

School of Electrical and Computer Engineering

Communication stream

**Vehicle Speed Detection Using Simple**

**Radar System**

A THESIS SUBMITTED IN PARTIAL FULFILMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Science

Aemiro Kebere………………… ATR/6566/07

Ashebir Shewangizaw……... ATR/0291/07

Dessalew Alemneh…………. ATR/2051/07

Advisor’s Name:

Rediet Million

JUNE, 2019

**Acknowledgement**

Many people deserve our cordial thanks for their help in completing this project. First and foremost, we would like to thank our honorable advisor Mr. Rediet Million who is so dedicated to help us when we need. We are really honored to make this project under his guidance who provide us his valuable time and knowledge, motivating thought and encouragement. we would like to extend our thanks to Mr. Matios Kassa and laboratory stuff of school of electrical and computer engineering department for the fast response and cooperation with us to get some material we need for our cause. Finally, we would like thank all persons who were involved with this project for their valuable help and professionalism during this project.

**Abstract**

Road accidents have been very common in the present world with the prime cause being the careless driving. The necessity to check this has been very essential and different methods have been used so far. However, with the advancement in the technology, different governing bodies are demanding some sort of computerized technology to control this problem of over speed driving. At this scenario, we are proposing a system to detect the vehicle which are being driven above the given maximum speed limit that the respective roads or highway limits.

The overall project is divided in three categories; speed detection, image acquisition and image processing. Speed detecting device works on the principle of Doppler Effect using microwave Doppler radar sensor. The speed is compared with the preset threshold and camera is triggered if the speed limit exceeds. Acquisition of image is done by a HD camera interfaced with raspberry pi. An Image processing has three modules: Plate Detection, Character Segmentation and Character Recognition. For plate detection, median filter, binarization, morphological closing operation and connected component analysis was used to detect legitimate plate region. For character segmentation process a connected component analysis method is used. But before the actual segmentation process, the plate image passes through a number of preprocessing tasks that dramatically increase the accuracy of the segmentation outcome. Finally, a correlation based template matching method is used for character recognition. We use python tool to develop this system.

**Keywords:** speed detection, image acquisition, License Plate Recognition, Plate Detection, Plate Characters Segmentation, Character Recognition,

### Content

[ACKNOWLEDGEMENT](#_Toc1858853)

[ABSTRACT ii](#_Toc1858854)

[CONTENTS iii](#_Toc1858855)

[LIST OF FIGURES v](#_Toc1858856)

[ABBRIVATION vi](#_Toc1858857)

[1 INTRODUCTION 1](#_Toc1858858)

[1.2 STATEMENT OF THE PROBLEM 2](#_Toc1858859)

[1.3 objectives 2](#_Toc1858860)

[1.3.1. General objective 2](#_Toc1858861)

[1.3.2 Specific objective 2](#_Toc1858862)

[1.4 Scope 2](#_Toc1858863)

[1.5 Literature Review 3](#_Toc1858864)

[2 BACKGROUNDS 4](#_Toc1858865)

[2.1 radar system 4](#_Toc1858866)

[2.2 HB 100 sensor 4](#_Toc1858867)

[2.2.1 Operation 5](#_Toc1858868)

[2.2.2 Frequency Variation 5](#_Toc1858869)

[2.2.3 Output of the sensor 6](#_Toc1858870)

[2.3 Amplifier 6](#_Toc1858871)

[2.3.1 Features 7](#_Toc1858872)

[2.3.2 Advantage 7](#_Toc1858873)

[2.4 Arduino platform for atmega 328 7](#_Toc1858874)

[2.5 Raspberry pi 8](#_Toc1858875)

[2.5.1 Why raspberry pi? 8](#_Toc1858876)

[2.5.2 Raspberry pi Camera 9](#_Toc1858877)

[2.6 LCD 9](#_Toc1858878)

[2.7 Number plate recognition 10](#_Toc1858879)

[2.7.1 Image Acquisition 11](#_Toc1858880)

[2.7.2 Preprocessing 11](#_Toc1858881)

[2.7.3 Localization of the numbers 11](#_Toc1858882)

[2.7.4 Connected Component Analysis 12](#_Toc1858883)

[2.7.5 Character Segmentation 12](#_Toc1858884)

[2.7.6 Character Recognition 12](#_Toc1858885)

[3.1 System Block diagram 13](#_Toc1858886)

[3.2 Diagram Description 13](#_Toc1858887)

[3.3 Flow chart 14](#_Toc1858888)

[3.4 Speed Detection 15](#_Toc1858889)

[3.5 Amplification of signal 16](#_Toc1858890)

[3.6 Measurement of frequency 16](#_Toc1858891)

[3.6.1 Algorithm for measuring frequency 17](#_Toc1858892)

[3.7 Image acquisition and processing 17](#_Toc1858893)

[3.8 SOFTWARE AND TOOLS USED 21](#_Toc1858894)

[3.8.1 Tools used for simulation 21](#_Toc1858895)

*[3.8.1.1 Proteus](#_Toc1858896)* [21](#_Toc1858896)

[3.8.2 Software used 22](#_Toc1858897)

*[3.8.2.2 Python (programming language](#_Toc1858898)*[) 22](#_Toc1858898)

[4 Simulation results 23](#_Toc1858899)

[4.1 calculation of speed 23](#_Toc1858900)

[5.2 Number Plate recognition 23](#_Toc1858901)

[5 LIMITATION AND FUTURE ENHANCEMENT 24](#_Toc1858902)

[6 CONCLUSION AND PROBLEM FACED 25](#_Toc1858903)

[6.1 conclusion 25](#_Toc1858904)

[6.2 problem faced 26](#_Toc1858905)

[REFERENCES 26](#_Toc1858906)

[APPENDICES 27](#_Toc1858907)

[Appendix A 27](#_Toc1858908)

[APENDIX B 29](#_Toc1858909)

**List of Figure**

[Figure 1 vehicle over speed detection and recognition 1](#_Toc2022063)

[Figure 2 block diagram and connection of hb100 sensor 6](#_Toc2022064)

[Figure 3 lm324 block diagram 7](#_Toc2022065)

[Figure 4 arduino uno [6] 8](#_Toc2022066)

[Figure 5 raspberry pi [7] 8](#_Toc2022067)

[Figure 6 raspberry pi camera [ 7] 9](#_Toc2022068)

[Figure 7 lcd jhd162a 9](#_Toc2022069)

[Figure 8 block diagram of the system 13](#_Toc2022070)

[Figure 9 system flow diagrams 14](#_Toc2022071)

[Figure 10 simplified diagram speed detection 15](#_Toc2022072)

[Figure 11 license plate region extraction steps 18](#_Toc2022073)

[Figure 12 character segmentation and recognition steps 20](#_Toc2022074)

[Figure 13 aurdino –lcd interface 21](#_Toc2022075)

[Figure 14 real time (input) image and license plate region 23](#_Toc2022076)

[Figure 15 real time (input) image and license plate region 24](#_Toc2022077)

**Abbreviation**

LPR License plate recognition

GND Ground

HD High Definition

IC Integrated Circuit

LCD Liquid Crystal Display

LED Light Emitting Diode

VCC Voltage Constant Current

VOUT Output Voltage

ITS Intelligent Transportation Systems

PD Plate Detection

CC Connected Component

CS Character Segmentation

CR Character Recognition

CCL Connected Component Labeling

OCR Optical Character Recognition

**1.Introduction**

Over speeding vehicles are major issues for road safety and needs proper addressing to minimize the accidents. Excessive Speed is a major factor in all fatal crashes.

Vehicle speed detection is based on the use of Doppler Radar to find the speed of the moving vehicles. Doppler Effect can be exploited to measure the speed of vehicles and identify those crossing speed limit. The shift in frequency between the transmitted and reflected high frequency wave is the key factor used to calculate speed. The Doppler radar based speed detector can be interfaced to a microprocessor based system for measurement and comparison. HD camera attached to the system can be used to provide a real time view of the road, so that that image can be processed for plate recognition.

