International Journal of Multidisciplinary Engineering in Current Research - IJMEC
Volume 9, Issue 5, May-2024, http://ijmec.com/, ISSN: 2456-4265

TRAFFIC CONGESTION CONTROL WITH AUTOMATIC SIGNAL CLEARANCE FOR EMERGENCY VEHICLES

Mr. Mohd Abdullah [*1], Mr. Anurag Sharma [*2], Mr. Md Dilshad Alam [*3],
Mr. K. Sampath Kumar[*4],
[*1, 2, 3] B. Tech. Student Dept. of ECE, Shadan College of Engineering and Technology
[*4] Associate Professor, Dept. of ECE, Shadan College of Engineering and Technology
mohd.abdullah06102002@gmail.com[*1], anuragsharmawaboy357@gmail.com[*2],
mohammeddilshad481@gmail.com*[3], sampath.shadan@gmail.com[*4],

Abstract-The increasing daily fatalities are due to traffic jams and critical situation which result in delay of ambulance reaching the hospitals and desired locations. This is due to vehicle traffic on the roads. The ascent of the Internet of Things has offered urban communities a chance to definitely advance their traffic circumstance by utilizing the web to help ease their traffic issues. The internet has likewise assisted by introducing advancement, and with development came effectiveness and common sense. Numerous peoples have started utilizing programming and applications for ease of traveling. This paper aims at creating an ambulance friendly traffic system, by regulating the traffic with the help of the mobile application installed in the ambulance driver's phone works according to the traffic density on the specific street.

Keywords: NODEMCU, Arduino, Traffic Signaling.

1 Introduction

In the present world, everybody is in a rush. This eventually turns out as uncontrollable traffic on the streets. Metro urban areas, specifically, are where population on roads and traffic congestion are a regular occurrence. This couldn't influence any person or thing, more than ambulances. We have seen a tons of cases, where ambulances get caught in the mid of the practically unfaltering traffic, leading to delay in arrival at the spot of crisis on schedule. This is something abominable, as it could cost lives. This project aims at creating an ambulance friendly traffic system, by regulating the traffic with the help of Mobile application installed in the mobile of ambulance driver. According to the traffic density present on the particular road the driver can change traffic light to green until the passage of ambulance through traffic. The ascent of the Internet of Things has offered urban areas a chance to radically advance their traffic circumstance by utilizing the internet to help reduce their traffic issues by introducing development and with advancement came proficiency and cost-effectiveness. Numerous people have started utilizing programming and applications to design their courses to sidestep clogged streets and expressways.

2. Literature Review

The literature survey conveyed 64 percent of the world's population will be living in cities by 2025, approximately 5.2 billion people. By 2035, roughly 66%, or 6 billion people will live in urban areas. This not

DOI: https://doi-ds.org/doilink/05.2024-16752678

just addresses a monstrous test by the way we construct and oversee urban areas however critical freedom to improve the existences of billions of individuals.

[1].Thus various techniques of solving the problem statement like using IoT, cloud computing, detections using sensors. Many smart city projects are working on this system using centralized systematic planning for achieving a smooth ambulance friendly city

[2]. The applications like apache help in establishing IoT communication interface. Another paper illustrates about revolutionary development in the field of Internet of Things (IoT) and how it can be seamlessly & widely in large number of end system where subset of a large amount of data can be accessed and processed easily and powerfully

[3] .The Internet of Things (IoT) is the communication of exceptionally recognizable installed calculating gadgets inside the current Internet architecture. Emergency services must be delivered efficiently and on schedule. He/she should be taken to the hospital as speedy as possible and treatment as to carry out fast to save his/her life.

[4]Integration of cloud computing with internet of things : challenges faced and open issues gives us a deep perspective and definition of the cloud computing and internet of things . The communication is established, the storing of data and the processing capabilities of both the technologies, also explores various new models and their application. The paper likewise clarifies the difficulties which we can face and how to beat them.

[5] low profile and minimal expense material for pressure detecting and position planning we became acquainted with ,the plan and alignment of the pressing factor cushions and its interfacing with Atmega is clarified exhaustively get acknowledge with the application of pressure sensor and its structure . The capacities and efficiency of sensor is also tested in this paper.

