

# Intelligent Traffic Control System For Emergency Vehicle Using RF Technology

J.Rohit Varma

Mr. Sudhansu Sekhar Behera,

M.Tech Assistant Professor (Ece)

Department Of Electronics And Communication Engineering

Maharaj Vijayaram Gajapathi Raj College Of Engineering (A)

---

## Abstract

Today's scenario is people on the earth are rapidly increased and according to that the number of vehicles on the road is also increased. Therefore, the problem of traffic management is arising specially for emergency vehicles. The idea behind this paper is to implement a system which would easily control the traffic and helps for the emergency vehicles to reach at their destination. This scheme relies completely on automatic intelligent control. Here the goal is to reduce the latency of emergency vehicles with minimum or less disruption to regular traffic flow is possible. However, there is still problem for an emergency vehicle to bypass near the traffic junction. The emergency vehicles could not be going as fast as it can. So to overcome that problem we have to find the new methods

### Softwares:

- ✓ Embedded c
- ✓ Arduino
- ✓ Avr isp flash
- ✓ Usb programming
- ✓ Express pcb

### Hardware:

- ✓ Micro controller –atmega2560
  - ✓ Power supply
  - ✓ Rf modules
  - ✓ 330 ohm resistors
  - ✓ Red leds-3
  - ✓ Green leds-3
  - ✓ Orange leds-3
  - ✓ Emergency switch
  - ✓ Lcd
  - ✓ Orange led
- 

Date of Submission: 03-02-2024

Date of acceptance: 13-02-2024

---

## I. INTRODUCTION

The traffic lights are used mainly for pedestrians to be protected when they cross the roads. The normal function of traffic system is to control the coordination to ensure that traffic moves as smoothly and safely as possible. It was reducing collisions, both vehicular and pedestrians. It was encourage travel within the speed limit to meet the green lights. The emergency will occur any way, any time and on any location. In that case the speedily response is required. The number of vehicles using the limited road networks infrastructure which was slowly increased. I feel that the major consequence of this increase is the traffic management problem. One of the most critical consequences of traffic problem is the delay of emergency vehicles such as, ambulance during accidents to reach hospitals on time, Fire brigade vehicles, police van to catch the thief, and VIP (minister or president) vehicles. There are traffic jams occur on main way in special seasons and rush hours. That was lead to a long waiting time of peoples and high cost of fuel consumption on the road. And in that delay the Emergency vehicles are stuck in traffic jams. Sometimes even if there is no traffic then also people have to wait because there is a certain time limit of traffic signal. So road users have to wait till the traffic signal turned to green light. Therefore, we have to find new methods which solve this problem. The EMERGENCY services began six years ago in India. With in starting total 14 ambulances are started in Ahmadabad and Gandhinagar,

which was slowly increased. Recently most of deaths are caused due to the traffic congestion. And ambulance also could not go fast as because of traffic jams near to the traffic junction. Solution of this problem is to control the traffic system so that it would be helpful to protect someone's life by giving first priority to the ambulance. In accordance with this, now a day's turning to the manual control it will sometimes solve the problem. But to do this automatic control is a very big task in today's scenario. Particularly in India, Most of people cannot give a way for an ambulance because of traffic. The Indian ambulance experiment was done around 400 cars on the road. There was 0% response to an Indian ambulance. Suppose someone has suffered a heart attack and needs ambulance immediately.

## II. TECHNOLOGY BEHIND THIS PROJECT

### Component Consideration

The information and data for component consideration are based on the reviewed journal and patents. All type of components must be consider first in order to ensure that it is suitable for this project.

### Choosing Component

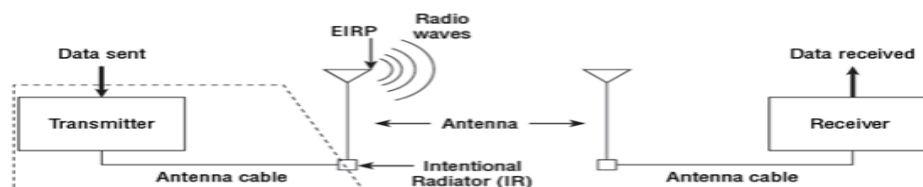
After consideration is done, there is a section where the component should be look and fully understood about their advantages and disadvantages. The component must be chosen properly before it will be proceed to design process. If the component is not compatible due to disadvantages, then it will go back to the consideration component process again.

### Configuration Design

The circuit of this project is designed and constructed roughly using the entire chosen component during this phase. The components are assembled on a breadboard to ensure that the circuit work properly.

### Radio Frequency Transmission

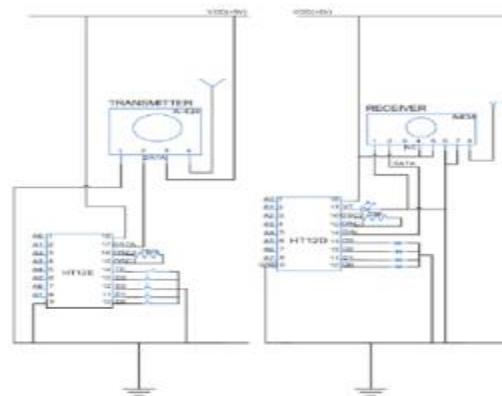
A radio frequency (RF) signal begins as an electrical alternating current (AC) signal that is originally generated by a transmitter. This AC signal is sent via a copper conductor which usually a coaxial cable and radiated out of an antenna element in the form of an electromagnetic wave. Changes of current flow in the antenna produce changes in the electromagnetic fields around the antenna.



**Fig: RF components with circuit**

### RF Based Wireless Remote using TX-RX Modules

This circuit utilizes the RF module, transmitter and receiver (TX-RX) for making a wireless remote, which could be used to drive an output from a distant place. RF module, as the name suggests, uses radio frequency to send signals. These signals are transmitted at a particular frequency and a baud rate. A receiver can receive these signals only if it is configured for that frequency. A four channel encoder/decoder pair has also been used in this system. The input signals, at the transmitter side, are taken through four switches while the outputs are monitored on a set of four light emitting diodes (LED) corresponding to each input switch. The circuit can be used for designing remote appliance control system. The outputs from the receiver can drive corresponding relays connected to any household appliance. This RF transmission system employs Amplitude Shift Keying (ASK) with transmitter/receiver (Tx/Rx) pair operating at 434 MHz. The transmitter module takes serial input and transmits these signals through RF. The transmitted signals are received by the receiver module placed away from the source of transmission. The system allows one way communication between two nodes, namely, transmission and reception. The RF module has been used in conjunction with a set of four channel encoder/decoder ICs. The encoder converts the parallel inputs (from the remote switches) into serial set of signals. These signals are serially transferred through RF to the reception point. The decoder is used after the RF receiver to decode the serial format and retrieve the original signals as outputs. These outputs can be observed on corresponding LEDs.



**Fig: RF Tx-Rx Module**

Transmission through RF is better than infrared (IR) because of many reasons. Firstly, signals through RF can travel through larger distances making it suitable for long range applications. Besides, while IR mostly operates in line-of-sight mode, RF signals can travel even when there is an obstruction between transmitter & receiver. Next, RF transmission is more strong and reliable than IR transmission. RF communication uses a specific frequency unlike IR signals which are affected by other IR emitting sources. The RF module comprises of an RF Transmitter and an RF Receiver. The transmitter/receiver (Tx/Rx) pair operates at a frequency of 434 MHz. An RF transmitter receives serial data and transmits it wirelessly through RF through its antenna connected at pin4. The transmission occurs at the rate of 1Kbps - 10Kbps. The transmitted data is received by an RF receiver operating at the same frequency as that of the transmitter

### EXISTING MODELS

There are so many models like manual controlling, automatic controlling, GPS, image processing traffic control system. Let's discuss about those

#### MANUAL CONTROLLING:

Manual controlling the name instance it require man power to control the traffic. Depending on the countries and state the traffic polices are allotted for a required area or city to control traffic. The traffic policies will carry sign board, sign light and whistle to control the traffic. They will be instructed to wear specific uniforms in order to control the traffic.



#### AUTOMATIC CONTROLLING:

Automatic traffic light is controlled by timers and electrical sensors. In traffic light each phase a constant numerical value loaded in the timer. The lights are automatically getting ON and OFF depending on

the timer value changes. While using electrical sensors it will capture the availability of the vehicles and signals on each phase, depending on the signal the lights automatically switch ON and OFF.



**DRAWBACKS:**

In the manual controlling system we need more man power. As we have poor strength of traffic police we cannot control traffic manually in all area of a city or town. So we need a better solution to control the traffic. On the other side, automatic traffic controlling a traffic light uses timer for every phase. Using electronic sensors is another way in order to detect vehicles, and produce signal that to this method the time is being wasted by green light on an empty road. Traffic congestion also occurred while using the electronic sensors for controlling the traffic. All these drawbacks are supposed to be eliminated by using image processing.



**IMAGE PROCESSING:**

Image processing is a technique to enhance raw images received from cameras/sensors placed on space probes, aircrafts and satellites or pictures taken in normal day-to-day life for various applications. An image is rectangular graphical object. Image processing involves issues related to image representation, compression techniques and various complex operations, which can be carried out on the image data. The operations that come under image processing are image enhancement operations such as sharpening, blurring, brightening, edge enhancement etc. Image processing is any form of signal processing for which the input is an image, such as photographs or frames of video; the output of image processing can be either an image or a set of characteristics or parameters related to the image. Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it. Image processing usually refers to digital image processing, but optical and analog image processing are also possible.



### III. COMPONENTS DESCRIPTION

#### HARDWARE COMPONENTS:

##### MICROCONTROLLER-ATMEGA2560:

Atmega2560, commonly found in the Arduino Mega 2560 as its main microcontroller. It's an AVR RISC-based microcontroller that executes powerful instructions in a single clock cycle. This allows it to strike a fine balance between power consumption and processing speed.

- ATmega2560 features, the good and the bad, comparisons
- Atmega2560 microcontroller boards, Arduino Mega 2560, Seeduino Mega

#### Features & Parametrics:

##### Features:

1. **High Endurance Non-volatile Memory Segments**
  - Write/Erase Cycles: 10,000 Flash
2. **Atmel QTouch library support**
3. **JTAG (IEEE std. 1149.1 compliant) interface**
4. **Peripheral Features**
  - Real-time Counter with Separate Oscillator
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
  - Interrupt and Wake-up on Pin Charge
5. **Other special features**
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby

#### PARAMETRICS:

Name	Value
Program Memory Type	Flash
Program Memory Size (KB)	256
CPU Speed (MIPS/DMIPS)	16
SRAM (KB)	8,192
Data EEPROM/HEF (bytes)	4,096
Digital Communication Peripherals	4-UART, 5-SPI, 1-I2C
Capture/Compare/PWM Peripherals	4 Input Capture, 4 CCP, 16PWM

Timers	2 x 8-bit, 4 x 16-bit
Number of Comparators	1
Temperature Range (°C)	-40 to 85
Operating Voltage Range (V)	1.8 to 5.5
Pin Count	100

**What’s Good and Bad?**

**Advantages:**

- Low power consumption with fast start-up
- Easier to use, with 8-bit microcontroller being less complex than 32/64 bit versions
- QTouch Suite allows for ease of exploring, developing and debugging own touch applications
- Patented Adjacent Key Suppression technology allows for unambiguous detection of key events

**Disadvantages:**

- Limited amount of flash memory write cycles restricts the number of times images can be flashed when programmed to pc
- Naturally lacks incremental performance compared to higher bit microcontrollers

**Microcontroller Comparisons:**

ATmega2560 belongs in an umbrella of microcontrollers; ATmega640/1280/1281/2560/2561. It does share common configurations such as the EEPROM and RAM but still consists of differences as shown below:

Device	Flash	General Purpose I/O pins	16-bit resolution PWM channels	Serial USARTs	ADC Channels
ATmega640	64KB	86	12	4	16
ATmega1280	128KB	86	12	4	16
ATmega1281	128KB	54	6	2	8
ATmega2560	256KB	86	12	4	16
ATmega2561	256KB	54	6	2	8

The closest microcontroller that comes close to the ATmega2560 is the ATmega1280, with the only difference being the additional 128KB of flash.

**This additional amount of flash 2560 brings will be better for projects that require a larger programming space.**

*Note: Arduino Based Mega 1280 is no longer manufactured by Arduino, making 2560 the only option*

**Microcontroller Boards that use the ATmega2560:  
Arduino Mega 2560**



Known for its capabilities in handling more complex projects, the Arduino Mega 2560 gives your projects plenty of room and opportunities. It’s recommended for 3D printers and robotics projects with its 54 digital I/O pins, 16 analog inputs, and a large space.

**Features/Specifications**

- Operating voltage: 5V
- Input voltage (recommended): 7-12V

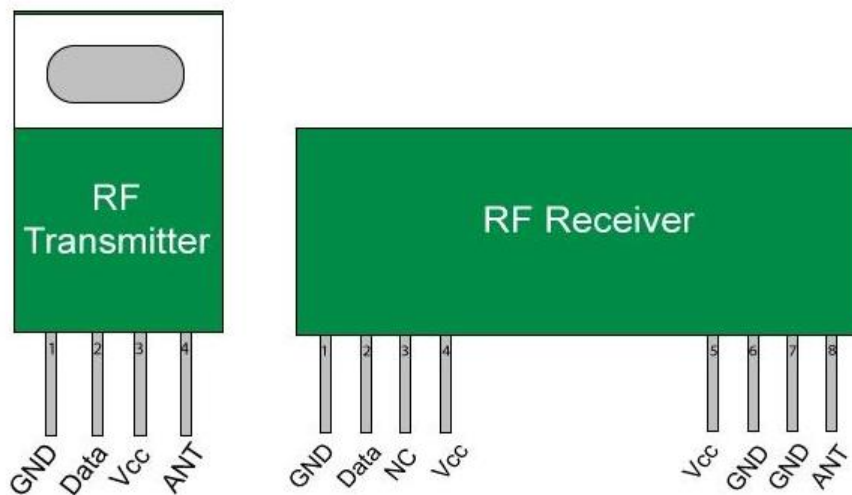
- Input voltage (limits): 6-20V
- Digital I/O pins: 54 (of which 14 provide PWM output)
- Analog input pins: 16
- DC current per I/O pin: 40mA
- DC current for 3.3V pin: 50mA
- Flash Memory: 256 KB, 8KB used by bootloader
- SRAM: 8 KB
- EEPROM: 4 KB
- Clock Speed: 16 MHz

Arduino Mega 2560 is an all-around good option. There's no denying that this board brings performance with the running of an ATmega2560, but it provides a substantial number of I/O pins and program space as well.

