has a led setup up connected to Arduino microcontroller with LCD for displaying information. ultrasonic sensor for detecting the range and a buzzer for Alarming purpose if the distance measured is less than threshold value. This Processed by Arduino and sent to receiver via led which emits light as a beam. An there Is an alerting system which sends emergency messages by using gsm module.

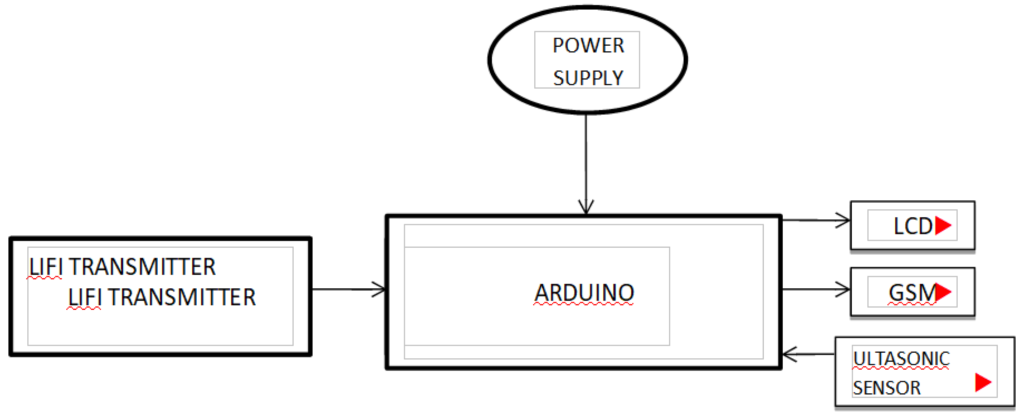


FIG 1.4.2 LIFI TRANSMITTER

1.4.3 LIFI-RECEIVER:

LIFI-RECEIVER on the other hand has a led which receives the voltage change and it convert In to user data by Arduino microcontroller. This distance should be converted in to speed and this speed should be converted in to a degree of rotation used in the speedometer. If the distance is less than threshold value GPS system enables the live tracking of the Auto mobiles.

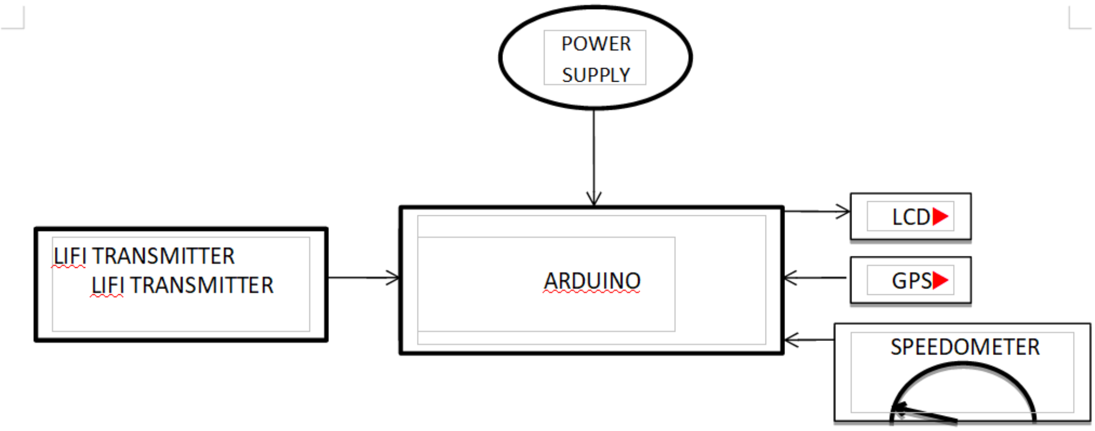


FIG 1.4.3 LIFI RECEIVER

1.5 ADC CONVERTER:

WORKING OF ADC CONVERTER:

The Analog to Digital Converter (ADC) is used to convert an analogy voltage (a voltage that vary continuously within a known range) to a 10-bit digital value. For instance, it can be used to log the output of a sensor (temperature, pressure, etc) at regular intervals, or to take some action in function of the measured variable value. There are several types of ADC. The one used by Arduino is of the "successive approximation ADC" kind. The following is a simplified scheme of the ADC.

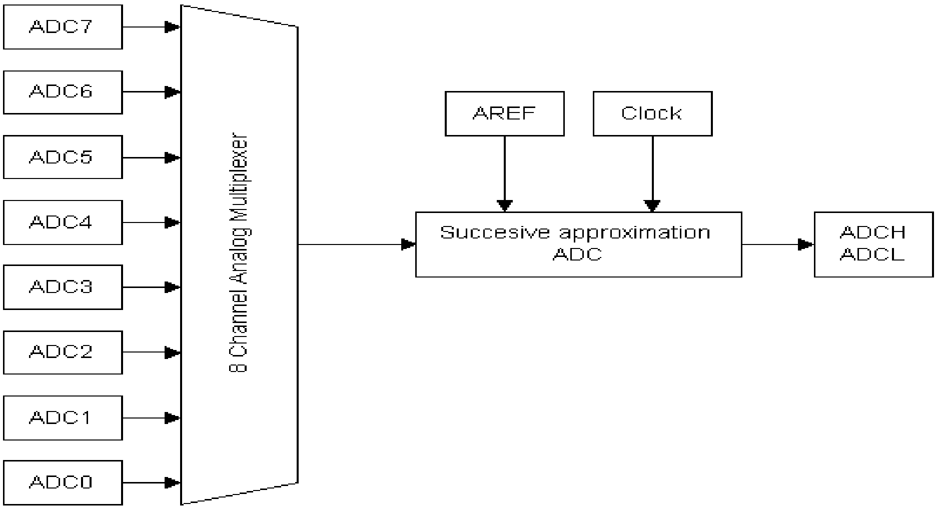


FIG 1.5.1 ADC CONVERTER REGISTER

The analog voltage at the input of the ADC must be greater than 0V, and smaller than the ADC's reference voltage AREF. The reference voltage is an external voltage you must supply at the Aref pin of the chip. The value the voltage at the input is converted to can be calculated with the following formula:

ADC conversion value = round ((vin/vref) *1023)

There are three types of registers in ADC naming

1) ADMUX
2) ADCSR
3) ADCL/ADCH

1)ADMUX (ADC MULTIPLEXER AND SELECT REGISTER):

----------	----------	---------	--------	-----------	MUX2	MUX1	MUX0

This register is used to select which of the 8channel (between ADC0 to ADC7) will be the input to the ADC. Since there are 8 possible inputs, only the 3 least significant bits of this register are used. The following table describe the setting of ADMUX.

INPUT FORMAT IS AS SPECIFIED IN THE TABLE BELOW:

MUX 2	MUX 1	MUX 0	SPECIFIED
0	0	0	ADC0
0	0	1	ADC1
0	1	0	ADC2
0	1	1	ADC3
1	0	0	ADC4
1	0	1	ADC5
1	1	0	ADC6
1	1	1	ADC7

2)ADSCR (ADC CONTROL AND STATUS REGISTER):

ADEN	ADFC	ADFR	ADIF	ADIE	ADPS2	ADPS1	ADPS0

ADEN (ADC Enable) bit: Setting this bit enables the ADC. By clearing this bit to zero, the ADC is turned off. Turning the ADC off while a conversion is in progress will terminate this conversion.

ADSC (ADC Start Conversion) bit: In Free Running Mode, you must set this bit to start the first conversion. The following conversions will be started automatically. In Single Conversion Mode, you must set it to start each conversion. This bit will be cleared by hardware when a normal conversion is completed. Remember that the first conversion after the ADC is enabled is an extended conversion. An extended conversion will not clear this bit after completion.

ADFR (ADC Free Running Select) bit: If you want to use the Free Running Mode, you must set this bit. ADIF (ADC Interrupt Flag) bit: This bit is set when an ADC conversion is completed. If the ADIE bit is set and global interrupts are enabled, the ADC Conversion Complete interrupt is executed. ADIF is cleared by hardware when executing the corresponding interrupt handling vector. Alternatively, ADIF is cleared by writing a logical 1 (!) to the flag. This has a nasty side effect: if you modify some other bit of ADCSR using the SBI or the CBI instruction, ADIF will be cleared if it has become set before the operation.

ADIE (ADC Interrupt Enable) bit: When the ADIE bit is set and global interrupts are enabled, the ADC interrupt is activated and the ADC interrupt routine is called when a conversion is completed. When cleared, the interrupt is disabled.

