"ZIGBEE BASED VEHICLE ACCESS CONTROL & PROTECTING SYSTEM"

A Report on Mini project

Submitted in the partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology

In

"ELECTRICAL & ELECTRONICS ENGINEERING"

By

NIMMALA JEEVAN SAI
(ROLL NO: B19EE129L)

Under the guidance of

M . SPANDANA
Asst.Professor , Dept. of EEE

Department of Electrical & Electronics Engineering
KAKATIYA INSTITUTE OF TECHNOLOGY & SCIENCE, WARANGAL
(An Autonomous Institute under Kakatiya University, Warangal)
(2020-2021)

Department of Electrical & Electronics Engineering

KAKATIYA INSTITUTE OF TECHNOLOGY & SCIENCE, WARANGAL

(An Autonomous Institute under Kakatiya University, Warangal)

CERTIFICATE

This is to certify that, this is the bonafide record of Mini project entitled "ZIGBEE BASED VEHICLE ACCESS CONTROL & PROTECTING SYSTEM " carried out by NIMMALA JEEVAN SAI bearing Roll No. B19EE129L ,student of B.Tech, EEE, VI Semester in partial fulfillment for the Degree of Bachelor of Technology in Department of "ELECTRICAL & ELECTRONICS ENGINEERING".

MINI PROJECT GUIDE

M. SPANDANA

Asst Professor

Dept. of EEE,KITSW

HEAD OF THE DEPARTMENT

Prof. C. VENKATESH

Professor& Head

Dept. of EEE,KITSW

(i)

ACKNOWLEDGEMENT

I wish to express sense of gratitude to my guide to Asst.Prof. M. SPANDHANA ,Asst.Professor, Electrical & Electronics Engineering Department, Kakatiya institute of Technology & Science, Warangal, who guided me at every moment during my entire Mini project and giving valuable suggestions. Her continuous encouragement at each of work and effort to push the work through are grateful acknowledged.

I sincerely thank the Mini project coordinator, Mr. G.SUNIL KUMAR, Asst.Professor, Dept. of EEE for timely conduction of Mini project .

I am indebted to Prof. C. VENKATESH,Professor &Head of the Department, Electrical & Electronics Engineering. I also extend my gratitude to all the faculty members of the department without whose support at various stages this report will not be materialized.

Also, I offer my sincere admiration to Prof. K. ASHOKA REDDY, Principal, Kakatiya institute of Technology & Science, Warangal for his kind patronage and permission to utilize the resource of institute to carry out the work.

Last but not the least I wish to thanks my friends, seniors who helped me directly or indirectly in the successful completion of this work.

NIMMALA JEEVAN SAI **(B19EE129L)**

(ii)

ABSTRACT

.

Current driver assistance systems are based on a number of technologies, such as radar, computer vision and sensors. Integrating all of these technologies into a single system is normally a costly and complex solution. We propose a complete Zigbee based driver assistance system solution that leverages the cost-effective, low power and secure wireless networking features of the Zigbee protocol. By using a Zigbee technology we can transfer data very fast and reliable. This project gives protection to vehicle driver in all aspect like vehicle accessing using RFID tag, smoke detection in engine, sudden obstacle detection, detecting alcohol consumption by driver. The project consists of RFID reader, LCD, Smoke sensor, proximity sensor,alcohol sensor and a motor driver with Zigbee communication.

.

INDEX

(iii)

LIST OF FIGURES

ZIGBEE BASED VEHICLE ACCESS CONTROL SYSTEM

1. INTRODUCTION

1. INTRODUCTION TO EMBEDDED SYSTEMS

1.1 Embedded System

An embedded system is a special-purpose computer system designed to perform one or a few dedicated functions, sometimes with real-time computing constraints. It is usually embedded as part of a complete device including hardware and mechanical parts. In contrast, a general-purpose computer, such as a personal computer, can do many different tasks depending on programming. Embedded systems have become very important today as they control many of the common devices we use.

Since the embedded system is dedicated to specific tasks, design engineers can optimize it, reducing the size and cost of the product, or increasing the reliability and performance. Some embedded systems are mass-produced, benefiting from economies of scale. Embedded means something that is attached to another thing. An embedded system can be thought of as a computer hardware system having software embedded in it. An embedded system can be an independent system or it can be a part of a large system. An embedded system is a microcontroller or microprocessor based system which is designed to perform a specific task. For example, a fire alarm is an embedded system; it will sense only smoke.

And embedded system has three components −

It has hardware.

It has application software.

It has Real Time Operating system (RTOS) that supervises the application software and provide mechanism to let the processor run a process as per scheduling by following a plan to control the latencies. RTOS defines the way the system works. It sets the rules during the execution of application program. A small scale embedded system may not have RTOS.

So we can define an embedded system as a Microcontroller based, software driven, reliable, real-time control system.In general, "embedded system" is not an exactly defined term, as many systems have some element of programmability. For example, Handheld computers share some elements with embedded systems — such as the operating systems and microprocessors which power them — but are not truly embedded systems, because they allow different applications to be loaded and peripherals to be connected.

An embedded system is some combination of computer hardware and software, either fixed in capability or programmable, that is specifically designed for a particular kind of application device. Industrial machines, automobiles, medical equipment, cameras, household appliances, airplanes, vending machines, and toys (as well as the more obvious cellular phone and PDA) are among the myriad possible hosts of an embedded system. Embedded systems that are programmable are provided with a programming interface, and embedded systems programming is a specialized occupation.

Certain operating systems or language platforms are tailored for the embedded market, such as Embedded Java and Windows XP Embedded. However, some low-end consumer products use very inexpensive microprocessors and limited storage, with the application and operating system both part of a single program. The program is written permanently into the system's memory in this case, rather than being loaded into RAM (random access memory), as programs on a personal computer are.

1.2 Applications of Embedded System

We are living in the Embedded World. You are surrounded with many embedded products and your daily life largely depends on the proper functioning of these gadgets. Television, Radio, CD player of your living room, Washing Machine or Microwave Oven in your kitchen, Card readers, Access Controllers, Palm devices of your work space enable you to do many of your tasks very effectively. Apart from all these, many controllers embedded in your car take care of car operations between the bumpers and most of the times you tend to ignore all these controllers.

These days designers have many choices in microprocessors/microcontrollers. Especially, in 8 bit and 32 bit, the available variety really may overwhelm even an experienced designer. Selecting a right microprocessor may turn out as a most difficult first step and it is getting complicated as new devices continue to pop-up very often.

In the 8 bit segment, the most popular and used architecture is Intel's 8031. Market acceptance of this particular family has driven many semiconductor manufacturers to develop something new based on this particular architecture. Even after 25 years of existence, semiconductor manufacturers still come out with some kind of device using this 8031 core.

[1] Military and aerospace software applications

Security is main requirement of military applications . So there is a wide range of use of embedded systems in military applications.

[2] Communications applications

Communication has became the a basic need of todays world. So there is a wide development in this communication by field of embedded systems. Such as real time uses like home automation etc.

[3] Electronics applications and consumer devices

As we know that electronics are ruling the today world as they are flexible, controllable and small in size. These technology is further developed by embedded systems.

[4] Industrial automation and process control software

Every industries need a wide range of speed control applications in any aspect devices. And also wireless communication for deceresing the man power and cost. This requirement is achived by embedded systems and iot devices.

1.3 These are the general characteristics of an embedded system:

- Performs a particular task or has a limited task set.
- Has a very simple UI or is completely devoid of a UI.
- Is a feedback-oriented system.
- In most of the cases, is are part of a larger system.
- Unlike computers and mobile phones, normal users can’t modify the software.

CHAPTER 2

ARDUINO UNO

2. ARDUINO UNO

2.1 ARDUINO

Arduino boards were originally created in 2005 by Massimo Benzi of IVRAE Institute for the need to learn of the computer and electronic students. Arduino is a development board that integrates a microcontroller and its support circuitry with digital and analog inputs and outputs.

It has an open source computing development platform based on an environment for programs creation. The software is written in C or C++ programming language. The Arduino development board is an implementation of wiring, a similar physical computing platform, which is based on the processing multimedia programming environment. This single chip microcontroller has a microprocessor, which comes from a company called Atmel. The chip is known as an AVR. The AVR chip is running at only 16 MHz with an 8-bit core, and has a very limited amount of available memory, with 32 kilobytes of storage and 2 kilobytes of random access memory. Arduino setup build around Atmel microprocessor causes it to be easy and popular to be used in all different kinds of DIY projects.

Arduino IDE is programming environment that allows the user to draft different kind of programs and load them into the Arduino microcontroller. Arduino uses user-friendly programming language, which is based on programming language called Processing. After the user has written his code, IDE compiles and translates the code to the assembler language. After translating the code, the IDE uploads the program to the Arduino microcontroller. Arduino IDE has a built-in code parser that will check the user written code before sending it to the Arduino. IDE software includes the set of different kind of programs that are ready to be tested on the device.

2.2 Different Types Of Arduino Boards

The list of Arduino boards includes the following such as

- Arduino Uno (R3)
- Arduino Nano
- Arduino Micro

- Arduino Due
- LilyPad Arduino Board
- Arduino Bluetooth
- Arduino Diecimila
- RedBoard Arduino Board
- Arduino Mega (R3) Board
- Arduino Leonardo Board
- Arduino Robot
- Arduino Esplora
- Arduino Pro Mic
- Arduino Ethernet
- Arduino Zero
- Fastest Arduino Board

2.3 ARDUINO UNO :

The Uno is a huge option for your initial Arduino. This Arduino board depends on an ATmega328P based microcontroller. As compared with other types of arduino boards, it is very simple to use like the Arduino Mega type board. .It consists of 14-digital I/O pins, where 6-pins can be used as PWM(pulse width modulation outputs), 6-analog inputs, a reset button, a power jack, a USB connection, an In-Circuit Serial Programming header (ICSP), etc. It includes everything required to hold up the microcontroller; simply attach it to a PC with the help of a USB cable and give the supply to get started with an AC-to-DC adapter or battery.

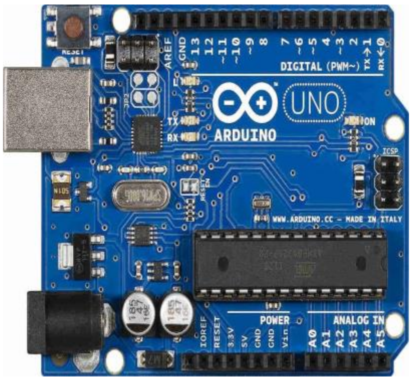


Fig 2.1 ARDUINO UNO BOARD

Circuit diagram :

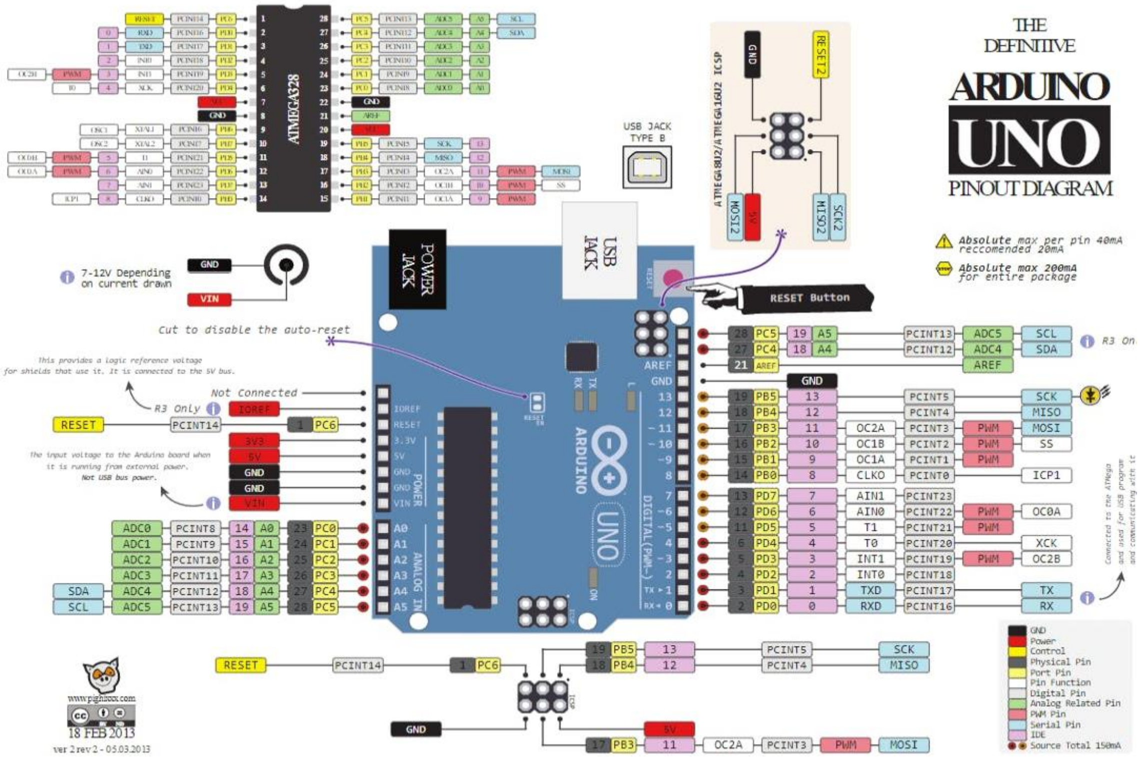


Fig 2.2 .circuit diagram of arduino uno

2.4 Why to choose Arduino?

So, you know that Arduino is easy to use. But why should you spend time learning how to use this microcontroller board? Well, it's incredibly flexible. With an Arduino, you can do everything from control a robot to manage a home automation project---and plenty besides.