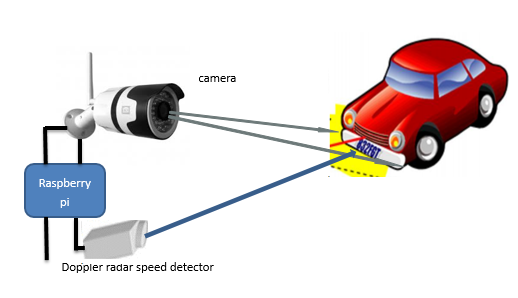


Figure 1 vehicle over speed detection and recognition

License plate recognition is an image processing technology that uses number plate to identify the vehicle. The objective is to isolate the number plate of the vehicle from the image and use optical character recognition to identify the characters of the number plate. A typical LPR system constitutes four major phases: Image Acquisition, Plate Detection, Character Segmentation and Character Recognition .

**1.1 Statement of problem**

Many Number of ways are being implemented to check and identify the over speeding vehicle. But no automatic system has been developed so far that can perform the task of speed detection and vehicle identification without human assistance.

* Road accidents are increasing day by day with prime cause being the over speeding of the vehicle.
* The use of human resources to check this issue can be very tedious and time consuming and sometimes become irrelevant.
* Commercially available automatic LPR systems are very costly and difficult to implement for the Ethiopian license number plate due to its character being different from the rest of the world.
* the number of vehicles on the road is increasing dramatically from time to time, which leads to difficulty of traffic management.

**1.2 Objectives**

**1.2.1 General Objective**

The general objective of our project is to address the issues listed above. The general objectives are as follows:

* To design an automatic and versatile system that addresses the issues of over speeding.
* To reduce inaccuracy of over speed detection due to human resources.
* To overcome the challenge of traffic accident with the help of this digitized technology.

**1.2.2 Specific Objective**

* Develop advanced license plate recognition system.
* developing methods for plate detection, character segmentation and character recognition tasks.
* Reviewing different techniques and algorithms for developing the CPR system.
* Selecting the appropriate mechanism for localization and detection of plate region from the image.

**1.3 Scope**

The scope of this project lies in detecting over speed of a single car with the principle of Doppler radar effect. It identifies the plate region and the characters.

Generally, the scope of this project is limited to the following:

* The system detects over speed only for one car at a time.
* The system recognizes only single car plate in a given input image.
* Car plates that are affected by dirt are not addressed.
* The system is not tested on all existed Ethiopian plates due to the non-uniformity of license plate position.

**1.4 Literature Review**

A number of different approaches have been used by many researchers to detect speed of moving vehicle. One of the most popular ways of speed detection is in our country as well as in international level is based on radar speed gun. But also some country use other way of speed detection was LASER crossing.

However, those approaches are now less frequently used for speed detection especially in developed countries. When we come to Ethiopia we still use radar speed gun, which is not accurate and suitable in the current situation. To overcome those limitations of this radar speed gun, researchers come up with the approach of Doppler radar guns. Speed detection using Doppler sensor is based on the principle of Doppler Effect. Today’s Commercially available speed guns are based on this approach of speed detection and provide high degree of accuracy.

The history of automatic vehicle number detection dates back to mid-1970s. In 1976 in United Kingdom. The issue of Number Plate Detection is still not solved completely because the accuracy in Number Plate digitization achieved so far is not satisfiable. Different approaches for LPR have been developed but the research still continues for the best result.

Morphological processing’s done to detect similar properties in the image so as to locate the position of number plate. Another approach was by using statistical properties of the characters in the plate. This approach is based on finding the regions for Number Plate character based on the variance of gray level, number of edges, edge densities in the region. This method is very accurate if the number of characters on the plate is fixed. The research and upgrading of these algorithms continue with many researchers working on the project over a long time [4]. In the case of our country the overall system that we are proposing is not even in the discussion.

# 

**2 Background**

**2.1 Radar System**

It is an object detection system which uses electromagnetic waves—specifically radio waves to determine the range, altitude, direction, or speed of both moving and fixed objects such as aircraft, ships, spacecraft, guided missiles, motor vehicles, weather formations, and terrain. The radar dish, or antenna, transmits pulses of radio waves or microwaves which bounce off any object in their path. The object returns a tiny part of the wave's energy to a dish or antenna which is usually located at the same site a s the same area. A radar system has a transmitter that emits radio waves called radar signals in predetermined directions. Radar signals are reflected especially well by materials of considerable electrical conductivity especially by most metals.

modern radar systems use the principle of Doppler Effect. Return signals from targets are shifted away from this base frequency via the Doppler Effect enabling the calculation of the speed of the object. The Doppler Effect is only able to determine the relative speed of the target along the line of sight from the radar to the target. It is also possible to make a radar without any pulsing, known as a continuous-wave radar (CW radar), by sending out a very pure signal of a known frequency. CW radar is ideal for determining the radial component of a target's velocity, but it cannot determine the target's range.

**2.2 HB100 Sensor**

HB100 miniature Microwave sensor is a x-band Bi static Doppler transceiver module. Its built in Dielectric Resonator Oscillator and a pair of Micro strip patch antenna array, make it ideal for use in motion and speed detection.

**Features:**

* Low current consumption
* Simplicity in interfacing
* Continuous Wave or Pulse operation
* Detection range of 10m
* Application in motion detection and speed measurement
* Application in motion detection and speed measurement.



Figure 2 HB100 Sensor

**2.2.1 Why Hb100 sensor for speed detection**

Modern speed detectors like speed gun use radar sensor built in to their circuit. It is difficult to hack one of them to use the sensor part for our purpose. As a separate unit, very few sensors are available in market that works on the principle of Doppler effect for speed detection. HB100 is one such sensor which meets most of our requirement and is a low cost device. There are other sensors like Xb100 with range of up to 100m but are expensive for use in this project.

**2.2.2 Operation**

It works on the principle of Doppler Effect. Doppler Effect is the difference between the observed frequency and emitted frequency of a wave for an observer moving relative to the source of waves. The shift in frequency between the emitted wave and the reflected wave is the factor that determines the speed of the object. This phenomenon is exploited in Doppler radar sensors to measure the speed of vehicles or any other object. The magnitude of the Doppler Shift is proportional to reflection of transmitted energy and is in the range of few mill volts.

**2.2.3 Frequency Variation**

If c be the velocity of light v the target velocity, then the shifted frequency (fr) can be expressed as a function of the original frequency () [9]

Assuming the direction of the source and the direction of motion of the target is perfectly aligned, Change in the frequency between the transmitted and reflected wave, (Doppler frequency) ( ), is thus:

Since the speed is much much smaller than the speed of light we ca aproximate the above equation as:

If the direction of the moving object makes a certain angle with the source direction, Doppler frequency is reduced by a value equal to the cosine of the angle between source and target’s direction.

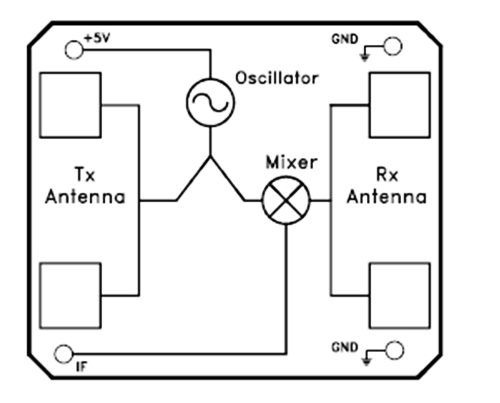


Figure 3 block diagram and connection of hb100 sensor

The sensor circuit consists of a Dielectric Resonator Oscillator, microwave Mixer and patch antenna. The transmitted frequency and power of the module is set by factory. There is no user adjustable part in this device. The module is a low power radio device (LPRD) or intended radiator. The Oscillator produces a sinusoidal wave of frequency 10.525 Ghz. The patch antenna connected to the oscillator radiate the wave towards the target. The reflected wave is received by another set of patch antenna built in the circuit. The microwave mixer mixes these two signals to generate a sinusoidal wave with frequency equal to the difference between the two.