3. Project Description

AIM:

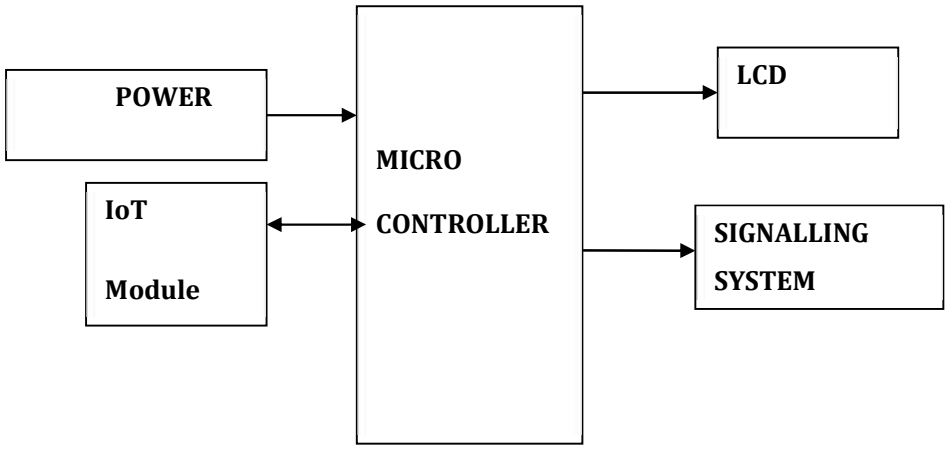
The main aim of this project is to control the traffic whenever the AMBULANCE vehicle will come in that way.

PURPOSE:

The purpose of this project is to overcome the draw backs in the normal traffic controlling system and to design traffic controlling system to enter the SWITCHES which overcomes the problem of heavy traffic in the cities for AMBULANCE.

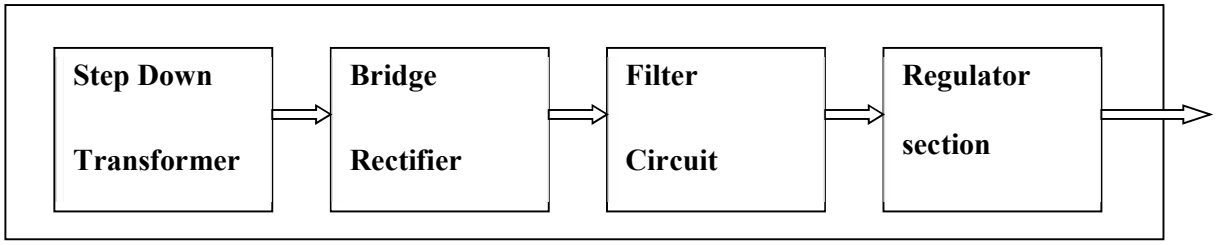
BLOCKDIAGRAM:

DOI: https://doi-ds.org/doilink/05.2024-16752678



REMOTE SECTION:

POWER SUPPLY:



ARDUINO

Overview:

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter. The Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.

The board has the following new features:

1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible with both the board that uses the AVR, which operates with 5V and with the Arduino Due that operates with 3.3V. The second one is a not connected pin, that is reserved for future purposes.

- Stronger RESET circuit.

DOI: https://doi-ds.org/doilink/05.2024-16752678

- Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

Node MCU:

NodeMCU is a low-cost open source IoT platform. It initially included firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which was based on the ESP-12 module.[6][7] Later, support for the ESP32 32-bit MCU was added.

NodeMCU is an open source firmware for which open source prototyping board designs are available. The name "NodeMCU" combines "node" and "MCU" (micro-controller unit). The term "NodeMCU" strictly speaking refers to the firmware rather than the associated development kits.

Both the firmware and prototyping board designs are open source.

The firmware uses the Lua scripting language. The firmware is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and SPIFFS. Due to resource constraints, users need to select the modules relevant for their project and build a firmware tailored to their needs. Support for the 32-bit ESP32 has also been implemented.