The key benefits of learning Arduino are:

You can build awesome projects

Arduino is great for programming

You can learn electronics easily

It's a cheap hobby to start

Arduino might suit you better than a Raspberry Pi

2.5 Arduino Uno Board Description :

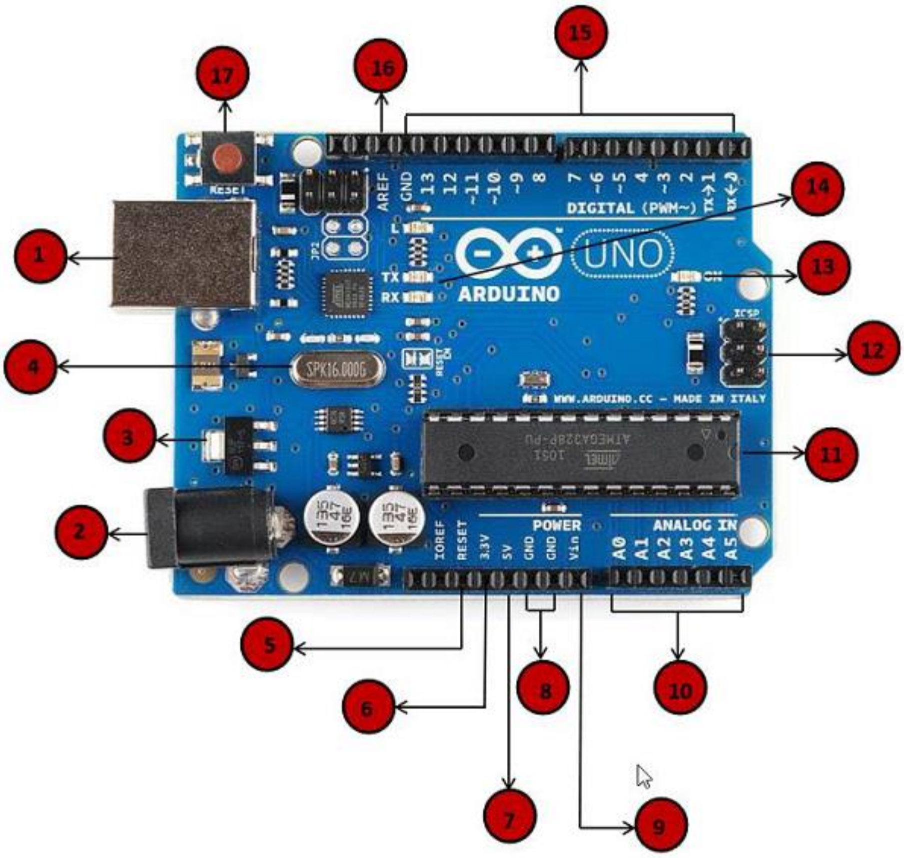


Fig 2.3 arduino uno board discription

1. **Power pin:**

   Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection.

2. **Power (Barrel Jack)**

Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack

3. **Voltage Regulator :** The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.

4. **Crystal Oscillator :**

   The crystal oscillator helps Arduino in dealing with time issues. How does Arduino calculate time? The answer is, by using the crystal oscillator. The number printed on top of the Arduino crystal is 16.000H9H. It tells us that the frequency is 16,000,000 Hertz or 16 MHz.

5. **,** 17 Arduino Reset :

   You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET.

6,7,8,9 **Pins (3.3, 5, GND, Vin)**

- 3.3V (6) − Supply 3.3 output volt
- 5V (7) − Supply 5 output volt
- Most of the components used with Arduino board works fine with 3.3 volt and 5 volt.
- GND (8)(Ground) − There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- Vin (9) − This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.

10 . **Analog pins :**

The Arduino UNO board has six analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be read by the microprocessor.

11. **Main microcontroller :**

Each Arduino board has its own microcontroller (11). You can assume it as the brain of your board. The main IC (integrated circuit) on the Arduino is slightly different from board to board. The microcontrollers are usually of the ATMEL Company. You must know what IC your board has before loading up a new program from the Arduino IDE. This information is available on the top of the IC. For more details about the IC construction and functions, you can refer to the data sheet.

12. **ICSP pin :**

Mostly, ICSP (12) is an AVR, a tiny programming header for the Arduino consisting of MOSI, MISO, SCK, RESET, VCC, and GND. It is often referred to as an SPI (Serial Peripheral Interface), which could be considered as an "expansion" of the output. Actually, you are slaving the output device to the master of the SPI bus.

13 . **Power LED indicator :**

This LED should light up when you plug your Arduino into a power source to indicate that your board is powered up correctly. If this light does not turn on, then there is something wrong with the connection.

14. **TX and RX LEDs :**

On your board, you will find two labels: TX (transmit) and RX (receive). They appear in two places on the Arduino UNO board. First, at the digital pins 0 and 1, to indicate the pins responsible for serial communication. Second, the TX and RX led (13). The TX led flashes with different speed while sending the serial data. The speed of flashing depends on the baud rate used by the board. RX flashes during the receiving process.

15. **Digital I/O :**

The Arduino UNO board has 14 digital I/O pins (15) (of which 6 provide PWM (Pulse Width Modulation) output. These pins can be configured to work as input digital pins to read logic values (0 or 1) or as digital output pins to drive different modules like LEDs, relays, etc. The pins labeled “~” can be used to generate PWM.

16. **AREF :**

AREF stands for Analog Reference. It is sometimes, used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

2.6 Block Diagram arduino uno :

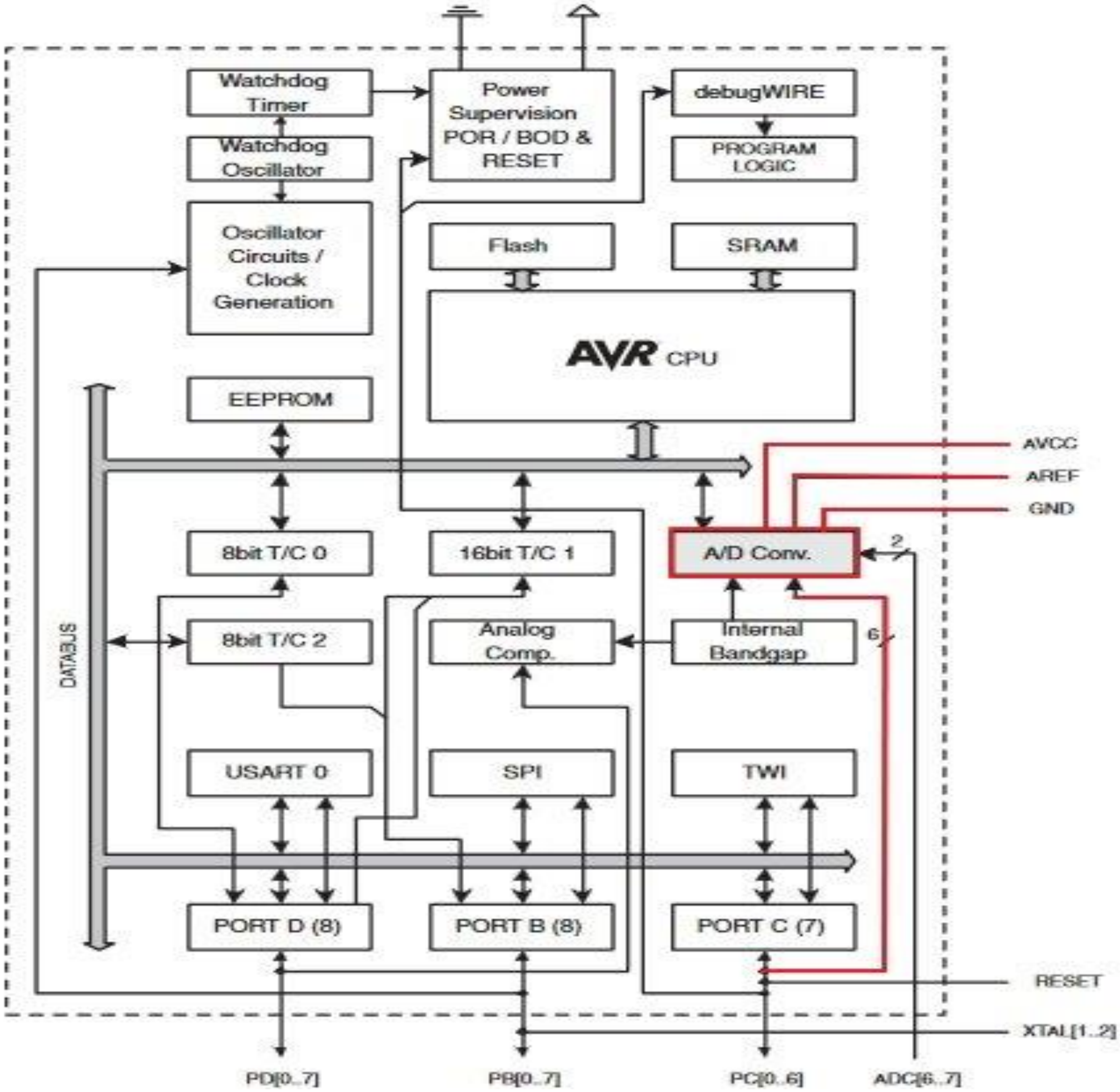
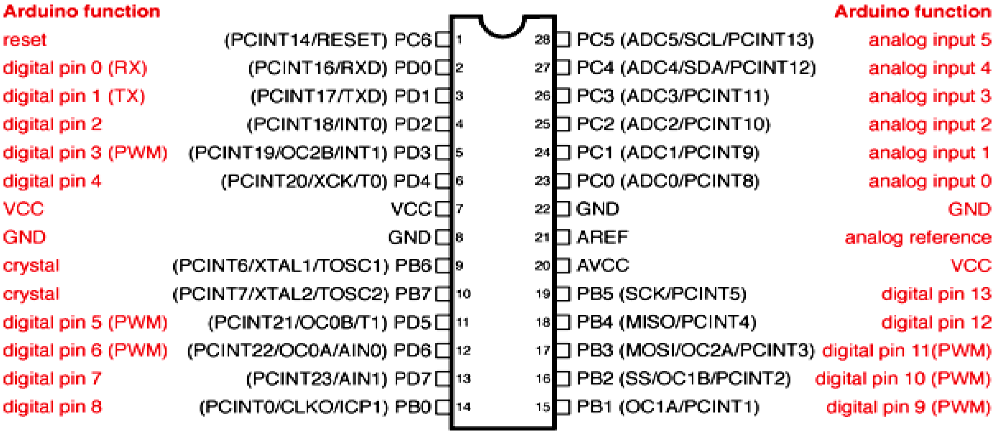