**2.2.4 Output of The Sensor**

The magnitude of the Doppler Shift is proportional to reflection of transmitted energy and is in the range of few mill volts. A high gain low frequency amplifier is usually connected to the IF terminal in order to amplify the Doppler shift to a processable level. Frequency of Doppler shift is proportional to velocity of motion. The Received Signal Strength (RSS) is the voltage measured of the Doppler shift at the IF output. It can be used to predict the distance between the source and the target. Target closer to the source produce a high level of RSS. But the exact value may vary greatly depending on the type of the surface and reflection coefficient of the target. Noises are of major concern when designing an amplifier. Noise may be the result of the internal circuitry of the sensor and amplifier or due to fluctuation in the power supply. High frequency noise from power supply may affect the output significantly.

**2.3 Amplifire**

Wide ranges of amplifiers are available for use in the circuit and the one that best fits the system requirement needs to be used. The amplifier IC used for this purpose is the LM324 Quad op amp. This IC has four operational amplifiers. The LM324-N series consists of four independent, high gain, internally frequency compensated operational amplifiers which were designed specifically to operate from a single power supply over a wide range of voltages. Operation from split power supplies is also possible and the low power supply current drain is independent of the magnitude of the power supply voltage.

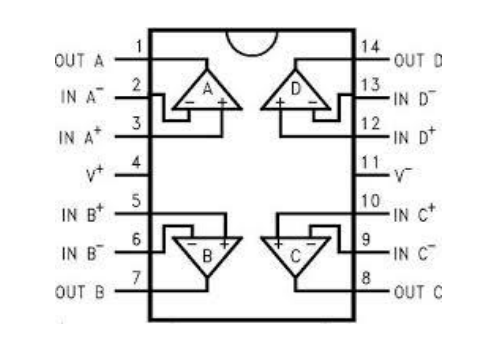


Figure 4 lm324 block diagram

**2.3.1 Feature**

* Internally frequency compensated for unity gain
* Large DC Voltage Gain 100 dB
* Wide Bandwidth (Unity Gain) 1 MHz (Temperature Compensated)
* Low Input Biasing Current 45 nA (Temperature compensated)
* Wide Power Supply Range:
* Very Low Supply Current Drain (700 μA) (Essentially Independent of Supply voltage)
* Low Input Offset Voltage 2 mV and Offset Current: 5 nA
* Input Common-Mode Voltage Range Includes ground
* Differential Input Voltage Range Equal to the power supply voltages
* Large Output Voltage Swing 0V to V+ − 1.5V

**2.3.2 Advantage**

* Eliminates Need for Dual Supplies
* Four Internally Compensated Op Amps in a Single Package
* Allows Directly Sensing Near GND and VOUT (Temperature Compensated) also Goes to GND
* Compatible with All Forms of Logic
* Power Drain Suitable for Battery Operation

**2.4 Arduino Platform for Atmega 328**

Arduino is a single-board microcontroller, intended to make the application of interactive objects or environments more accessible. The hardware consists of an open-source hardware board designed around an 8-bit Atmel AVR microcontroller, or a 32-bit Atmel ARM. Current models feature a USB interface, 6 analog input pins, as well as 14 digital I/O pins which allow the user to attach various extension boards. It comes with a simple integrated development environment (IDE) that runs on regular personal computers and allows users to write programs for Arduino using C or C++.

The frequency of the output signal from the amplifier can be measured using the digital pin of microcontroller. Microcontroller can be used to measure frequency, calculate the speed and make necessary comparisons. The output is displayed on the display device interfaced with the microcontroller.

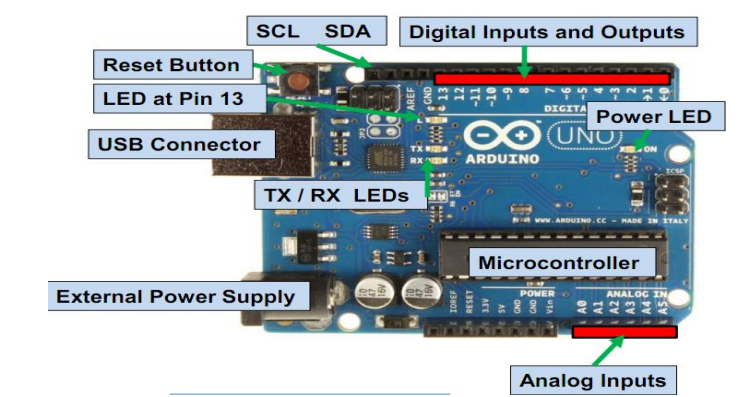


Figure 5 Arduino Uno [6]

Arduino Uno is a development board with an on-board atmega 328 microcontroller. The input output pins of the microcontroller are mapped to different sets of digital and analog I/O pins on the board. The five analog pins are capable of sensing analog voltage down to 4mv. The digital pins 2-13 can be used as input or output. Pins 0 and 1 are dedicated for serial data transmission. Arduino works with a supply voltage of +5 volts and ground. The power supply can be given via a USB cable connected to USB plug or by using an external adapter. The processor runs on a 16Mhz crystal soldered to the board.

* + 1. **Features**
* Easy to interface sensors and display devices
* Debugging and testing is easy with Arduino IDE
* Easy to program in the target circuit using USB cable
* Open source project
* Wide range of libraries available in the repository for interfacing commonly used peripherals

**2.4.2 Atmega 328**

Atmega 328 is the microcontroller unit attached to the board and performs all the processing. The high-performance Atmel 8-bit AVR RISC-based microcontroller combines 32KB ISP flash memory with read-while-write capabilities, 1KB EEPROM, 2KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timers/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, 6-channel 10-bit A/D converter.

**2.5 Raspberry pi**

The Raspberry Pi is a credit-card-sized single-board computer developed in the UK by the Raspberry Pi Foundation with the intention of promoting the teaching of basic computer science in schools.

The secret sauce that makes this computer so small and powerful is the Broadcom BCM2835, a System-on-Chip that contains an ARM1176JZFS with floating point, running at 700 MHz, and a Video core 4 GPU. The GPU provides Open GL ES 2.0, hardware-accelerated OpenVG, nd 1080p30 H.264 high-profile decode and is capable of 1Gpixel/s, 1.5Gtexel/s or 24 GFLOPs of general purpose computer. It comes with both model A and model B. If you want a Raspberry Pi for general use, then you should buy a model B, revision 2(the latest). With twice as much memory, it will cope with most tasks much better than the model A. If, on the other hand, you are embedding a Raspberry Pi in a project for a single purpose, then using a model A will be better option.

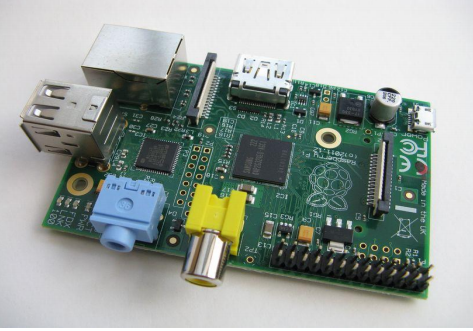


Figure 6 raspberry pi [7]

**2.5.1 Hardware**

**2.5.1.1 RAM**

On the older boards 128 MB RAM was allocated by default leaving 128 MB for CPU. Later 256 MB RAM raspberry pi was released which was sufficient for 1080p decoding, or for simple 3D. 224 MB was for Linux only, with just a 1080p frame buffer, and was likely to fail for any video or 3D. 128 MB was for heavy 3D, possibly also with video decoding .For the new model B with 512 MB RAM initially there were new standard memory split files released( arm256\_start.elf, arm384\_start.elf, arm496\_start.elf) for 256 MB, 384 MB and 496 MB CPU RAM (and 256 MB, 128 MB and 16 MB video RAM).

