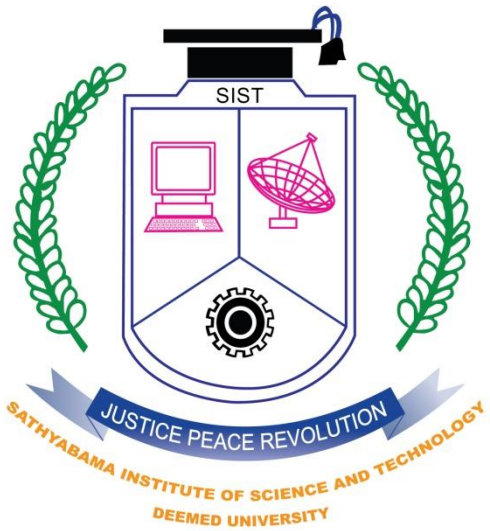
BLUETOOTH HOME AUTOMATION USING ARDUINO UNO

Submitted in partial fulfillment of the requirements for the award of

Bachelor of Engineering Degree in

Electrical and Electronics Engineering



ANUSHAA.A(39140003)

HARINI.C(39140017)

JOTHIKA.P(39140020)

PRADEENA.G(39140041)

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

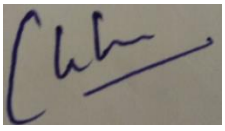
SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

APRIL 2022

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of ANUSHAA.A (39140003) HARINI.C (39140017) JOTHIKA.P (39140020)PRADEENA(39140041) who carried out the project entitled "BLUETOOTH HOME AUTOMATION USING ARDUINO UNO" under our supervision from november 2022 to april 2022



Dr. C.Bhuvaneshwari
Internal Guide

Dr. V. SIVACHIDAMBARANATHAN
Head of the Department

Submitted for Viva voce Examination held on 29.04.2022

Name:

Signature:

Internal Examiner **External Examiner**

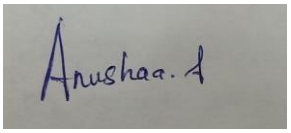
DECLARATION

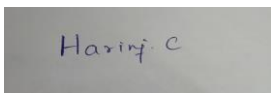
We **ANUSHAA.A(39140003), HARINI.C(39140017), JOTHIKA.P(39140020), PRADEENA.G (39140041)** hereby declare that the Project Report entitled **"WIRELESS POWER TRANSFER SYSTEM FOR ELECTRIC VEHICLE APPLICATION"** done by us under the guidance of **Dr. C.BHUVANESWARI** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electrical and Electronics Engineering.

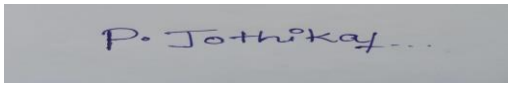
1. ANUSAA.A (39140003)
2. HARINI.C (39140017)
3. JOTHIKA.P(39140060)
4. PRADEENA.G(39140041)

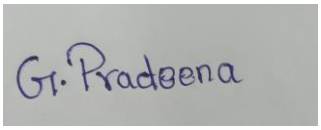
DATE:

PLACE: **SIGNATURE OF THE CANDIDATE**









ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. Nandhita., Dean, School of Electrical and Electronics Engineering** and **Dr. V. Sivachidambaranathan, Head of the Department, Dept. of Electrical and Electronics Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. C.Bhuvaneswari** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electrical and Electronics Engineering who were helpful in many ways for the completion of the project.

TABLE OF CONTENTS

LIST OF FIGURES

BLUETOOTH HOME AUTOMATION USING ARDUINO UNO

ABSTRACT

In order to maintain comfortable living conditions within a home, home monitoring and automation are utilized. Smart home concept brings comfort and convenience in our lives with the aid of IoT. Major issues in current smart home scenario are automation and security. Problem in security arises due to network of devices in the home with internet. Focus is sifted towards providing confidentiality, authenticity, and integrity of data sensed and exchanged by smart home objects. Computation overhead is also a concern for smart home solutions. Comfort and user requirements as per scenario or situation are basic need for automation. Automation with learning human behavior is also a major concern with smart home concept. This project represents IoT based smart home automation approach which is secure and also reduces computation overhead. Automation can be accomplished by using the Bluetooth technology. This gives the inhabitant accesses to certain data in the house and the ability to control some parameters remotely

CHAPTER-1

INTRODUCTION

The proposed method presents the design and implementation of a robust, low cost and user-friendly home automation system using Bluetooth technology. The design of proposed method is based on Arduino board, Bluetooth module, sensors and smartphone application. Bluetooth module HC-06 is interfaced with Arduino board and home appliances are connected with Arduino board via relay. Smartphone application is used for serial communication between smartphone and Bluetooth module which is further connected with Arduino board. Proposed method has ability to not only remotely control the appliances but it also monitors the sensors. Nowadays most of conventional home automation systems are designed for elderly, handicapped people or for any special purpose. The proposed method is not only suitable for elderly and handicapped people but it also provides a general-purpose home automation system, which can easily implement in existing home. An ultrasonic sensor is used for water level detection and soil moisture sensor is used for automatic irrigation system to provide more ease and facilities to users.

1.1 EMBEDDED SYSTEM:

An embedded system is a combination of computer hardware and software, either fixed in capability or programmable, designed for a specific function or functions within a larger system. Industrial machines, agricultural and process industry devices, automobiles, medical equipment, cameras, household appliances, airplanes, vending machines and toys, as well as mobile devices, are possible locations for an embedded system.

Embedded systems are computing systems, but they can range from having no user interface (UI) -- for example, on devices in which the system is designed to perform a single task -- to complex graphical user interfaces (GUIs), such as in mobile devices. User interfaces can include buttons, LEDs, touchscreen sensing and more. Some systems use remote user interfaces as well.

Embedded system hardware (microprocessor-based, microcontroller-based)

Embedded system hardware can be microprocessor- or microcontroller-based. In either case, an integrated circuit is at the heart of the product that is generally designed to carry out computation for real-time operations. Microprocessors are visually indistinguishable from microcontrollers, but while the microprocessor only implements a central processing unit (CPU) and, thus, requires the addition of other components such as memory chips, microcontrollers are designed as self-contained systems.

Microcontrollers include not only a CPU, but also memory and peripherals such as flash memory, RAM or serial communication ports. Because microcontrollers tend to implement full (if relatively low computer power) systems, they are frequently put to use on more complex tasks. For example, microcontrollers are used in the operations of vehicles, robots, medical devices and home appliances, among others. At the higher end of microcontroller capability, the term system on a chip (SoC) is often used, although there's no exact delineation in terms of RAM, clock speed and so on.