ADPS (ADC Prescaler Select) bits: These bits determine the division factor between the AVR clock frequency and the ADC clock frequency. The following table describe the setting of these bits:

ADPS2	ADPS1	ADPS0	DIVISION
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

3)ADCL AND ADCH (ADC DATA REGISTER HIGH AND ADC DATA REGISTER LOW)

These registers hold the result of the last ADC conversion. ADCH holds the two most significant bits, and ADCL holds the remaining bits.

When ADCL is read, the ADC Data Register is not updated until ADCH is read. Consequently, it is essential that both registers are read and that ADCL is read before ADCH.

The AT mega series of AVRs have a more complex ADC. They are similar to the ADC explained here, but have some additional features like (see the datasheet for the details):

- 7 Differential Input Channels
- 2 Differential Input Channels with Optional Gain of 10x and 200x(1)
- Optional Left adjustment for ADC Result Readout
- Selectable 2.56V ADC Reference Voltage
- ADC Start Conversion by Auto Triggering on Interrupt Sources

1.6 MANCHESTER ENCODING:

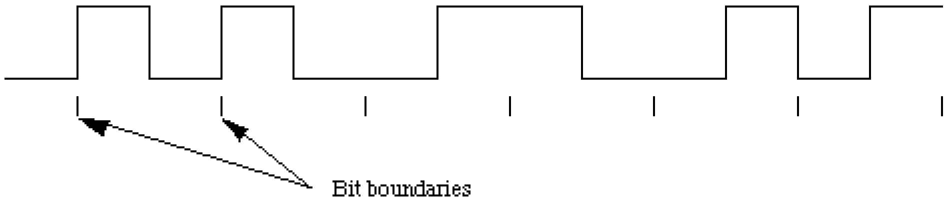
Manchester encoding (first published in 1949) is a synchronous clock encoding technique used by the physical layer to encode the clock and data of a synchronous bit stream. In this technique, the actual binary data to be transmitted over the cable are not sent as a sequence of logic 1's and 0's (known technically as non-return to zero (NRZ). Instead, the bits are translated into a slightly different format that has a number of advantages over using straight binary encoding (i.e. NRZ).

In the Manchester encoding shown, a logic 0 is indicated by a 0 to 1 transition at the centre of the bit and a logic 1 is indicated by a 1 to 0 transition at the centre of the bit. Note that signal transitions do not always occur at the ‘bit boundaries’ (the division between one bit and another), but that there is always a transition at the centre of each bit. The Manchester encoding rules are summarised below:

ORGINAL DATA	VALUE SENT
LOGIC O	O TO 1 UPWARD TRANSITION BIT AT BIT CENTER
LOGIC 1	1 TO 0 DOWNWARD TRANSITION BIT AT BIT CENTER

Note that in some cases you will see the encoding reversed, with 0 being represented as a 0 to 1 transition. The two definitions have co-existed for many years. The Ethernet Blue-Book and IEEE standards (10 Mbps) describe the method in which a Logic 0 is sent as 0 to 1 transition, and a Logic 1 as a one to zero transition (where a zero is represented by a less negative voltage on the cable). Note that because many physical layers employ an inverting line driver to convert the binary digits into an electrical signal, the signal on the wire is the exact opposite of that output by the encoder. Differential physical layer transmission, (e.g.10BT) does not suffer this inversion.

The following diagram shows a typical Manchester encoded signal with the corresponding binary representation of the data (1,1,0,1,0,0) being sent.



The waveform for a Manchester encoded bit stream carrying the sequence of bits 110100.

FIG 1.6.1 MANCHESTER ENCODING

CHAPTER-2

LITERATURE SURVEY

2.1 WHAT IS LIFI:

AUTHORS: HARALD, HASS.

PUBLICATION YEAR:2014.

DESCRIPTION:

In this research paper author describes the difference between VLC and LIFI and also in details about the LIFI construction Architecture along with-it characteristics with the layered network Protocol designed for the transmission of data via LIFI.

2.2 LIFI: CONCEPTIONS MISCONCEPTIONS AND OPPORTUNITIES [5].

AUTHORS: HARALD HASS.

PUBLICATION YEAR:2015.

DESCRIPTION:

In this research paper author details about the concepts of lifi and also Disclaim about the misconceptions of LIFI which are not true as per People's knowledge and also describes the opportunities of the LIFI in the technologically developing world.

2.3 EVALUATION OF COST-EFFECTIVE SENSOR COMBINATIONS FOR A VEHICLE PRE-CRASH DETECTION SYSTEM [10].

AUTHOR: JOHN CARLIN, WILLIAM THOMSON, DANIEL KAWANO, CHARLES BIRDSONG, PETER SCHUSTER.

PUBLICATION YEAR:2012

DESCRIPTION:

This paper describes about the the sensors which are used for vehicle Collision Detection. Author describes the range of sensors which are used in the field Of automobiles for distance measuring and providing communication. He tells the Characteristic of the sensors ranging their functions with cost effective solutions etc.

2.4 SMART VEHICULAR COMMUNICATION SYSTEM USING LIFI TECHNOLOGY [4].

AUTHOR: POOJA BAHATELEY, S. BALAJI FROM VIT INSTITUTE

PUBLICATION YEAR:2015

DESCRIPTION:

Authors in this paper describes the idea behind the project used he Describes that a white light which is fixed to the automobiles is enough. To establish a communication via lifi. The two leds each at front and Rear of the automotive is used as light source and photodetector data. Can be encoded by using encryption algorithms.

2.5 VEHICLE TO VEHICLE COMMUNICATION PROTOCOL FOR COOPERATIVE COLLISION WARNING.

AUTHOR: NITHIN H VAIDYA, JIE LIU, FENG ZHAO, MICROSOFT CORP.

PUBLICATION YEAR: 2015

DESCRIPTION:

This paper proposes a vehicle-to-vehicle communication protocol for Cooperative collision warning this paper describes the importance of achieving low Latency in delivering emergency warnings in various road situations. Based on careful Analysis of application requirements, we design an effective protocol, comprising Congestion control policies, service differentiation mechanisms and methods for Emergency warning dissemination.

2.6 LIFI TECHNOLOGY-VEHICLE TO VEHICLE DATA TRANSMISSION [3]

AUTHOR: JAGADISH A, PATEL, SAGAR C, SHUBHANGI S, SHINDE, YASWANT N, EMANE, SUVARNA B, MAHAJAN.

PUBLICATION YEAR: 2015

DESCRIPTION:

In this research paper authors dealt with the vehicle to vehicle date Transmission. They had presented initial designs and result of a small-Scale prototype using LIFI Technology. They used LED bulb for Communication establishment. They aim for designing This system is Highly reliable which give desired data transmission between vehicle To-Vehicle by using transmitter and receiver mounted on vehicle.

CHAPTER 3

PROBLEM DEFINITION:

AUTOMOBILE COLLISION DETECTION SYST USING LIFI TECHNOLOGY.

3.1 EXISTING SYSTEM:

DESCRIPTION:

Existing system deals with the communication establishment between auto mobiles using LIFI Technology. LIFI system is achieved by led lights and photo-transistors. There is a led in Transmitter system and a photo-transistor in receiver module. Transmitter module has an LCD, ultrasonic sensor for detecting distance. And the receiver module has Servo-motor attached to It to regulate the speed. Which in turn slows the auto mobile. And an LCD which displays status.