If you're currently an Arduino Uno user, it's perhaps time to step up and take a look at the Arduino Mega 2560 for that extra "juice" of performance and I/O pins.

#### **RF MODULE:**

As the name suggests, RF module operates at Radio Frequency. This frequency range varies between 30 kHz & 300 GHz. In this RF system, the digital data is represented as variations in the amplitude of carrier wave. This type of modulation is an Amplitude Shift Keying (ASK).



This **RF module** is a combination of **RF Transmitter** and **RF Receiver**. The transmitter/receiver (Tx/Rx) pair operates at a frequency of **433 MHz**.

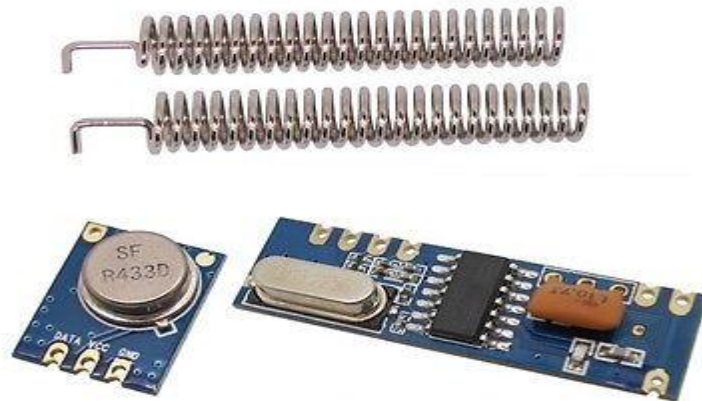
The RF transmitter receives serial data and transmits it wirelessly through its RF antenna. The transmission occurs at the rate of 1 Kbps – 10 Kbps. RF receiver receives the transmitted data and it is operating at the same frequency as that of the transmitter.

#### **Features of RF Module:**

- The Receiver frequency 433MHz
- Receiver typical frequency 105 Dbm
- Receiver supply current 3.5 mA
- Low power consumption
- operating voltage of receiver is 5V
- The transmitter frequency range 433.92MHz
- Supply voltage of transmitter is between 3V to 6V
- Output power of transmitter is between 4Dbm to 12Dbm

433 MHz RF Transmitter and Receiver:

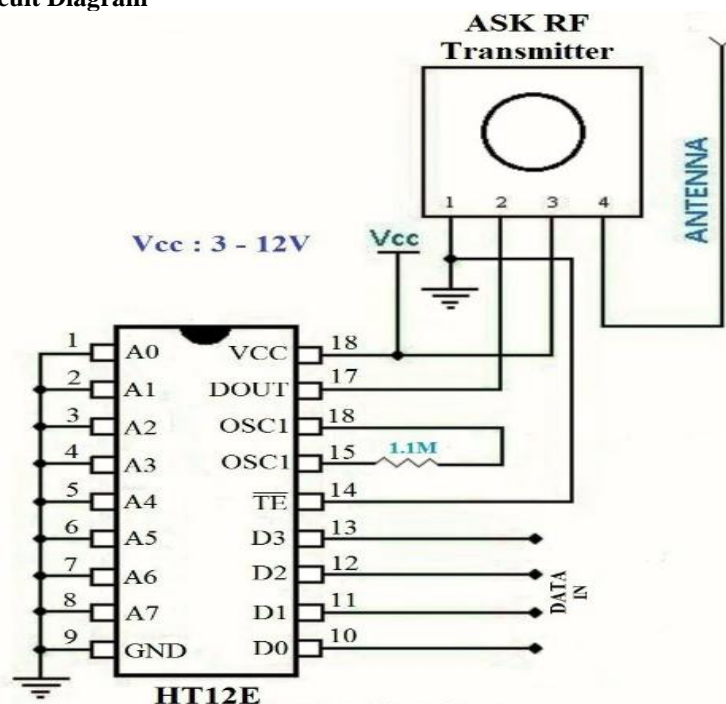
In many projects, we use RF modules to transmitting and receiving the data because it has a high volume of applications than IR. RF transceiver module will always work in a pair that is it needs a Transmitter and Receiver to send and receive the data. A transmitter can only send information and a Receiver and can only receive it, so data can send from one end to another and not the other way around.



The **Transmitter module consists of three pins namely Vcc, Din and ground** as shown above. The Vcc pin has a wide range input voltage from 3V to 12V. The transmitter consumes a minimum current of 9mA and can go as high as 40mA during transmission. The center pin is the data pin to transmit the signal. This signal modulated using the ASK and then sent on air at a frequency of 433MHz.

RF receiver module has four pins namely Vcc, Dout, Linear out and Ground as shown above. The Vcc pin should be powered with a regulated 5V supply. The operating current of this module is less than 5.5mA. The pins Dout and Linear out is shorted together to receive the 433Mhz signal from air. This signal is then demodulated to get the data and sent out through the data pin.

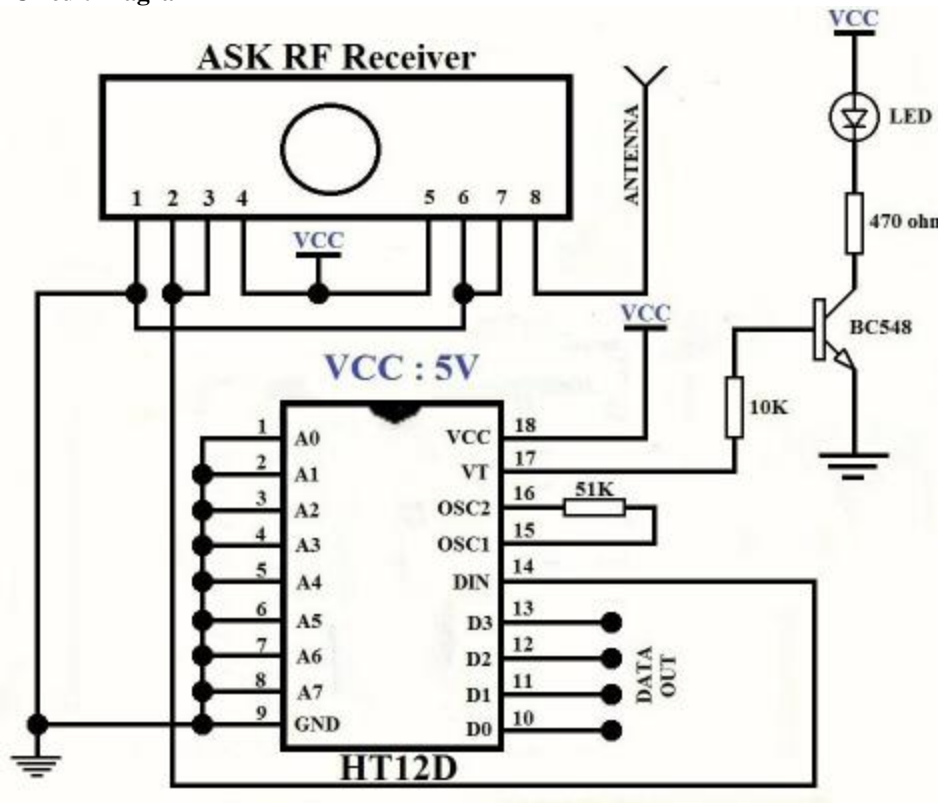
### RF Transmitter Circuit Diagram



HT12E is an encoder IC that converts the 4-bit parallel data from the 4 data pins into serial data in order to transmit over RF link using transmitter



**RF Receiver Circuit Diagram**



HT12D is a decoder IC that converts the serial data received by the RF Receiver into 4-bit parallel data and drives the output accordingly.

**Why Encoders and Decoders are Necessary?**

The RF modules can also function without the need of Encoder and Decoder modules. Simply power on both the modules with the corresponding voltage mentioned above. But, there is a big drawback in this method. You can have only one button on the sender side and one output on the receiver side. So to have more inputs and outputs, the encoder and decoder modules are required.

The HT12D and HT12E are 4-data bit encoder and decoder modules. This means that we can make ( $2^4 = 16$ ) 16 different combinations of inputs and outputs. These are 18 pin IC's which can operate between 3V to 12V input power supply. As said they have 4-data bit and 8-address bit, these 8 address bits has to be set same on both the encoder and decoder to make them work as a pair.

**Applications of RF Module**

- Vehicle monitoring
- Remote control
- Telemetry
- Small-range wireless network
- Wireless meter reading
- Access control systems
- Wireless home security systems
- Area paging
- Industrial data acquisition system
- Radio tags reading
- RF contactless smart cards
- Wireless data terminals
- Wireless fire protection systems
- Biological signal acquisition
- Hydrological and meteorological monitoring
- Robot remote control

- Wireless data transmissions
- Digital video/audio transmission
- Digital home automation, such as remote light/switch
- Industrial remote control, telemetry and remote sensing
- Alarm systems and wireless transmission for various types of low-rate digital signal
- Remote control for various types of household appliances and electronics projects
- Many other applications field related to RF wireless controlling
- Mobile web server for elderly people monitoring

**ENCODER:**

An **encoder** (or "simple encoder") in digital electronics is a one-hot to binary converter. That is, if there are  $2^n$  input lines, and at most only one of them will ever be high, the binary code of this 'hot' line is produced on the  $n$ -bit output lines. A binary encoder is the dual of a binary decoder.

For example, a 4-to-2 simple encoder takes 4 input bits and produces 2 output bits. The illustrated gate level example implements the simple encoder defined by the truth table, but it must be understood that for all the non-explicitly defined input combinations (i.e., inputs containing 0, 2, 3, or 4 high bits) the outputs are treated as don't cares.

**4 to 2 Simple Encoder**

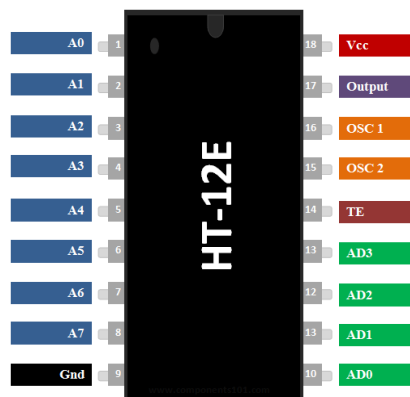
I <sub>3</sub>	I <sub>2</sub>	I <sub>1</sub>	I <sub>0</sub>	O <sub>1</sub>	O <sub>0</sub>	V
0	0	0	0	x	X	0
0	0	0	1	0	0	1
0	0	1	0	0	1	1
0	1	0	0	1	0	1
1	0	0	0	1	1	1

If the input circuit can guarantee at most a single-active input, a simple encoder is a better choice than a priority encoder, since it requires less logic to implement. However, a simple encoder can generate an incorrect output when more than a single input is active, so a priority encoder is required in such cases.

**HT 12E ENCODER:**



**HT12E Pin Configuration**



Pin Number	Pin Name	Description
1 to 8	A0, A1, A2, A3, A4, A5, A6 & A7	These are the 8-bit address bits, which is used to protect your data. We should set the bits in same pattern on Encoder and Decoder IC to pair them.
9	Ground/Vss	Connected to the ground of circuit
10 to 13	AD0, AD1, AD2 & AD3	These four pins are used to send data, the data encoded here will be decoded on HT12D IC sharing the same address bits.
14	Transmission Enable (TE)	This pin has to be connected to Ground (0V) to enable the transmission
15 and 16	Oscillator pins 1 & 2	The IC has a built in oscillator. This oscillator can be used by connecting these two pins through a 1M resistor.
17	Output	The encoded 12 bit output data can be obtained from this pin
16	Vcc/Vdd	This pin powers the IC, typically +5V is used. Can range from 2.4V to 12V

**Features**

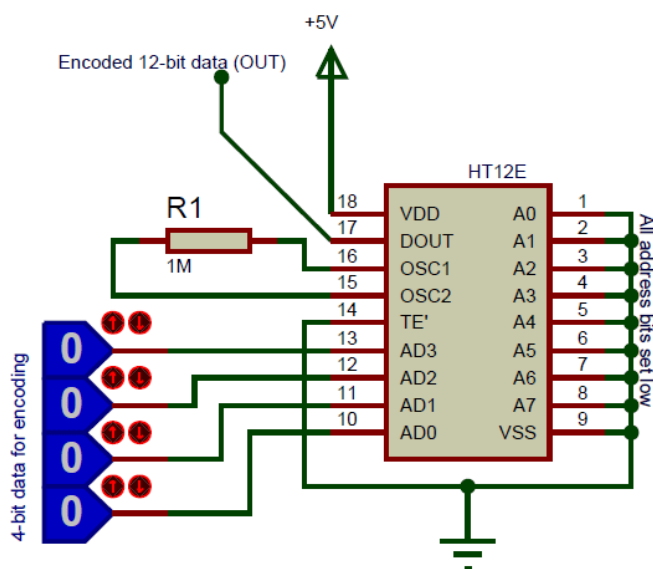
- 12-bit Encoder IC to be used with HT12D
- Encoded data has 4 Data bits and 8 Address bits (8+4=12-bits)
- Commonly used for **RF and IR wireless transmission**
- Wide supply voltage range from 2.4V to 12V, typically 5V is used
- Low stand by current of 0.1uA at Vcc=5V
- Available in 16-pin DIP, 20-pin SOP

**Where to use HT12E Encoder IC**

The IC HT12E can be used only with its pair HT12D. These two ICs together form an **Encoder and Decoder pair**. They are 12-bit Encoders/Decoders, meaning they can transmit 12-bit a data among them. But your encoder IC should not communicate with someone else decoder IC, so an Encoder and Decoder IC pair will share a common Address which is an 8-bit data. So out of the 12-bits 8-bits will be used to set address and the remaining 4-bit will be used to transmit data. With 4-bit data we can create 16 types ( $2^4 = 16$ ) of combinations. These IC's are commonly used with RF pairs or IR pairs. So if you are working on a project which has to transmit a 4-bit data from one end to other either by wire or wireless then this IC pair will be best suited for you.