Figure 2.4- Block Diagram of the arduino uno

2.7 Pin configuration


Arduino function
reset (PCINT14/RESET) PC6 1
digital pin 0 (RX) (PCINT16/RXD) PD0 2
digital pin 1 (TX) (PCINT17/TXD) PD1 3
digital pin 2 (PCINT18/INT0) PD2 4
digital pin 3 (PWM) (PCINT19/OC2B/INT1) PD3 5
digital pin 4 (PCINT20/XCK/T0) PD4 6
VCC VCC 7
GND GND 8
crystal (PCINT6/XTAL1/TOSC1) PB6 9
crystal (PCINT7/XTAL2/TOSC2) PB7 10
digital pin 5 (PWM) (PCINT21/OC0B/T1) PD5 11
digital pin 6 (PWM) (PCINT22/OC0A/AIN0) PD6 12
digital pin 7 (PCINT23/AIN1) PD7 13
digital pin 8 (PCINT0/CLKO/ICP1) PB0 14
Arduino function
28 PC5 (ADC5/SCL/PCINT13) analog input 5
27 PC4 (ADC4/SDA/PCINT12) analog input 4
26 PC3 (ADC3/PCINT11) analog input 3
25 PC2 (ADC2/PCINT10) analog input 2
24 PC1 (ADC1/PCINT9) analog input 1
23 PC0 (ADC0/PCINT8) analog input 0
22 GND GND
21 AREF analog reference
20 AVCC VCC
19 PB5 (SCK/PCINT5) digital pin 13
18 PB4 (MISO/PCINT4) digital pin 12
17 PB3 (MOSI/OC2A/PCINT3) digital pin 11(PWM)
16 PB2 (SS/OC1B/PCINT2) digital pin 10 (PWM)
15 PB1 (OC1A/PCINT1) digital pin 9 (PWM)

Digital Pins 11,12 & 13 are used by the ICSP header for MOSI, MISO, SCK connections (Atmega168 pins 17,18 & 19). Avoid low-impedance loads on these pins when using the ICSP header.

Fig 2.5 pin configuration

Pin Description :

- Vin: This is the input voltage pin of the Arduino board used to provide input supply from an external power source.
- 5V: This pin of the Arduino board is used as a regulated power supply voltage and it is used to give supply to the board as well as onboard components.
- 3.3V: This pin of the board is used to provide a supply of 3.3V which is generated from a voltage regulator on the board
- GND: This pin of the board is used to ground the Arduino board.
- Reset: This pin of the board is used to reset the microcontroller. It is used to Resets the microcontroller.
- Analog Pins: The pins A0 to A5 are used as an analog input and it is in the range of 0-5V.
- Digital Pins: The pins 0 to 13 are used as a digital input or output for the Arduino board.
- Serial Pins: These pins are also known as a UART pin. It is used for communication between the Arduino board and a computer or other devices. The transmitter pin number 1 and receiver pin number 0 is used to transmit and receive the data resp.

- External Interrupt Pins: This pin of the Arduino board is used to produce the External interrupt and it is done by pin numbers 2 and 3.
- PWM Pins: This pins of the board is used to convert the digital signal into an analog by varying the width of the Pulse. The pin numbers 3,5,6,9,10 and 11 are used as a PWM pin.
- SPI Pins: This is the Serial Peripheral Interface pin, it is used to maintain SPI communication with the help of the SPI library. SPI pins include:
- SS: Pin number 10 is used as a Slave Select
- MOSI: Pin number 11 is used as a Master Out Slave In
- MISO: Pin number 12 is used as a Master In Slave Out
- SCK: Pin number 13 is used as a Serial Clock
- LED Pin: The board has an inbuilt LED using digital pin-13. The LED glows only when the digital pin becomes high.
- AREF Pin: This is an analog reference pin of the Arduino board. It is used to provide a reference voltage from an external power supply.

2.8 Arduino Uno 2D Model :

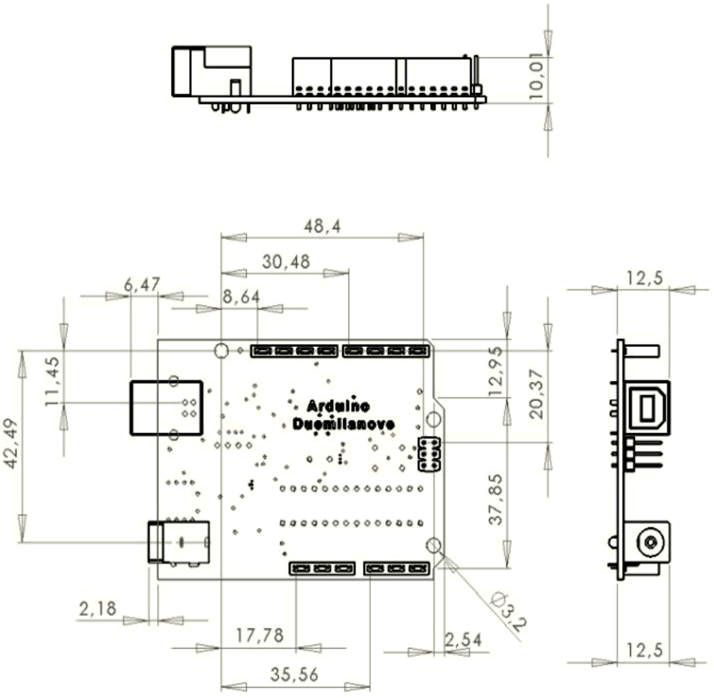


Fig 2.6 2D modal of arduino uno

2.9 Arduino Uno Technical Specifications :

	SPECIFICATION
Microcontroller	ATmega328P – 8 bit AVR family microcontroller
Operating Voltage	5V
Recommended Input Voltage	7-12V
Input Voltage Limits	6-20V
Analog Input Pins	6 (A0 – A5)
Digital I/O Pins	14 (Out of which 6 provide PWM output)
DC Current on I/O Pins	40 mA
DC Current on 3.3V Pin	50 mA
Flash Memory	32 KB (0.5 KB is used for Bootloader)
SRAM	2 KB
Frequency (Clock Speed)	16 MHz

2.10 APPLICATIONS :

- Prototyping of Electronics Products and Systems
- Multiple DIY Projects.
- Easy to use for beginner level DIYers and makers.
- Projects requiring Multiple I/O interfaces and communications

CHAPTER 3

3.1. WORKING AND COMPONENTS

- When the authorized RFID tag brought near to the reader, reader detects the card and switch ON the relay so that engine get started.
- When the vehicle came near to gate, Zigbee module transmit the data to receiver, so that motor gets turned ON and gate gets opened automatically.
- Alcohol sensor was placed near the steering. It reads the amount of alcohol consumed and if it is above the limited value. CPU resist the starting of engine.
- A Smoke sensor was placed in engine and connected to arduino. So when there is abnormal condition on engine, Smoke gets detected and indication was given to driver through buzzer.
- Proximity sensor was placed in front of vehicle and When a obstacle or person came across suddenly it gives indication and alert the driver.

3.2. Components

1. Arduino UNO
2. RFID Reader and Tag
3. Switching relay
4. Zigbee module
5. Smoke sensor
6. Proximity sensor
7. Alcohol sensor
8. DC Motor
9. 16x2 LCD

3.3 PARTS USED :

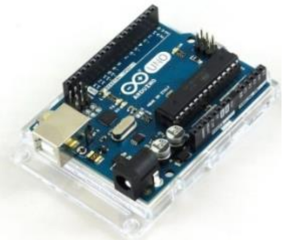


Fig 3.1 arduino uno

fig 3.2 RFID cards

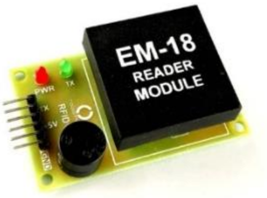


fig b3.3 RFID reader

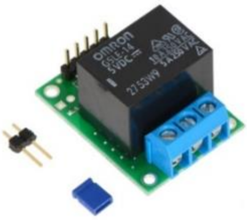


Fig 3.4 switching relay

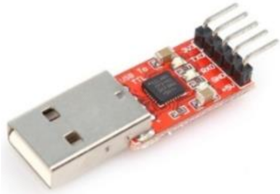


fig 3.5 zigbee module

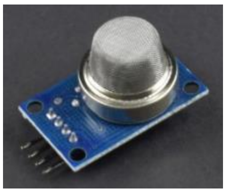


fig 3.6 smoke sensor

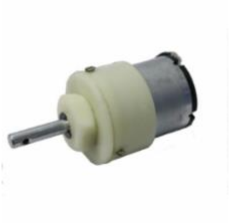


Fig 3.7 dc motor

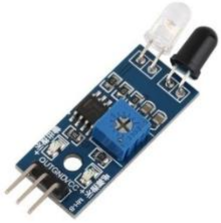


fig 3.8 proximity sensor

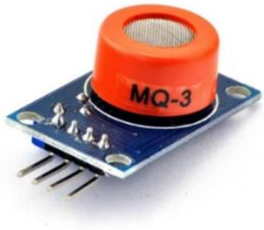


fig 3.9 alchohol sensor

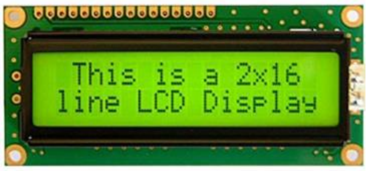


Fig 3.10 LCD display

CHAPTER 4

HARDWARE IMPLEMENTATION

4.1 Block diagram:

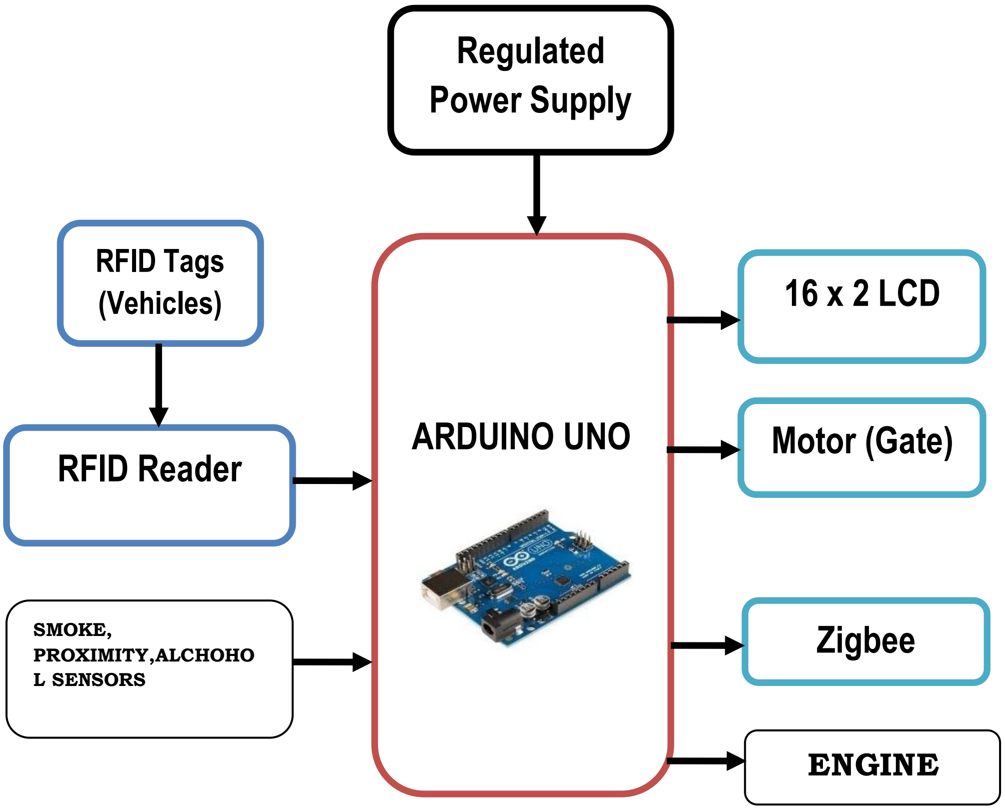
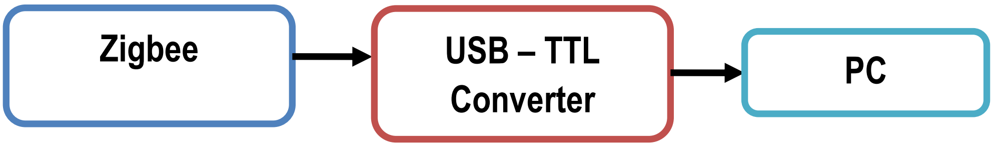


fig 4.1. block diagram



4.2 Regulated Power Supply:

A variable regulated power supply, also called a variable bench power supply, is one where you can continuously adjust the output voltage to your requirements. Varying the output of the power supply is the recommended way to test a project after having double checked parts placement against circuit drawings and the parts placement guide.