The prototyping hardware typically used is a circuit board functioning as a dual in-line package (DIP) which integrates a USB controller with a smaller surface-mounted board containing the MCU and antenna. The choice of the DIP format allows for easy prototyping on breadboards. The design was initially was based on the ESP-12 module of the ESP8266, which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IoT applications

NodeMCU was created shortly after the ESP8266 came out. On December 30, 2013, Espressif Systems began production of the ESP8266. NodeMCU started on 13 Oct 2014, when Hong committed the first file of nodemcu-firmware to GitHub. Two months later, the project expanded to include an open-hardware platform when developer Huang R committed the gerber file of an ESP8266 board, named devkit v0.9. Later that month, Tuan PM ported MQTT client library from Contiki to the ESP8266 SoC platform, and committed to NodeMCU project, then NodeMCU was able to support the MQTT IoT protocol, using Lua to access the MQTT broker. Another important update was made on 30 Jan 2015, when Devsaurus ported the u8glib[16] to the NodeMCU project,[17] enabling NodeMCU to easily drive LCD, Screen, OLED, even VGA displays.

ESP8266 Arduino Core

As Arduino.cc began developing new MCU boards based on non-AVR processors like the ARM/SAM MCU and used in the Arduino Due, they needed to modify the Arduino IDE so that it would be relatively easy to change the IDE to support alternate toolchains to allow Arduino C/C++ to be compiled for these new processors. They did this with the introduction of the Board Manager and the SAM Core. A "core" is the collection of software components required by the Board Manager and the Arduino IDE to compile an Arduino C/C++ source file for the target MCU's machine language. Some ESP8266 enthusiasts developed an Arduino

DOI: https://doi-ds.org/doilink/05.2024-16752678

core for the ESP8266 WiFi SoC, popularly called the "ESP8266 Core for the Arduino IDE".[18] This has become a leading software development platform for the various ESP8266-based modules and development boards, including NodeMCUs.

WI-FI: The WI-FI module used in this project is ESP8266. It follows TCP/IP stack and is a microchip which is less in cost. This microchip allows microcontroller to connect to a WI-FI network, by using Hayes style command connections are done or made through TCP/IP connection. ESP8266 has 1MB of built in flash, single chip devices able to connect WI-FI. Espressif systems are the manufacturers of this module, it is a 32 bit microcontroller. There are 16 GPIO pins in this module. This module follows RISC processor. It has 10 bit DAC. Later Espressif systems released a software development kit(SDK) which is used to programme on the chip, so that another microcontroller is not used. Some of the SDK's are Node MCU, Arduino, Micro Python, Zerynth and Mongoose OS. SPI, I2C, I2S, UART are used for communicating between two sensors or modules.