**2.5.1.2 Networking**

The Model A does not have an RJ45 Ethernet port; it can connect to a network by using an external user-supplied USB Ethernet or Wi-Fi adapter. On the model B the Ethernet port is provided by a built-in USB Ethernet adapter**.**

**2.5.2 Model B Features**

* Dimensions: 85.60mm x 56mm x 21mm
* Broadcom BCM2835 SoC
* 700 MHz ARM1176JZF-S core CPU
* Broadcom VideoCore IV GPU
* 512 MB RAM
* 2 x USB2.0 Ports
* Audio Out via 3.5mm Jack or Audio over HDMI
* 10/100 Ethernet (RJ45)
* Low-Level Peripherals:
* 8 x GPIO
* UART
* 2C bus
* SPI bus with two chip selects
* +3.3V
* +5VGround
* Power Requirements: 5V @ 700 mA via MicroUSB or GPIO Header
* Supports Debian GNU/Linux, Fedora, Arch Linux, RISC OS and More!

**2.5.3 Python Programs with IDLE in raspberry pi**

The common Raspberry Pi distributions come with the IDLE Python development tool. In both the Python and Python 3 versions. If you are using Raspbian. You will find shortcuts to both versions of IDLE on your Raspberry Pi desktop. We have used python programming to transfer captured image of vehicle which has crossed speed limits.

**2.5.4 Why Raspberry pi**

* The biggest advantage of raspberry pi is its small size as credit card
* One of other advantage of raspberry pi is usually speeding: ‘
* Lower power consumption.

**2.6 Raspberry pi Camera**

RPI Camera board plugs directly into the CSI connector on the Raspberry Pi. It's able to deliver a crystal clear 5MP resolution image or 1080p HD video recording at 30fps with latest v1.3. Board features a 5MP (2592 × 1944 pixels) Omni vision 5647 sensor in a fixed focus module. The module attaches to Raspberry Pi, by way of a 15 pin Ribbon Cable, to the dedicated 15 pin MIPI Camera Serial Interface (CSI), which was designed especially for interfacing to cameras. The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data to the BCM2835 processor.

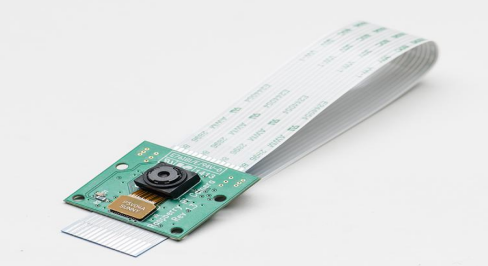


Figure 7 raspberry pi camera [ 7]

**Feature**

* Fully compatible with both the Model A and Model B Raspberry Pi
* 5MP Omni vision 5647 camera module
* Still picture resolution of 2592 x 1944
* Video supports 1080p at 30fps, 720p at 60fps and 640x480p 60/90 recording
* 15 pin MIPI camera serial interface plugs directly into the Raspberry Pi Board
* Size is 20mm x 25mm x 9mm
* Weight of 3g

**2.6.1 Installing Camera Module**

The Raspberry Pi camera module is attached to a Raspberry Pi by a ribbon cable. This cable attaches to a special connector just behind the Ethernet socket. To fit it, pull up the levers on either side of the connector, so that they unlock, and then press the cable into the slot with the connector pads of the cable facing away from the Ethernet socket. Press the two levers of the connector back down to lock the cable in place. The camera module also requires some software configuration. For that run the following command in terminal session:

$ sudo raspi-config

Then from the options there select camera option and enable it. Two commands are available for capturing still images and videos: raspiStill and raspivid. To capture a single still image, use the raspiStill command as shown here:

$ raspistill -o image1.jpg

A preview screen displays for about five seconds and then takes a photograph and stores it in the file image1.jpg in the current directory.

To capture video, use the command raspivid:

$ raspivid -o video.h264 -t 10000

The number on the end is the recording duration in milliseconds in this case, 10 seconds.



Figure 8. Raspberry Pi with camera module fitted

**2.7 LCD**

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi segment LEDs. The reasons being LCDs are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments) animations and so on.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data. We will be using this LCD to display speed and frequency given by HB100 microwave dopplar radar speed sensor. It has 3 control pins RS(register select), E(Enable) and RW(read/write).

The LCD have basic two operating modes Instruction mode and Character mode depending upon the status of the pin, the data on the 8 data pins (D0-D7) is either treated as instruction or character data. The enable pin has a simple function. It is just a clock input for the LCD. It is just held at VCC by a pull up resistor. The instruction or the character data at the data pins (D0-D7) is processed by the LCD on the falling edge of this pin. Generally we always use LCD to display something but in some cases we need to read from LCD in that case R/W pin is used



Figure 9 LCD jhd162a

|  |  |  |
| --- | --- | --- |
| **Pin number** | **Name** | **Function** |
| 1 | VSS | Ground voltage |
| 2 | VCC | +5v |
| 3 | VEE | Contrast voltage |
| 4 | RS | Register select  0 =instruction register  1 =data register |
| 5 | R/W | Read/write, to choose write or read mode  0 =write mode  1 =read mode |
| 6 | E | Enabled  0 =start to latch data to lcd character  1 =disable |
| 7 | DB0 | Data bit 0(LSB) |
| 8 | DB1 | Data bit 1 |
| 9 | DB2 | Data bit 2 |
| 10 | DB3 | Data bit 3 |
| 11 | DB4 | Data bit 4 |
| 12 | DB5 | Data bit 5 |
| 13 | DB6 | Data bit 6 |
| 14 | DB7 | Data bit 7(MSB) |
| 15 | BPL | Blanck plane light +5v or lower(optional) |
| 16 | GND | Ground voltage |

Table1 Pin configuration of LCD JHD162A

**2.8 Number Plate Recognition**

After detection of speed of the vehicle, the system checks for the over speed according to the threshold set and if crossed it will trigger to capture the moment of the event. The positioning of the camera will be structured previously to get the image of back part of over speeding vehicle. The image thus captured will be sending to detect and recognize the license plate of the vehicle.

**2.8.1 Image Acquisition**

Image acquisition is done by the raspberry pi camera which captures the image when it is triggered. The positioning of the camera will be structured previously to get the image of back part of over speeding vehicle. A better choice is an HD Infrared (IR) camera.

**2.8.2 Processing**

The aim of this stage is to perform some basic tasks in order to render the resulting image more suitable for the job to follow. Preprocessing stage incorporated several image processing concepts such as resizing, noise removal, filtering, contrast enhancement techniques, color conversion and others.

**.2.8.2.1 Size Modification**

The image size from the camera might be large and can drive the system slow. It is to be resized to a feasible aspect ratio.

**2.8.2.2 Conversion of Color Space**

Images captured using IR or photographic cameras will be either in raw format or encoded into some multimedia standards. Normally, these images will be in RGB mode, with three channels (viz. red, green and blue). Number of channels defines the amount color information available on the image. Thus, the image has to be converted to gray scale to make it appropriate for further processing.

**2.9.1 Localization of Numbers**

The localization is the process of making the binary image, localization is done by an image processing technique called Thresholding. The pixels of the image are truncated to two values depending upon the value of threshold. This is an easy and convenient way to perform image segmentation based on different intensities or colors in the foreground and background part of an image.

Not all images can be segmented successfully into foreground and background using simple thresholding. Its accuracy depends on the distribution of the intensity histogram. If the intensity distribution of foreground objects is quite distinct from the intensity distribution of background, it will be clear to apply thresholding for image segmentation.

**2.9.2 Connected Component Analysis**

After localization of the image, connected component algorithm is applied to eliminate undesired areas of the image. This connected-component labeling is alternatively referred as connected-component analysis, blob extraction, region labeling, and blob discovery. This algorithm is used in computer vision to detect connected regions in binary digital images, although color images and data with higher dimensionality can also be processed. When integrated into an image recognition system or human-computer interaction interface, connected component labeling can operate on a variety of information.