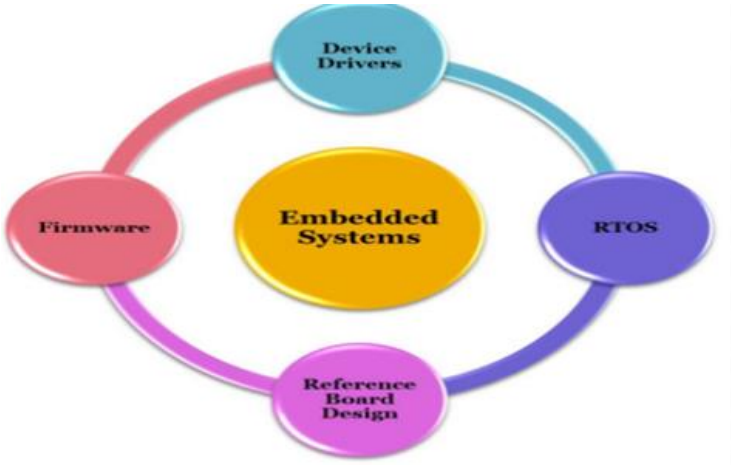


Fig .1.1 Embedded systems

The embedded market was estimated to be in excess of $140 billion in 2013, with many analysts projecting a market larger than $20 billion by 2020. Manufacturers of chips for embedded systems include many mainstays of the computer world, such as Apple, IBM, Intel and Texas Instruments, as well as numerous other companies less familiar to those outside the field. Arm has been a highly influential vendor in this space. The company began as an outgrowth of Acorn, a U.K. maker of early PCs. Arm

chips, produced under license by other companies, are based on the reduced instruction set computer (RISC) architecture and are often used in mobile phones; they remain the most widely deployed SoC in the embedded world, with billions of units fielded.

1.2 Embedded system software

A typical industrial microcontroller is unsophisticated compared to the typical enterprise desktop computer and generally depends on a simpler, less-memory-intensive program environment. The simplest devices run on bare metal and are programmed directly using the chip CPU's machine code language.

Often, embedded systems use operating systems or language platforms tailored to embedded use, particularly where real-time operating environments must be served. At higher levels of chip capability, such as those found in SoCs, designers have increasingly decided the systems are generally fast enough and the tasks tolerant of slight variations in reaction time that near-real-time approaches are suitable. In these instances, stripped-down versions of the Linux operating system are commonly

deployed, although other operating systems have been pared down to run on embedded systems, including Embedded Javaand Windows IoT (formerly Windows Embedded).

Generally, storage of programs and operating systems on embedded devices make use of either flash or rewritable flash memory.

1.3 Embedded firmware

The firmware on embedded systems, referred to as embedded firmware, is specific software written into the memory of a device that serves the purpose of ROM, but can be updated more easily. Firmware can be stored in non-volatile memory devices including ROM, programmable ROM, erasable PROM or flash memory. Embedded firmware is used to control various device and system functions, for example, telling the device how to communicate with other devices, perform specific functions and provide input and output functionality.

The delineation between the terms embedded firmware and embedded software are blurring, but embedded software often refers to the only code running on a piece of hardware, while firmware can also refer to the chip that houses a device's basic input/output system (BIOS) or Unified Extensible Firmware Interface (UEFI), which connect software and a system's operating system.

1.4 Embedded systems vs. VLSI

Very-large-scale integration, or VLSI, is a term that describes the complexity of an integrated circuit. VLSI is the process of embedding hundreds of thousands of transistors into a chip, whereas LSI (large-scale integration) microchips contain thousands of transistors, MSI (medium-scale integration) contain hundreds of

transistors, and SSI (small-scale integration) contain tens of transistors. ULSI, or ultra-large-scale integration, refers to placing millions of transistors on a chip.

VLSI circuits are common features of embedded systems. Many ICs in embedded systems are VLSI, and the use of the VLSI acronym has largely fallen out of favour.

1.5 Debugging embedded systems

One area where embedded systems part ways with the operating systems and development environments of other, larger-scale computers is in the area of debugging. While programmers working with desktop computer environments have systems that can run both the code being developed and separate debugger applications that monitor the actions of the development code as it is executed, embedded system programmers generally cannot.

Some programming languages run on microcontrollers with enough efficiency that rudimentary interactive debugging is available directly on the chip. Additionally, processors often have CPU debuggers that can be controlled -- and, thus, control program execution -- via a JTAG or similar debugging port.

In many instances, however, programmers of embedded systems need tools that attach a separate debugging system to the target system via a serial or other port. In this scenario, the programmer can see the source code on the screen of a conventional personal computer just as would be the case in the debugging of software on a desktop computer. A separate, frequently used approach is to run software on a PC that emulates the physical chip in software, thus making it possible to debug the performance of the software as if it were running on an actual, physical chip.

Broadly speaking, embedded systems have received more attention to testing and debugging because a great number of devices using embedded controls are designed for use in situations where safety and reliability are top priorities.

1.6 The internet of things builds on an embedded systems base

While some embedded systems can be relatively simple, a growing number either supplant human decision-making or offer capabilities beyond what a human could provide. For instance, some aviation systems, including those used in drones, are able to integrate sensor data and act upon that information faster than a human could, permitting new kinds of operating features.

CHAPTER-2

LITERATURE SURVEY

IOT BASED SMART SECURITY AND HOME AUTOMATION

This project focuses on building a home security system which will be wireless. Security over a network is achieved using AES encryption. Security of house is managed by sending notifications to the user using Internet in case of any trespasser and it can also ring an alarm if required. Home automation is utilized by using appropriate sensors installed around house. Raspberry pi is used as a server and controller. Raspberry pi has task of controlling electrical appliances and providing authentication and security to user.

2.1 A SMART HOME AUTOMATION AND METERING SYSTEM USING INTERNET OF THINGS (IOT)

This paper discussed a smart low-cost home automation system which is designed using IoT. With the help of this system, all the home appliances and electronic machines can be controlled and observed through a website very easily. The metering method of a home can also be supervised using this system. Not only the consumers but also the dealers will get the opportunity to observe the anomalies of the power distribution system by watching the metering system. Online billing system also included in the designed system.

2.2 MOBILE BASED HOME AUTOMATION USING INTERNET OF THINGS(IOT)

This paper discusses about IoT and how it can be used for realizing smart horne automation using a micro-controller-based Arduino board and Android mobile app. In this paper, two prototypes namely Horne automation using Bluetooth in an indoor environment and Horne automation using Ethernet in an outdoor environment are presented.

2.3 REAL-TIME SENSOR ACQUISITION INTERFACING USING MATLAB

One of the most important procedures which must be done in the dam's monitoring process, from the dam behavior point of view, is the analysis of the data acquired from the sensors mounted inside or in the proximity of the dam. The analysis can be done in two ways: real-time - using a direct dedicated interfacing algorithm with one or more sensors, respectively a so-called historical analysis - using large data packets acquired during long time periods. The first acquisition procedure shows the behavior of a sensor at an exact moment, while the second reveals - in the end - the dam behavior and suggests the proper actions that must be taken to secure the dam infrastructure.