**How to use a HT12E Encoder IC**

The primary function of HT12E is to encode a 12-bit and send it out through the output pin. Since the IC comes with an in-built Oscillator it is very easy to make this IC work. The IC is has a wide range of operating voltage from 2.4V to 12V, but normally the Vcc pin (pin 18) is powered by +5V and the ground pin (pin 9) is grounded. Pull the Transmission Enable pin (pin 14) to ground to activate transmission. For decoding a data the IC will requires an oscillator, luckily this IC has one in-built. We just have to connect the OSC1 and OSC2 (pin 15 & 16) through a 1M resistor to invoke it. The 4-bit data that has to be sent has to be given to the pins AD0 to AD1 and an address of 8-bit has to be set using the pins A0 to A7. It is very important that your Decoder should also have this same address for them to talk to each other. A basic **connection diagram for the HT12E IC** is shown below



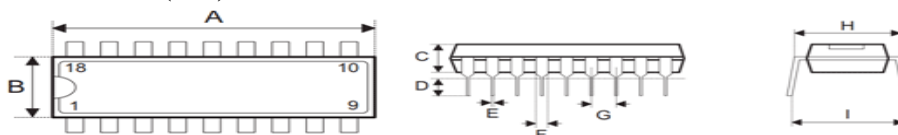
In the above circuit I have set the 8-bit address data as 0b00000000, by connecting all address pins to ground. If you want security you can connect any of the 8 pins to 5V to make it high. The complete IC is powered by a +5V supply which could be obtained from a voltage regulator like 7805. The pins AD3, AD2, AD1 and AD0 are connected to any Digital IC which can provide the 4-bit data. They can also be connected to switches to manually send and receive data. In this image I have made all four data bits as zero (low), when this is decoded we will get the same kind of bits on the output side of HT12D, similarly we can make any changes on these 4-bits and they will be reflected on the output side of the HT12D decoder IC.

The encoded 12-bit can be obtained from the Dout pin (pin 17). This data should be sent to the HT12D for decoding, it can either be sent directly through a wire or by using a wireless medium like RF or IR. You can know to set up the HT12D after this from here.

**Applications**

- Used to convert Parallel 4-bit data to series data
- Highly useful in wireless communication projects involving RF or IR
- Remote controlled systems like garage doors, Car alarm system, Car door controls etc.
- Can be used in Home automation for short range remote switching
- Safety systems like Burglar alarm system, Smoke or Fire alarm system etc..

**2D Model of HT12E (DIP)**

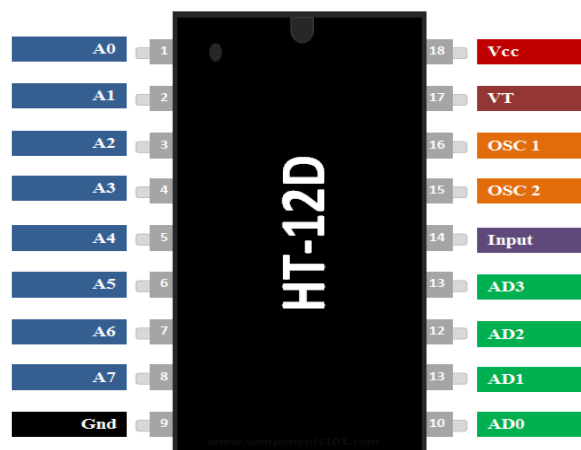


Symbol	Dimensions in mil		
	Min.	Nom.	Max.
A	880	—	920
B	240	—	280
C	115	—	195
D	115	—	150
E	14	—	22
F	45	—	70
G	—	100	—
H	300	—	325
I	—	—	430

**HT 12D DECODER**



**HT12D Pin Configuration**



Pin Number	Pin Name	Description
1 to 8	A0, A1, A2, A3, A4, A5, A6 & A7	These are the 8-bit address bits, which is used to protect your data. We should set the bits in same pattern on Encoder and Decoder IC to pair them.
9	Ground/Vss	Connected to the ground of circuit
10 to 13	AD0, AD1, AD2 & AD3	These four pins are used to obtain the data bits by decoding the data from HT12E IC
14	Input	The encoded 12 bit output data obtained from HT12E has to be given here
15 and 16	Oscillator pins 1 & 2	The IC has a built in oscillator can be used by connecting these two pins through a 1M resistor.
17	Valid Transmission (VT)	This pin will go high when a data is received. It is not mandatory to use it.
16	Vcc/Vdd	This pin powers the IC should use only 5V

**Features**

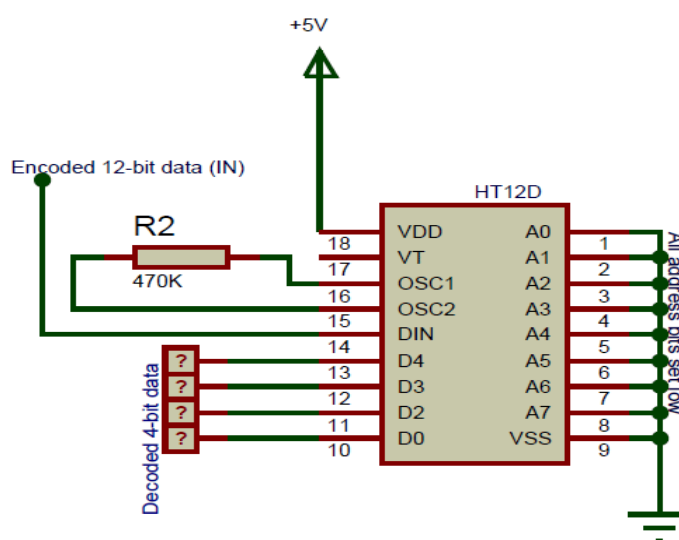
- 12-bit Decoder IC to be used with HT12E
- Decoded data has 4 Data bits and 8 Address bits (8+4=12-bits)
- Commonly used for RF and IR wireless transmission
- Operating Voltage 5V
- Low stand by current of 0.1uA at Vcc=5V
- Available in 16-pin DIP, 20-pin SOP

**Where to use HT12D**

The IC **HT12D** can be used only with its pair HT12E. These two ICs together form an **Encoder and Decoder pair**. They are 12-bit Encoders/Decoders, meaning they can transmit 12-bit a data among them. But your encoder IC should not communicate with someone else's decoder IC, so an Encoder and Decoder IC pair will share a common Address which is an 8-bit data. So out of the 12-bits 8-bits will be used to set address and the remaining 4-bit will be used to transmit data. With 4-bit data we can create 16 types ( $2^4 = 16$ ) of combinations. These IC's are commonly used with RF pairs or IR pairs. So if you are working on a project which has to transmit a 4-bit data from one end to other either by wire or wireless then this IC pair will be best suited for you.

**How to use a HT12D**

The primary function of **HT12D** is to decode the 12-bit that is received by the input pin. Since the IC comes with an in-built Oscillator it is very easy to make this IC work. The IC is should be powered by 5V (pin 18) and the ground pin (pin 9) is grounded. For decoding a data the IC will requires an oscillator, luckily this IC has one in-built. We just have to connect the OSC1 and OSC2 (pin 15 & 16) through a 470K resistor to invoke it. The 4-bit data that is received can be obtained on pins AD0 to AD1 and an address of 8-bit has to be set using the pins A0 to A7. It is very important that your Decoder should have the same address of that of the encoder. A basic **circuit diagram for the HT12D IC** is shown below



In the above circuit I have set the 8-bit address data as 0b00000000, by connecting all address pins to ground. If you want security you can connect any of the 8 pins to 5V to make it high. The complete IC is powered by a +5V supply which could be obtained from a voltage regulator like 7805. The pins AD3, AD2,

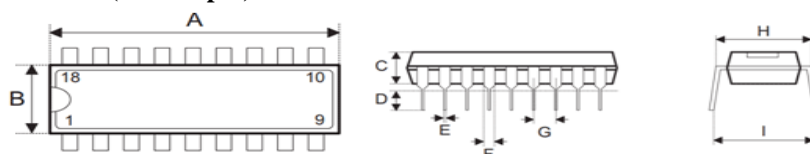
AD1 and AD0 are connected to any Digital IC that can read the 4-bit data. They can also be connected to LED to physically view the received data. In this the 4-bit output data is just shown as “?” since we do not know what the Encoder IC has sent to the Input pin. If any valid data is received they can be obtained from these four pins.

To know how a 4-bit data can be encoded and then send to the input pin of this IC, we have to understand the working of the **HT12E Encoder IC** which is given here.

**Applications**

- Used to convert Parallel 4-bit data to series data
- Highly useful in wireless communication projects involving RF or IR
- Remote controlled systems like garage doors, Car alarm system, Car door controls etc.
- Can be used in Home automation for short range remote switching
- Safety systems like Burglar alarm system, Smoke or Fire alarm system etc.

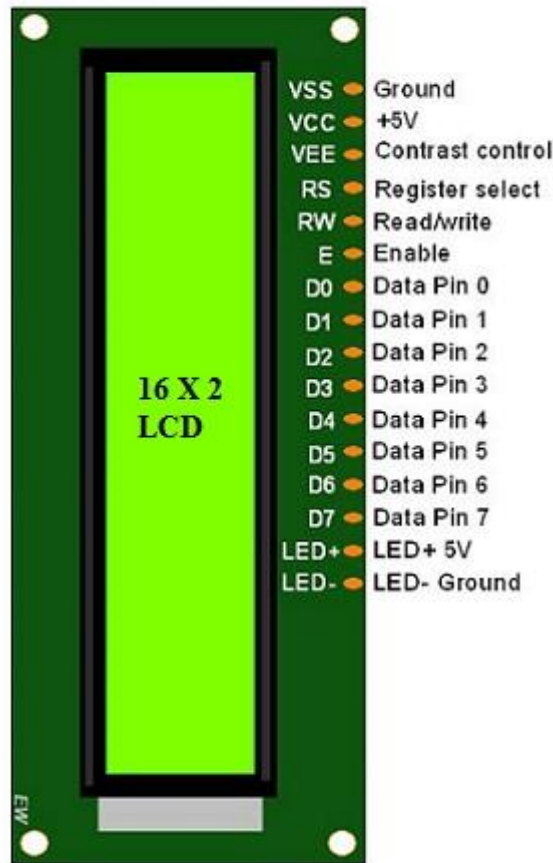
**2D Model of HT12D (DIP 18-pin)**



Symbol	Dimensions in mil		
	Min.	Nom.	Max.
A	880	—	920
B	240	—	280
C	115	—	195
D	115	—	150
E	14	—	22
F	45	—	70
G	—	100	—
H	300	—	325
I	—	—	430

**LCD 16 x 2 DISPLAY**

- Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
- Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.
- Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.
- Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1 (0 = data mode, and 1 = command mode).
- Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).
- Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit & constantly held high.
- Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.
- Pin15 (+ve pin of the LED): This pin is connected to +5V
- Pin 16 (-ve pin of the LED): This pin is connected to GND.



LCD-16x2-pin-diagram

### Features of LCD16x2

The features of this LCD mainly include the following.

- The operating voltage of this LCD is 4.7V-5.3V
- It includes two rows where each row can produce 16-characters.
- The utilization of current is 1mA with no backlight
- Every character can be built with a 5x8 pixel box
- The alphanumeric LCDs alphabets & numbers
- Is display can work on two modes like 4-bit & 8-bit
- These are obtainable in Blue & Green Backlight
- It displays a few custom generated characters

### Registers of LCD

A 16x2 LCD has two registers like data register and command register. The RS (register select) is mainly used to change from one register to another. When the register set is '0', then it is known as command register. Similarly, when the register set is '1', then it is known as data register.

### Command Register

The main function of the command register is to store the instructions of command which are given to the display. So that predefined tasks can be performed such as clearing the display, initializing, set the cursor place, and display control. Here commands processing can occur within the register.

### Data Register

The main function of the data register is to store the information which is to be exhibited on the LCD screen. Here, the ASCII value of the character is the information which is to be exhibited on the screen of LCD.

Whenever we send the information to LCD, it transmits to the data register, and then the process will be starting there. When register set =1, then the data register will be selected.

### **16×2 LCD Commands**

The commands of LCD 16X2 include the following.

- For Hex Code-01, the LCD command will be the clear LCD screen
- For Hex Code-02, the LCD command will be returning home
- For Hex Code-04, the LCD command will be decrement cursor
- For Hex Code-06, the LCD command will be Increment cursor
- For Hex Code-05, the LCD command will be Shift display right
- For Hex Code-07, the LCD command will be Shift display left
- For Hex Code-08, the LCD command will be Display off, cursor off
- For Hex Code-0A, the LCD command will be cursor on and display off
- For Hex Code-0C, the LCD command will be cursor off, display on
- For Hex Code-0E, the LCD command will be cursor blinking, Display on
- For Hex Code-0F, the LCD command will be cursor blinking, Display on
- For Hex Code-10, the LCD command will be Shift cursor position to left
- For Hex Code-14, the LCD command will be Shift cursor position to the right
- For Hex Code-18, the LCD command will be Shift the entire display to the left
- For Hex Code-1C, the LCD command will be Shift the entire display to the right
- For Hex Code-80, the LCD command will be Force cursor to the beginning ( 1st line)
- For Hex Code-C0, the LCD command will be Force cursor to the beginning ( 2nd line)
- For Hex Code-38, the LCD command will be 2 lines and 5×7 matrix

### **LCD 16×2 Arduino**

Thus, this is all about LCD 16×2 datasheet, which includes what is a 16X2 LCD, pin configuration, working principle, and its applications. The main advantages of this LCD device include power consumption is less and low cost. The main disadvantages of this LCD device include it occupies a large area, slow devices and also lifespan of these devices will be reduced due to direct current. So these LCDs use AC supply with less than 500Hz frequency.