**2.9.3 Character Segmentation**

Character segmentation is the technique in which individual character present in the image is separated out. Here all character is checked out individually.it is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. mage segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

**2.9.4 Character Recognition**

Character recognition system is the system, which is used to extract the characters on the plate. The segmented characters are passed to optical character recognition engine. OCR engine is used to extract text from an image or a scanned document.

**3 Methodology**

**3.1 System Block Diagram**

The overall system has a number of components, with specific tasks, working together. Connection of the different components is as shown in the block diagram below.

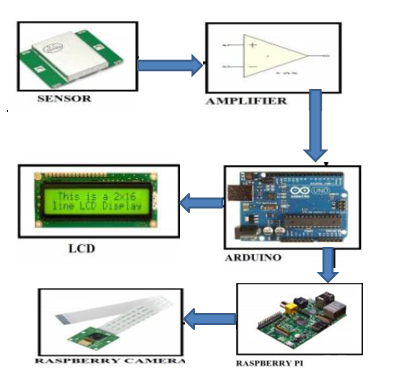


Figure 10 block diagram of the system

**3.2 Diagram Description**

Firstly, the HB100 sensor is a major part of the system. It senses the moving vehicle and produces output sinusoidal signal. The output signal is amplified by a very high gain amplifier. The output of the amplifier is fed to the digital pin of Arduino. The microcontroller measures the frequency of the input signal and calculates the speed of the vehicle from the value of frequency. The value of speed is displayed on the LCD panel. It also checks if the speed limit is exceeded and signals raspberry pi if the speed limit is crossed. The raspberry pi triggers the camera attached to the board. The image is then sent to the server via internet connection. The server processes the image to extract the characters in the License plate. The extracted character is then sent to the next station.

## 3.3 Flow chart

Check the frequency shift of vehicle with microwave Doppler radar sensor

Amplify the signal

Send the signal to Arduino to calculate speed

Compare if the speed is > speed limit

**YES**

Send high voltage to raspberry pi pin to trigger the camera

Send image to the server

Figure 11 system flow diagram

The overall system can be broadly divided into three parts:

* Speed Detection
* image acquisition and transfer
* image processing

**3.4 Speed Detection**

Radar sensor was used to detect the speed of the vehicles. Among the various methods researched for speed detection, radar detection seemed to be the most accurate way of detecting speed of vehicles. It ++is widely used in the commercial speed gun used by Traffic Police for checking on speeding vehicle. The radar sensor has a built in oscillator and mixer.

The oscillator outputs a sinusoidal signal of frequency 10.525 Ghz and radiates the signal to a direction using microstrip patch antenna. The signal strikes the moving vehicle and reflected back to the sensor with shift in frequency according to Doppler principle. The mixer generates a signal that is the difference between the transmitted and the received signal and the frequency of the output signal is proportional to the speed of vehicle. The output is a low amplitude frequency signal in audio range.

The sensor is a 3 pin circuit. Vcc, Gnd and output pin. +5 volt is supplied between Vcc and Gnd and the output is taken from the output pin and Gnd.

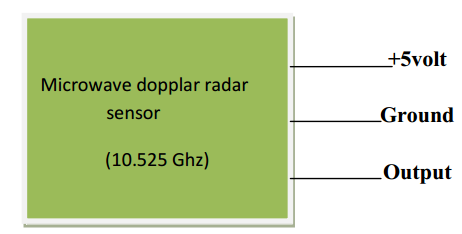


Figure 12 simplified diagram speed detection

Using the Doppler shift equation to calculate the relation between the shift in frequency and velocity of the moving object.

Where: fd= shift in frequency in Hz (i.e. frequency of the output signal from radar)

v=velocity of moving object

c=velocity of electromagnetic wave in vacuum

f=frequency of transmitted wave from the sensor (10.525 Ghz

Assuming the object is directly ahead so that the value of angle cosA=1, the expression for

velocity of the object reduces to:

v= fd/19.49074 km/hour

For v=0, fd=0

For speed limit of 60 Km/hr., i.e. v=60 Km/hr., output frequency fd=1169. 428Hz.If the output signal of the sensor has frequency exceeding 1169.444Hz, the vehicle exceeds the speed limit of 60 km/hr.

**3.5 Amplification of Signal**

The output of the sensor is sinusoidal wave with very small amplitude and needs to be amplified before further processing. The output of sensor is in the audio frequency range. However, we are concerned with the frequency below the speed limit. i.e. nearly 1.1 KHz. So an audio amplifier with low pass active filter is desired for the design. The amplifier IC used for this purpose is the LM324 Quad op amp. This IC has four operational amplifiers.

The output signal from the radar is in the range of few millivolts. So high gain of the amplifier is needed to shift the voltage level to the point that is able to drive a TTL logic. The amplification is achieved in two stages to minimize the effect of noise. The complete diagram of the amplifier circuit used is as shown below

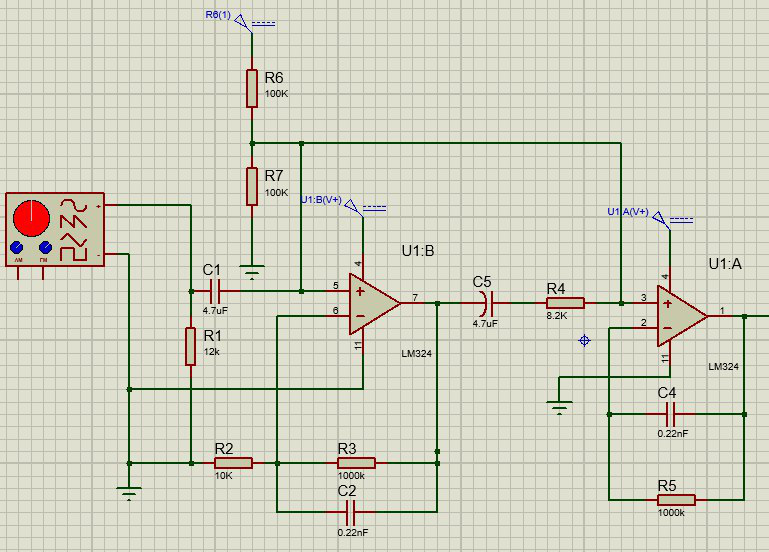


Figure 13 Amplifier Design

Amplification of the radar output is achieved using two of the four stages of LM324 IC.

**3.5.1 Non-Inverting Stage**

The first stage is configured as Non-inverting amplifier. The output of the sensor has a small dc value (0.01 to 0.2 Vdc) and so AC coupling of the output is necessary. The input is supplied to the non-inverting terminal via a capacitor (4.7 µF) and a resistor(12K) is connected to the ground. Dc offset of nearly 2.5 volts is generated using a resistive divider network. Two resistors of 100K are connected between +5 volt and Ground to generate 2.5volt at the center. The gain of the first stage is given by the values of the resistor in the feedback path(1M) and the resistor from the inverting terminal to ground(10k). The gain in the non-inverting amplifier is

**Gain= (1+1M/10K) = 100**

**3.5.2 Inverting Stage**

The output of first stage is a sinusoidal signal with a small dc offset. The dc component at the output is filtered by the capacitor at the output terminal. The second stage is a op-amp configured as inverting amplifier. The signal is applied to the inverting terminal via 8.2K resistor and dc offset of 2.5 volt is supplied to the non-inverting terminal. The gain for this stage is given by the values of the feedback resistor (1M) and the resistor at the inverting input terminal (8.2 K). The gain in this stage is given by the equation

**Gain= -1M/8.2K =122**

The 2.2nF capacitor in the feedback path of the amplifier makes this configuration an active low pass filter. The high frequency noises are filtered and the accuracy of the system is improved.

The two stages are cascaded to achieve the overall gain of the amplifier. The overall gain is given by the product of the gain in individual stages.

**Overall gain of amplifier= 100 \* 122 =12,200**

The gain of the amplifier is so high that the op-amp in the second stage reaches saturation. Since we are concerned only with the frequency of the signal and not the shape of the wave form, the output is driven to saturation point with very high gain. The second stage acts as a comparator and the output produced is a digital waveform that can be sensed by the digital pin of a microcontroller.