2.4 HAND GESTURE BASED SWITCHING USING MATLAB

Controlling the Electrical Appliances and Electronic Gadgets through an Infrared Remote Control is now in general. But the same controlling tasks can be done more easily. Primary motive of proposing the new system of Virtual Switch Control is to remove the need to look in to the hand held remote and to search for a specific key for specific function. Here we present a novel system to control Various Appliances by just using hand gestures recognition like showing first finger will switch the first device. It uses real time image processing for hand gestures Recognition using a simple web-camera and microcontroller based embedded system. This project/paper proposes a possible solution to control the gadgets for physically challenged or people in industries, who cannot touch electric panels too often during their work, so better they just by showing hand gestures, devices get controlled. There will be a computer application designed in MATLAB to have a real time image processing. A web camera giving images to the computer application. After processing the images and recognizing the command for switch, decision data is sent to a microcontroller hardware based on AVR environment. This hardware sends the data to the gadgets in same way as a remote control does for general use. To implement the algorithm

and its logic for run time image processing, a MATLAB based processing environment is used. To process the images, Image Processing & Acquisition toolboxes are used.

2.5 THINGSPEAK BASED SENSING AND MONITORING SYSTEM FOR IOT WITH MATLAB ANALYSIS

As the expeditious of Internet of Things (IoT) is emerging and is accustom for remote monitoring of the surrounding parameters and other stuffs with the use of sensors that acquaint for wireless sensing of real time data and transfer them into the desired form and help to forward the sensed data across the network cloud via 'Internet Connection'. Here the project work deals with The IoT 'Thingspeak' web service which is a generous open API service that act as a host for the variety of sensors to monitor the sensed data at cloud level and composite a special feature of porting the sensed data to the MATLAB R2016a using a channel ID and read API key that is assigned by services and able to track data value at picky sample at particular intervals. This project also uses an Arduino UNO board, ESP8266 Wi-Fi Module that helps to process and transfer the sensed data to the Thingspeak Cloud.

2.6 HAND GESTURE RECOGNITION FOR HUMAN COMPUTER INTERACTION

The use of a physical controller like mouse, keyboard for human computer interaction hinders natural interface as there is a strong barrier between the user and computer. In this paper, we have designed a robust marker- less hand gesture recognition system which can efficiently track both static and dynamic hand gestures. Our system translates the detected gesture into actions such as opening websites and launching applications like VLC Player and PowerPoint. The dynamic gesture is used to shuffle through the slides in presentation. Our results show that an intuitive HCI can be achieved with minimum hardware requirements.

2.7 MONITOR AND CONTROL USING VIRTUAL SWITCH TECHNOLOGY

This paper presents a way of replacing mechanical switches by Virtual Switch technology. It mainly aims in designing a virtual switchboard with the help of Laser and LDR which can be used as a replacement of the mechanical switches. Mechanical switches produce sparks and after some span of time they get damaged. As the mechanical switches have limited use they need to be replaced and a better option is using Virtual Switches. We have created an assembly of Virtual Switches which can be used to monitor and control various electrical appliances at home and industrial parameters in industries. Microcontroller used is atmega 16 and Zigbee is used for wireless communication.

2.8 A NEW AND INTELLIGENT MAN-MACHINE INTERFACE

Although investigated from early days of research in the domain of human-computer interfaces, gesture-based control of computer application entered in the everyday life of computer users with the advent of 3D infrared cameras. The usage of real-time depth-mapping cameras and of robust image processing applied the flow of depth map streams triggered the production of a plethora of applications ranging from controlling electronic games and electronic devices such as TV sets or set-top boxes, to e-learning, and sterile environments such as operating rooms. Gesture and motion-based control of computer applications received increased attention from both academic and industrial research groups for the unique interactive experiences it offers. Of particular research interest has been the control of games through gesture-based interface. In this chapter after a brief survey of the methods and technologies used in the gesture control area, a new and intelligent user interface based on a sequence of gestures linked in a gesture language through a sign grammar will be introduced and described. The gesture recognition language is based on image processing functions which are grouped in a combination of algorithmic and learning approaches. Applications of the gesture language in gaming, TV and set-top box control, e-learning and virtual reality-reality interaction illustrate the validity of the approach.

2.9 WIRELESS HOME SECURITY SYSTEM WITH MOBILE

Wireless security is the prevention of unauthorized access or damage to MOBILE using wireless networks Systematic solution for home. The aim of this paper is to investigate a cost-effective solution that will provide controlling of home appliances remotely and will also enable home security against intrusion in the absence of home owner. The system uses latest wireless communication like Bluetooth, Infrared and Wi-Fi access to the system for security and automated appliance control. Home security has been a major issue where crime is increasing and everybody wants to take proper measures to prevent intrusion. System will work on different wireless communications and latest 3 of 10 mobiles uses for security purpose. The proposed system characteristics involve remote controlling of appliances, intrusion detection, system security and auto-configuration such that system automatically adjusts the system settings on running hardware support check.

2.10 IOT BASED MONITORING AND CONTROL SYSTEM FOR HOME AUTOMATION

The project proposes an efficient implementation for IoT (Internet of Things) used for monitoring and controlling the home appliances via World Wide Web. Home automation system uses the portable devices as a user interface. They can communicate with home automation network through an Internet gateway, by means of low power communication protocols like Zigbee, Wi-Fi etc. This project aims at controlling home appliances via Smartphone using Wi-Fi as communication protocol and raspberry pi as server system. The user here will move directly with the system through a web-based interface over the web, whereas home appliances like lights, fan and door lock are remotely controlled through easy website. An extra feature that enhances the facet of protection from fireplace accidents is its capability of sleuthing the smoke in order that within the event of any fireplace, associates an alerting message and an image is sent to Smartphone. The server will be interfaced with relay hardware circuits that control the appliances running at home. The communication with server allows the user to select the appropriate device. The communication with server permits the user to pick out the acceptable device. The server communicates with the corresponding relays. If the web affiliation is down or the server isn't up, the

embedded system board still will manage and operate the appliances domestically. By this we provide a climbable and price effective Home Automation system.

CHAPTER-3

PROJECT DESCRIPTIONS

3.1 EXISTING SYSTEM:

- The existing systems have certain deficiencies namely:
- Lack of an intuitive UI, high base cost, lack of a good security system.
- The existing infra-red controls present in the market are in general appliance specific and the same cannot be used interchangeably.
- Pic microcontroller used in the existing system, the dumping and programming facility in pic microcontroller is difficult.

3.2 PROPOSED SYSTEM:

- First, using the IoT connectivity, we can monitor and access our smart home easily from anywhere, which will definitely will prove to be energy efficient.
- Secondly, it acts has a helping hand for the old age and differently abled person.
- For future work, we would like to add up more controlling units that can make our smart home more intelligent that can be practically deployed in the real time situation.
- In this proposed system we can control the appliances via manual control using IOT and automated control using sensor.