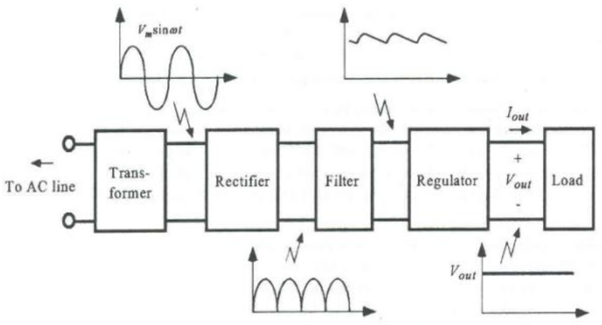
This type of regulation is ideal for having a simple variable bench power supply. Actually this is quite important because one of the first projects a hobbyist should undertake is the construction of a variable regulated power supply. While a dedicated supply is quite handy ,it's much handier to have a variable supply on hand, especially for testing.

Mainly the microcontroller needs 5 volt power supply. To use these parts we need to build a regulated 5 volt source. Usually you start with an unregulated power To make a 5 volt power supply, we use a 7805 voltage regulator IC (Integrated Circuit).

Circuit Features:-

V_{out} range	1.25V - 37V
V_{in} - V_{out} difference	3V - 40V
Operation ambient temperature	0 - 125°C
Output I_{max}	<1.5A
Minimum Load Current$_{max}$	10Ma

Block Diagram



Components of a typical linear power supply

Fig.4.2. regulated power supply

4.3 LCD Interfacing

Fig. 4.3. LCD display

This section describes the operation modes of LCDs, then describes how to program and interface an LCD to art .8051 using Assembly and C.

LCD operation

In recent years the LCD is finding widespread use replacing LEDs (seven-segment LEDs or other multisegment LEDs). This is due to the following reasons:

1. The declining prices of LCDs.
2. The ability of display numbers, characters, and graphics. This is ain contrast to LEDs, which are limited to numbers and a few characters.
3. Incorporation of a refreshing controller into the LCD, thereby relieving the CPU of the task of refreshing the LCD. In contrast, the LED must be refreshed by the CPU (or in some other way) to keep displaying the data.
4. Ease of programming for characters and graphics.

Interfacing an LCD to the 8951 Microcontroller

LCD pin descriptions:

The LCD discussed in this section has 14 pins. The function of each pin is given in table.

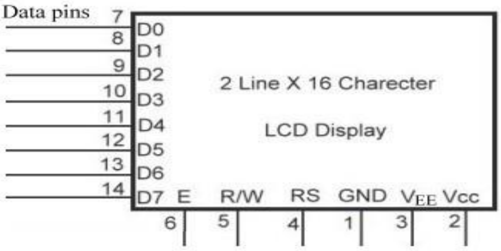
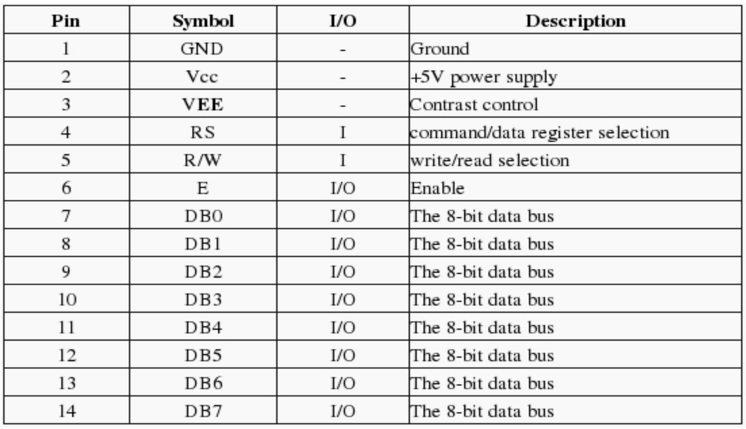


Fig. 4.4 LCD pin description



Pin	Symbol	I/O	Description
1	GND	-	Ground
2	Vcc	-	+5V power supply
3	VEE	-	Contrast control
4	RS	I	command/data register selection
5	R/W	I	write/read selection
6	E	I/O	Enable
7	DB0	I/O	The 8-bit data bus
8	DB1	I/O	The 8-bit data bus
9	DB2	I/O	The 8-bit data bus
10	DB3	I/O	The 8-bit data bus
11	DB4	I/O	The 8-bit data bus
12	DB5	I/O	The 8-bit data bus
13	DB6	I/O	The 8-bit data bus
14	DB7	I/O	The 8-bit data bus

Vcc, Vss, and VEE:

While Vcc and Vss provide +5V and ground, respectively, VEE is used for controlling LCD contrast.

RS – register select:

There are two very important registers inside the LCD. The RS pin is used for their selection as follows. If RS = 0, the instruction command code register is selected, allowing the user to send a command such as clear display, cursor at home, etc. If RS = 1 the data register is selected, allowing the user to send data to be displayed on the LCD.

R/W – read/write:

R/W input allows the user to write information to the LCD or read information from it. R/W = 1 when reading; R/W =0 when writing.

E – enable:

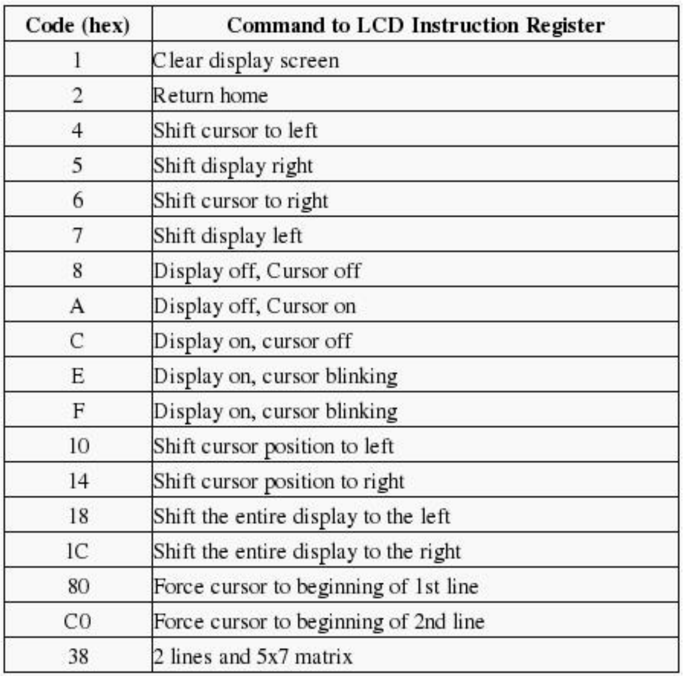
The enable pin is used by the LCD to latch information presented to its data pins. When data is supplied to data pins, a high to low pulse must be applied to this pin in order for the LCD to latch in the data present at the data pins. This pulse must be a minimum of 450 ns wide.

D0 – D7:

The 8 bit data pins, D0 – D7, are used to send information to the LCD or read the contents of the LCD’s internal registers.

To display letters and numbers, we send ASCII codes for the letters A – Z, a – z, and numbers 0 – 9 to these pins while making RS = 1.

There are also instructions command codes that can be sent to the LCD to clear the display or force the cursor to the home position or blink the cursor. Table below lists the instruction command codes.



Code (hex)	Command to LCD Instruction Register
1	Clear display screen
2	Return home
4	Shift cursor to left
5	Shift display right
6	Shift cursor to right
7	Shift display left
8	Display off, Cursor off
A	Display off, Cursor on
C	Display on, cursor off
E	Display on, cursor blinking
F	Display on, cursor blinking
10	Shift cursor position to left
14	Shift cursor position to right
18	Shift the entire display to the left
1C	Shift the entire display to the right
80	Force cursor to beginning of 1st line
C0	Force cursor to beginning of 2nd line
38	2 lines and 5x7 matrix

LCD Commands table

We also use RS = 0 to check the busy flag bit to see if the LCD is ready to receive information. The busy flag is D7 and can be read when R/W =1 and RS = 0, as follows: if R/W =1, RS =0. When D7 = 1(busy flag = 1), the LCD busy taking care of internal operations and will not accept any new information. When D7 = 0, the LCD is ready to receive new information. Note: It is recommended to check the busy flag before writing any data to the LCD.

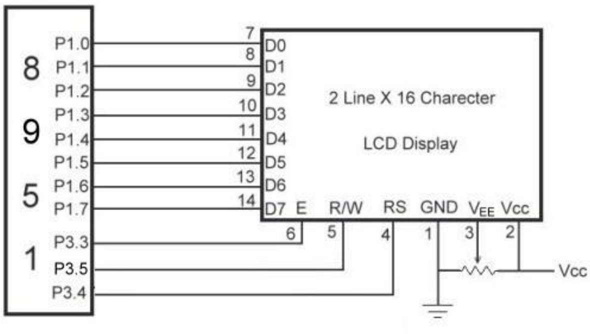


Fig 4.5. Interfacing LCD to 89S52

4.4 RFID MODULE:

Radio frequency identification is a powerful emerging technology that enables companies to achieve total business visibility. By knowing the identity, location and conditions of assets, tools, inventory, people and more, companies can optimize business processes and reduce operational costs. Radio frequency identification (RFID) is a generic term that is used to describe a system that transmits the identity (in the form of a unique serial number) of an object or person wirelessly, using radio waves.

RFID reader module, are also called as interrogators. They convert radio waves returned form the RFID tag into a form that can be pressed on to controllers, which can make use of it. RFID tags and readers have to be tuned to the same frequency in order to communicate. RFID systems use many different frequencies, but the most common and widely used and supported by our Reader 125 KHz.

An RFID reader typically contains a module (transmitter and receiver), a control unit and a coupling element (antenna). The reader has three main functions: energizing, demodulating and decoding. In addition, readers can be fitted with an additional interface that converts the radio waves returned from the RFID tag into a form that can then be passed on to another system, like a computer or any programmable logic controller. Anti-Collision algorithms permit the simultaneous reading of large numbers of tagged objects, while ensuring that each tag is read only once.

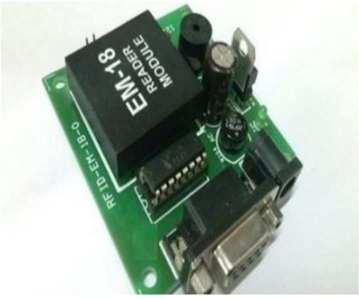


Fig 4.6. RFID Reader

Fig 4.7 RFID Tag

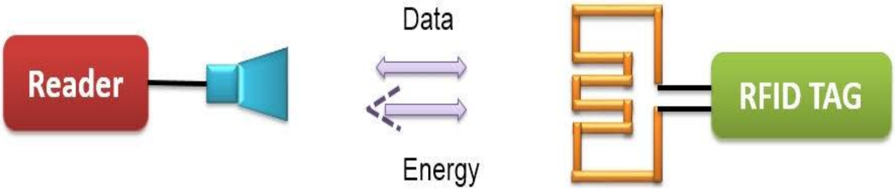


Fig 4.8 RFID working

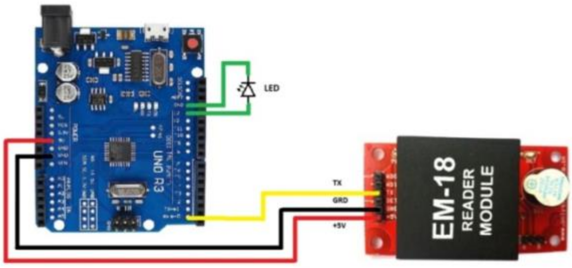
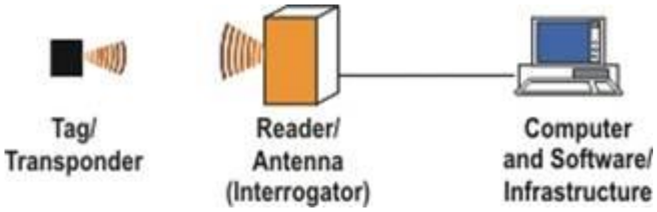


Fig 4.9 circuit for EM18

Components:

A basic RFID system consists of three components:

- An antenna or coil
- A transceiver (with decoder)
- A transponder (RF tag) electronically programmed with unique information.