BLOCK DIAGRAMS:

TRANSMITTER MODULE:

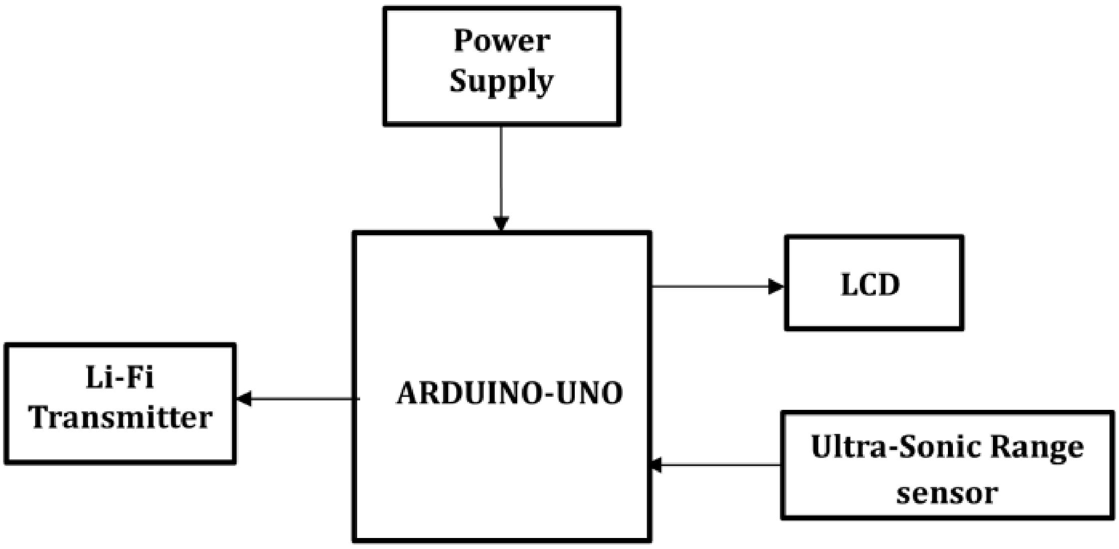


FIG 3.1.1 TRANSMITTER MODULE

RECEIVER MODULE:

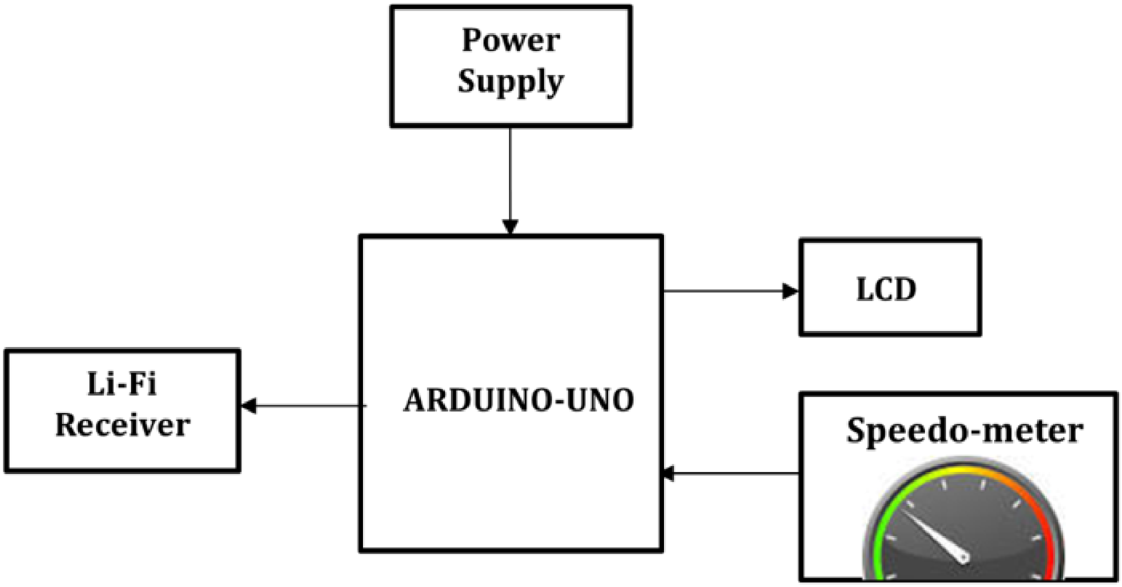


FIG 3.1.2 RECEIVER MODULE

WORKING PRINCIPLE:

TRANSMITTER SIDE:

Set the baud rate for data communication to and FRO. Initialize LCD, ultrasonic sensor. If any Object Is found it gets detected. After detecting the object, it determines the distance in Between. If distance is less than 100m LCD print's warning message, this distance is converted in to speed by the transmitter and is sent to receiver.

RECEIVER SIDE:

Set the baud rate similar to the transmitter system. Initialize LCD, servo motor. The photodetector at the receiver side receives data form transmitter. If the current position of the servo angle marking Parallel to the speed is less than the received speed then speeds increases. Or else speed decreases.

FOLLOWING FLOW CHARTS DESCRIBES THE WORK FLOW OF TRANSMITTER AND RECEIVER SYSTEMS:

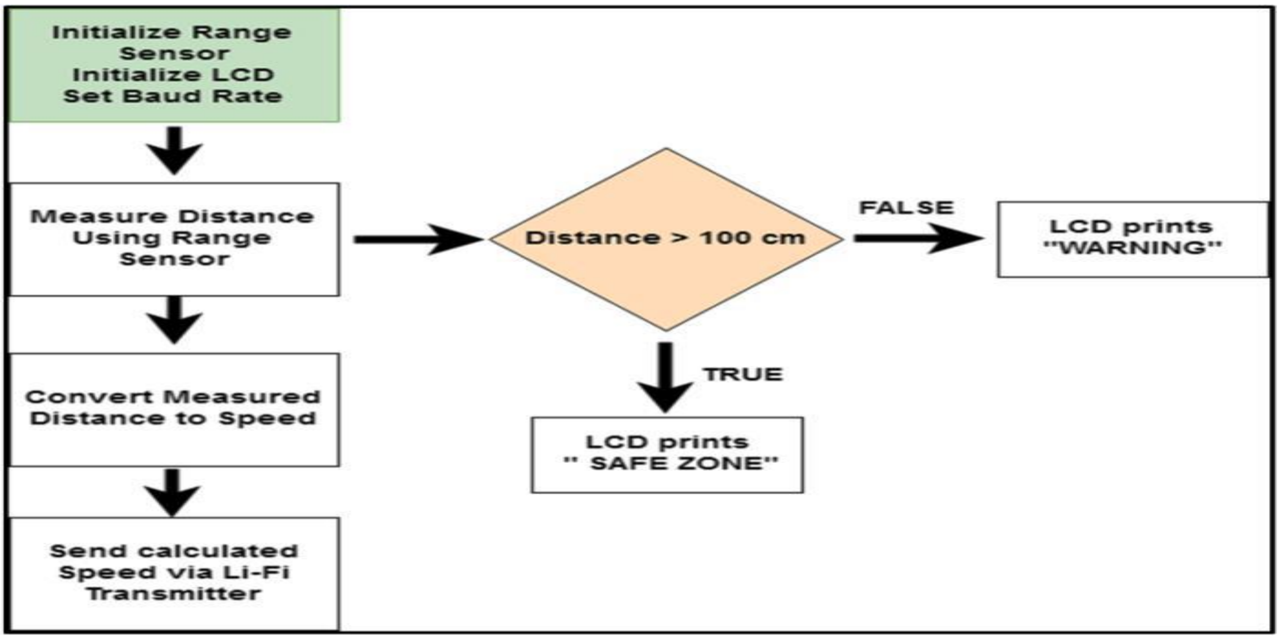
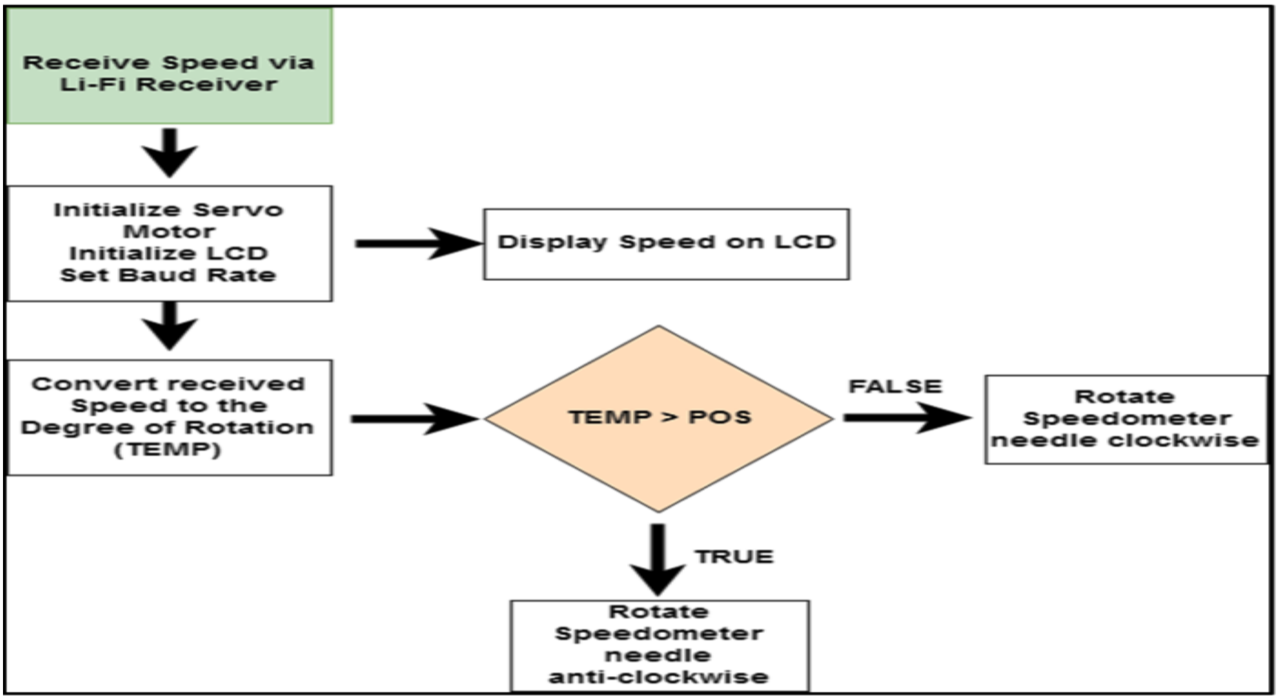