**3.6 Measurement of Frequency**

Frequency of the signal is the factor representing the speed of the moving object. So the next stage is the measurement of the frequency of the incoming pulse from the amplifier. Output from the amplifier is connected to the external interrupt pin of atmega 328. The corresponding pin in Arduino Uno is digital pin 8. Counter of atmega328 is used for the measurement of the frequency of input signal. External interrupt pin of atmega328 can be programmed to work with either rising edge or falling edge. So, Interrupt pin can be used to detect rising edge in the input signal. The rising edge of the input signal provides interrupt to the microcontroller. The interrupt subroutine starts the counter. On the next rising edge, the counter stops. The frequency of the input signal can be obtained from counted value and the clock of the counter. A number of such measurements can be taken and averaged for better accuracy.

**3.6.1 Algorithm for Measuring Frequency**

1) Enable external interrupt and check for rising edge in the input signal

2) At rising edge, start counter

3) Wait for next rising edge

4) At next rising edge, stop counter

5) Divide clock frequency of counter by output of counter

6) Repeat steps 1-5 for a number of times and take average

The microcontroller is run with a 16 MHz crystal. So if the counter inside it counts ‘n’ between two successive rising edges of the input signal then the frequency of the input signal is 16/n Mhz. This approach of measuring frequency is accurate when the frequency of the input signal is very small compared to the clock of microcontroller

**3.7 Display**

The Arduino board not only calculates and displays speed but also compares it with a preset threshold. The threshold for our design is 60 km/hr. When it detects over speed, Digital Pin 8 is set high for a short instance. Output pin of microcontroller is connected to raspberry Pi to signal the Pi to trigger the camera whenever speed limit is crossed. Direct connection between atmega328 and raspberry Pi cannot be made due to difference of logic level for the two processors. So a resistive voltage divider network is used to shift voltage level of 5V at output pin to 3.3V. The output pin of atmega is connected to ground via two resistors 270 ohm and 470 ohms in series. Voltage across 470-ohm resistor and ground is fed as input to raspberry Pi’s digital pin. Digital pin 23 of raspberry Pi is configured as input pin. When the digital pin in raspberry pi goes high it triggers its HD camera and captures image of the approaching vehicle

The value of the speed calculated from the frequency was displayed on a 2\*16 LCD display. The LCD was interfaced to the microcontroller as per the instruction provided in the datasheet.

|  |  |  |  |
| --- | --- | --- | --- |
| Pin  No. | Symbol | Function | Remarks |
| 1 | Vss | GND | Connected to Ground |
| 2 | Vdd | +3V or +5V | Connected to +5 volt |
| 3 | Vo | Contrast Adjustment | Connected to ground for constant contrast. |
| 4 | RS | H/L Register Select Signal | Digital pin 12 of Arduino |
| 5 | R/W | H/L Read/Write | Connected to ground |
| 6 | E | H → L Enable Signal | Digital pin 11 |
| 7 | DB0 | H/L Data Bus Line | Not connected |
| 8 | DB1 | H/L Data Bus Line | Not connected |
| 9 | DB2 | H/L Data Bus Line | Not connected |
| 10 | DB3 | H/L Data Bus Line | Not connected |
| 11 | DB4 | H/L Data Bus Line | Digital pin 5 |
| 12 | DB5 | H/L Data Bus Line | Digital pin 4 |
| 13 | DB6 | H/L Data Bus Line | Digital pin 3 |
| 14 | DB7 | H/L Data Bus Line | Digital pin 2 |
| 15 | A/Vee | + 4.2V for backlight  LED | Connected to Vcc with 570 ohm resistor |
| 16 | K | Power Supply for  B/L (OV) | GND |

Table 2 display

**3.8 Image Acquisition and Processing**

* Take an image
* Resize the image keeping the aspect ratio same
* Convert to gray scale
* Apply median filtering to remove noise

 Preprocess 

Original image gray image

All contours possible chars

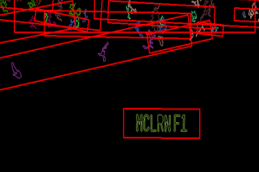
 

Figure 14 license plate region extraction steps

 Preprocess 



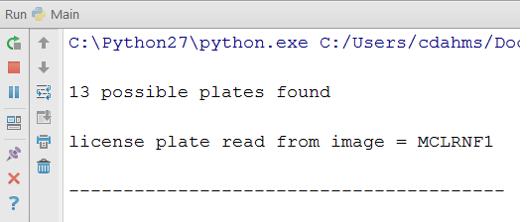


Figure 15- character segmentation and recognition steps

**4. Software and Tools Used**

### 4.1 Tools used for simulation

#### **4..1.1 Proteus**

The Proteus VSM combines mixed mode SPICE circuit simulation, animated components and  
microcontroller models to facilitate co-simulation of complete microcontroller based design. It is possible to develop and test such designs before a physical prototype is constructed. This is possible because interaction with the design is possible using circuit indicators like LED, display panels, actuators etc. It also provides extensive debugging facility by employing breakpoints, single stepping and variable display of both assembly code and high level language source code.

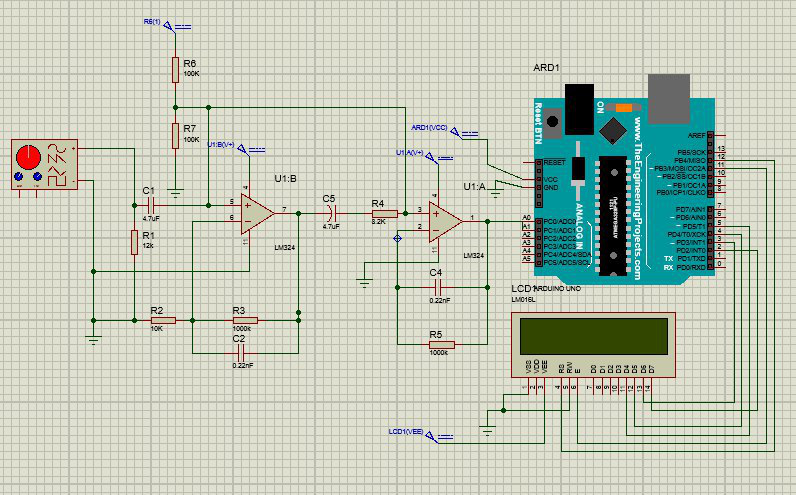


Figure 16 Arduino –lcd interface

#### **4.2 Software Used**

**4.2.1 Arduino IDE**

#### The Arduino development environment contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions, and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them. It has many built in libraries along with libraries for LCD display. We can also install third party library in the latest IDE of Arduino.

### **4.2.2 Python (programming language*)***

Python is a widely used general purpose high level programming language which emphasizes program readability and reduces complexity in coding. Its syntax allows programmers to express concepts in fewer lines of code than would be possible in other languages. The language provides constructs intended to enable clear programs on both a small and large scale. It supports multiple programming including object oriented, imperative and functional programming. It has feature like automatic memory management and has a large library.

Python uses dynamic typing and a combination of reference counting and a cycle detecting garbage collector for memory management. An important feature of Python is dynamic name resolution (late binding), which binds method and variable names during program execution.

Features of python programming language:

* Simple, easy to learn, free and open source
* Interpreted
* Object Oriented
* Extensible.

Python is indeed an exciting and powerful language. It has the right combination of performance and features that make writing programs in Python both fun and easy.

**4.2.3 Real VNC**

It is a company that provides remote access software. The software consists of a server and client application for the Virtual Network Computing (VNC) protocol to control another computer’s screen remotely. Real VNC clients using can VNC viewer run in full-screen mode they use the F8 function-key as the default key for bringing up an options menu. The server component of Real VNC allows a computer to be remotely controlled by another computer. The software can be installed for legitimate purposes, but it can also be installed from a remote location by an attacker with malicious intent.