The antenna emits radio signals to activate the tag and read and write data to it. Antennas are available in a variety of shapes and sizes; they can be built into a doorframe to receive tag data from persons or things passing through the door, or mounted on an interstate tollbooth to monitor traffic passing by on a freeway.

Transceiver: Often the antenna is packaged with the transceiver and decoder to become a reader (a.k.a. interrogator), which can be configured either as a handheld or a fixed-mount device. The reader emits radio waves in ranges of anywhere from one inch to 100 feet or more, depending upon its power output and the radio frequency used. When an RFID tag passes through the electromagnetic zone, it detects the reader's activation signal.

The purpose of an RFID system is to enable data to be transmitted by a portable device, called a tag, which is read by an RFID reader and processed according to the needs of a particular application. The data transmitted by the tag may provide identification or location information, or specifics about the product tagged, such as price, color, date of purchase, etc. RFID technology has been used by thousands of companies for a decade or more. . RFID quickly gained attention because of its ability to track moving objects. As the technology is refined, more pervasive - and invasive - uses for RFID tags are in the works.

A typical RFID tag consists of a microchip attached to a radio antenna mounted on a substrate. The chip can store as much as 2 kilobytes of data.

To retrieve the data stored on an RFID tag, you need a reader. A typical reader is a device that has one or more antennas that emit radio waves and receive signals back from the tag. The reader then passes the information in digital form to a computer system.

More About Antennas

Antennas are the conduits between the tag and the transceiver, which controls the system's data acquisition and communication. Antennas are available in a variety of shapes and sizes; they can be built into a door frame to receive tag data from persons or things passing through the door, or mounted on an interstate toll booth to monitor traffic passing by on a freeway. The electromagnetic field produced by an antenna can be constantly present when multiple tags are expected continually. If constant interrogation is not required, the field can be activated by a sensor device.

Often the antenna is packaged with the transceiver and decoder to become a reader (a.k.a. interrogator), which can be configured either as a handheld or a fixed-mount device.

RFID Tags

RFID tags come in a wide variety of shapes and sizes. Animal tracking tags, inserted beneath the skin, can be as small as a pencil lead in diameter and one-half inch in length. Tags can be screw-shaped to identify trees or wooden items, or credit-card shaped for use in access applications. The anti-theft hard plastic tags attached to merchandise in stores are RFID tags. In addition, heavy-duty 5- by 4- by 2-inch rectangular transponders used to track intermodal containers or heavy machinery, trucks, and railroad cars for maintenance and tracking applications are RFID tags.

There are a variety of RFID tag types. Selecting the correct tag will be imperative to ensure a proper functioning system. Selecting the proper tag will be discussed later. Tags can be placed on wooden or plastic pallets, clothing, embedded into traditional barcode labels, animals, metal surfaces, and much more.

The data associated with a tag is programmed into the chip. The tag is placed on merchandise and is activated and read when it is energized by the reader and antenna system.

The IC contains an actual microchip where data is stored. Chips are available in many sizes and configurations. They can be extremely small to be incorporated into small form factor RFID tags. The chips' capability to carry data and have that data amended is defined by their Read/Writecharacteristics.

An RFID tag can take on many form factors and power levels. The unique identifier is encoded onto the integrated circuit and travels with this data. The data on the RFID IC is transmitted to a reader through the antenna incorporated onto the tag. RFID tags can be as tiny as an ant's head, larger than the palm of an adult hand, or any size in between. The form factor that the RFID tag takes is dictated by factors including power, durability, and lifetime requirements. Tag characteristics are defined by the application, and can vary in power requirements, read/write capability, and frequency. RFID tags are developed using a frequency according to the needs of the system including read range and the environment in which the tag will be read.

Active or Passive RFID …

RFID tags are categorized as either active or passive. Active RFID tags are powered by an internal battery and are typically read/write, i.e., tag data can be rewritten and/or modified. An active tag's memory size varies according to application requirements; some systems operate with up to 1MB of memory. In a typical read/write RFID work-in-process system, a tag might give a machine a set of instructions, and the machine would then report its performance to the tag. This encoded data would then become part of the tagged part's history. The battery-supplied power of an active tag generally gives it a longer read range. The trade off is greater size, greater cost, and a limited operational life (which may yield a maximum of 10 years, depending upon operating temperatures and battery type).

Passive RFID tags operate without a separate external power source and obtain operating power generated from the reader. Passive tags are consequently much lighter than active tags, less expensive, and offer a virtually unlimited operational lifetime. The trade off is that they have shorter read ranges than active tags and require a higher-powered reader. Read-only tags are typically passive and are programmed with a unique set of data (usually 32 to 128 bits) that cannot be modified. Read-only tags most often operate as a license plate into a database, in the same way as linear barcodes reference a database containing modifiable product-specific information.

Frequencies

RFID systems are also distinguished by their frequency ranges.

- Low-frequency or LF (30 KHz to 500 KHz) systems have short reading ranges and lower system costs. They are most commonly used in security access, asset tracking, and animal identification applications.
- High-frequency or HF (850 MHz to 950 MHz and 2.4 GHz to 2.5 GHz) systems, offering long read ranges (greater than 90 feet) and high reading speeds, are used for such applications as railroad car tracking and automated toll collection. However, the higher performance of high-frequency RFID systems incurs higher system costs.
- Ultra high frequency or UHF

Advantages

The significant advantage of all types of RFID systems is the noncontact, non-line-of-sight nature of the technology. Tags can be read through a variety of substances such as snow, fog, ice, paint, crusted grime, and other visually and environmentally challenging conditions, where bar codes or other optically read technologies would be useless. RFID tags can also be read in challenging circumstances at remarkable speeds, in most cases responding in less than 100 milliseconds. The read/write capability of an active RFID system is also a significant advantage in interactive applications such as work-in-process or maintenance tracking. Though it is a costlier technology (compared with bar code), RFID has become indispensable for a wide range of automated data collection and identification applications that would not be possible otherwise.

Developments in RFID technology continue to yield larger memory capacities, wider reading ranges, and faster processing. It is highly unlikely that the technology will ultimately replace bar code — even with the inevitable reduction in raw materials coupled with economies of scale, the integrated circuit in an RF tag will never be as cost-effective as a barcode label. However, RFID will continue to grow in its established niches where bar code or other optical technologies are not effective.

Inventory efficiency - Because line of sight is not required to read RFID tags, inventory can be performed in a highly efficient method. For example, pallets in a warehouse can be read, inventoried, and their location can be determined no matter where the tag is placed on the pallet. This is because the radio waves from the reader are strong enough for the tag to respond regardless of location.

Return on investment - Though the cost may be high at first, the total cost of ownership should go down over the years and provide a return on investment (ROI), if the implementation provides a significant method to improve business processes.

Vulnerability to damage minimized - barcodes can be damaged in many ways. Although, 2D barcode types such as Data Matrix can be read even when up to 40% of the barcode is damaged.

4.5 DC MOTOR

A **DC motor** is any of a class of rotary electrical machines that converts direct current electrical power into mechanical power. The most common types rely on the forces produced by magnetic fields. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current flow in part of the motor.

DC motors were the first type widely used, since they could be powered from existing direct-current lighting power distribution systems. A DC motor's speed can be controlled over a wide range, using either a variable supply voltage or by changing the strength of current in its field windings. Small DC motors are used in tools, toys, and appliances. The universal motor can operate on direct current but is a lightweight motor used for portable power tools and appliances. Larger DC motors are used in propulsion of electric vehicles, elevator and hoists, or in drives for steel rolling mills. The advent of power electronics has made replacement of DC motors with AC motors possible in many applications.

A coil of wire with a current running through it generates an electromagnetic field aligned with the center of the coil. The direction and magnitude of the magnetic field produced by the coil can be changed with the direction and magnitude of the current flowing through it. A simple DC motor has a stationary set of magnets in the stator and an armature with one or more windings of insulated wire wrapped around a soft iron core that concentrates the magnetic field. The windings usually have multiple turns around the core, and in large motors there can be several parallel current paths. The ends of the wire winding are connected to a commutator. The commutator allows each armature coil to be energized in turn and connects the rotating coils with the external power supply through brushes. (Brushless DC motors have electronics that switch the DC current to each coil on and off and have no brushes.) The total amount of current sent to the coil, the coil's size and what it's wrapped around dictate the strength of the electromagnetic field created.

The sequence of turning a particular coil on or off dictates what direction the effective electromagnetic fields are pointed. By turning on and off coils in sequence a rotating magnetic field can be created. These rotating magnetic fields interact with the magnetic fields of the magnets (permanent or electromagnets) in the stationary part of the motor (stator) to create a force on the armature which causes it to rotate. In some DC motor designs the stator fields use electromagnets to create their magnetic fields which allow greater control over the motor. At high power levels, DC motors are almost always cooled using forced air.

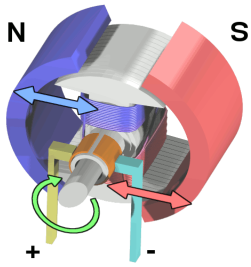


Fig 4.10 dc motor model

Fig 4.11 DC motor

4.6 ZIGBEE

ZigBee is an IEEE 802.15.4-based specification for a suite of high-level communication protocols used to create personal area networks with small, low-power digital radios, such as for home automation, medical device data collection, and other low-power low-bandwidth needs, designed for small scale projects which need wireless connection. Hence, ZigBee is a low-power, low data rate, and close proximity (i.e., personal area) wireless ad hoc network.

The technology defined by the ZigBee specification is intended to be simpler and less expensive than other wireless personal area networks (WPANs), such as Bluetooth or Wi-Fi. Applications include wireless light switches, electrical meters with in-home-displays, traffic management systems, and other consumer and industrial equipment that requires short-range low-rate wireless data transfer.

Its low power consumption limits transmission distances to 10–100 meters line-of-sight, depending on power output and environmental characteristics. ZigBee devices can transmit data over long distances by passing data through a mesh network of intermediate devices to reach more distant ones. ZigBee is typically used in low data rate applications that require long battery life and secure networking (ZigBee networks are secured by 128 bit symmetric encryption keys.) ZigBee has a defined rate of 250 kbit/s, best suited for intermittent data transmissions from a sensor or input device.

ZigBee is a low-cost, low-power, wireless mesh network standard targeted at the wide development of long battery life devices in wireless control and monitoring applications. Zigbee devices have low latency, which further reduces average current. ZigBee chips are typically integrated with radios and with microcontrollers that have between 60-256 KB of flash memory. ZigBee operates in the industrial, scientific and medical (ISM) radio bands: 2.4 GHz in most jurisdictions worldwide; 784 MHz in China, 868 MHz in Europe and 915 MHz in the USA and Australia. Data rates vary from 20 kbit/s (868 MHz band) to 250 kbit/s (2.4 GHz band).

The ZigBee network layer natively supports both star and tree networks, and generic mesh networking. Every network must have one coordinator device, tasked with its creation, the control of its parameters and basic maintenance. Within star networks, the coordinator must be the central node. Both trees and meshes allow the use of ZigBee routers to extend communication at the network level.

ZigBee builds on the physical layer and media access control defined in IEEE standard 802.15.4 for low-rate WPANs. The specification includes four additional key components: network layer, application layer, *ZigBee device objects* (ZDOs) and manufacturer-defined application objects which allow for customization and favor total integration. ZDOs are responsible for some tasks, including keeping track of device roles, managing requests to join a network, as well as device discovery and security.