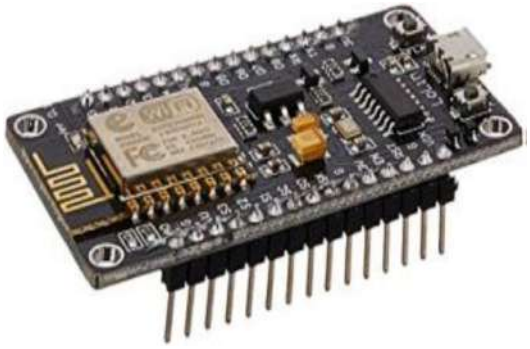


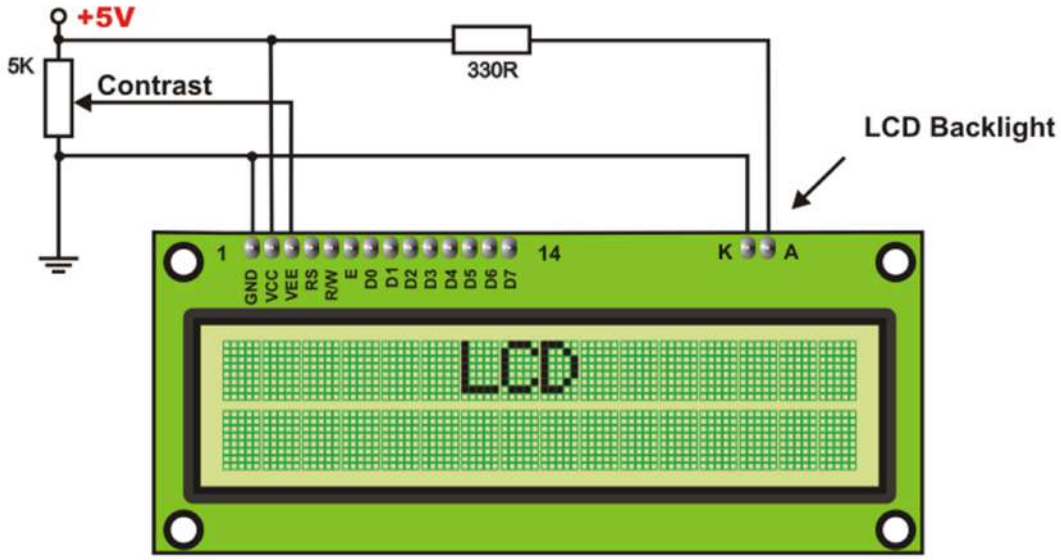
Figure : Wi-Fi module

LIQUID CRYSTAL DISPLAY

LCD screen:

LCD screen consists of two lines with 16 characters each. Each character consists of 5x7 dot matrix. Contrast on display depends on the power supply voltage and whether messages are displayed in one or two lines. For that reason, variable voltage 0-Vdd is applied on pin marked as Vee. Trimmer potentiometer is usually used for that purpose. Some versions of displays have built in backlight (blue or green diodes). When used during operating, a resistor for current limitation should be used (like with any LE diode).

DOI: https://doi-ds.org/doilink/05.2024-16752678



Arduino IDE compiler:

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards can read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT
Applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

DOI: https://doi-ds.org/doilink/05.2024-16752678

The advantages of Arduino IDE application are

1. Inexpensive
2. The simple clear programming environment
3. extensible software and hardware

PROCEDURE FOR CREATION AND CONNECTING BLYNK TO MICRO CONTROLLER

1. Create a Blynk Account

After you download the Blynk App, you'll need to create a New Blynk account. This account is separate from the accounts used for the Blynk Forums, in case you already have one.

We recommend using a real email address because it will simplify things later.

Fig 5.1 : Creation of blynk account

2. Create a New Project

After we have successfully logged into our account, start by creating a new project.

Fig 5.2 : Creation of a project

3. Choose Hardware

Select the hardware model that we will use.

Fig 5.3 : Selection of controller

4. Auth Token

Auth Token is a unique identifier which is needed to connect our hardware to our smart phone. Every new project that we create will have its own Auth Token. We will get Auth Token automatically on your email after project creation. we can copy it manually. Click on devices section and selected required device : and we see the token.

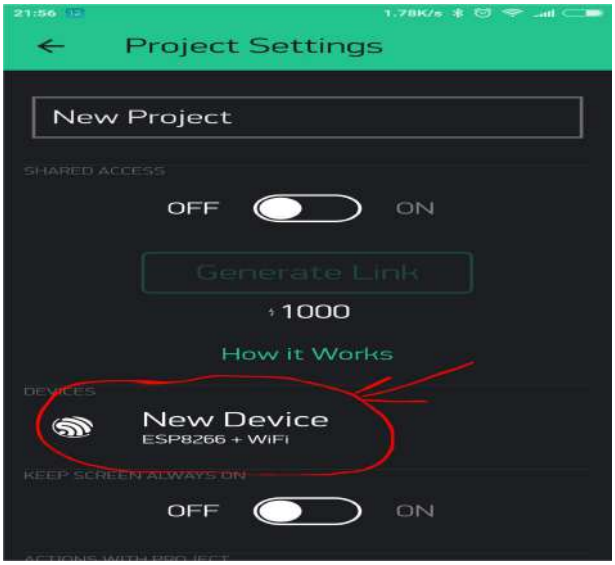


Fig 5.4 : Generation of auth token

5. Blynk Tools Creation

Required tools are taken and laced in proper positions. Various tools, Switches, joysticks, sliders, etc are available. The required video display, two joysticks, one button and GPS tools are taken.