**4.2.4 PuTTY**

PuTTY is a free and open-source terminal emulator, serial console and network file transfer application. It supports several network protocols, including SCP, SSH, Telnet, rlogin, and raw socket connection.

Some features of PuTTY are:

* The storing of hosts and preferences for later use.
* Control over the SSH encryption key and protocol version.
* Public key authentication support.
* Support local serial port connection.
* Self-contained executables required on installation.

**5. Simulation Result**

**5.1 Calculation of speed**

If ‘f’ be the value of frequency measured by the microcontroller, then the corresponding speed in Km/hr. is obtained by dividing it with a factor 19.49. Some of the values of frequency and their corresponding speed are presented in the table below.

|  |  |
| --- | --- |
| Frequency(hz) | Speed((km/hr) |
| 22 | 1.128 |
| 31 | 1.56 |
| 44 | 2.29 |
| 95 | 4.9 |
| 336 | 17.29 |
| 494 | 25.35 |

Table 3 Calculation of speed

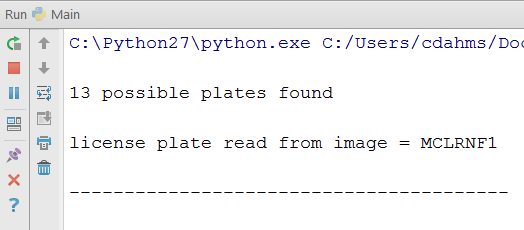
From the above result we can notice that, the frequency is directly proportional to the speed.so to estimate the cars speed which is beyond the limit we can find approximate value of corresponding frequency.

**5.2 Number Plate Recognition**

The original input image

Figure 17 real time (input) image and license plate region



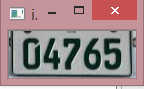
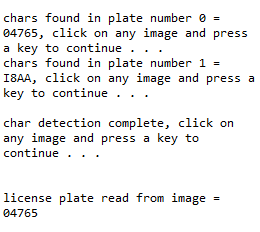
 

Figure 18 real time (input) image and license plate region



# 

**6 Hardware Testing**

**6.1 Acquiring Testing Equipment**

**6.1.1 Signal Testing Equipment**

DC offset and amplifier circuit testing was originally planned to be performed using function generator and oscilloscope. But it was difficult to find all the components at a time or to adjust a proper testing environment

**6.2 Speed Detection Test**

After simulation we have tested the hardware by setting up the circuit and visualizing the variation of frequency and corresponding speed with the help of HB 100 Sensor. By varying the speed of the motion of the object we were able to read the frequency and the speed on the LCD

The figure shown below is the result of Hardware testing with frequency of 10.525 GHz

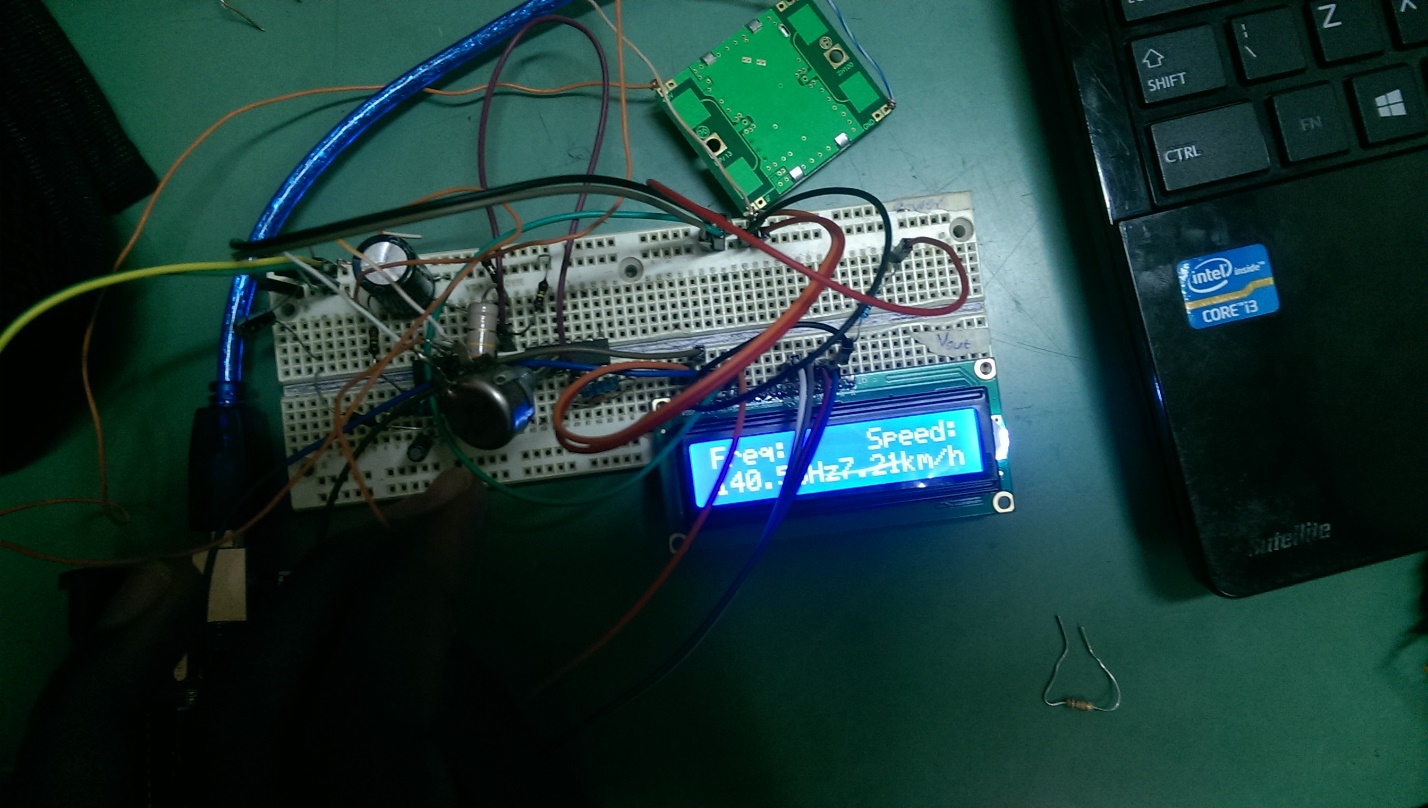


Figure 19 speed detection test

**6.3 Camera Acquisition Test**

From the result of the speed detection, the raspberry pi camera is triggered to capture the back side of the vehicle, when the speed which is calculated from the frequency exceed the threshold limit.

Figure 20 Camera Acquisition test

**6.4 Assembled Hardware**

Figure 21 overall assembled Hardware

**7. Discussion and Conclusion**

**7.1. Discussion**

As it is mentioned in the chapters above the project contains different subsystems that have their own concept and fundamental principles. So it is the success of the discreet subsystems that will result for the success of the whole project.

The results of the first and the second test, speed detection and camera acquisition, are almost perfect. As shown in the simulation and hardware test results above, the speed detection test has the result as required. As we can notice from speed detection result, the frequency and the speed are directly proportional. The speed was expected to increase with respect to the frequency.

In the image acquisition hardware test, the raspberry pi camera as shown in the figure is triggered by the output of the Arduino to capture the expected image of the vehicle’s license plate .in order to do the acquisition process the speed of the vehicle has to exceed the threshold limit. This process is takes place by Arduino. The output of the Arduino is the input to the raspberry pi camera.

The third test is overall assembled hardware. This was the most challenging part since the operating system of the raspberry pi and interconnection between Arduino and raspberry pi was very challenging.

* 1. **conclusion**

The main objective was to detect over speed detection and processes in developing the license plate recognition systems. The designed speed detection system was capable of continuously monitoring the speed of the approaching vehicle. The output was more accurate with no other moving objects in the surrounding or when we only consider the single car. The value of speed of each passing vehicle was displayed in the LCD display. With each over speeding vehicle passing by the sensor, the camera was triggered and the image was saved in the SD card. The Number Plate Recognition system was not perfect and requires modification, especially for Ethiopian plate. It accurately identified few of the characters but not all.