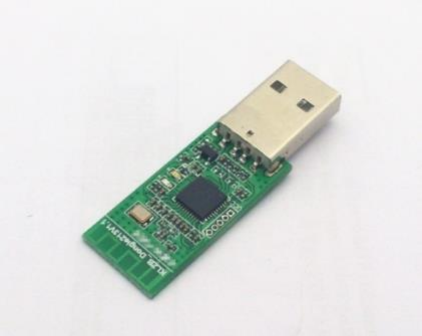


Fig 4.12 Zigbee module

ZigBee is one of the global standards of communication protocol formulated by the significant task force under the IEEE 802.15 working group. The fourth in the series, WPAN

Low Rate/ZigBee is the newest and provides specifications for devices that have low data rates, consume very low power and are thus characterized by long battery life. Other standards like Bluetooth and IrDA address high data rate applications such as voice, video and LAN communications.

ZigBee is a technology of data transfer in wireless network. It also can be described as a wireless technology developed as an open global standard to address the unique needs of low-cost, low-power wireless M2M networks. ZigBee has a low energy consumption and its designed for multichannel control systems, alarm system and lighting control. It operates on IEEE 802.15.14 physical radio specification and operates in unlicensed bands. It is also more economical than Wi-Fi and Bluetooth which makes it simpler. ZigBee provides the ability to run for years on inexpensive batteries for a host of monitoring and control applications. The ZigBee network layer support star and tree networks and mesh networking. It ensures that networks remain operable in the conditions of a constantly changing quality between communication nodes. In mesh and tree topologies, The ZigBee network is extended with several routers where coordinator is responsible for staring them. They allow any devices to communicate with any other adjacent node for providing redundancy to the data. If any node fails, the information is routed automatically to other device by these topologies. In star topology, the network consists of one coordinator responsible for initiating and managing the devices over a network. All devices of the protocol can interact because it has a unified standard of data transfer. The ZigBee advantage is the ZigBee protocol which is designed to communicate data through hostile RF environment that are common in commercial and industrial application. Its protocol feature include support for multiple network topologies such as; point to point and mesh network, collision avoidance and retries, and low latency. Another defining feature of ZigBee is to provide facilities for carrying out secure communications, protecting establishment and transport of cryptographic keys, cyphering frames, and controlling device. It builds on the basic security framework defined in IEEE 802.15.4.

USB to TTL converter:

This is an USB2.0 to TTL UART Converter module which is based on CP2102 Bridge by SiLabs. This module can be used with Laptop's which don't have standard serial port. This module creates a virtual COM port using USB on your computer which can support various standard Baud Rates for serial communication. You just need to install the driver using a setup file which automatically installs correct driver files for Windows XP/Vista/ 7. After driver installation, plug the module into any USB port of your PC. Finally a new COM port is made available to the PC. The feature which makes it more convenient is the TTL level data i/o. So you don't need to make a RS232 to TTL converter using chips like MAX232. The Rx and Tx pin can be connected directly to the MCUs pins (assuming 5v i/o).

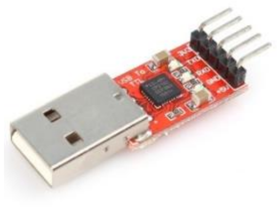


Fig 4.13 Zigbee TTL

Pinouts:

This module has 5 pin breakout which includes

TXD = Transmit Output - Connect to Receive Pin(RXD) of Micro controller. This pin is TX pin of CP2102 on board.

RXD = Receive Input - Connect to Transmit Pin(TXD) of Micro controller. This pin is RX pin of CP2102 on board.

GND = Should be common to microcontroller ground.

3V3 = Optional output to power external circuit upto 50mA.

5V = Optional output to power external circuit upto 500mA

Features:

Stable and reliable chipset CP2102.

USB specification 2.0 compliant with full-speed 12Mbps.

Standard USB type A male and TTL 6pin connector.

6pins for 3.3V, RST, TXD, RXD, GND & 5V.

All handshaking and modem interface signals.

Baud rates: 300 bps to 1.5 Mbps.

Byte receive buffer; 640 byte transmit buffer.

Hardware or X-On/X-Off handshaking supported.

Event character support Line break transmission.

USB suspend states supported via SUSPEND pins.

Temperature Range: -40 to +85.

Size: 42mm X 15mm.

Weight: 4g

4.7 SMOKE SENSOR

A smoke sensor is a device that senses smoke, typically as an indicator of fire. Commercial and residential security devices issue a signal to a fire alarm control panel as part of a fire alarm system, while household detectors, known as smoke alarms, generally issue a local audible or visual alarm from the detector itself. The Analog Smoke/LPG/CO Gas Sensor (MQ2) module utilizes an MQ-2 as the sensitive component and has a protection resistor and an adjustable resistor on board. The MQ-2 gas sensor is sensitive to LPG, i-butane, propane, methane, alcohol, Hydrogen and smoke. It could be used in gas leakage detecting equipments in family and industry. The resistance of the sensitive component changes as the concentration of the target gas changes.

Specifications:

- Dimension: 32mm x 22mm x 30mm Specification:
- Operating voltage: 5v
- Detection Zone: 300 - 10000ppmm

- Characteristic Gas: 1000ppmm
- Sensitivity: R in air/ R in typical gas > 5
- Response Time: <10s
- Recovery time: <30s
- Heating Resistance: 31ohm

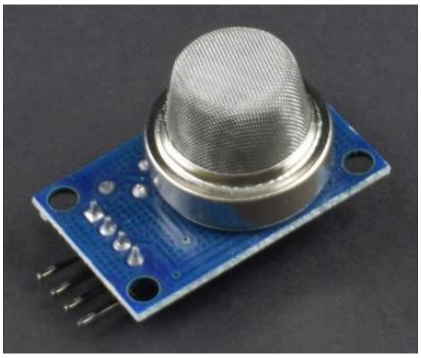


FIG 4.14 SMOKE SENSOR

Pin Configuration:

1. VCC

2. D0 pin

3. A0 pin

4. Ground

How to test :

1. Connect your Arduino microcontroller to the computer.

2. Connect the VCC pin of your module to the to the 5V pin of your Arduino.

3. Connect the GND pin of your module to the GND pin of your Arduino.

4. Connect the Output pin of your module to the A0 pin of you Arduino.

5. Enter this program to your Arduino Integrated Development Environment (IDE):

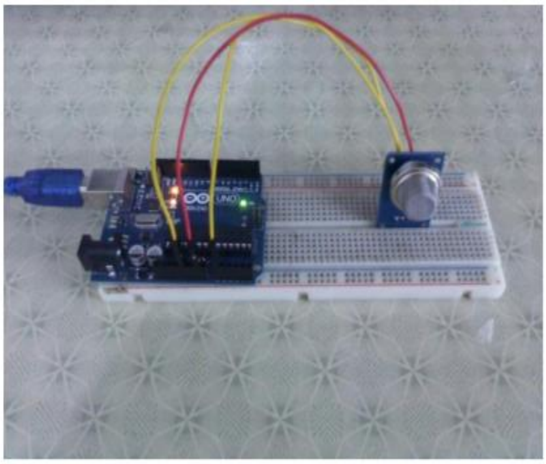
```
void setup()
{
 Serial.begin(9600);
}
void loop()
{
 if (analogRead(A0) < 100) Serial.println("No Smoke");
 else Serial.println("Smoke Detected!");
 delay(100);
}
```

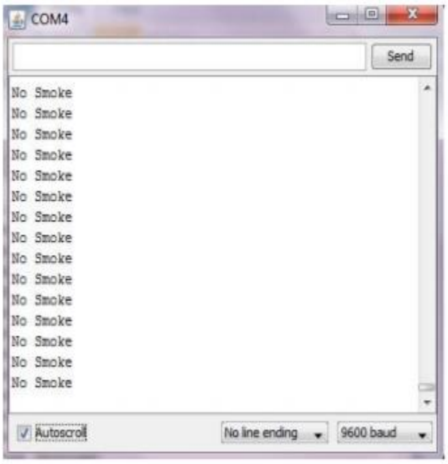
6. Click the Upload Button

7. Lastly, click the Serial Monitor button

Testing Results:

When there is no smoke exposure:

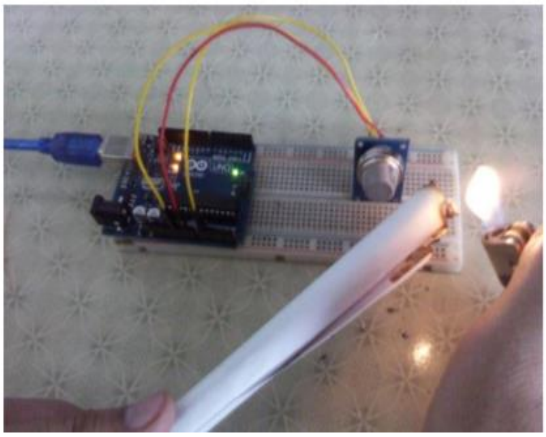


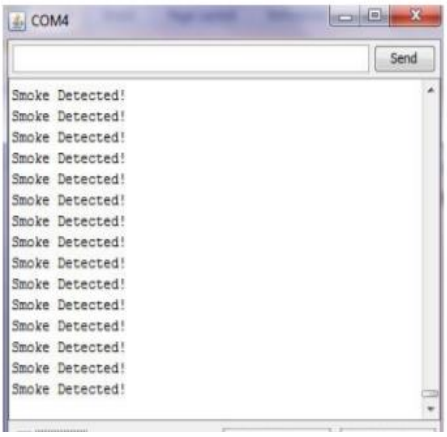


```
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
No Smoke
```

☑ Autoscroll   No line ending   9600 baud

When exposed to a burning paper:





```
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
Smoke Detected!
```

Fig 4.15 smoke sensor working

4.8 PROXIMITY SENSOR

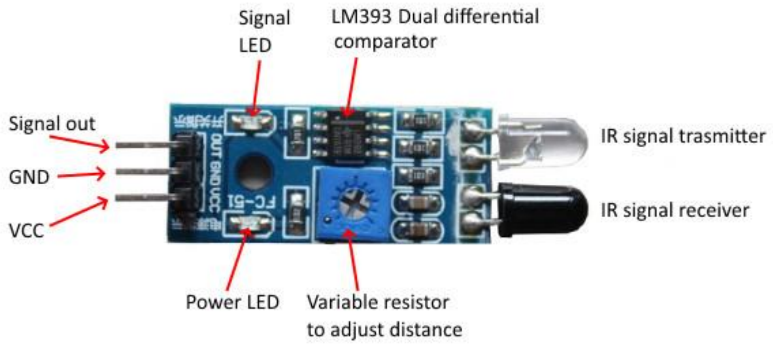


Fig 4.16 proximity sensor

"Proximity Sensor" includes all sensors that perform non-contact detection in comparison to sensors, such as limit switches, that detect objects by physically contacting them. Proximity Sensors convert information on the movement or presence of an object into an electrical signal. There are three types of detection systems that do this conversion: systems that use the eddy currents that are generated in metallic sensing objects by electromagnetic induction, systems that detect changes in electrical capacity when approaching the sensing object, and systems that use magnets and reed switches. The Japanese Industrial Standards (JIS) define proximity sensors in JIS C 8201-5-2 (Low-voltage switchgear and controlgear, Part 5: Control circuit devices and switching elements, Section 2: Proximity switches), which conforms to the IEC 60947-5-2 definition of non-contact position detection switches

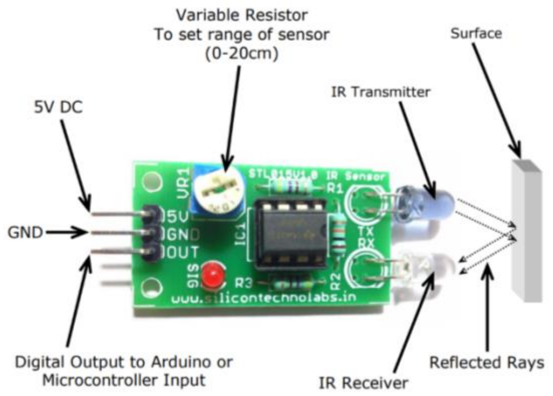


Fig 4.17 proximity sensor working

4.9 SWITCHING RELAY

The advantage of using a 5V relay in this project is that the power supply for the relay can be directly given from the Arduino UNO board itself. Let us now see some basics of a relay. A relay is a type of a switch that acts as an interface between microcontrollers and AC Loads. A simple Single Pole – Single Throw (SPST) relay, like the one used in this project consists of 5 Terminals: 5V, GND, Normally Open (NO), Normally Close (NC) and Common (COMM). Since we need to control this relay with the help of Arduino, a transistor is used with an additional pin called Control Pin on the Relay Module. We can control high voltage electronic devices using relays. A Relay is actually a switch which is electrically operated by an electromagnet. The electromagnet is activated with a low voltage, for example 5 volts from a microcontroller and it pulls a contact to make or break a high voltage circuit.