DOI: https://doi-ds.org/doilink/05.2024-16752678

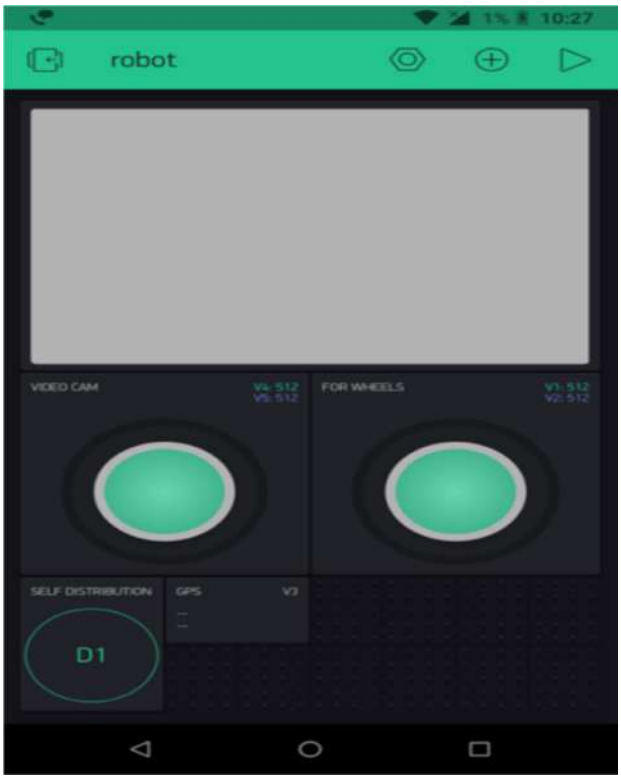


Fig 5.5 : App Appearance

Conclusion:

Human life is precious and must follow safety measures very conscious in all aspect this of course includes ambulance services too. In this by using ATSC we can achieve the Uninterrupted services of the traffic control system by implemented for signal changes to allow flow control. This system is cost effective, multiple usage and deployed using of IoT, Which is more efficient. This system will reduce accidents which often happen at the traffic signal intersection because other vehicles have to huddle to give way to the ambulance services. This life saver project must be implemented in traffic forums to aid the public in good manner.

Future Scope:

Further improvement we can add GPS module o it. The GPS track the latitude and longitude value and send to GSM, and the tracked value is to traffic light signal through internet. We can fixed a GPS module in ambulance to track the location, when it reached our fixed point the signals starts to open by the indication of plus and green signal in the traffic light pole. We have another fixed value for return back to normal position of the signal. We can control this anywhere anyplace through control room

DOI: https://doi-ds.org/doilink/05.2024-16752678

References

[1] K.Sangeetha, P.Archana , M.Ramya , P.Ramya,Automatic Ambulance Rescue With Intelligent Traffic Light System, IOSR Journal of Engineering (IOSRJEN), Issue 02 (February. 2014)

[2] Manav Kandhari, Svetlin Antonov,Smart Traffic Control System for Ambulance,September 2016

[3] Gargi Beri, Pankaj Ganjare, Amruta Gate, Ashwin Channawar, Vijay Gaikwad, Intelligent Ambulance with Traffic Control, International Journal of Electrical, Electronics and Computer Systems, Pune, February 2016

[4] Venkatesh H, Shrivatsa D Perur, Jagadish M C, An Approach to Make Way for Intelligent Ambulance Using IoT,International Journal of Electrical and Electronics Research, March 2015

[5] Takwa Tlilia, Marwa Harzi, Saoussen Krichena, Swarm based approach for solving Ambulance routing problem, International Conference on Knowlegde based and Intelligent Information and Engineering Sytems

[6] Saurabh Kapoor, Parul Gupta, Pooja Sharma, Prabhu Nath Singh, Intelligent Ambulance with Automatic Traffic Control, , International Research Journal of Engineering and Technology (IRJET), Apr - 2017

[7] Priyanka Nalawade, Prajakta Waghere, Nisha Vanare,Prajakta Kalbhor, A. J. Jadhav,"Dynamic Traffic Controlsystem Using Rfid Technology "IJARCCE,Vol.6, Issue.1, Jan2017

[8] Ruihua Zhang, and Dongfeng Yuan, "A Health Monitoring System for Wireless Sensor Networks," in Proc. of 2ed IEEE Conference on Industrial Electronics and Applications (ICIEA), pp. 1648-1652, Harbin, China, May 2007

DOI: https://doi-ds.org/doilink/05.2024-16752678