FIG 3.1.3 TRANSMITTER SYSTEM FLOW CHART



Receive Speed via Li-Fi Receiver
Initialize Servo Motor
Initialize LCD
Set Baud Rate
Display Speed on LCD
Convert received Speed to the Degree of Rotation (TEMP)
TEMP > POS
FALSE
Rotate Speedometer needle clockwise
TRUE
Rotate Speedometer needle anti-clockwise

FIG 3.1.4 RECEIVER SYSTEM FLOW CHART

DRAWBACKS:

1. It is only a pre-caution system.
2. The system encoding scheme are not highly efficient for receiving data.
3. It has only alerting display system. Lack of real time alerting systems like smart phone.
4. Messaging, alarm buzzer or a blinking led light.
5. It can only tell the receiver to maintain this speed but this is suitable for self-driven cars.
6. Which is a head of future technology.
7. There is no information regarding the status of the transmitter.
8. In case of manual driving there are chances of emergency cases where location address is required for apt timing. So, location tracking.
9. There are no noise filters which can reduce the noise in message.
10. Range of communication is very low.

CHAPTER 4

DESIGN OF PROPOSED SYSTEM

4.1 PROPOSED SYSTEM

The Smart Car unlike any traditional car consists of number of sensors and complex circuitry to process real time data in order to make V2V communication more efficient and to ensure the passengers' safety. The proposed system uses Li-Fi technology to transfer the real time data between two vehicles.

The Li-Fi transmitter circuitry and receiver circuitry are installed in the Car represented by C1 where I=1, 2, N, for N cars, In the proposed system which will be investigated in this work, I consider the case with N=2. The tail light of C1 is equipped with Li-Fi transmitter and the front side C2 of is equipped with Li-Fi receiver. The proposed system model is shown in Fig. 1. The devices and sensors which are used in the Li-Fi transmitter and receiver sections are described as follows.

I Li-Fi transmitter module: To transmit the calculated information from C1 to C2.

II Li-Fi receiver module: To receive the information from C1.

III Ultrasound range Sensor: To measure the distance between C1 and C2.

IV Servo motor: To act as a speedometer, this will show the variation in speed.

V GSM: To intimate alert message.

VI GPS: To live track the automobile.

VII BUZZER: To alert alarm.

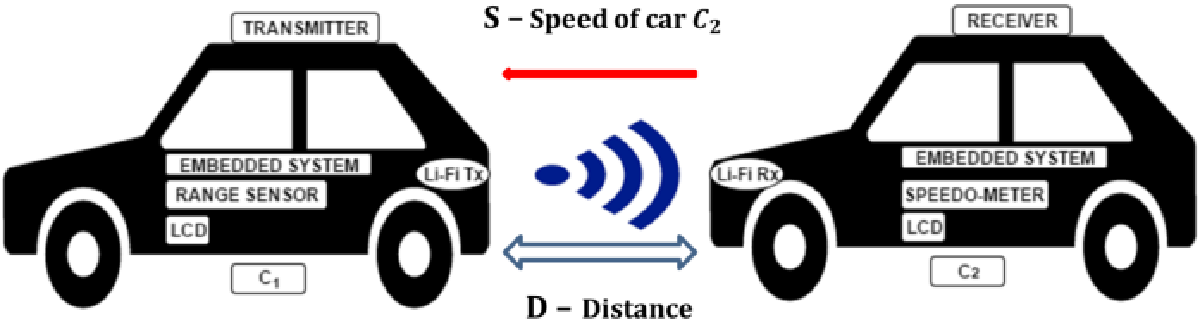


FIG 4.1.1 PICTURE OF PROPOSED MODEL

The proposed system is categorized in two sub sections, namely "Transmitter" and "Receiver". The former deals with the circuitry related to the transmitter side while the latter deals with the circuitry related to the receiver side. The elaborated explanations of both the sections are described below.

A. The transmitter module

This module is equipped with Li-Fi Transmitter module, Ultrasonic Range sensor and Arduino-Uno Board which in turn helps the leading car C1 to transmit the information to following car C2 to detect and avoid collision. The block diagram of transmitter section and its real time module is shown in Figs. 2 and 3, while the work flow diagram of transmitter section is shown in Fig 4. When a following car C2 approaches the leading car C1, the range sensor installed in Car gets initialized and measure the distance between two cars. If the measured distance between two cars is less than the threshold d th distance then the LCD display attached to the transmitter section will appear the indication “WARNING” otherwise the LCD display will inform “SAFE ZONE”. Threshold depends on several different parameters like the relative velocity between two cars and distance between two vehicles, braking safety distance, "two seconds" rule. In this work has been considered one meter as the threshold distance or safety margin between two cars due to limitation of measuring range of an ultrasonic range sensor. In this prototype, the ultrasonic sensors used to measure distance in range of 10 cm-100 cm.

This graph describes about the accuracy of the distance measured by ultrasonic sensor

With actual original distance.

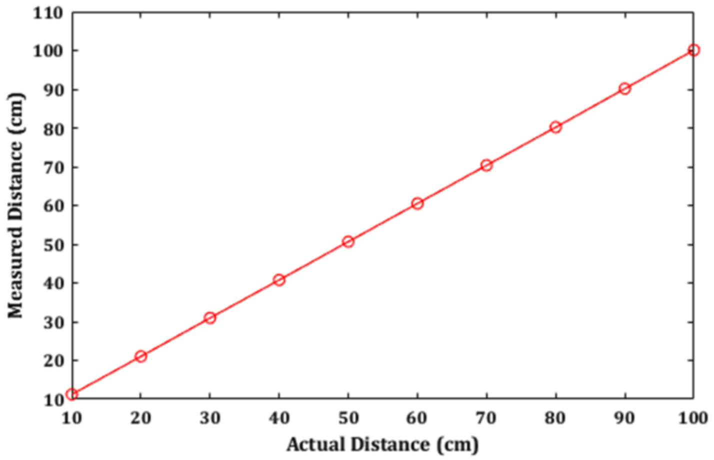


FIG 4.1.2 GRAPH BETWEEN ACTUAL AND MEASURED DISTANCE

FLOWCHART WORKING OF TRANSMITTER SYSTEM:

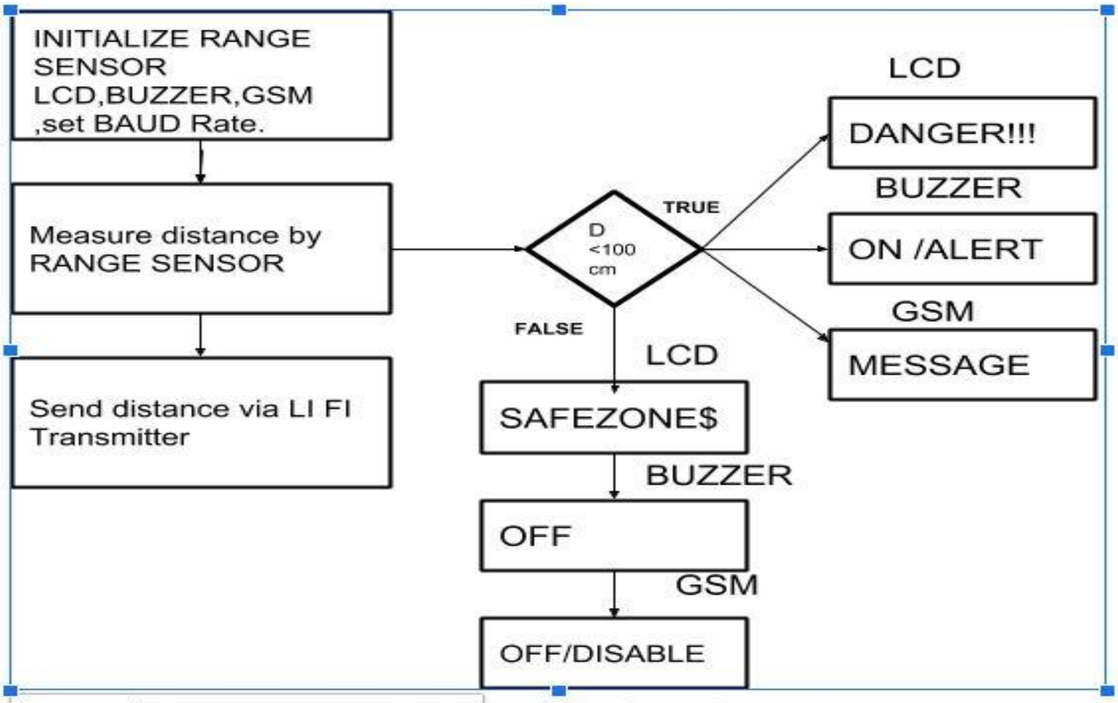

INITIALIZE RANGE SENSOR LCD,BUZZER,GSM ,set BAUD Rate.
Measure distance by RANGE SENSOR
Send distance via LI FI Transmitter
D <100 cm
TRUE
FALSE
LCD
DANGER!!!
BUZZER
ON /ALERT
GSM
MESSAGE
LCD
SAFEZONE$
BUZZER
OFF
GSM
OFF/DISABLE

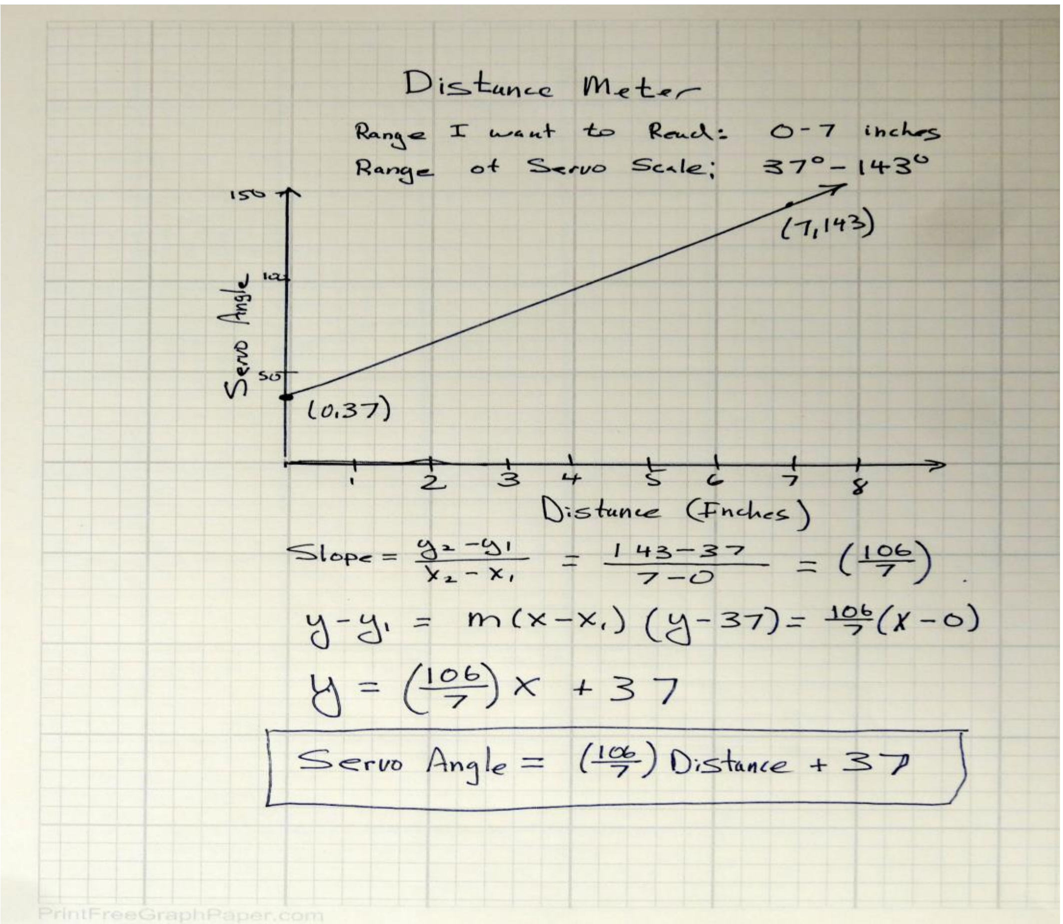
FIG 4.1.3 TRANSMITTER SYSTEM

B) RECEIVER MODULE:

Receiver at the other end receives the data in the form of bits as the transmission of the data is via light receiver module has an led which detects the change in Variation of the voltage through which it detects the fluctuations in air. This is taken Upon by ADC which is known as analog to digital convertor which converts analog Signal to digital signal. This converted signal is decoded by Manchester decoder which Maintains the perceived intensity of the light for data communication.

Distance from the ultrasonic sensor is ranged to be converted in to the speed, This Speed should be converted or scaled to an angle of the servo by knowing the range of the servo we can determine the apt angle parallel to speed.

Here is a Demo graph which scales the distance of the vehicle in to servo-angle with the following line equation.



$$\text{Slope} = \frac{y_2 - y_1}{x_2 - x_1} = \frac{143-37}{7-0} = \left(\frac{106}{7}\right)$$

$$y - y_1 = m(x - x_1) \quad (y-37) = \frac{106}{7}(x-0)$$

$$y = \left(\frac{106}{7}\right)x + 37$$

$$\boxed{\text{Servo Angle} = \left(\frac{106}{7}\right)\text{Distance} + 37}$$

FIG 4.1.4 GRAPH PLOT AGAINST SERVO ANGLE AND DISTANCE

The above graph scales the distance and converts it to servo angle. GPS on other hand activate activates the tracking if and only if the servo angle pin goes down suddenly.

Following below describes the working flow of receiver module.

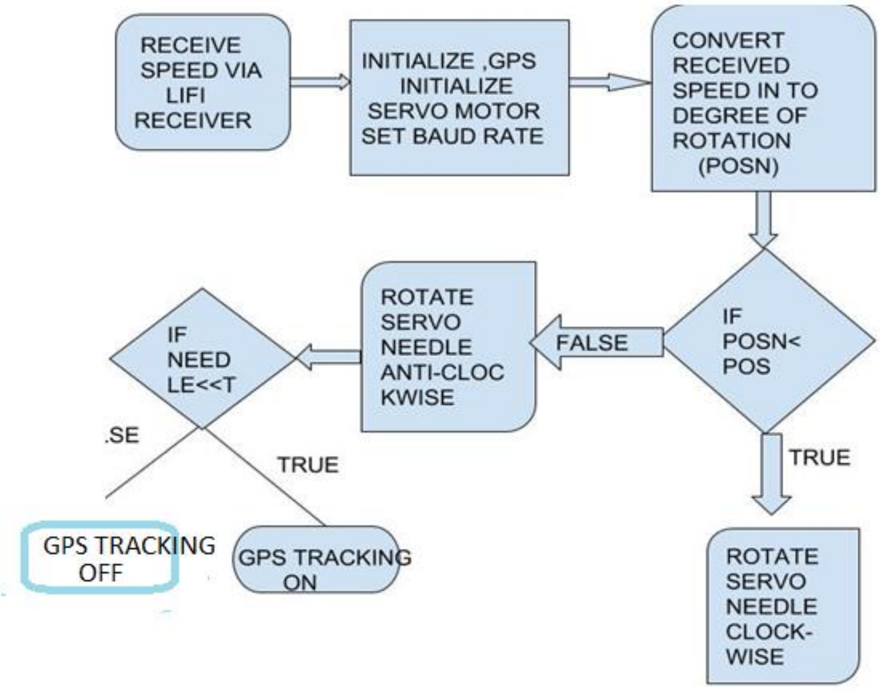

RECEIVE SPEED VIA LIFI RECEIVER
INITIALIZE ,GPS INITIALIZE SERVO MOTOR SET BAUD RATE
CONVERT RECEIVED SPEED IN TO DEGREE OF ROTATION (POSN)
IF POSN< POS
FALSE
ROTATE SERVO NEEDLE ANTI-CLOC KWISE
IF NEED LE<<T
.SE
TRUE
GPS TRACKING OFF
GPS TRACKING ON
TRUE
ROTATE SERVO NEEDLE CLOCK-WISE