### **9V BATTERY:**

The **nine-volt battery**, or **9-volt battery**, is a common size of battery that was introduced for the early transistor radios. It has a rectangular prism shape with rounded edges and a polarized snap connector at the top. This type is commonly used in smoke detectors, gas detectors, clocks, walkie-talkies, electric guitars and effects units.

The nine-volt battery format is commonly available in primary carbon-zinc and alkaline chemistry, in primary lithium iron disulfide, and in rechargeable form in nickel-cadmium, nickel-metal hydride and lithium-ion. Mercury-oxide batteries of this format, once common, have not been manufactured in many years due to their mercury content. Designations for this format include *NEDA 1604* and *IEC 6F22* (for zinc-carbon) or *MN1604 6LR61* (for alkaline). The size, regardless of chemistry, is commonly designated **PP3**—a designation originally reserved solely for carbon-zinc, or in some countries, *E* or *E-block*.

Most nine-volt alkaline batteries are constructed of six individual 1.5 V LR61 cells enclosed in a wrapper. These cells are slightly smaller than LR8D425 AAAA cells and can be used in their place for some devices, even though they are 3.5 mm shorter. Carbon-zinc types are made with six flat cells in a stack, enclosed in a moisture-resistant wrapper to prevent drying. Primary lithium types are made with three cells in series

9-volt batteries accounted for 4% of alkaline primary battery sales in the United States in 2007, and 2% of primary battery sales and 2% of secondary battery (rechargeable) sales in Switzerland in 2008.

### **Connectors**

The battery has both terminals in a snap connector on one end. The smaller circular (male) terminal is positive, and the larger hexagonal or octagonal (female) terminal is the negative contact. The connectors on the battery are the same as on the load device; the smaller one connects to the larger one and vice versa.<sup>[2]</sup> The same snap-style connector is used on other battery types in the Power Pack (PP) series. Battery polarization is normally obvious, since mechanical connection is usually only possible in one configuration.



A problem with this style of connector is that it is very easy to connect two batteries together in a short circuit, which quickly discharges both batteries, generating heat and possibly a fire. Because of this hazard, nine-volt batteries should be kept in the original packaging until they are going to be used.

### **RESISTOR:**

The resistor is a passive electrical component to create resistance in the flow of electric current. In almost all electrical networks and electronic circuits they can be found. The resistance is measured in ohms. An ohm is the resistance that occurs when a current of one ampere passes through a resistor with a one volt drop across its terminals. The current is proportional to the voltage across the terminal ends. This ratio is represented by Ohm's law



Resistors are used for many purposes. A few examples include delimit electric current, voltage division, heat generation, matching and loading circuits, control gain, and fix time constants. They are commercially available with resistance values over a range of more than nine orders of magnitude. They can be used to as electric brakes to dissipate kinetic energy from trains, or be smaller than a square millimeter for electronics

### **RESISTOR DEFINITION AND SYMBOL**

A resistor is a passive electrical compontherent with the primary function to limit the flow of electric current



The international IEC symbol is a rectangular shape. In the USA the ANSI standard is very common, this is a zigzag line.

### **OVERVIEW OF TYPES AND MATERIALS**

Resistors can be divided in construction type as well as resistance material. The following breakdown for the type can be made:

- Fixed resistors
- Variable resistors, such as the:
  - Potentiometer
  - Rheostat
  - Trimpot
- Resistance dependent on a physical quantity:
  - Thermistors (NTC and PTC) as aresult of temperature change
  - Photo resistor (LDR) as a result of a changing light level
  - Varistor (VDR) as a result of a changing voltage
  - Magneto resistor (MDR) as a result of a changing magnetic field
  - Strain gauges as a result of mechanical load

The choice of material technology is a specific to the purpose. Often it is a trade-off between costs, precision and other requirements. For example, carbon composition is a very old technique with a low precision, but is still used for specific applications where high energy pulses occur. Carbon composition resistors have a body of a mixture of fine carbon particles and a non-conductive ceramic. The carbon film technique has a better tolerance. These are made of a non-conductive rod with a thin carbon film layer around it.

This layer is treated with a spiral cut to increase and control the resistance value. Metal and metal oxide film are widely used nowadays, and have better properties for stability and tolerance. Furthermore, they are less influenced by temperature variations. They are just as carbon film resistors constructed with a resistive film around a cylindrical body. Metal oxide film is generally more durable. Wirewound resistors are probably the oldest type and can be used for both high precision as well as high power applications. They are constructed by winding a special metal alloy wire, such as nickel chrome, around a non-conductive core. They are durable, accurate and can have very low resistance value. A disadvantage is that they suffer from parasitic reactance at high frequencies. For the highest requirements on precision and stability, metal foil resistors are used. They are constructed by cementing a special alloy cold rolled film onto a ceramic substrate.

**Resistor Characteristics**

Dependent on the application, the electrical engineer specifies different properties of the resistor. The primary purpose is to limit the flow of electrical current; therefore the key parameter is the resistance value. The manufacturing accuracy of this value is indicated with the resistor tolerance in percentage. Many other parameters that affect the resistance value can be specified, such as long term stability or the temperature coefficient. The temperature coefficient, usually specified in high precision applications, is determined by the resistive material as well as the mechanical design.

In high frequency circuits, such as in radio electronics, the capacitance and inductance can lead to undesired effects. Foil resistors generally have a low parasitic reactance, while wirewound resistors are amongst the worst. For accurate applications such as audio amplifiers, the electric noise must be as low as possible. This is often specified as microvolts noise per volt of applied voltage, for a 1 MHz bandwidth. For high power applications the power rating is important. This specifies the maximum operating power the component can handle without altering the properties or damage. The power rating is usually specified in free air at room temperature. Higher power ratings require a larger size and may even require heat sinks. Many other characteristics can play a role in the design specification. Examples are the maximum voltage, or the pulse stability. In situations where high voltage surges could occur this is an important characteristic.

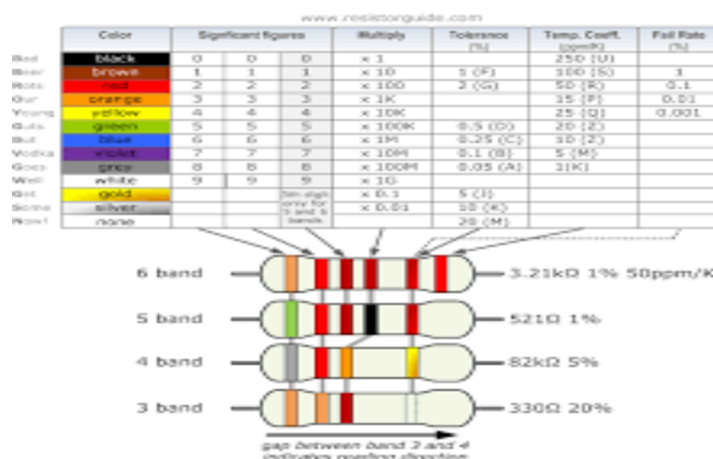
Sometimes not only the electrical properties are important, but the designer also has to consider the mechanical robustness in harsh environments. Military standards sometimes offer guidance to define the mechanical strength or the failure rate.

**Resistor Standards**

Many standards exist for resistors. The standards describe ways to measure and quantify important properties. Other norms exist for the physical size and resistance values. Probably, the most well known standard is the color code marking for axial leaded resistors.

**Resistor Color Code**

The resistance value and tolerance are indicated with several colored bands around the component body. This marking technique of electronic components was already developed in the 1920's. Printing technology was still not far developed, what made printed numerical codes too difficult on small components. Nowadays, the color code is still used for most axial resistors up to one watt. In the figure an example is shown with four color bands. In this example the two first bands determine the significant digits of the resistance value, the third band is the multiplying factor and the fourth band gives the tolerance. Each color represents a different number and can be looked up in a resistor color code chart.



### **Resistor applications**

There is a huge variation in fields of applications for resistors; from precision components in digital electronics, till measurement devices for physical quantities.

### **USB PROGRAMMING CABLE:**

USB bridge cables, or data transfer cables can be found within the market, offering direct PC to PC connections. A bridge cable is a special cable with a chip and active electronics in the middle of the cable. The chip in the middle of the cable acts as a peripheral to both computers, and allows for peer-to-peer communication between the computers. The USB bridge cables are used to transfer files between two computers via their USB ports.

Popularized by Microsoft as Windows Easy Transfer, the Microsoft utility used a special USB bridge cable to transfer personal files and settings from a computer running an earlier version of Windows to a computer running a newer version. In the context of the use of *Windows Easy Transfer* software, the bridge cable can sometimes be referenced as *Easy Transfer cable*.



Many USB bridge / data transfer cables are still USB 2.0, but there are also a number of USB 3.0 transfer cables. Despite USB 3.0 being 10 times faster than USB 2.0, USB 3.0 transfer cables are only 2 - 3 times faster given their design.

The USB 3.0 specification introduced an A-to-A cross-over cable without power for connecting two PCs. These are not meant for data transfer but are aimed at diagnostic uses.

### **Dual role USB connections**

USB bridge cables have become less important with USB dual-role-device capabilities introduced with the USB 3.1 specification. Under the most recent specifications, USB supports most scenarios connecting systems directly with a Type-C cable. For the capability to work, however, connected systems must support role-switching. Dual-role capabilities requires there be *two* controllers within the system, as well as a *role controller*. While this can be expected in a mobile platform such as a tablet or a phone, desktop PCs and laptops often will not support dual roles.

### **EXPRESS PCB**

**PCB** is a free and open-source software suite for electronic design automation (EDA) - for printed circuit boards (PCB) layout. It uses GTK+ for its GUI widgets.

### **FEATURES**

- Scalable fonts
- Layer groups to keep signals together
- Add on device drivers
- Gerber RS-274X and NC Drill output support

- Centroid (X-Y) data output
- PostScript and Encapsulated PostScript output
- Rats-nest generation from simple net lists
- Automatic clearance around pins that pierce a polygon
- Flags for pins and vias
- Groups of action commands can be undone by a single undo
- Simple design rule checker (DRC) - checks for minimum spacing and overlap rules
- Drawing directly on the silk layer
- Viewable solder-mask layers and editing
- Netlist window
- Netlist entry by drawing rats
- Auto router
- Snap to pins and pads
- Element files and libraries that can contain whole sub-layouts, metric grids
- Up to 16 copper layer designs by default
- Trace optimizer
- Connectivity verification
- Can interoperate with free schematic capture tools such as gEDA and Xcircuit
- GNU autoconf/automake based build system
- PCB is Free Software

## **SOFTWARES USED**

### **PROTEUS 8**

The **Proteus Design Suite** is a proprietary software tool suite used primarily for electronic design automation. The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards.

### **History**

The first version of what is now the Proteus Design Suite was called PC-B and was written by the company chairman, John Jameson, for DOS in 1988. Schematic Capture support followed in 1990, with a port to the Windows environment shortly thereafter. Mixed mode SPICE Simulation was first integrated into Proteus in 1996 and microcontroller simulation then arrived in Proteus in 1998. Shape based auto routing was added in 2002 and 2006 saw another major product update with 3D Board Visualisation. More recently, a dedicated IDE for simulation was added in 2011 and MCAD import/export was included in 2015. Support for high speed design was added in 2017. Feature led product releases are typically biannual, while maintenance based service packs are released as it is required.

### **Product modules**

The Proteus Design Suite is a Windows application for schematic capture, simulation, and PCB (Printed Circuit Board) layout design. It can be purchased in many configurations, depending on the size of designs being produced and the requirements for microcontroller simulation. All PCB Design products include an auto router and basic mixed mode SPICE simulation capabilities.

### **Schematic capture**

Schematic capture in the Proteus Design Suite is used for both the simulation of designs and as the design phase of a PCB layout project. It is therefore a core component and is included with all product configurations.

### **Microcontroller Simulation**

The micro-controller simulation in Proteus works by applying either a hex file or a debug file to the microcontroller part on the schematic. It is then co-simulated along with any analog and digital electronics connected to it. This enables its use in a broad spectrum of project prototyping in areas such as motor control, temperature control and user interface design. It also finds use in the general hobbyist community and, since no hardware is required, is convenient to use as a training or teaching tool. Support is available for co-simulation of:

- Microchip Technologies PIC10, PIC12, PIC16, PIC18, PIC24, dsPIC33 Microcontrollers.
- Atmel AVR (and Arduino), 8051 and ARM Cortex-M3 Microcontrollers

- NXP 8051, ARM7, ARM Cortex-M0 and ARM Cortex-M3 Microcontrollers.
- Texas Instruments MSP430, PICCOLO DSP and ARM Cortex-M3 Microcontrollers.
- Parallax Basic Stamp, Freescale HC11, 8086 Microcontrollers.

### **PCB Design**

The PCB Layout module is automatically given connectivity information in the form of a netlist from the schematic capture module. It applies this information, together with the user specified design rules and various design automation tools, to assist with error free board design. PCB's of up to 16 copper layers can be produced with design size limited by product configuration.

### **3D Verification**

The 3D Viewer module allows the board under development to be viewed in 3D together with a semi-transparent height plane that represents the boards enclosure. STEP output can then be used to transfer to mechanical CAD software such as Solidworks or Autodesk for accurate mounting and positioning of the board.

### **EMBEDDED C**

Embedded C is one of the most popular and most commonly used Programming Languages in the development of Embedded Systems. So, in this article, we will see some of the Basics of Embedded C Program and the Programming Structure of Embedded C.