3.3 BLOCK DIAGRAM:

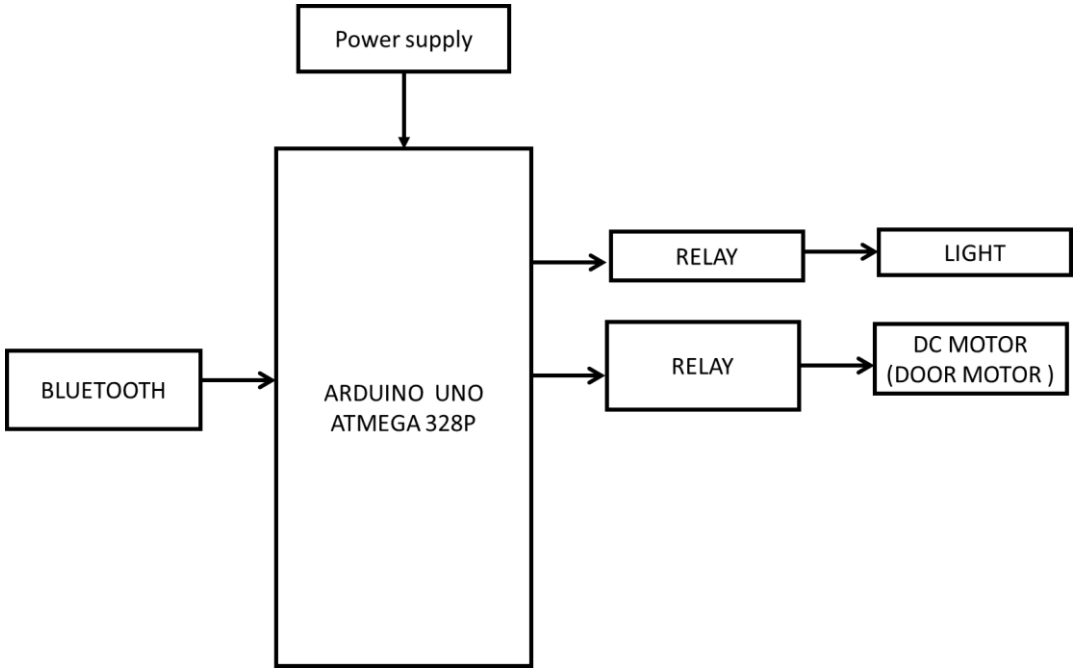


Fig 3.1 Block Diagram

3.4 HARDWARE REQUIREMENTS:

- Arduino uno
- Bluetooth
- Relay
- Led
- Dc motor
- Power supply

3.5 SOFTWARE REQUIREMENTS:

- Arduino IDE,
- Embedded C.

CHAPTER-4

HARDWARE DESRIPTIONS

4.1 ARDUINO UNO

The UNO is the best board to get started with electronics and coding. If this is your first experience tinkering with the platform, the UNO is the most robust board you can start playing with. The UNO is the most used and documented board of the whole Arduino family.

Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button.

It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

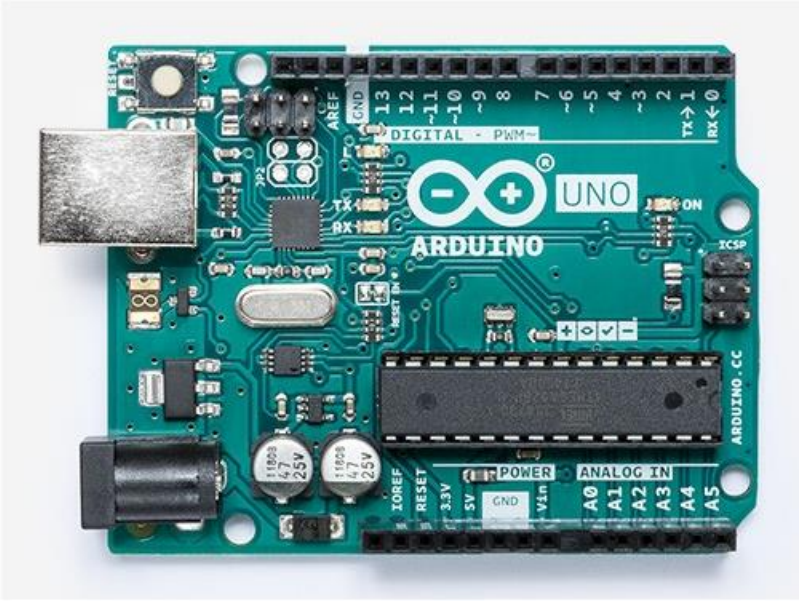


Fig 4.1 Arduino Uno

There are many versions of Arduino boards introduced in the market like Arduino Uno, Arduino Due, Arduino Leonardo, Arduino Mega, however, most common versions are

Arduino Uno and Arduino Mega. If you are planning to create a project relating to digital electronics, embedded system, robotics, or IoT, then using Arduino Uno would be the best, easy and most economical option.

4.1.1 ARDUINO BOARD

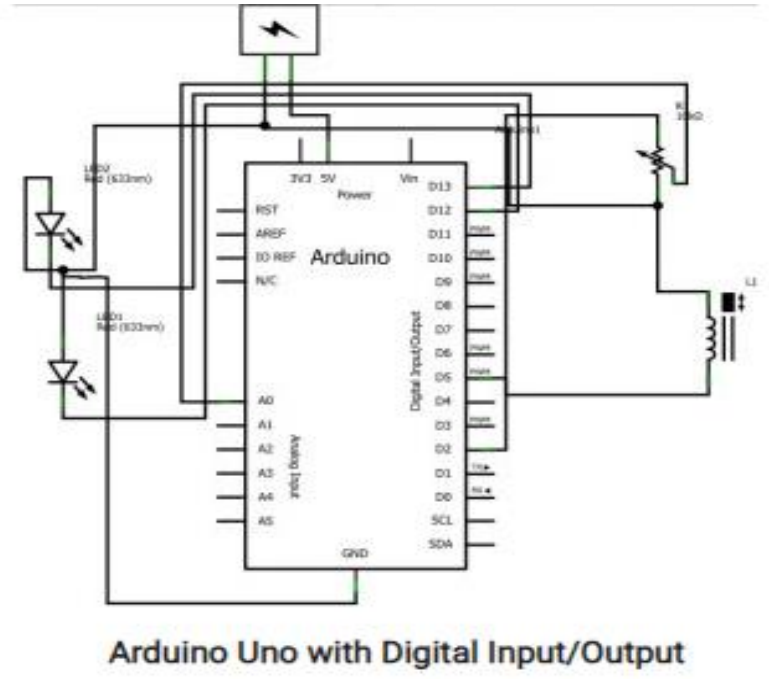
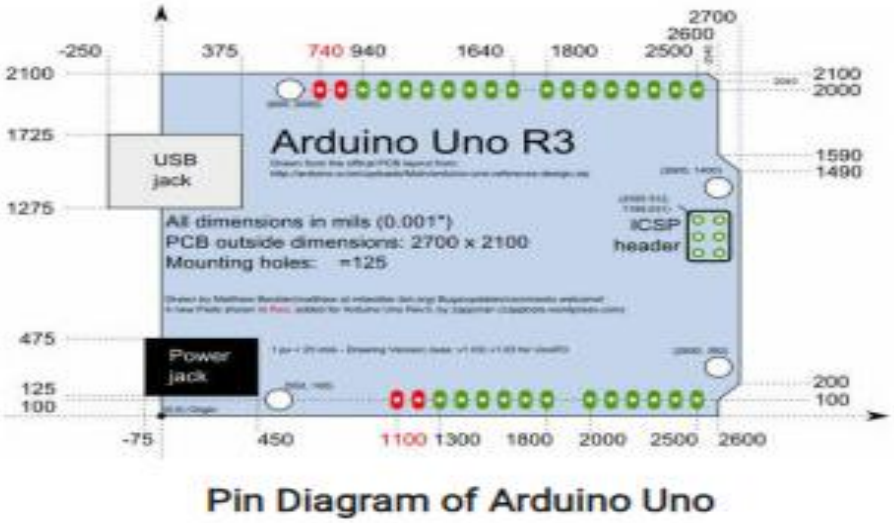