FIG 4.1.5 RECEIVER SYSTEM FLOWCHART

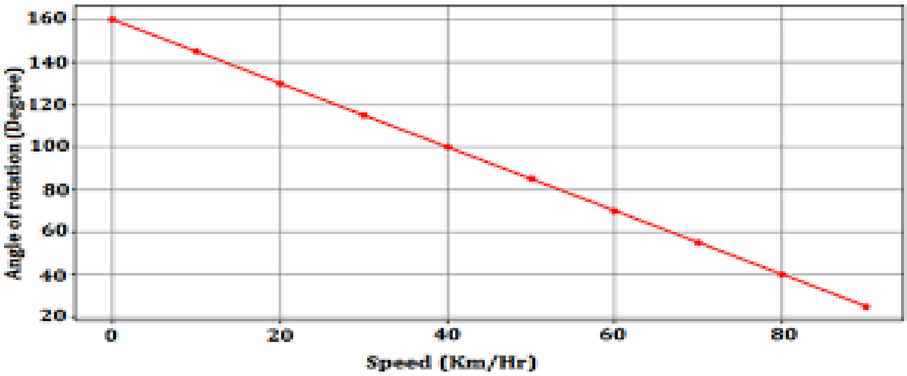


FIG 4.1.6 GRAPH PLOT AGAINST SPEED AND ANGLE OF ROTATION.

Below graph describes how speed varies with angle of rotation. As angle of 160 degree describes the least speed. whereas 25 degree describes the high speed.

4.2 ALGORITHM DESIGN

TRANSMITTER WORKING PROCESS

This algorithm is designed to control the speed of the following car C2. The algorithm is divided into two parts namely for Transmitter Side and Receiver Side. Depending upon the threshold distance, the calculated speed from the transmitter side is transmitted to the receiver side and according to it the speed of following car C2 changes. A. The transmitter section Functions as follows.

Initializations: Set LCD, Baud Rate, Range Sensor.

Step 1: Start execution of instruction and calculate the distance using range sensor

Step 2: Check If distance <dth.

Step 3: If Step 2 is TRUE, Display “WARNING” on LCD Step with buzzer indication followed by a text message.

step4: If Step2 is FALSE, Display “SAFE ZONE!” on LCD.

Step 5: Convert measured distance to the speed.

Step 6: Transmit calculated speed to receiver via LIFI transmitter in carC1(transmitter side).

The distance between the two cars C1 andC2 is measured using range sensor. The measured distance value is compared with the threshold distance (). If the measured distance is lesser than the threshold distance, the LCD displays “WARNING” else, it displays “SAFE ZONE”. At the same time the measured distance is converted to speed and it can be transmitted to the car C2. Therefore, the speed of the car changes according to the distance in between them.

RECEIVER WORKING PROCESS

In the receiver section following work flow takes place after receiving data from LIFI TX
Initializations: Set LCD, Servo motor, Baud Rate, GPS, Lpos, Cpos.
Step 1: Receive speed from transmitter via LIFI receiver
Step 2: Display speed on LCD
Step 3: Convert speed to DOR
Step 4: If Cpos>Lpos, is TRUE, rotate pointer anti-clockwise
Step 5: If Step 4 FALSE, rotate pointer clockwise
step6: In step4 if pointer drops in anti-clockwise suddenly through an angle of 75 then GPS live tracking is on.
step 7: if it drops at an angle less than <75 sudden drop. Then no need to on GPS live tracking.
The car receiver side, receives the transmitted speed and display the value in its LCD. In the proposed prototype, the received speed is converted in to DOR to indicate the speed of car. If is grater then the, the speedometer needle rotates in anticlockwise direction (that means speed of car is decreasing to avoid the collision), else the speedometer needle rotates in clockwise direction (that means speed of car is increasing since the collision chance is less.

4.3 SYSTEM ADVANTAGES

*Encrypted and Decryption is done smoothly by using Manchester encoding.
*Range of communication may extend proportionally with the usage of high voltage led bulbs.
*Immediate alerting system in real world scenario like messaging and alarm buzzer. *Can find the location it starts automatically tracking the location by indication of any emergency.
*This track location can be shared to known ones for immediate help.

CHAPTER 5

PROJECT IMPLEMENTATION

5.1 INTRODUCTION TO ARDUINO IDE:

- Arduino IDE is an open source software that is mainly used for writing and compiling the code into the Arduino Module.
- It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process.
- It is easily available for operating systems like MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role for debugging, editing and compiling the code in the environment.
- A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo Arduino Micro and many more.
- Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code.
- The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.
- The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.
- This environment supports both C and C++ languages.

HOW TO DOWNLOAD ARDUINO IDE:

You can download the Software from Arduino main website. As I said earlier, the software is available for common operating systems like Linux, Windows, and MAX, so make sure you are downloading the correct software version that is easily compatible with your operating system.

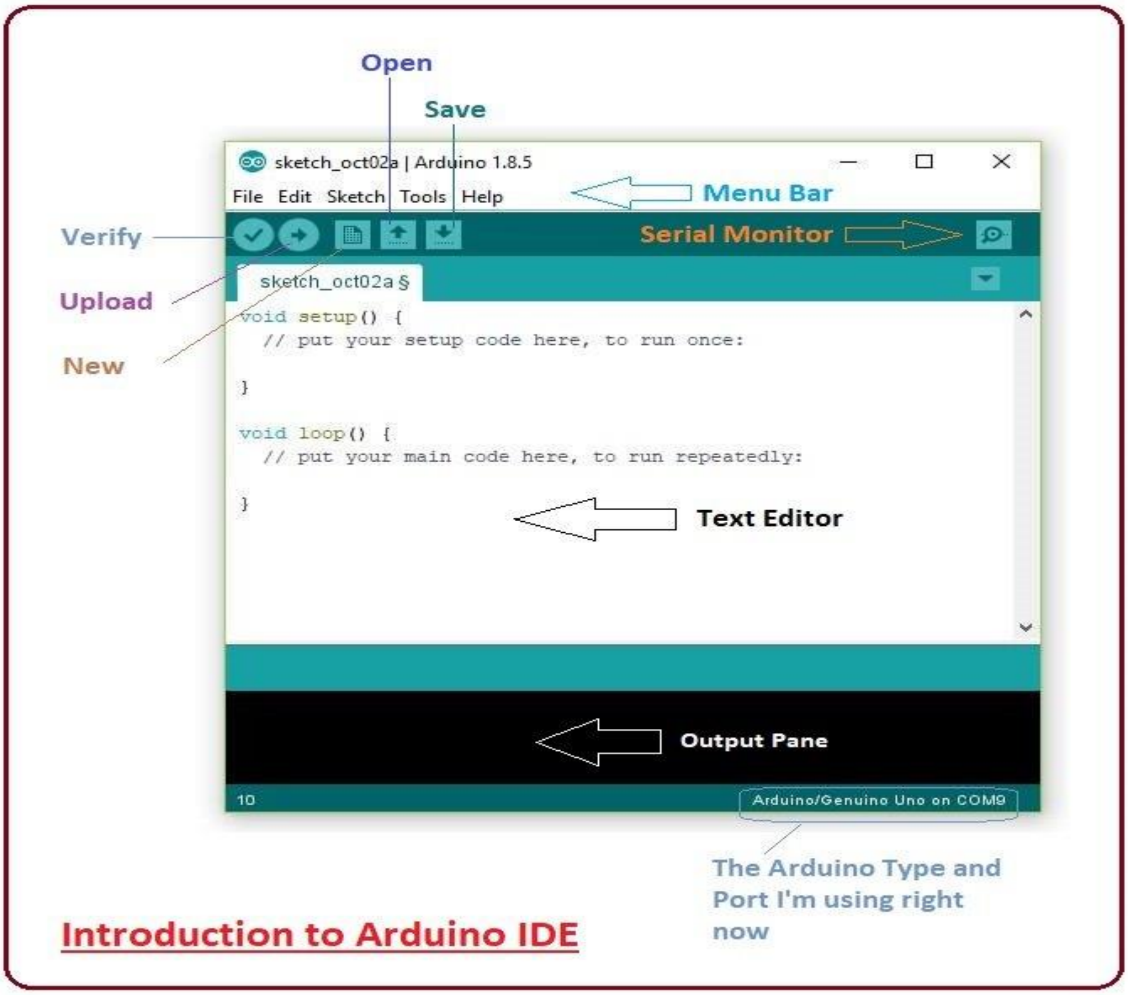
- If you aim to download Windows app version, make sure you have Windows 8.1 or Windows 10, as app version is not compatible with Windows 7 or older version of this operating system.