Embedded C is perhaps the most popular languages among Embedded Programmers for programming Embedded Systems. There are many popular programming languages like Assembly, BASIC, C++, Python etc. that are often used for developing Embedded Systems but Embedded C remains popular due to its efficiency, less development time and portability.

Before digging in to the basics of Embedded C Program, we will first take a look at what an Embedded System is and the importance of Programming Language in Embedded Systems.

### **What is an Embedded system?**

An Embedded System can be best described as a system which has both the hardware and software and is designed to do a specific task. A good example for an Embedded System, which many households have, is a Washing Machine.

### **Programming Embedded Systems**

As mentioned earlier, Embedded Systems consists of both Hardware and Software. If we consider a simple Embedded System, the main Hardware Module is the Processor. The Processor is the heart of the Embedded System and it can be anything like a Microprocessor, Microcontroller, DSP, CPLD (Complex Programmable Logic Device) or an FPGA (Field Programmable Gated Array).

All these devices have one thing in common: they are programmable i.e., we can write a program (which is the software part of the Embedded System) to define how the device actually works.

Embedded Software or Program allow Hardware to monitor external events (Inputs / Sensors) and control external devices (Outputs) accordingly. During this process, the program for an Embedded System may have to directly manipulate the internal architecture of the Embedded Hardware (usually the processor) such as Timers, Serial Communications Interface, Interrupt Handling, and I/O Ports etc.

From the above statement, it is clear that the Software part of an Embedded System is equally important as the Hardware part. There is no point in having advanced Hardware Components with poorly written programs (Software).

There are many programming languages that are used for Embedded Systems like Assembly (low-level Programming Language), C, C++, JAVA (high-level programming languages), Visual Basic, JAVA Script (Application level Programming Languages), etc.

In the process of making a better embedded system, the programming of the system plays a vital role and hence, the selection of the Programming Language is very important.

### **Factors for Selecting the Programming Language**

The following are few factors that are to be considered while selecting the Programming Language for the development of Embedded Systems.

- **Size:** The memory that the program occupies is very important as Embedded Processors like Microcontrollers have a very limited amount of ROM (Program Memory).
- **Speed:** The programs must be very fast i.e., they must run as fast as possible. The hardware should not be slowed down due to a slow running software.
- **Portability:** The same program can be compiled for different processors.

- Ease of Implementation
- Ease of Maintenance
- Readability

Earlier Embedded Systems were developed mainly using Assembly Language. Even though Assembly Language is closest to the actual machine code instructions and produces small size hex files, the lack of portability and high amount of resources (time and man power) spent on developing the code, made the Assembly Language difficult to work with.

There are other high-level programming languages that offered the above mentioned features but none were close to C Programming Language. Some of the benefits of using Embedded C as the main Programming Language:

- Significantly easy to write code in C
- Consumes less time when compared to Assembly
- Maintenance of code (modifications and updates) is very simple
- Make use of library functions to reduce the complexity of the main code
- You can easily port the code to other architecture with very little modifications

### **Introduction to Embedded C Programming Language**

Before going in to the details of Embedded C Programming Language and basics of Embedded C Program, we will first talk about the C Programming Language.

The C Programming Language, developed by Dennis Ritchie in the late 60's and early 70's, is the most popular and widely used programming language. The C Programming Language provided low level memory access using an uncomplicated compiler (A software that converts programs to machine code) and achieved efficient mapping to machine instructions.

The C Programming Language became so popular that it is used in a wide range of applications ranging from Embedded Systems to Super Computers.

Embedded C Programming Language, which is widely used in the development of Embedded Systems, is an extension of C Program Language. The Embedded C Programming Language uses the same syntax and semantics of the C Programming Language like main function, declaration of datatypes, defining variables, loops, functions, statements, etc.

The extension in Embedded C from standard C Programming Language include I/O Hardware Addressing, fixed point arithmetic operations, accessing address spaces, etc.

### **Keywords in Embedded C**

A Keyword is a special word with a special meaning to the compiler (a C Compiler for example, is a software that is used to convert program written in C to Machine Code). For example, if we take the Keil's Cx51 Compiler (a popular C Compiler for 8051 based Microcontrollers) the following are some of the keywords:

- bit
- sbit
- sfr
- small
- large

### **Basic Structure of an Embedded C Program (Template for Embedded C Program)**

The next thing to understand in the Basics of Embedded C Program is the basic structure or Template of Embedded C Program. This will help us in understanding how an Embedded C Program is written.

The following part shows the basic structure of an Embedded C Program.

- Multiline Comments . . . . Denoted using /\*.....\*/
- Single Line Comments . . . . Denoted using //
- Preprocessor Directives . . . . #include<...> or #define
- Global Variables . . . . Accessible anywhere in the program
- Function Declarations . . . . Declaring Function
- Main Function . . . . Main Function, execution begins here  
{  
Local Variables . . . . Variables confined to main function  
Function Calls . . . . Calling other Functions  
Infinite Loop . . . . Like while(1) or for(;;)

```

Statements
....
....
}
o Function Definitions . . . . . Defining the Functions
{
Local Variables . . . . . Local Variables confined to this Function
Statements
....
....
}

```

Before seeing an example with respect to 8051 Microcontroller, we will first see the different components in the above structure.

**Different Components of an Embedded C Program**

**Comments:** Comments are readable text that are written to help us (the reader) understand the code easily. They are ignored by the compiler and do not take up any memory in the final code (after compilation).

There are two ways you can write comments: one is the single line comments denoted by // and the other is multiline comments denoted by /\*...\*/.

**Preprocessor Directive:** A Preprocessor Directive in Embedded C is an indication to the compiler that it must look in to this file for symbols that are not defined in the program.

In C Programming Language (also in Embedded C), Preprocessor Directives are usually represented using # symbol like #include... or #define....

In Embedded C Programming, we usually use the preprocessor directive to indicate a header file specific to the microcontroller, which contains all the SFRs and the bits in those SFRs.

In case of 8051, Keil Compiler has the file “reg51.h”, which must be written at the beginning of every Embedded C Program.

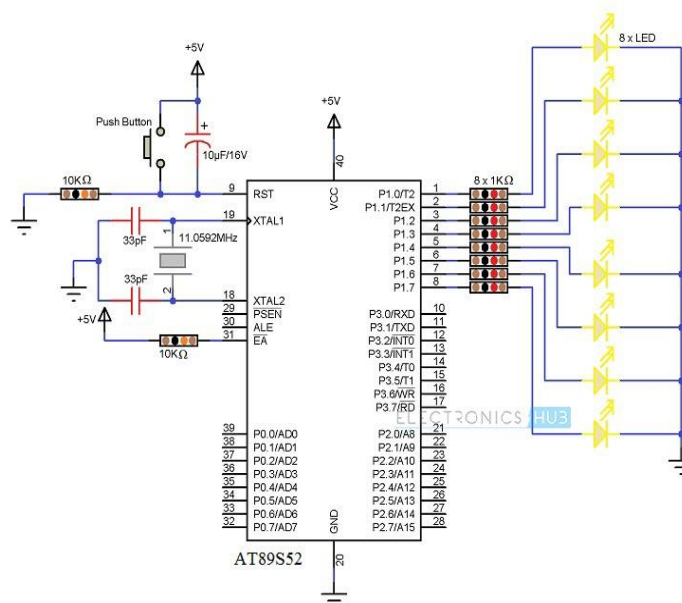
**Global Variables:** Global Variables, as the name suggests, are Global to the program i.e., they can be accessed anywhere in the program.

**Local Variables:** Local Variables, in contrast to Global Variables, are confined to their respective function.

**Main Function:** Every C or Embedded C Program has one main function, from where the execution of the program begins.

**Example of Embedded C Program**

The following image shows the circuit diagram for the example circuit. It contains an 8051 based Microcontroller (AT89S52) along with its basic components (like RESET Circuit, Oscillator Circuit, etc.) and components for blinking LEDs (LEDs and Resistors).



## PROJECT IMPLEMENTATION

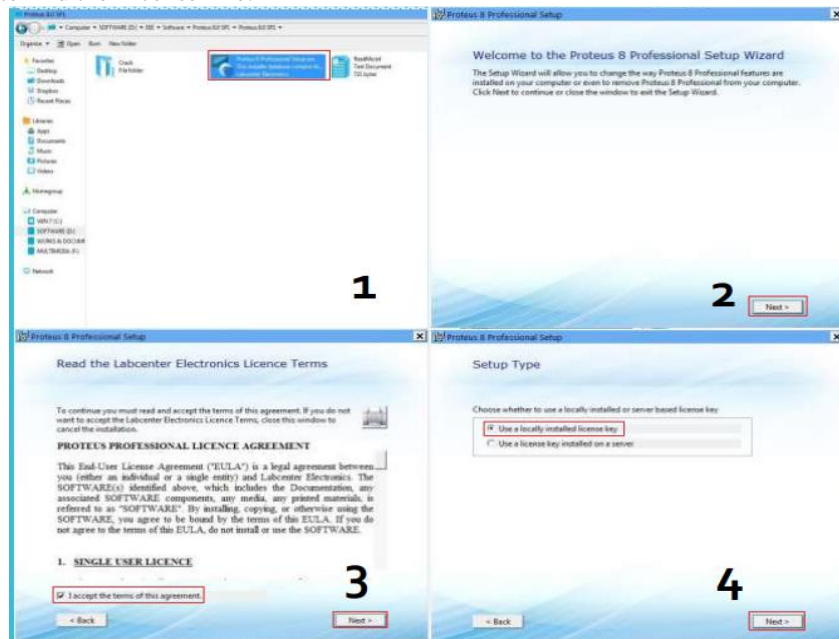
### SOFTWARE:

STEP 1: Download Proteus 8 software using any browser.

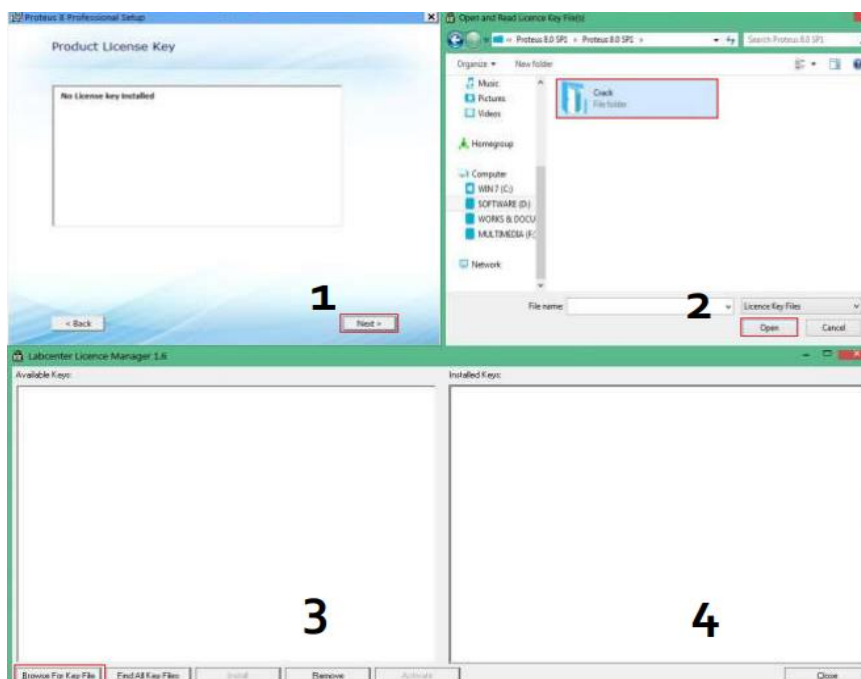
STEP 2: After downloading need to install the software.

Installation proper follows:

- Click twice on the installation exe file.
- Click “Next”.
- On the “License Agreement” page, check “I accept terms of this agreement and click “Next”.
- Select “Use locally installed license” and click next.
- Click Next to find the License File.



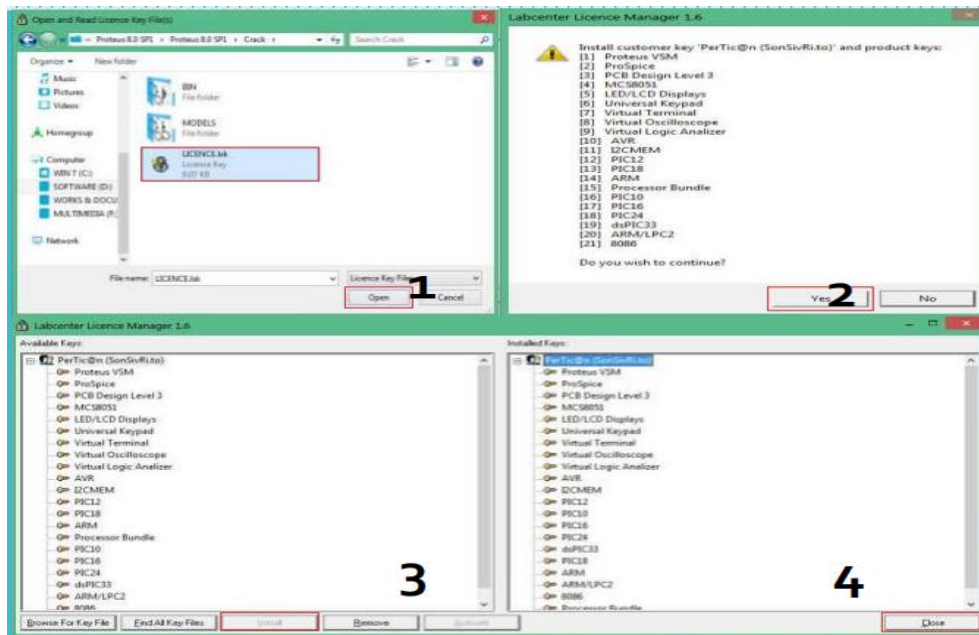
- After opening the “Labcenter License Manager” window, click on “Browse For Key file”.
- Go to the specific folder where you have copied all installation folder before starting installation of Proteus.
- Find the “Crack” named folder there.