Fig 4.2 Arduino Uno with digital input/output

There are various types of Arduino boards in which many of them were third-party compatible versions. The most official versions available are the Arduino Uno R3 and the Arduino Nano V3. Both of these run a 16MHz Atmel ATmega328P 8-bit microcontroller with 32KB of flash RAM 14 digital I/O and six analogue I/O and the 32KB will not sound like as if running Windows. Arduino projects can be stand-alone or they can communicate with software on running on a computer. For e.g., Flash, Processing, Max/MSP). The board is clocked by a 16 MHz ceramic resonator and has a USB connection for power and communication. You can easily add micro-SD/SD card storage for bigger tasks.

4.1.2 FEATURES OF ARDUINO:

- It is an easy USB interface. This allows interface with USB as this is like a serial device.
- The chip on the board plugs straight into your USB port and supports on your computer as a virtual serial port. The benefit of this setup is that serial communication is an extremely easy protocol which is time-tested and USB makes connection with modern computers and makes it comfortable.
- It is easy-to-find the microcontroller brain which is the ATmega328 chip. It has more number of hardware features like timers, external and internal interrupts, PWM pins and multiple sleep modes.



Pin Diagram of Arduino Uno

Fig 4.3 Pin diagram of arduino uno

You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware.

4.2 BLUETOOTH MODULE

HC-05 module is an easy-to-use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology

and with AFH (Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

BLUETOOTH MODULE

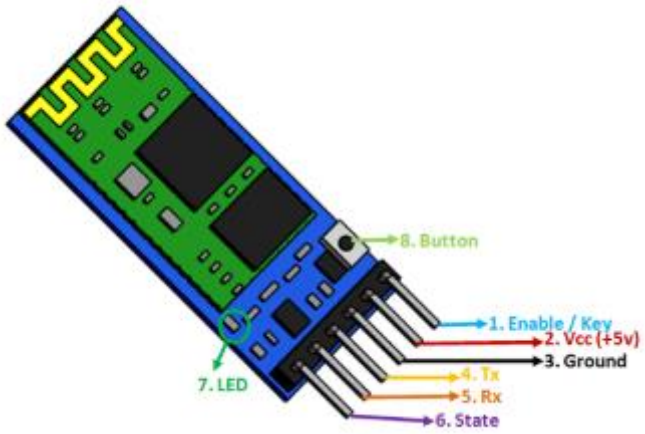


Fig 4.4 Bluetooth module

4.2.1 Hardware Features

- Typical -80dBm sensitivity
- Up to +4dBm RF transmit power
- Low Power 1.8V Operation ,1.8 to 3.6V I/O
- PIO control
- UART interface with programmable baud rate
- With integrated antenna
- With edge connector

4.2.2 Software Features

- Default Baud rate: 38400, Data bits:8, Stop bit:1,Parity:No parity, Data control: has
- Supported baud rate: 9600,19200,38400,57600,115200,230400,460800.

- Given a rising pulse in PIO0, device will be disconnected.
- Status instruction port PIO1: low-disconnected, high-connected;
- PIO10 and PIO11 can be connected to red and blue led separately. When master and slave

4.2.3 PIN CONFIGUARTION

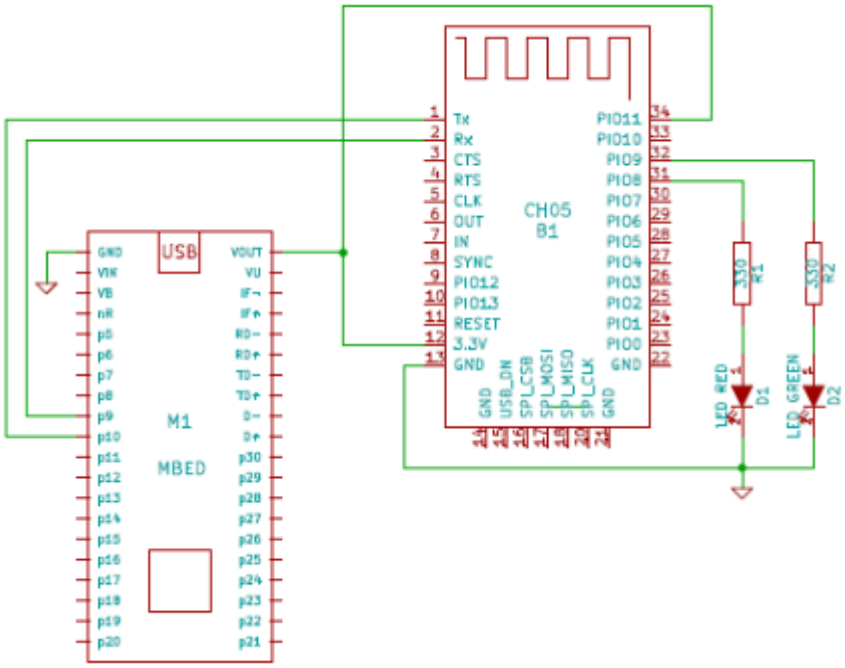


Fig 4.5 Pin configuration

are paired, red and blue led blinks 1time/2s in interval, while disconnected only blue led blinks 2times/s.

- Auto-connect to the last device on power as default.
- Permit pairing device to connect as default.
- Auto-pairing PINCODE:"0000" as default
- Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

The **HC-05** has two operating modes, one is the Data mode in which it can send and receive data from other Bluetooth devices and the other is the AT Command mode where the default device settings can be changed. We can operate the device in either of these two modes by using the key pin as explained in the pin description.