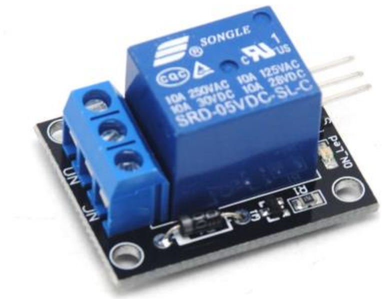


Fig 4.18 switch relay

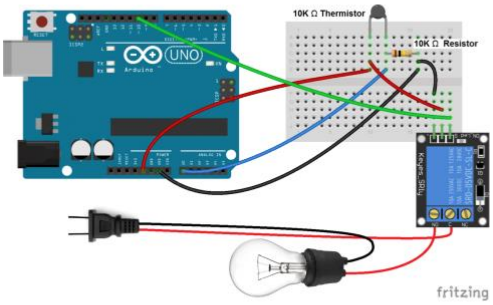


Fig 4.19 switch with arduino

As an example for this Arduino Relay Tutorial we will use the HL-52S 2 channel relay module, which has 2 relays with rating of 10A @ 250 and 125 V AC and 10A @ 30 and 28 V DC. The high voltage output connector has 3 pins, the middle one is the common pin and as we can see from the markings one of the two other pins is for normally open connection and the other one for normally closed connection.

On the other side of the module we have these 2 sets of pins. The first one has 4 pins, a Ground and a VCC pin for powering the module and 2 input pins In1 and In2. The second set of pins has 3 pins with a jumper between the JDVcc and the Vcc pin. With a configuration like this the electromagnet of the relay is directly powered from the Arduino Board and if something goes wrong with the relay the microcontroller could get damaged.

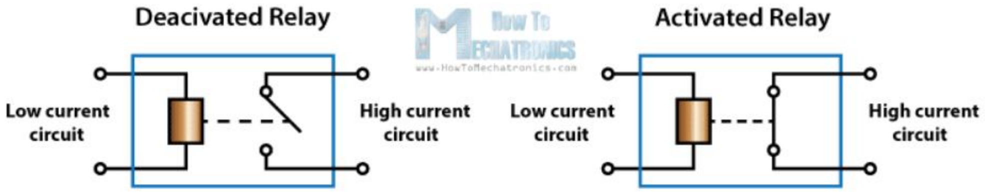


Fig 4.20 relay working

Advantages and Disadvantages :

Advantages: The main and important advantage of connecting a 5V relay with Arduino is that it can be powered by Arduino itself.

Disadvantages: A transistor based relay might not be ideal for long time use as there will always be noise in the relay coil. A suitable option will be using additional isolation like an opto-isolator or completely eliminating the electromechanical relay and replacing it with solid state relay.

Applications :

Interfacing a 5V Relay with Arduino opens up the door to a huge number of applications. Although the main task of the relay is to control a load, how that relay is being operated by the Arduino makes it an interesting project.

Some of the techniques and methods using which we can control the relay are: Bluetooth, Infrared (IR) remote, RF Transmitter – Receiver Pair or even using Internet.

4.10 ALCOHOL SENSOR

The analog gas sensor- MQ3 is suitable for alcohol detecting, this sensor can be used in a breath analyzer. It has a high sensitivity to alcohol and small sensitivity to benzene. The sensitivity can be adjusted by the potentiometer sensitive material of MQ3 gas sensor is SnO2, which with lower conductivity in clean air. When the target alcohol gas exist, the sensors conductivity is higher along with the gas concentration rising, use of simple electro circuit, convert change of conductivity to correspond output signal of gas concentration. The MQ-3 sensor is made of Tin Dioxide (SnO2) delicate layer. It is sorted out in such a structure to give high affectability to liquor and low affectability to Benzene. It has an immediate drive circuit to give lively reaction, quality, and longer lifetime. It is having a clear interface type. On the sensor, port pins 1, 2 and 3 tends to the yield, GND and VCC independently. The particular of the sensor is depicted in table underneath.

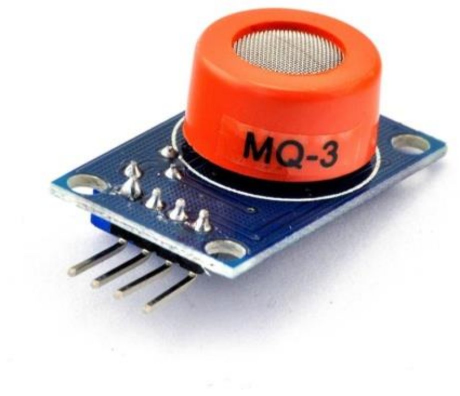
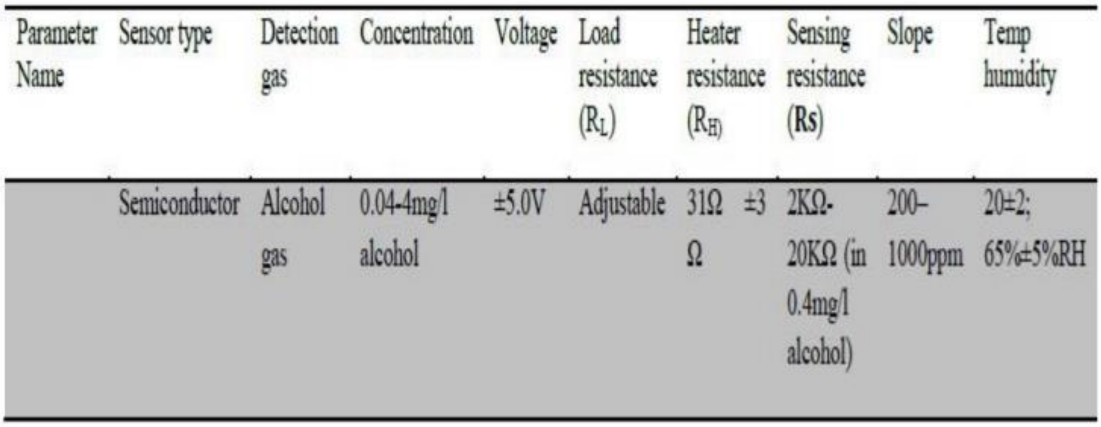


Fig 4.21 alchohol sensor



Parameter Name	Sensor type	Detection gas	Concentration	Voltage	Load resistance (R_L)	Heater resistance (R_H)	Sensing resistance (**Rs**)	Slope	Temp humidity
	Semiconductor	Alcohol gas	0.04-4mg/l alcohol	±5.0V	Adjustable	31Ω ±3 Ω	2KΩ-20KΩ (in 0.4mg/l alcohol)	200–1000ppm	20±2; 65%±5%RH

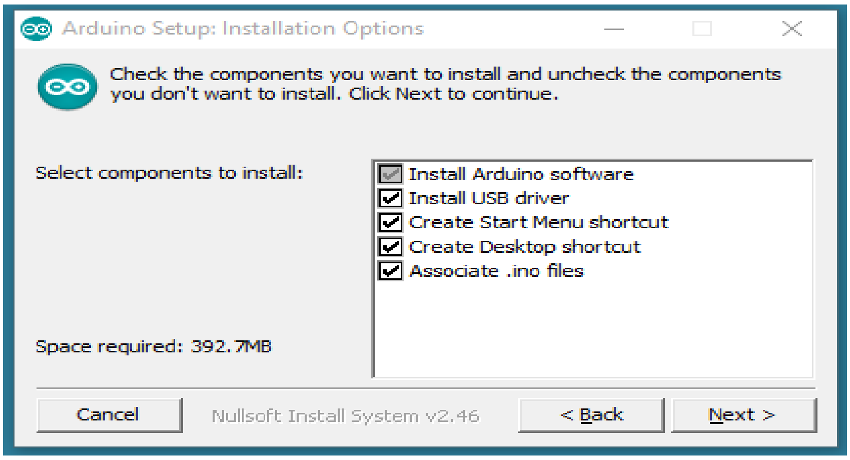
CHAPTER 5

SOFTWARE IMPLEMENTATION

5.1 INSTALLING ARDUINO IDE :

1. Install latest version of arduino ide from official arduino website.

2. When the download finishes, proceed with the installation and please allow the driver installation process when you get a warning from the operating system.



Arduino Setup: Installation Options

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

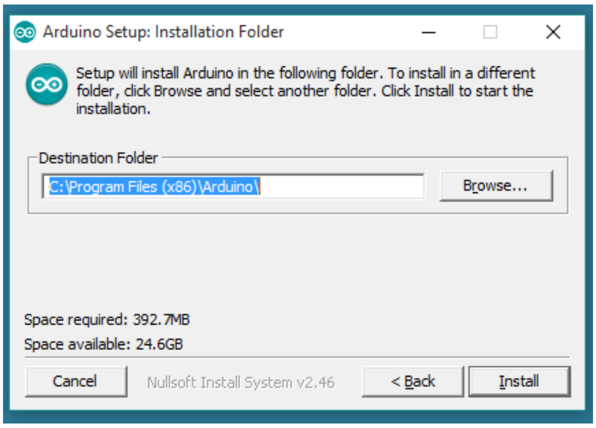
Select components to install:

☑ Install Arduino software
☑ Install USB driver
☑ Create Start Menu shortcut
☑ Create Desktop shortcut
☑ Associate .ino files

Space required: 392.7MB

Fig 5.1 arduino installation

3. Choose the components to install



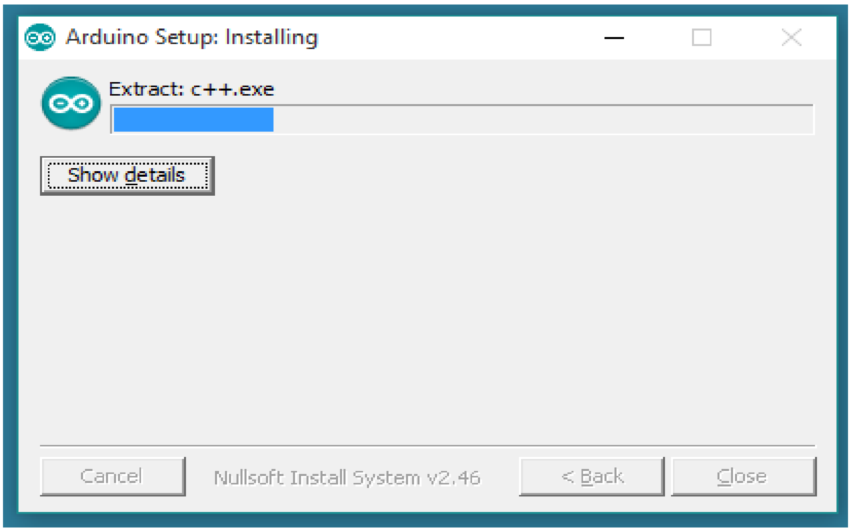
Arduino Setup: Installation Folder

Setup will install Arduino in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.

Destination Folder: C:\Program Files (x86)\Arduino\

Space required: 392.7MB
Space available: 24.6GB

4. Choose the installation directory (we suggest to keep the default one)



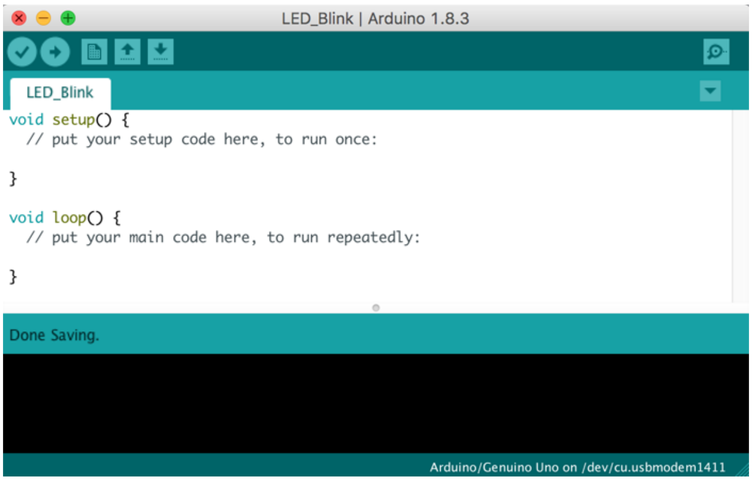
Arduino Setup: Installing

Extract: c++.exe

Fig 5.2 arduino extracting

5. The process will extract and install all the required files to execute properly the Arduino Software (IDE)

6. Once you've successfully installed Arduino IDE, it's time to start coding. The source code files for Arduino are called sketches. The [language]() used for arduino is based off C/C++ and is very similar. Open Arduino IDE and a new blank sketch will appear on your screen.