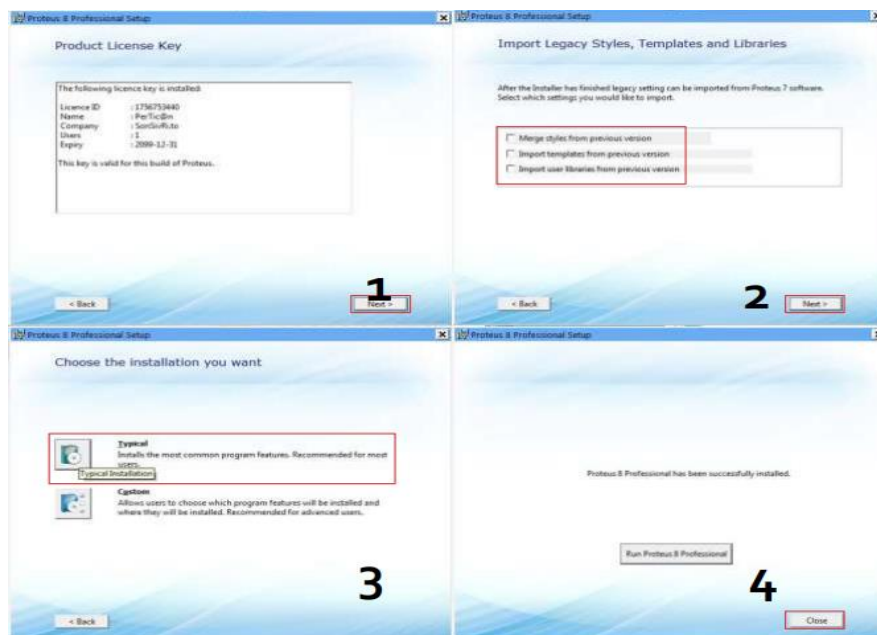
- Select the “license.lxk” file in the “Crack” folder and click to “Open” it.



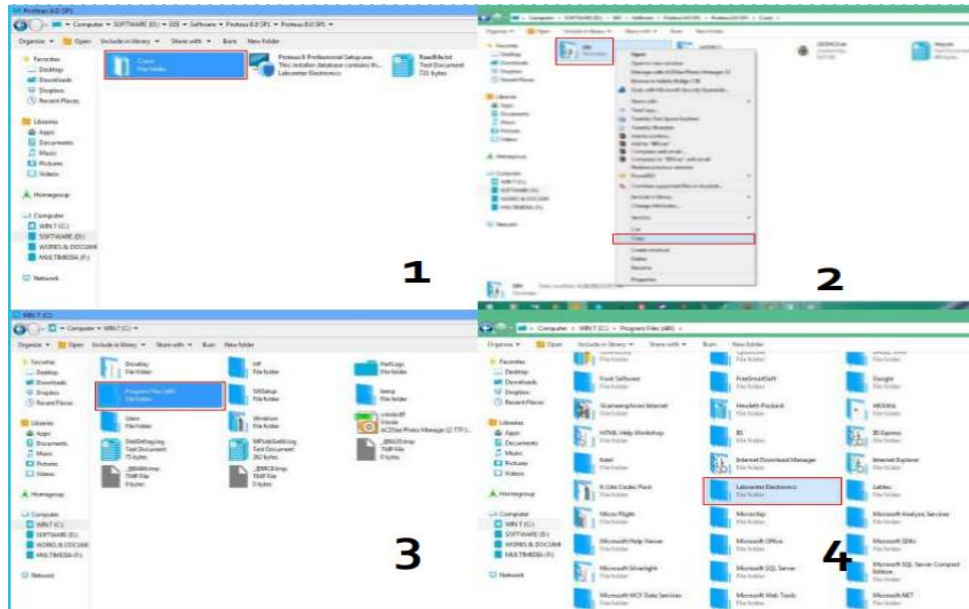
- Click Yes and then Click “Install”.
- Close the License Manager window.



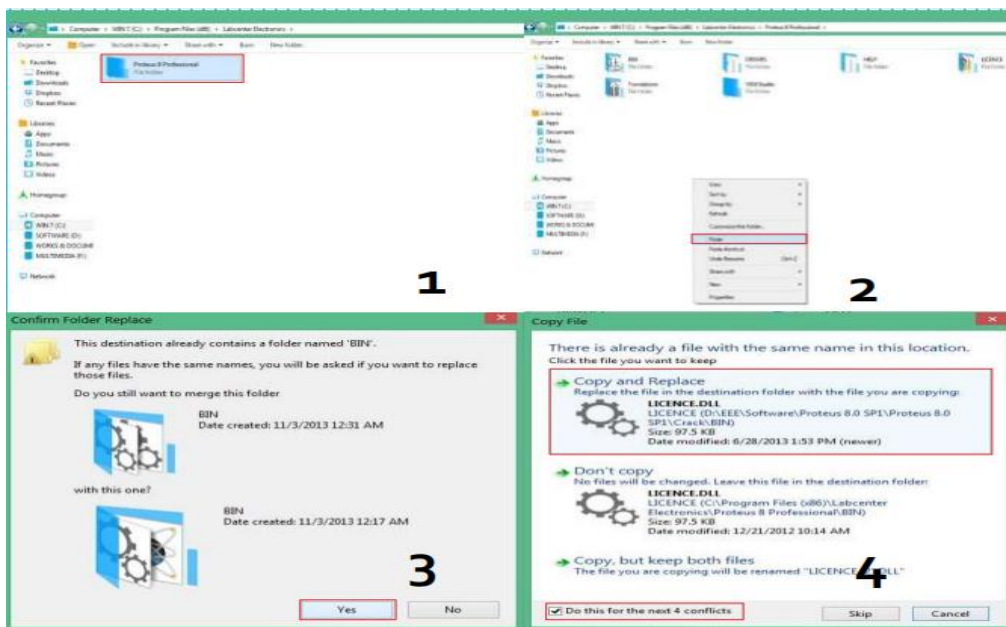
- If the license file is installed perfectly, you will see a window like 1.
- If you have any previous version of Proteus, then check all these 3 boxes and click “Next”.
- If it is the first time of your Proteus using, leave all three box uncheck and Click “next”.
- Click “Typical”.
- Proteus installation will be started now. After 2/3 minutes, the installation will be done. **DO NOT RUN Proteus NOW!** Click “Close”.



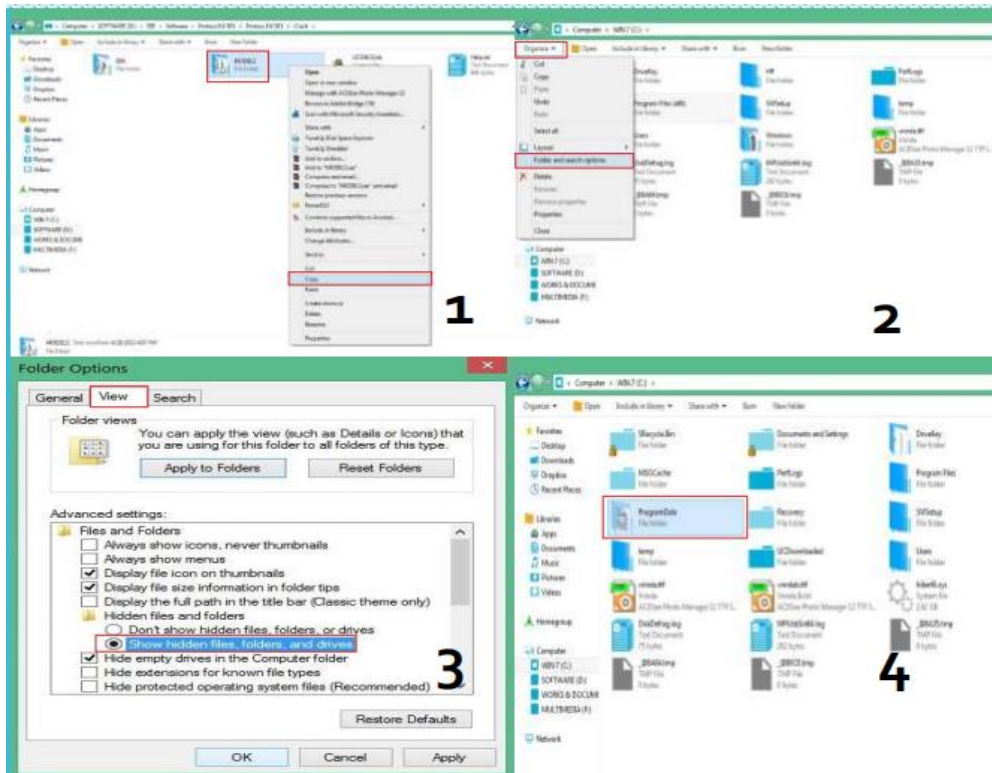
- To crack the installed software, to the “Crack” folder again
- There should be a folder named “BIN”.
- Now, go to “C:\Program Files Labcenter Electronics\Proteus 8 Professional” (32bit) Or, “C:\Program Files (x86)\Labcenter Electronics\Proteus 8 Professional” (64bit) - and pasted that BIN folder there.



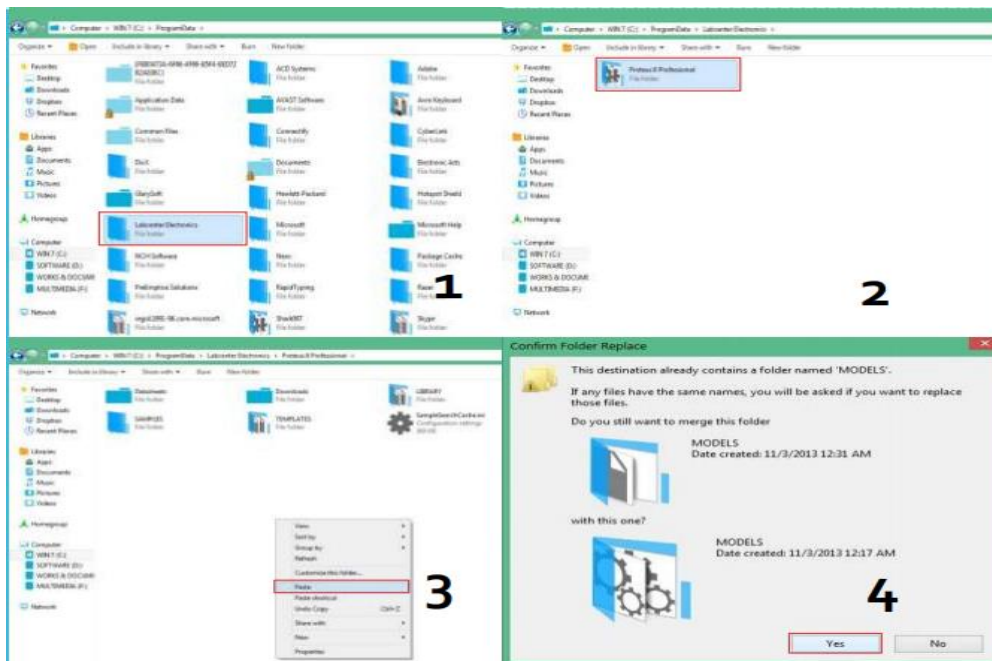
- Click “Yes” while pasting the BIN file at “Confirm Folder Replace” window.
- Check “Do this for the next 4 conflicts” option and click on “Copy and Replace” option.
- If any other permission seeking windows pops up, just click “okay/yes”.



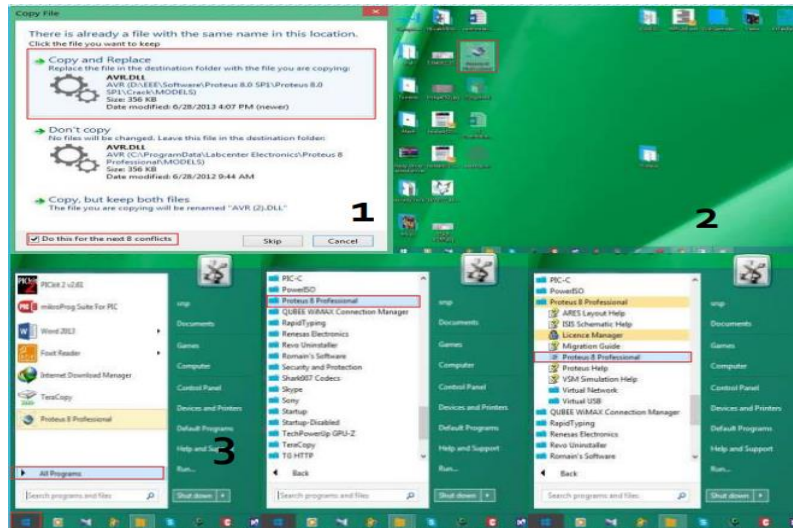
- After being done with copy pasting BIN folder, now come to the previous Crack folder and copy “MODELS” folder.
- The location of pasting this “MODELS” folder is hidden by default. You will have to make it visible to past MODELS.
- Go to “Organize>Folder and search option > Folder Option > View.”
- Select “Show hidden files, folder and drives” and click “OK”, the close the window.
- Now if you go to your “C” drive, you will see “Program Data” folder.