It is very easy to pair the HC-05 module with microcontrollers because it operates using the Serial Port Protocol (SPP). Simply power the module with +5V and connect the Rx pin of the module to the Tx of MCU and Tx pin of module to Rx of MCU

During power up the key pin can be grounded to enter into Command mode, if left free it will by default enter into the data mode. As soon as the module is powered you should be able to discover the Bluetooth device as “HC-05” then connect with it using the default password 1234 and start communicating with it. The name password and other default parameters can be changed by entering into the

4.2.4 **Applications**

1. Wireless communication between two microcontrollers

2. Communicate with Laptop, Desktops and mobile phones

3. Data Logging application

4. Consumer applications

5. Wireless Robots

6. Home Automation

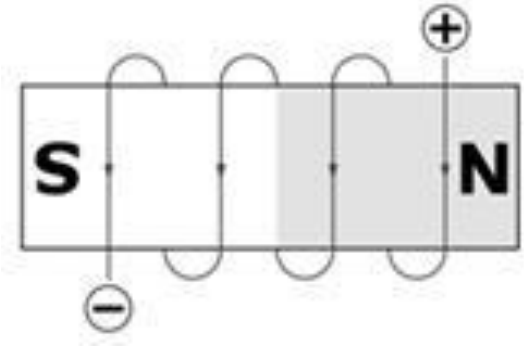
4.3 DC MOTOR:

A **DC motor** is any of a class of rotary electrical machines that converts direct current electrical energy into mechanical energy. 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.

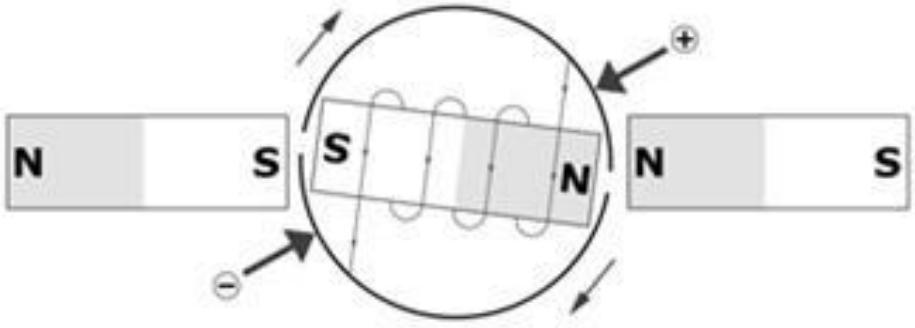
DC motors were the first type widely used, since they could be powered from existing direct-current lighting power distribution systems. A DC motor's speed can be controlled over a wide range, using either a variable supply voltage or by changing the strength of current in its field windings. Small DC motors are used in tools, toys, and appliances. The universal motor can operate on direct current but is a lightweight brushed 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.

4.3.1 WORKING:

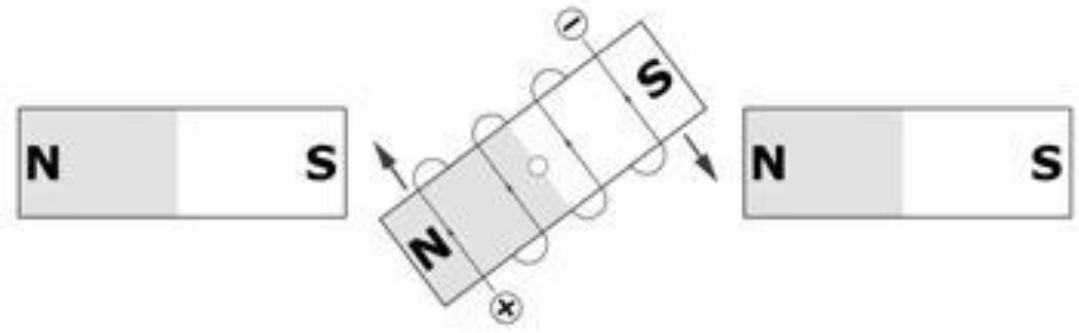
The DC motor is a machine that transforms electric energy into mechanical energy in form of rotation. Its movement is produced by the physical behavior of electromagnetism. DC motors have inductors inside, which produce the magnetic field used to generate movement. But how does this magnetic field changes if DC current is being used



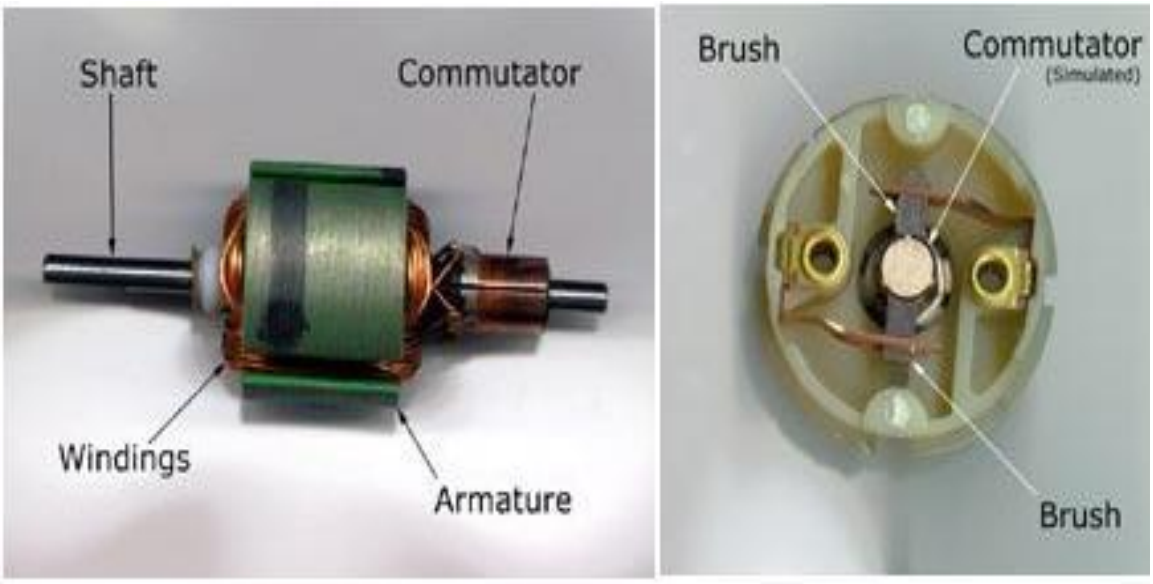
An electromagnet, which is a piece of iron wrapped with a wire coil that has voltage applied in its terminals. If two fixed magnets are added in both sides of this electromagnet, the repulsive and attractive forces will produce a torque.



Then, there are two problems to solve: feeding the current to the rotating electromagnet without the wires getting twisted, and changing the direction of the current at the appropriate time. Both of these problems are solved using two devices: a split-ring commutator, and a pair of brushes.