LED_Blink | Arduino 1.8.3

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

}

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

}
```

Done Saving.

Arduino/Genuino Uno on /dev/cu.usbmodem1411

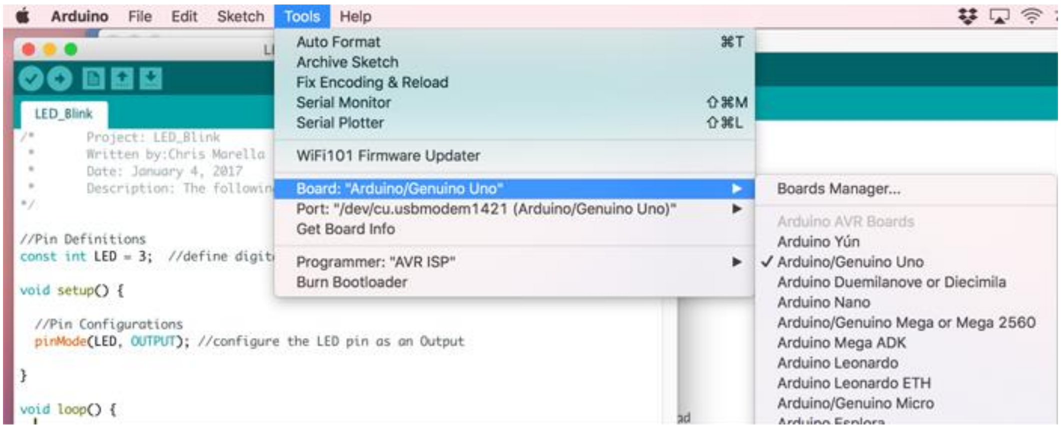
7. Uploading an Arduino Sketch :

Connect the Arduino board to your computer through USB. Once the Arduino is connected, follow these steps to upload the sketch:

1) Select the target board

2) The target board selection tells Arduino IDE which Arduino board you are uploading to

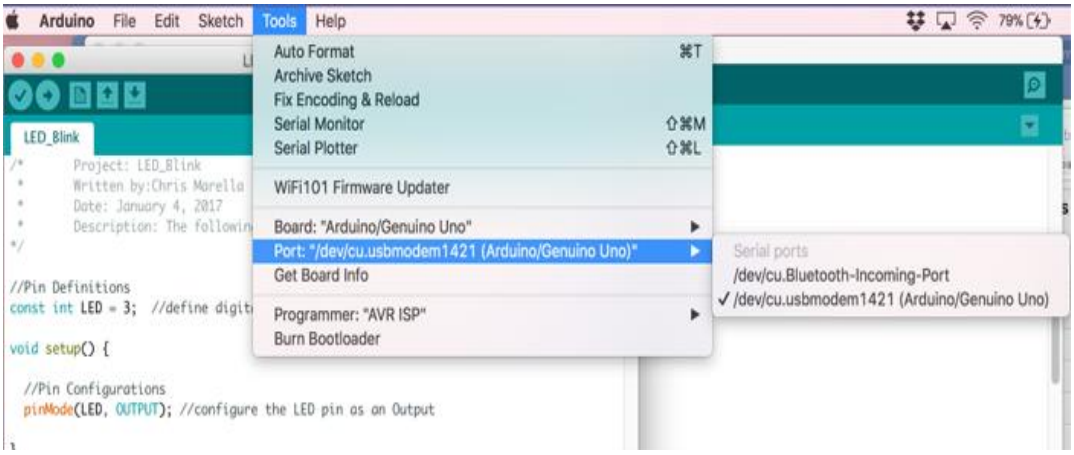
Note: If you have a different board then the Arduino Uno, select that board



```
/*
 *     Project: LED_Blink
 *     Written by:Chris Marella
 *     Date: January 4, 2017
 *     Description: The followin
*/

//Pin Definitions
const int LED = 3;  //define digit

void setup() {

  //Pin Configurations
  pinMode(LED, OUTPUT); //configure the LED pin as an Output

}

void loop() {
```

8. Select the serial port the board is connected to (Figure 10: Serial Port Connection)

Note: On Windows this will typically be a COM port



```
/*
 *     Project: LED_Blink
 *     Written by:Chris Marella
 *     Date: January 4, 2017
 *     Description: The followin
*/

//Pin Definitions
const int LED = 3;  //define digit

void setup() {

  //Pin Configurations
  pinMode(LED, OUTPUT); //configure the LED pin as an Output

}
```

9. Press the “Upload” button to upload the sketch to the Arduino.

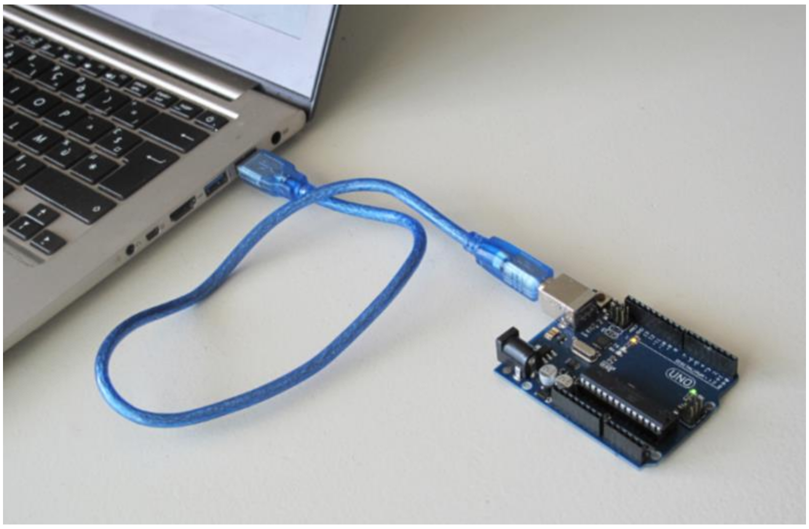


Fig 5.3 arduino uno connection

5.2 SOURCE CODE

```
//AccessControl.ino

#include<LiquidCrystal.h>

LiquidCrystallcd(7,6,5,4,3,2);

#define relay 8

char tag1[] ="6A007D57CC8C"; // Replace with your Tag ID

char input[12];

int count = 0;

boolean match1 = 0;    // A variable to store the Tag match status

void setup()

{

lcd.begin(16,2);

Serial.begin(9600);

pinMode(relay,OUTPUT);

pinMode(11,OUTPUT);

pinMode(14,INPUT);

pinMode(9,INPUT);

pinMode(10,INPUT);

digitalWrite(relay, HIGH);
digitalWrite(11, LOW);

Serial.print("\nWelcome\n\r");

lcd.clear();

lcd.print("Welcome");

}

void loop()

{

    if(digitalRead(14)==LOW)

      {

digitalWrite(11,HIGH);

Serial.print("\nObstacle detected\n\r");

lcd.setCursor(0,1);

lcd.print("Obstacle detected");

delay(1000);

digitalWrite(11, LOW);

      }

      else if(digitalRead(9)==LOW)

      {

digitalWrite(11,HIGH);

Serial.print("\nAlcohol detected\n\r");

lcd.setCursor(0,1);

lcd.print("Alcohol detected");

delay(1000);

digitalWrite(11, LOW);

      }

      else if(digitalRead(10)==LOW)

      {

digitalWrite(11,HIGH);

Serial.print("\nSmoke detected\n\r");

lcd.setCursor(0,1);

lcd.print("Smoke detected");

delay(1000);

digitalWrite(11, LOW);
```

```
      }    else

      {

lcd.setCursor(0,1);

lcd.print("                ");

      }

if(Serial.available())// check serial data ( RFID reader)

  {


    count = 0; // Reset the counter to zero

while(Serial.available() && count < 12)

    {

      input[count] = Serial.read(); // Read 1 Byte of data and store it in the
input[] variable

      count++; // increment counter

delay(5);

    }

if(count == 12) //

    {

      count =0; // reset counter varibale to 0

     if(input[count]==tag1[count])

        {

Serial.println("Authorised Vehicle");

digitalWrite(11,HIGH);

lcd.clear();

lcd.print("Authorised");

delay(300);

digitalWrite(11,LOW);

digitalWrite(relay, LOW);

delay(5000);

digitalWrite(relay, HIGH);

        }

        else

        {

Serial.println("Access Denied");
lcd.clear();
lcd.print("  Unauthorised  ");// Incorrect Tag Message
digitalWrite(11, HIGH);
delay(2000);
digitalWrite(11, LOW);
        }
          count++; // increment i
      }
    }
if(match1 == 1) // If match variable is 1, then it means the tags match
    {
Serial.println("Authorised Vehicle");
digitalWrite(11,HIGH);
lcd.clear();
lcd.print("Authorised Vehicle");
delay(300);
digitalWrite(11,LOW);
digitalWrite(relay, LOW);
delay(5000);
digitalWrite(relay, HIGH);
    }
    else
    {
Serial.println("Access Denied");
lcd.clear();
lcd.print("  Unauthorised  ");// Incorrect Tag Message
digitalWrite(11, HIGH);
delay(2000);
digitalWrite(11, LOW);
    }
    /* Fill the input variable array with a fixed value 'F' to overwrite
    all values getting it empty for the next read cycle */
for(count=0; count<12; count++)
```

```
    {
      input[count]= 'F';
    }
    count = 0; // Reset counter variable
  }
```

6 . ADVANTAGES & DISADVANTAGES

Advantages

- By this project accidents occurance can reduced
- Secure wireless identity for vehicles.
- Reducing accidents due to drunk and drive
- The zigbee has flexible network structure.
- It can be easily implemented.
- It has a very low cost.
- It is more reliable and self healing.
- Setting up the network is very simple and easy.

Disadvantages:

- It is not as secure as wi fi based secured system.

7. APPLICATIONS

1. It can be installed at Offices, homes, authorized areas etc.
2. Can be installed at any four wheel automobiles at low cost.

8. FUTURE SCOPE

This project can further be improved by giving the sensor signals to mechanical control which makes its more developed version of protection such that, When the obstacle was detected in front it sensor sences and access the brakes so that to stop vehicle. And when the alchohol consumption was detected by sensor it send a message to fanily members and track the location every time. RFID identification can further be made small such that it can be placed inside our skin so that owner himself can be the key to vehicle.

9. CONCLUSION

The project **"ZIGBEE BASED VEHICLE ACCESS & PROTECTING SYSTEM"** has been successfully designed and tested. It has been developed by integrating features of all the hardware components used. Presence of every module has been reasoned out and placed carefully thus contributing to the best working of the unit.

Project has high capabilities of detecting drunk and drive situations, Authurised vehicle identity gate opening system, RFID based vehicle accessing, Detecting abnormal conditions of engine like smoke detecting, detecting obstacles in front of vehicle and indication. By all this work it can be concluded that the project provides a secure and protected way for vehicle access and protection system.

10. REFARENCES

1. "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", by Pethuru Raj and Anupama C. Raman (CRC Press)

2. https://en.wikipedia.org/wiki/Zigbee
3. "Internet of Things: A Hands-on Approach", by Arshdeep Bahga and Vijay Madisetti (Universities Press)
4. Electronic Components -D.V.Prasad
5. Wireless Communications Theodore S. Rappaport
6. Mobile Tele Communications William C.Y. Lee

References on the Web:

1. www.national.com
2. www.nxp.com
3. www.microsoftsearch.com
4. www.geocities.com