COMPONENTS OF ARDUNIO IDE:

The IDE environment is mainly distributed into three sections

- **1. Menu Bar**
- **2. Text Editor**
- **3. Output Pane**

Below picture depicts the view of Arduino ide



```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Open
Save
sketch_oct02a | Arduino 1.8.5
File Edit Sketch Tools Help
Menu Bar
Verify
Upload
New
Serial Monitor
sketch_oct02a §
Text Editor
Output Pane
10
Arduino/Genuino Uno on COM9
The Arduino Type and Port I'm using right now
Introduction to Arduino IDE

FIG 5.1.1 INDETAIL VIEW OF ARDUINO IDE

VERIFY:

Verify button verifies the program which was saved in the Arduino ide.

ARDUINO/GENUINO UNO ON COM:

This tells us on which port of the system Arduino is interfaced for serial communication between Arduino and system.

UPLOAD:

Upload button is used to upload the program which is in Arduino ide to Arduino.

OUTPUT PANE AND TEXT EDITOR:

Output pane details the status of compilation and uploading programs. detects errors and display them

TOOLS:

In tools we get access to different libraries. Can create user defined libraries, add pre-defined libraries, change port option.

FILE:

In files we save the stored programs and also has demo programs for user experience.

SERIAL MONITOR:

Serial monitor displays the serial communication data between Arduino and monitor.

5.2 PROJECT MODULES DESCRIPTION:

MODULE 1:

DEVELOPING AN ALOGITHM FOR THE DATA TRANSMISSION KNOWN AS LIFI TECHNOLOGY.

This module describes about the type of algorithm used for data transmission. Data is packed in to a frame of 8 bits plus start and stop bit for message processing is encoded by Manchester Encoding and this data is sent via led light on the other end the other led light receives the data In the form of analogy voltage which then converts in to analogy signal to digital using ADC.an Number is sent of time for the identification of optimum threshold which describes the high and low of the signal which was sampled .data synchronization clock pulse is sent before the start of receiving data. And then it is decoded by Manchester decoding.

MODULE 2 (TRANSMITTER SYSTEM):

IMPLEMTION OF DESIGNING THE WORKING OF LCD, ULTRASONIC SENSOR, BUZZER AND GSM MODULE.

In this module we define the work flow of the components used in LIFI transmitter system We initialize the baud rate for serial data communication and initialize LCD, BUZZER AND GSM MODULE. Then according to our algorithm, we design the procedure of work flow

MODULE 3 (RECEIVER SYSTEM):

IMPLEMENTION OF THE DESIGNING THE WORKING OF SERVO MOTOR, GPS MODULE:

In this module we define the work flow of the components used in the LIFI receiver system We set the baud rate same as transmitter for receiving the correct data. Then we initialize the servo motor, GPS module and design the work flow and algorithm for its functioning.

5.3 SCHEMATIC VIEW PICTURES:

TRANSMITTER VIEW:

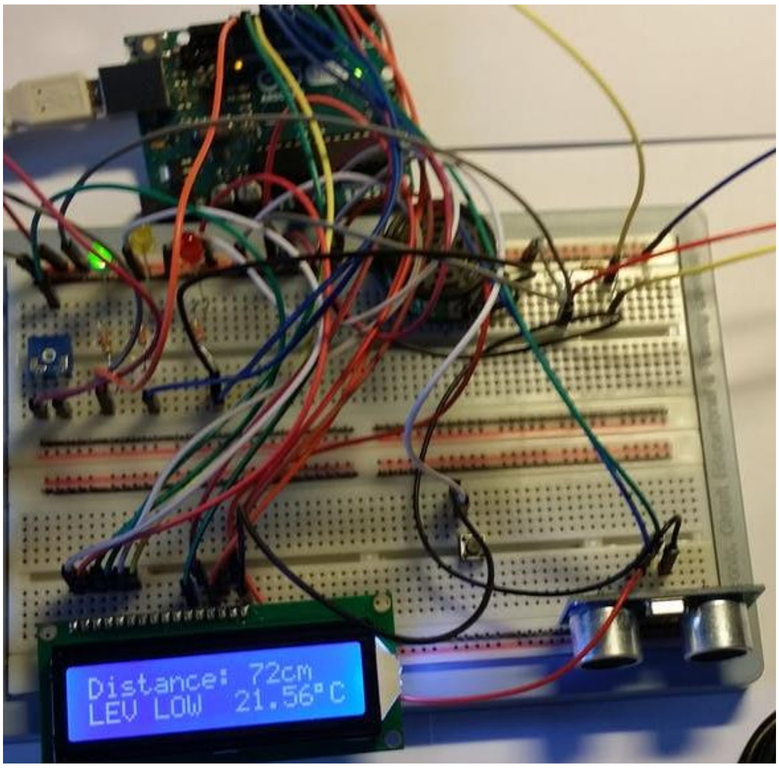


FIG 5.3.1 TRANSMITTER SYSTEM

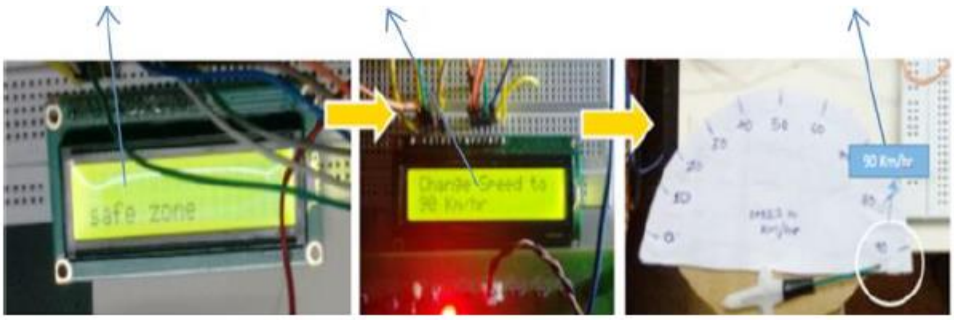


FIG 5.3.2 SERVO INDICATION (RECEIVER SYSTEM)

CHAPTER 6

CONCLUSION

About 1.25 million people are affected each year due to road accidents. The safety of the driver, the passengers and the public should be of primary concern in regard to various road safety measures that are being adopted by governments. Adopting a solution to prevent such fatal accidents due to inefficiency of human assisted driving is of utmost importance. With rise in technology and automation it has become possible to propose a plan to mitigate the adverse effects of Human assisted Driving. This paper presents a new and innovative collision detection system to prevent collision of cars, thereby reducing the probability of fatal road accidents. The proposed method in this paper relies on Li-Fi transmitter and receiver systems. This paper would encourage and motivate others to further explore the notion of Smart Cars and the use of Li-Fi Technology and hence, develop more efficient strategy to enhance the transmitting range of the Li-Fi system and propose innovative solutions to overcome the challenges due to line of sight (LOS) communication and white light interference i.e. associated with the Li-Fi. Apart from that, the ultrasonic range sensor can be modified to increase its detection range. The proposed system can be extended to detect the Side impact collision, Lane-Change assistance or Blind-Spot detection. With proper assistance from the concerned authorities it is highly possible to implement the proposed system in the Smart Cars of the future.

FUTURE ENHANCEMENTS:

Enhancement to be done in the following areas.

DATA TRANSFER:

Effective use of external hardware for noise filtration techniques with increase in bandwidth. Like adding noise filters amplifiers etc.

DATA SPEED:

Developing high effective data speeds by enhancing encryption techniques.

REALTIME IMPLEMENTATION:

Increase the scope of the project by enhancing the hardware and to test it with real time scenario

CHAPTER 7

REFERENCES:

[1] Changsheng Zhu, Victor C. M. Leung, Lei Shu and Edith C.-H. Ngai, “ Green Internet of Things for Smart World”, 2015 IEEE Access, Special Section On Challenges For Smart Worlds

[2] Mahadevapura Harish, “A Study on air pollution by automobiles in Bangalore city”, Management Research and Practice, vol. 4 issue 3 (2012) pp: 25-36, 2012

[3] P. Shruthi, et al., Analysis of Fatal Road Traffic Accidents in a Metropolitan City of South India”, J Indian Aced Forensic Med. October-December 2013, Vol. 35, No. 4, 2013

[4] Tao Jiang, Uma Ayer, Anand Tolani, Sajid Hussain, “Self-driving Cars: disruptive or incremental?”, Applied Innovation Review, Issue 1, June 2015

[5] John Carlin, Charles Birdsong, Peter Schuster, William Thompson, Danelaw, “Evaluation of Cost-Effective Sensor Combinations for a VehiclePrecrash Detection System”, SAE international, 2005

[6] Shivaji Kulkarni, Amoah Dabekar and Pavan Joshi, “A survey on Li-Fi technology”, 2016 International Conference on Wireless Communications, Signal Processing and Networking (Wisp NET), 2016

[7] Harald Haas, “Life: Conceptions, misconceptions and opportunities”,2016 IEEE Photonics Conference (IPC), 2016

[8] Harald Haas, Liang Yin, Yunluo Wang and Cheng Chen, “What is LIFI?”, Journal of Lightwave Technology, Volume: 34, Issue: 6, pp.1533 – 1544, March 2016

[9] Quintana, C., Victor Guerra, J. Ruffo, J. Ramadan, and R. Perez-Jimenez. "Reading lamp-based visible light communication system for in-flight entertainment." IEEE Transactions on Consumer Electronics 59, no. 1 (2013): 31-37.

[10] Imani, Mahdi, and Ulises M. Braga-Net. "Particle filters for partially-observed Boolean dynamical systems." Automatica87 (2018): 238-250.

APPENDIX:

CODE FOR TRANSMITTER MODULE:

```
#include <TimerOne.h>
#define SYMBOL_PERIOD 500 /* Defined a symbol period in us*/
#define WORD_LENGTH 10 /* Each byte is encoded on 10-bit with start, byte, stop */
#define SYNC_SYMBOL 0xD5 /* Synchronization symbol to send after a preamble, before data communication happens */
#define ETX 0x03  //end frame.
#define STX 0x02  //start frame.
//Fast manipulation of LED IO.
//These defines are for a LED connected on D13
#define OUT_LED() DDRB |= (1 << 5);
#define SET_LED() PORTB |= (1 << 5)
#define CLR_LED() PORTB &= ~(1 << 5)
unsigned char frame_buffer [38] ; //buffer for frame
char frame_index = -1; // index in frame
char frame_size = -1  ; // size of the frame to be sent
//state variables of the manchester encoder
unsigned char bit_counter = 0 ;
unsigned short data_word = 0 ;  //8bit data + start + stop
unsigned char half_bit = 0 ;
unsigned long int manchester_data ;
void to_manchester(unsigned char data, unsigned long int * data_manchester){
  unsigned int i ;
 (*data_manchester) = 0x02 ; // STOP symbol
 (*data_manchester) = (*data_manchester) << 2 ;
  for(i = 0 ; i < 8; i ++){
    if(data & 0x80) (*data_manchester) |=  0x02  ; // data LSB first
    else (*data_manchester) |= 0x01 ;
    (*data_manchester) = (*data_manchester) << 2 ;
    data = data << 1 ; // to next bit
  }
  (*data_manchester) |= 0x01 ; //START symbol
}
void setup() {
  // initialize serial communication at 115200 bits per second:
  Serial.begin(115200);
  OUT_LED();
  init_frame(frame_buffer);
  init_emitter();
  Timer1.initialize(SYMBOL_PERIOD); //1200 bauds
  Timer1.attachInterrupt(emit_half_bit);
  LCD.begin(16,2);
  LCD.setCursor(0,0);
  pinMode(trigPin,OUTPUT);
  pinMode(echoPin,INPUT);
  pinMode(buzzerPin,OUTPUT);
  LCD.print("Status:");
}
 #ifdef TRANSMIT_SERIAL
  if(Serial.available() && transmitter_available()){ //constructing the data frame only if transmitter is ready to transmit
    char c = Serial.read();
    com_buffer[com_buffer_nb_bytes] = c ;
    com_buffer_nb_bytes ++ ;
    if(com_buffer_nb_bytes >= 32 || c == '\n'){
```

```
   if(write(com_buffer, com_buffer_nb_bytes) < 0){
    Serial.println("Transmitter is busy");
   }else{
    com_buffer_nb_bytes = 0 ;
   }
  }
 }
 delay(10);
 #else
  static int i = 0 ;
  memcpy(com_buffer, msg, 11);
  com_buffer[11] = i + '0' ;
  if(write(com_buffer, 12) < 0){
   delay(10);
  }else{
   i ++ ;
   if(i > 9) i = 0 ;
  }
 #endif
```

RECEIVER SYSTEM CODE:

```
#include <TimerOne.h>
enum receiver_state {
 IDLE, //waiting for sync
 SYNC, //synced, waiting for STX
 START, //STX received
 DATA //receiving DATA
};
//#define DEBUG
```

```
//#define DEBUG_ANALOG
#define INT_REF /* Commen this to use AVCC reference voltage. To be used
when the receiver LED generate low levels */
enum receiver_state frame_state = IDLE ;
//This defines receiver properties
#define SENSOR_PIN 3
#define SYMBOL_PERIOD 500
#define SAMPLE_PER_SYMBOL 4
#define WORD_LENGTH 10 // a byte is encoded as a 10-bit value with start and
stop bits
#define SYNC_SYMBOL 0xD5 // this symbol breaks the premanble of the frame
#define ETX 0x03 // End of frame symbol
#define STX 0x02 //Start or frame symbol
// global variables for frame decoding
char frame_buffer[38] ;
int frame_index  = -1 ;
int frame_size = -1 ;
//state variables of the thresholder
unsigned int signal_mean = 0 ;
unsigned long acc_sum = 0 ; //used to compute the signal mean value
unsigned int acc_counter = 0 ;
//manechester decoder state variable
long shift_reg = 0
//Start of ADC managements functions
void ADC_setup()
  ADCSRA =  bit (ADEN);                      // turn ADC on
  ADCSRA |= bit (ADPS0) |  bit (ADPS1) | bit (ADPS2);  // Prescaler of 128
  #ifdef INT_REF
  ADMUX  =  bit (REFS0) | bit (REFS1);    // internal 1.1v reference
  #else
  ADMUX  =  bit (REFS0) ;   // external 5v reference
  #endif
}
#define EDGE_THRESHOLD 15 /* Defines the voltage difference between two
samples to detect a rising/falling edge. Can be increased depensing on the
environment */
int oldValue = 0 ;
int steady_count = 0 ;
int dist_last_sync = 0 ;
unsigned int detected_word = 0;
// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 115200 bits per second:
  int i;
  Serial.begin(115200);
  Serial.println("Start of receiver program");
  ADC_setup();
  ADC_start_conversion(SENSOR_PIN);
  //analogReference(INTERNAL); // internal reference is 1.1v, should give better
accuracy for the mv range of the led output.
  Timer1.initialize(SYMBOL_PERIOD/SAMPLE_PER_SYMBOL);        //1200
bauds oversampled by factor 4
  Timer1.attachInterrupt(sample_signal_edge);
}
if((*frame_state) != IDLE){ // we are synced
  frame_buffer[*frame_index] = data ;
  (*frame_index) ++ ;
    if(data == STX){
      //Serial.println("START");
      (*frame_state) = START ;
       return 0 ;
    }else if(data == ETX){
      //Serial.println("END");
      (*frame_size) = (*frame_index) ;
      (*frame_index) = -1 ;
      (*frame_state) = IDLE ;
      //Serial.println("END");
       return 1 ;
    }else if((*frame_index) >= 38){ //frame is larger than max size of frame ...
      (*frame_index) = -1 ;
      (*frame_size) = -1 ;
      (*frame_state) = IDLE ;
      return -1 ;
    }else{
      (*frame_state) = DATA ;
    }
    return 0 ;
  }
  return -1 ;
}
// the loop routine runs over and over again forever:
void loop() {
  int i;
  unsigned char received_data;
  char received_data_print ;
  int nb_shift ;
  int byte_added = 0 ;
```

```
if(new_word == 1){
  received_data = 0 ;
  for(i = 0 ; i < 16 ; i = i + 2){ //decoding Manchester
         received_data = received_data << 1 ;
         if(((detected_word >> i) & 0x03) == 0x01){
            received_data |= 0x01 ;
         }else{
            received_data &= ~0x01 ;
         }
  }
  received_data = received_data & 0xFF ;
  #ifdef DEBUG
    Serial.print(received_data & 0xFF, HEX);
    Serial.print(", ");
    Serial.println((char) received_data);
  #endif
  new_word = 0 ;
  if((byte_added = add_byte_to_frame(frame_buffer, &frame_index, &frame_size, &frame_state,received_data)) > 0){
    frame_buffer[frame_size-1] = '\0';
    Serial.println(&(frame_buffer[1]));
  }
  //if(frame_state != IDLE) Serial.println(received_data, HEX);
```