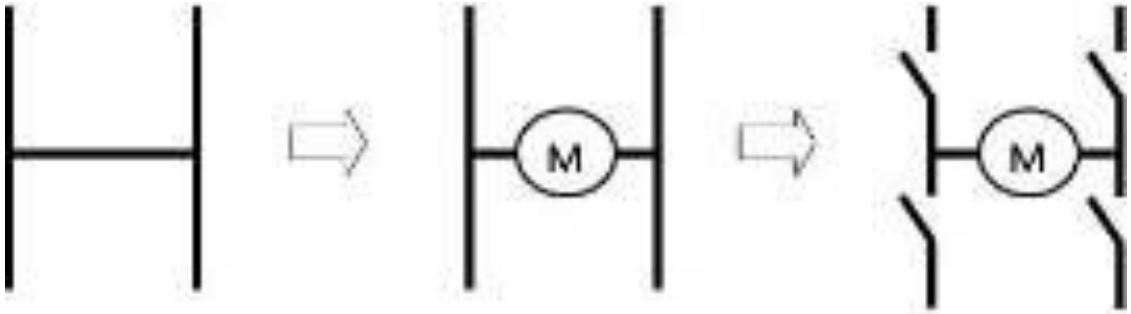
As it can be seen, the commutator has two segments which are connected to each terminal of the electromagnet, besides the two arrows are the brushes which apply electric current to the rotary electromagnet. In real DC motors it can be found three slots instead of two and two brushes.



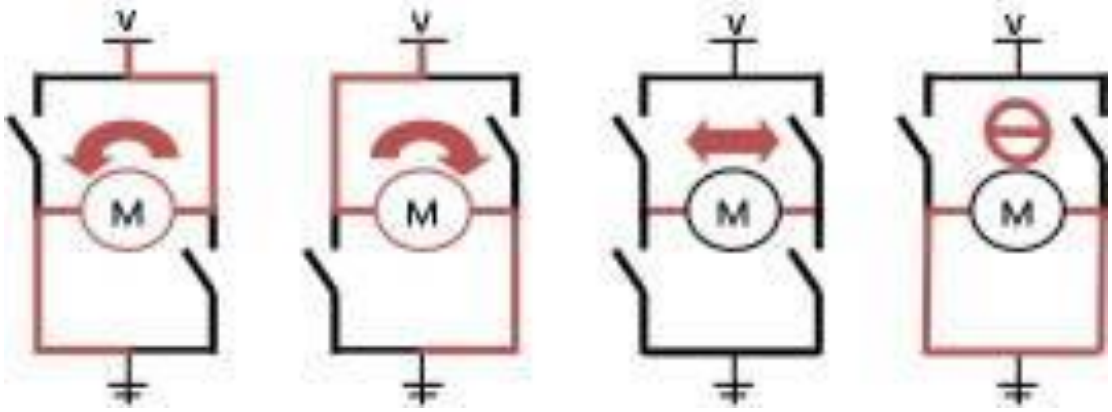
This way, as the electromagnet is moving its polarity is changing and the shaft may keep rotating. Even if it is simple and sounds that it will work great there are some issues which make these motors energy inefficient and mechanically unstable, the principal problem is due to the timing between each polarity inversion. Since polarity in the electromagnet is changed mechanically, at some velocities polarity is changing too soon, which result in reverse impulses and sometimes in changing too late, generating instantaneous “stops” in rotation. Whatever the case, these issues produce current peaks and mechanical instability.

4.3.2 CONTROL OF DC MOTOR:

DC motors have only two terminals. If you apply a voltage to these terminals the motor will run, if you invert the terminals position the motor will change its direction. If the motor is running and you suddenly disconnect both terminals the motor will keep rotating but slowing down until stopping. Finally if the motor is running and you suddenly short-circuit both terminals the motor will stop.
So there is not a third wire to control a DC motor, but knowing the previous behaviors it can be designed a way to control it, and the solution is an H-bridge.



Look at the last evolution of the DC Motor above, you can observe that there are four gates and a motor connected between them. This is the simplest H-bridge, where the four gates represent for transistors. By manipulating these gates and connecting the upper and lower terminals to a voltage supply, you can control the motor in all the behaviors as below.



4.3.3 Main Types of DC Motors

Understanding the different types of DC motors will also help you understand how they're used for different applications, and which type may apply to your application. There are 4 main types of DC motors:

1. Permanent Magnet DC Motors

The permanent magnet motor uses a permanent magnet to create field flux. This type of DC motor provides great starting torque and has good speed regulation, but torque is limited so they are typically found on low horsepower applications.

2. Series DC Motors

In a series DC motor, the field is wound with a few turns of a large wire carrying the full armature current. Typically, series DC motors create a large amount of starting torque, but cannot regulate speed and can even be damaged by running with no load.

These limitations mean that they are not a good option for variable speed drive applications.

3. Shunt DC Motors

In shunt DC motors the field is connected in parallel (shunt) with the armature windings. These motors offer great speed regulation due to the fact that the shunt field can be excited separately from the armature windings, which also offers simplified reversing controls.

4. Compound DC Motors

Compound DC motors, like shunt DC motors, have a separately excited shunt field. Compound DC motors have good starting torque but may experience control problems in variable speed drive applications.

4.4 POWER SUPPLY:

DEFINITION:

A **power supply** (sometimes known as a **power supply unit** or **PSU**) is a device or system that supplies electrical or other types of energy to an output load or group of loads. The term is most commonly applied to electrical energy supplies, less often to mechanical ones, and rarely to others.

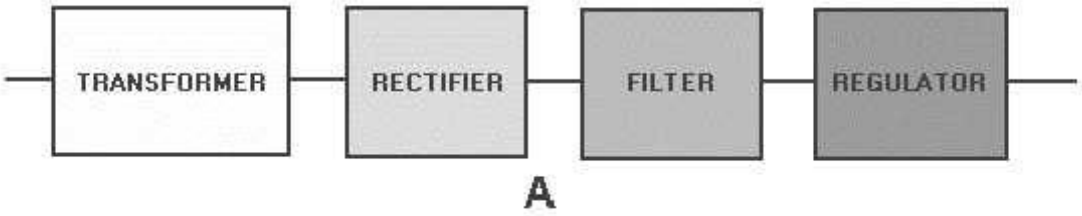


Fig 4.6 Block diagram of a basic power supply

The transformer steps up or steps down the input line voltage and isolates the power supply from the power line. The rectifier section converts the alternating current input signal to a pulsating direct current. However, as you proceed in this chapter you will learn that pulsating dc is not desirable. For this reason, a filter section is used to convert pulsating dc to a purer, more desirable form of dc voltage.