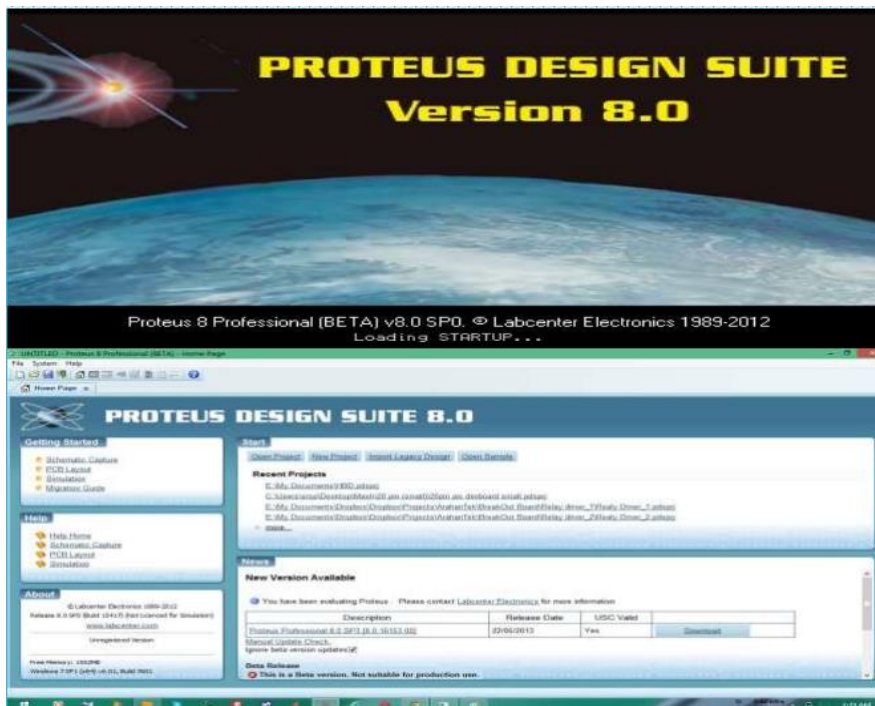
- Now go to this link location – “C:\ProgramData\Labcenter Electronics\Proteus 8 Professional” and paste that MODEL folder on there.
- Click “Yes” if “Confirm Folder Replace” window shows up.



- Check the “Do this for the next 8 conflicts” box and click on “Copy and Replace”.
- Now you can run the software by clicking on the software by clicking on the shortcut of “Proteus 8 Professional” created on your desktop.
- You also can launch is from Start Menu.



- Finally it is installed.



STEP 3: Need to dump the code in workspace.

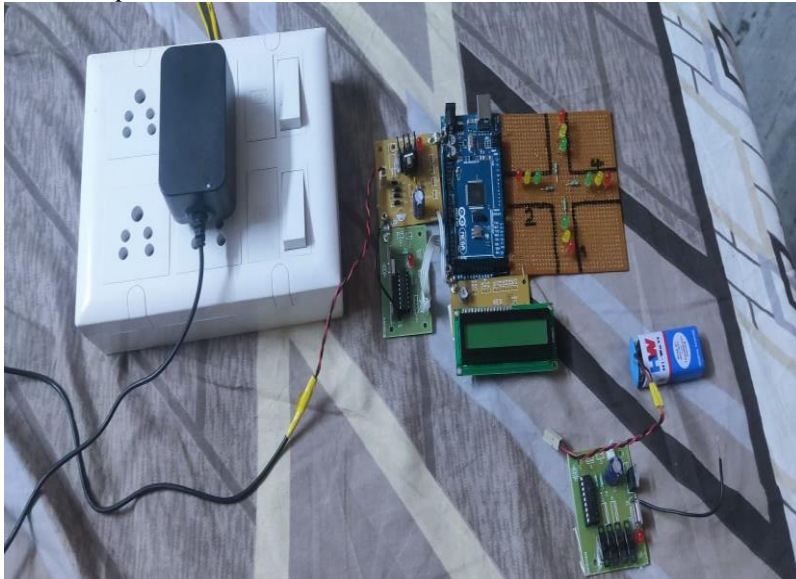
STEP 4: After dumping the code need to run it.

STEP 5: Using SPST (single pole single throw) need to trigger the lane in which ambulance is arriving.

STEP 6: After triggering the lane in which ambulance is arriving yellow light of that lane goes ON for two seconds and then green light goes ON after that.

**HARDWARE PROCEDURE**

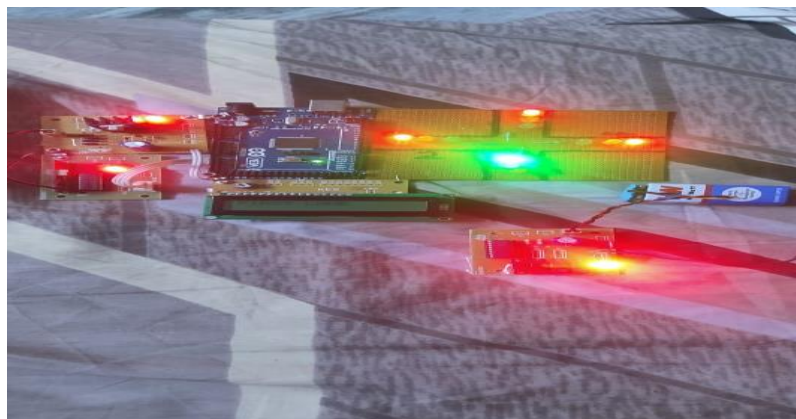
1. Need to connect the components as shown below.



2. After connections are done, need to give power supply.
3. It will be in normal mode when 0000 signal is transmitted. 0000 is nothing but normal traffic operation.



4. Whenever an ambulance is coming through first lane 1000 signal gets transmitted making first lane green lane ON.



5. Whenever an ambulance is coming through second lane 0100 signal gets transmitted making second lane green light ON.



6. Whenever an ambulance is coming through third lane 0010 signal gets transmitted making third lane green light ON.



7. Whenever an ambulance is coming through fourth lane 0001 signal gets transmitted making fourth lane green light ON.



**OPERATION  
NORMAL SEQUENCE**

The sequence of the traffic lights started as green light 1 and red light of other traffic lights are on. The duration for this mode lasted for 30 seconds unless the RF receiver triggers any signal from the transmitter to override the sequence.

Traffic light	Red	Yellow	Green
1	0	0	1
2	1	0	0
3	1	0	0
4	1	0	0

**Table 1 Truth Table for first 30 seconds of the sequence**

Here in first lane for 30 seconds the green light will be on and in other lanes the red light will be on. After 30 seconds the green light gets off in first lane and the yellow light of same lane gets on for 2 seconds.

Traffic light	Red	Yellow	Green
1	0	1	0
2	1	0	0
3	1	0	0
4	1	0	0

**Table 2 Truth Table when the yellow light of traffic light 1 is on**

Next, the sequence turned on the green light of traffic light 2 and the red light of other traffic light for duration of 30 seconds. The same thing happened to the traffic light 3 and traffic light 4 after an interval of the yellow light of each traffic light is on for 2 seconds. The microcontroller will keep repeating this sequence of the traffic light unless the emergency sequence is triggered.

### EMERGENCY MODE SEQUENCE

The emergency mode is triggered when the RF receiver received the transmitted signal from the RF transmitter to override the normal sequence of the traffic light. For example, an ambulance arrives at the traffic light 4 and the green light of the traffic light 1 is on.

When the push-on button no.4 is turned on, the RF receiver received the transmitted signal and changed the sequence to the emergency sequence mode. The emergency sequence mode started when the yellow of traffic light 1 is on for 2 seconds. Then the green of traffic light 4 is on for 10 seconds and then the yellow light of the same traffic light is turned on for 2 seconds.

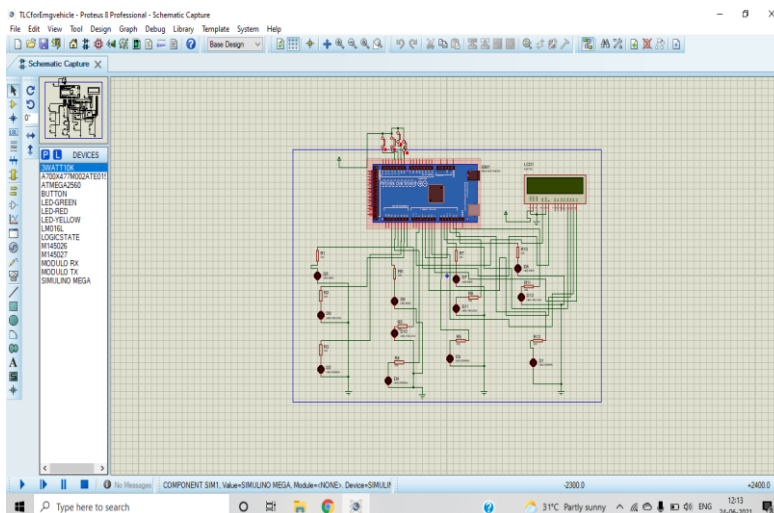
The emergency sequence mode is ended when the sequence of the traffic light is back to the normal sequence which the green light of traffic light 1 is turned back on for the remaining time before the emergency sequence mode is triggered.

### CONNECTION DIAGRAM

#### HARDWARE:



#### SOFTWARE:



**APPLICATIONS**

- This can be used in any Smart traffic system.
- This can be used in any Emergency vehicle.
- This can be used by VIP vehicles also.

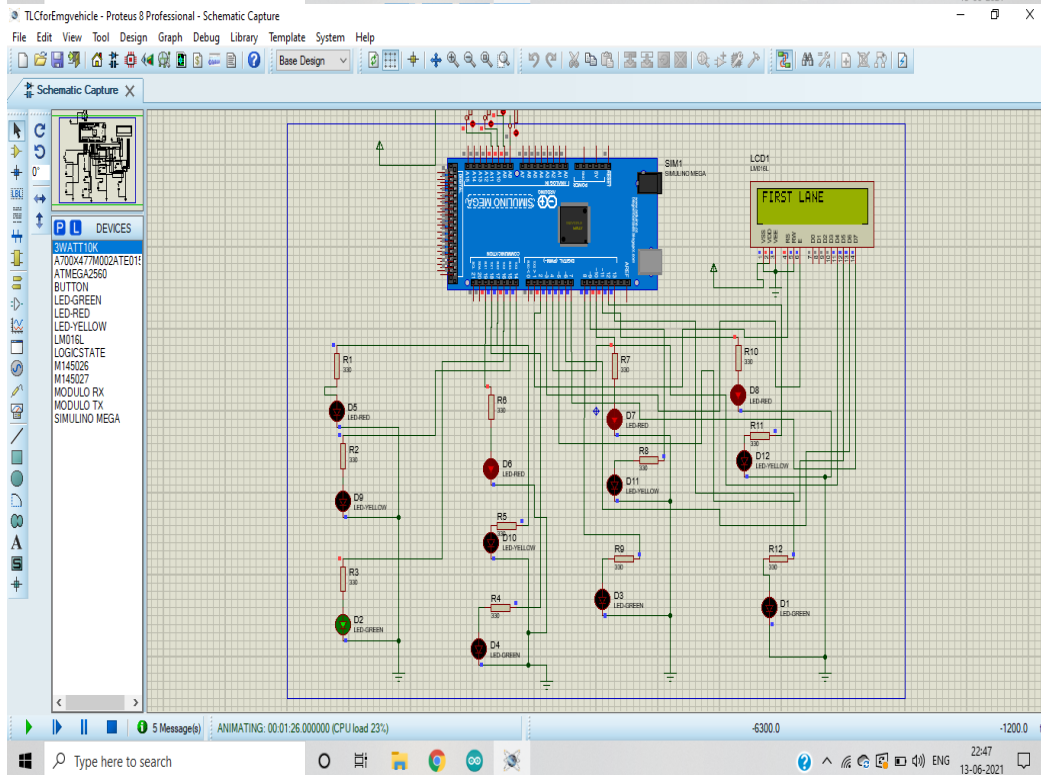
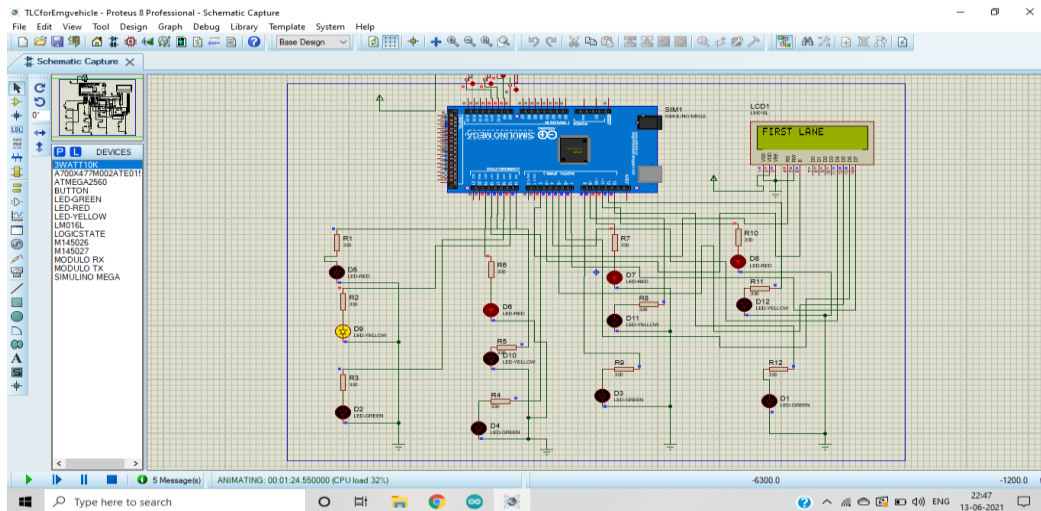
**ADVANTAGES**

- Economical
- Effective
- Range of communication between 3m to 100m
- Easy to operate since its manually done by the driver
- Less hardware requirement
- Can be used by VIP vehicles