The main processes in developing the systems are detecting and extracting the plate image and recognizing the characters on the plate image. In detecting the plate region, image processing techniques are incorporated. For preprocessing phase, gray scale conversion along with median filtering technique is used. For detecting the plate region, connected component analysis (CCA) is used to analyze each region and rate the potential to be assigned as a plate region. The technique is mainly focuses on finding and segmenting white regions. Finally, using structural properties, the plate region is identified from the candidate regions.

after the plate image is extracted, it is passed to the character recognition system. For this research, OCR was used for recognizing the characters on the plate image. For segmenting the individual characters, CCA is used. After the individual characters are segmented, a template matching technique was used to classify the characters into the correct character class. Template matching techniques was preferred because the characters on the plates are standard. This property helps the template matching technique to work better. In conclusion, in this work, we tried to develop the CPR system that recognizes more varieties of standard car plates to increase the accuracy of the traffic management. In addition, we succeed in detecting the over speed vehicle by the principle of Doppler radar effect.

**8. Problem Faced and Future Enhancement**

**8.1 Problem Faced**

**8.1.1 Sensor**

The microwave Doppler radar sensor is not available in the local market of Ethiopia and had to be ordered from abroad. It took about three months for the vendor to ship the sensor to Nepal. The sensor initially worked well but quickly broke down due to some unseen reasons. It took another few months for the next sensor to be available. Designing and real time testing was not possible without the sensor and a huge time was wasted waiting for the piece to arrive.

The Hb100 sensor has a range of 10 meters which is not sufficient for a good speed detection system. However, this problem can be solved by using other sensors like DF00 which work on the same principle but have effective range of up to 100 meters.

Also the sensor used has a wide angle of coverage. The sensor is affected not only by the objects moving directly ahead but by the objects in the surrounding as well. The directivity of the radiator can be improved by using a directional antenna.

The sensor detects the speed of not only moving vehicles but any moving object in its field of view. This could be prevented by using a sensor whose radiation wavelength is of the order of the dimension of vehicles.

**8.1.2 Camera**

The type of camera required for our system is a high resolution high speed camera. Few vendors could supply such camera but with cost that was far beyond our budget. There is a different category of cameras made for the specific purpose of License Plate Recognition. Such cameras use their own light of specific wave length and so are least affected by the external lighting condition and illuminance of surface. But such camera was not affordable for a student project like us and we had no resource to access and test one.

The camera we used is a 5 Megapixel High definition camera that comes with the raspberry pi. It is easy to interface with the Pi board but the image quality is very poor and unacceptable for image processing operation such as License Plate Recognition. The camera takes nearly one second to take a picture of good quality but this time is not acceptable when working on the road with vehicles running over 60 km/hr.

**8.1.3 Number Plate Recognition**

Another problem faced in the Ethiopian Number Plate is the dissimilarity in the font used by different persons. There is a standard for the font, height and width of the characters in the plate. the dimension of the character and the font varies among different number plates. Our approach of Character recognition in the number plate is by template matching. So the accuracy of the system depends on the prior knowledge of the dimension and style of the characters.

**8.2 Enhancement**

Although automatic systems are advantageous regarding the simplicity of the task, the reliability won’t be like that from human work. At such, our system has different limitations whose prime cause being the ones we discussed in problem Faced. Resolving these issues will help to enhance our system to a level where following objectives could be met: The stolen vehicle can be detected by comparing with the registered entry of stolen vehicles.

Accuracy can be increased using Neural Networks. Advanced image processing algorithms and libraries could be used so that the system can be used efficiently even during unfavorable lighting conditions and during the night time as well.

Can be used as traffic counters to count the number of vehicles plying on a highway.

The Number plate of vehicle which has crossed over speed limit can be sent to mobile of traffic police in the area using Android application.

Adding more vehicle detection wireless sensing networks and implementing a wireless sensor network will be another interesting application which will open up much more applications areas.

Security of data during communication can be another important work for the future development of this project.

**9 Cost Estimation**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NO | Item | Specification | Birr | Quantity | Total  (Rs.) |
| 1 | Raspberry pi | model B | 3500 | 1 | 3500 |
| 2 | camera | rev1.3 | 1500 | 1 | 1500 |
| 3 | Arduino | UNO | 600 | 1 | 600 |
| 4 | Sensor | HB100 | 200 | 1 | 200 |
| 5 | Microcontroller | atmega328 |  |  |  |
| 6 | Resistors | pack | 3 | 6 | 18 |
| 7 | Capacitors | pack | ? | 4 |  |
| 8 | Crystal | 16MHZ |  |  |  |
| 9 | Matrix Board |  | ----------- | 1 | ----------- |
| 10 | memory card | 16GB | 160 | 1 | 160 |
| 11 | Male female header |  | 3 | 12 | 36 |
| 12 | Connectors |  | 5 | 3 | 15 |
| 13 | LCD | 16\*2 | 200 | 1 | 200 |
| 14 |  |  |  |  |  |
| **Total** |  |  |  |  |  |

Table 4 estimated cost estimation

**Reference**

[1] Huda Zuber Ahmed “Design and Implementation of Car Plate Recognition  
System for Ethiopian Car Plates”

[2] A License Plate Recognition and Speed Detection System” (C. Leo, D. David, 2008)

[3] H. Asha, R UdayShankar, Rashmin “Radar Based Cost Effective Vehicle Speed

Detection Using Zero Cross Detection”

[4] Final thesis(aradar system for voiding car collisions on u-turn roads ,,,,,,,by Rediet)

[5] "HB100 Motion Sensor Module" AgilSense. Web. 28 Apr. 2010

[6 rediet final thesisa

[7] “Raspberry pi”

http://en.wikipedia.org/wiki/Raspberry\_Pi

[8] “JHD162A SERIES”

http://www.itron.com.cn/PDF\_file/JHD162A SERIES.pdf

[9] Doppler effect

http://en.wikipedia.org/wiki/Doppler\_effect

[10] automatic License Plate Recognition using Python and OpenCV, K.M. Sajjad

[11] Seble nigusse “Recognition of Ethiopian Car Plate”

[12] Parul Shah, Sunil Karamchandani, Taskeen Nadkar, Nikita Gulechha, Kaushik Koli, Ketan

Lad. “OCR-based Chassis-Number Recognition using Artificial Neural Networks”. IEEE 2009.

[13 ] Satadal Saha, Subhadip Basu, Mita Nasipuri and Dipak Kr. Basu. “A Hough Transform

**Appendices**

**Appendix I**

#include <LiquidCrystal.h>

#include "arduinoFFT.h"

#define SAMPLES 128 //Must be a power of 2

#define SAMPLING\_FREQUENCY 1000 //Hz, must be less than 10000 due to ADC

arduinoFFT FFT = arduinoFFT();

unsigned int sampling\_period\_us;

unsigned long microseconds;

double vReal[SAMPLES];

double vImag[SAMPLES];

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

const int sensor=A1; // Assigning analog pin A1 to variable 'sensor'

float speedKM;

float freq;

void setup() {

Serial.begin(9600);

sampling\_period\_us = round(1000000\*(1.0/SAMPLING\_FREQUENCY));

pinMode(sensor,INPUT); // Configuring pin A1 as input

lcd.begin(16,2);

delay

**Appendix II**

# Main.py

import cv2

import numpy as np

import os

import DetectChars

import DetectPlates

import PossiblePlate

# module level variables

showSteps = True

def main():

blnKNNTrainingSuccessful = DetectChars.loadKNNDataAndTrainKNN() # attempt KNN training

if blnKNNTrainingSuccessful == False: # if KNN training was not successful

print("\nerror: KNN traning was not successful\n") # show error message

retur # and exit program

# end if

imgOriginalScene = cv2.imread("LicPlateImages/10.png") # open image

if imgOriginalScene is None: # if image was not read successful

print("\nerror: image not read from file \n\n") # print error message to std out

os.system("pause") # pause so user can see error message

return # and exit program