**OUTPUT**

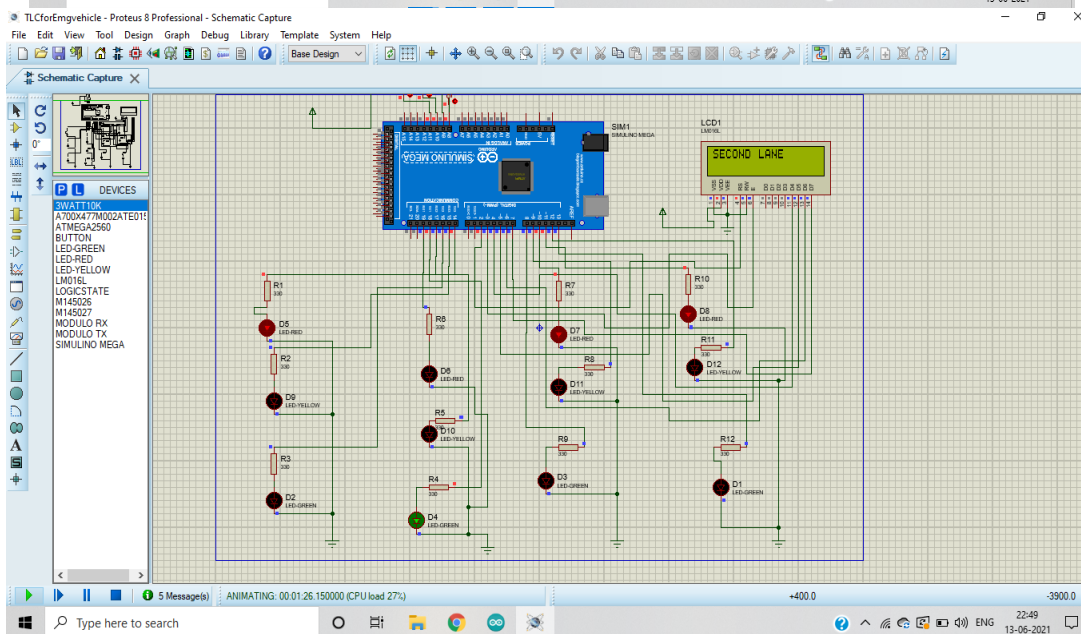
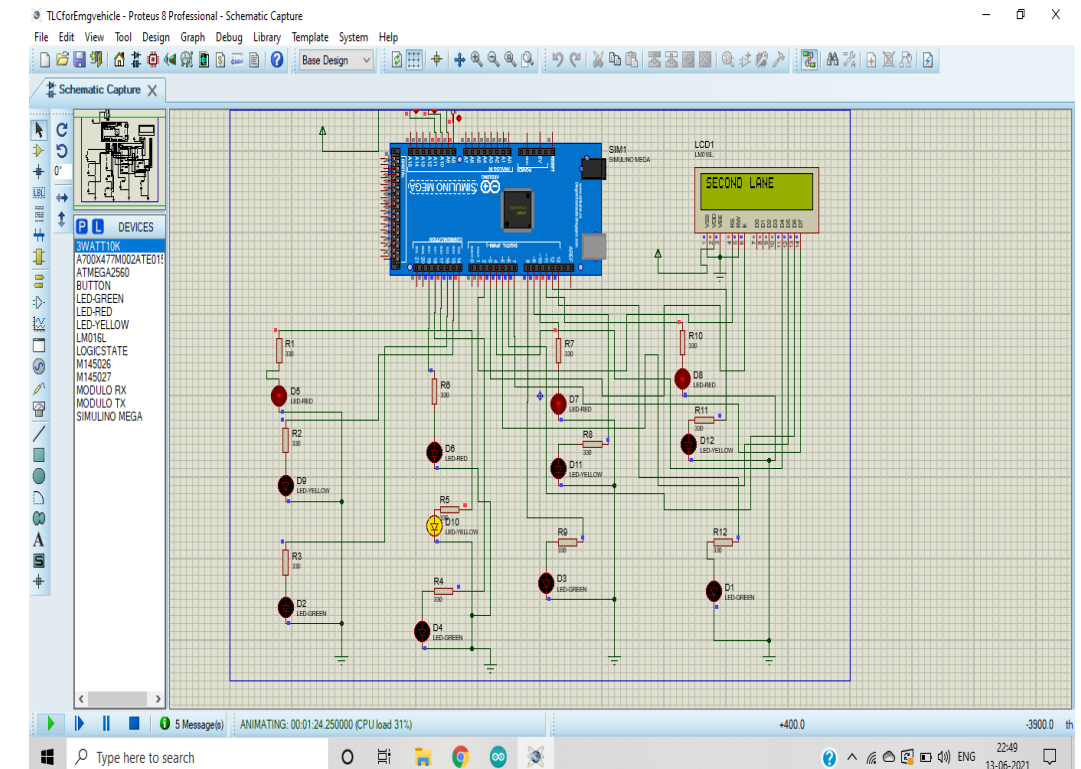
**SOFTWARE:**

**FIRST LANE**

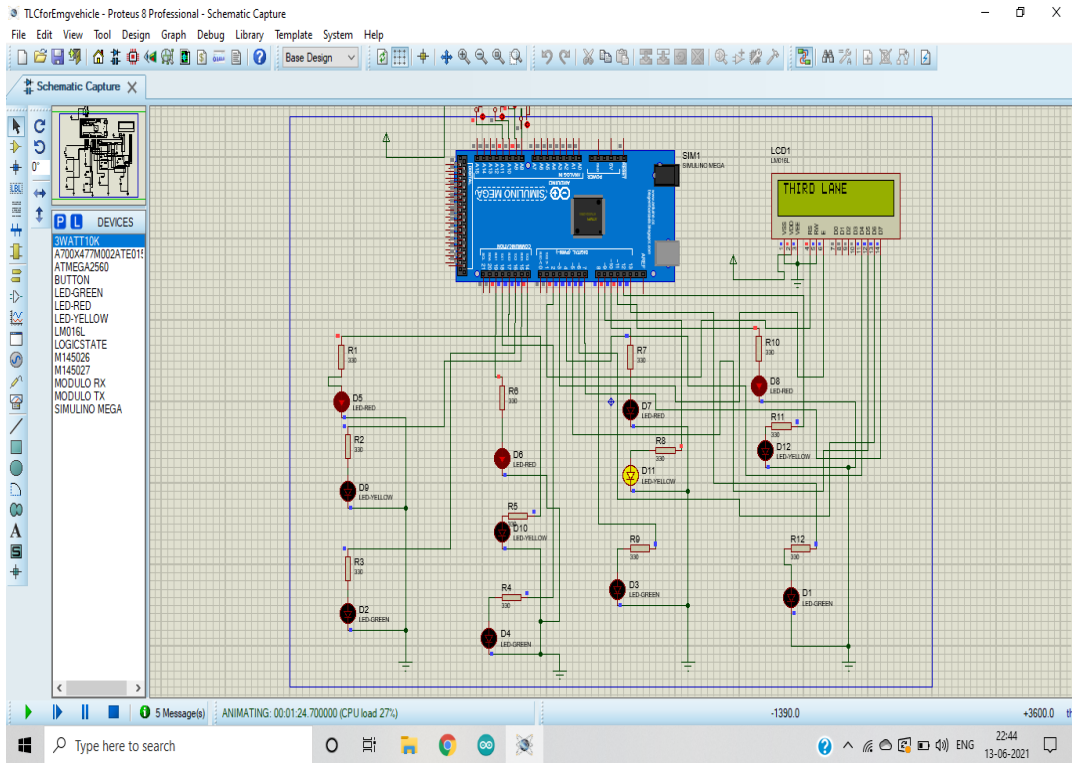




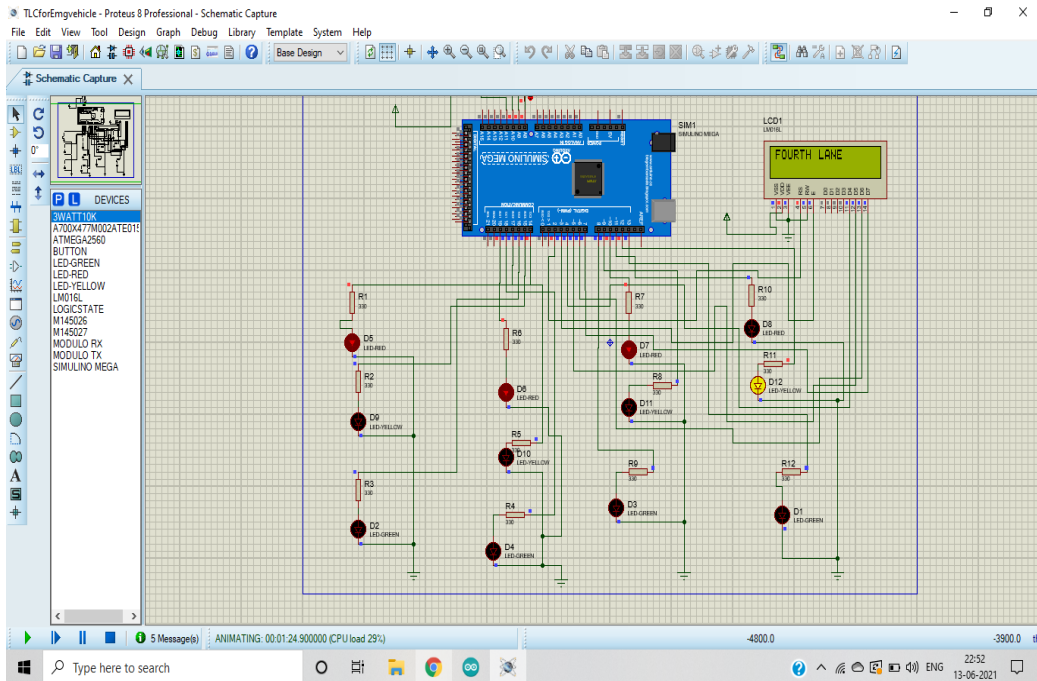
SECOND LANE

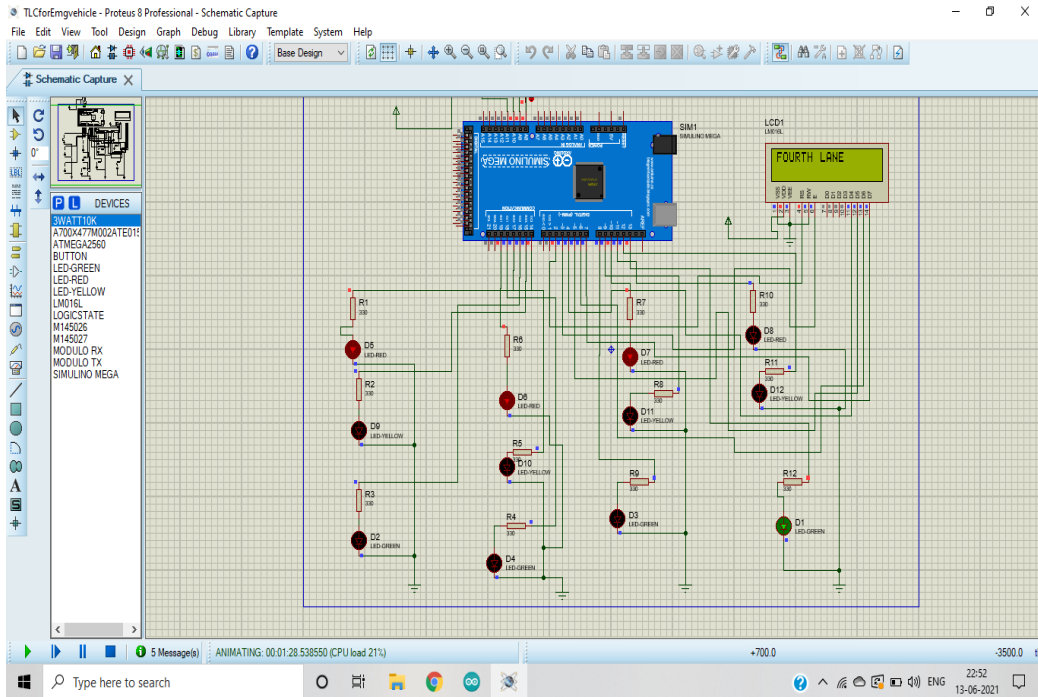


THIRD LANE



FOURTH LANE





**HARDWARE:**  
**FIRST LANE:**



SECOND LANE



THIRD LANE



FOURTH LANE



#### **IV. CONCLUSION**

As a conclusion, this project have achieved the main objective stated earlier which is analyzing and implementing the wireless communication; the radio frequency (RF) transmission in the traffic light control system for emergency vehicles. The prototype of this project is using the frequency of 434 MHz compared to the range of about 3 kHz to 300 GHz of frequency which have been reserved for the RF theoretically. Besides, the functionality of this project proved that the other objectives have been successfully attained which are designing an emergency sequence mode of traffic light when emergency vehicles passing by an intersection and changing the sequence back to the normal sequence before the emergency mode was triggered. The sequences for this project have been developed using the programming in the microcontroller PIC 16F877A. In future, this prototype system can be improved by controlling the real traffic situation and the study can be done by investigating the length, reception and transmission issue for the system to be operated with this traffic light system

#### **FUTURE SCOPE**

- In order to increase the range other wireless communication techniques such as ZIGBEE can be used.
- A model which can be manage multiple emergency vehicles based on their priority.

#### **REFERENCES**

- [1]. Vidya Bhilawade, L.K.Ragha, Intelligent Traffic Control System: International Journal Of Scientific And Research Publications, Volume 8, Issue 2, February 2018, Issn 2250-3153
- [2]. M. Abdoos, N. Mozayani, And A. L. C. Bazzan, "Traffic Light Control In Non-Stationary Environments Based On Multi Agent Q-Learning," In Proc.14th Int. Ieee Conf. Intell. Transp. Syst., Pp. 580–1585, Oct. 2011.
- [3]. Zigbee Specifications, Zigbee Alliance Ieeestandard 802.15.4k [Online].Available: [Http://Www.Zigbee.Org/Specifications.aspx](http://www.zigbee.org/Specifications.aspx), 2013,2014.
- [4]. Traffic Congestion In Bangalore—A Rising Concern. [Online]. Available: [Http://Www.Commonfloor.Com/Guide/Traffic-Congestion-In-Bangalore-Arising-Concern-27238.html](http://www.commonfloor.com/Guide/Traffic-Congestion-In-Bangalore-Arising-Concern-27238.html),Accessed 2013.