The final section, the regulator, does just what the name implies. It maintains the output of the power supply at a constant level in spite of large changes in load current or input line voltages. Now that you know what each section does, let's trace

an AC signal through the power supply. At this point you need to see how this signal is altered within each section of the power supply. Later on, in the chapter you will see how these changes take place. In view B of figure 4-1, an input signal of 115 volts AC is applied to the primary of the transformer. The transformer is a step-up transformer with a turn's ratio of 1:3. We can calculate the output for this transformer by multiplying the input voltage by the ratio of turns in the primary to the ratio of turns in the secondary; therefore, 115 volts AC´ 3 = 345 volts ac (peak-to- peak) at the output. Because each diode in the rectifier section conducts for 180 degrees of the 360-degree input, the output of the rectifier will be one-half, or approximately 173 volts of pulsating DC. The filter section, a network of resistors, capacitors, or inductors, controls the rise and fall time of the varying signal. Consequently, the signal remains at a more constant DC level. We will see the filter process more clearly in the discussion of the actual filter circuits. The output of the filter is a signal of 110 volts dc, with ac ripple riding on the dc. The reason for the lower voltage (average voltage) will be explained. The regulator maintains its output at a constant 110-volt dc level, which is used by the electronic equipment (more commonly called the load).

4.4.1 Simple 5v supply for digital circuits

- Brief description of operation: Gives out well-regulated +5V output, output current capability of 100 mA.
- Circuit protection: Built-in overheating protection shuts down output when regulator IC gets too hot.
- Circuit complexity: Very simple and easy to build.
- Circuit performance: Very stable +5V output voltage, reliable operation.
- Availability of components: Easy to get, uses only very common basic components.
- Applications: Part of electronics devices, small laboratory power supply.
- Power supply voltage: Unregulated DC 8-18V power supply.
- Power supply current: Needed output current + 5 mA.

4.4.2 Circuit description

This circuit is a small +5V power supply, which is useful when experimenting with digital electronics. Those transformers are easily available, but usually their

voltage regulation is very poor, which makes then not very usable for digital circuit experimenter unless a better regulation can be achieved in some way.

This circuit can give +5V output at about 150 mA current, but it can be increased to 1 A when good cooling is added to 7805 regulator chip. The circuit has over overload and terminal protection.

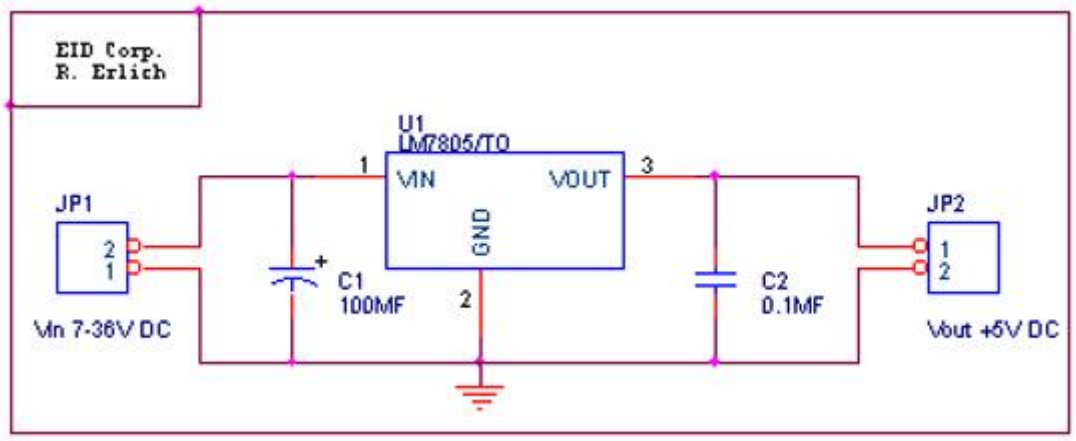


Fig 4.7 Circuit diagram

4.4.3 Circuit diagram of the power supply:

The capacitors must have enough high voltage rating to safely handle the input voltage feed to circuit. The circuit is very easy to build into a piece of Vero board.

Pinout of the 7805 regulator IC.

1. Unregulated voltage in
2. Ground
3. Regulated voltage out

COMPONENT LIST

- 7805 regulator IC.
- 100 uF electrolytic capacitor, at least 25V voltage rating.

- 10 uF electrolytic capacitor, at least 6V voltage rating.
- 100 nF ceramic or polyester capacitor.

MORE OUTPUT CURRENT

If we need more than 150 mA of output current, we can update the output current up to 1A doing the following modifications:

- Change the transformer from where we take the power to the circuit to a model which can give as much current as we need from output
- Put a heat sink to the 7805 regulators (so big that it does not overheat because of the extra losses in the regulator)

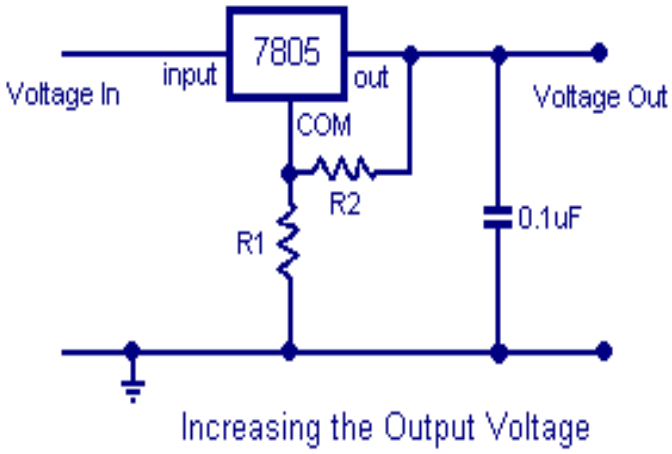


FIG 4.8 MORE OUTPUT CURRENT

OTHER OUTPUT VOLTAGES

If we need other voltages than +5V, we can modify the circuit by replacing the 7805 chips with another regulator with different output voltage from regulator 78xx chip family. The last numbers in the chip code tells the output voltage. The input voltage

must be at least 3V greater than regulator output voltage to otherwise the regulator does not work well.

4.5 LED

A **light-emitting diode** (**LED**) is a semiconductor light source that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons. This effect is called electroluminescence. The color of the light (corresponding to the energy of the photons) is determined by the energy required for electrons to cross the band gap of the semiconductor. White light is obtained by using multiple semiconductors or a layer of light-emitting phosphor on the semiconductor device.

Appearing as practical electronic components in 1962, the earliest LEDs emitted low-intensity infrared light.[8] Infrared LEDs are used in remote-control circuits, such as those used with a wide variety of consumer electronics. The first visible-light LEDs were of low intensity and limited to red. Modern LEDs are available across the visible, ultraviolet, and infrared wavelengths, with high light output.

Early LEDs were often used as indicator lamps, replacing small incandescent bulbs, and in seven-segment displays. Recent developments have produced white-light LEDs suitable for room lighting. LEDs have led to new displays and sensors, while their high switching rates are useful in advanced communications technology.

LEDs have many advantages over incandescent light sources, including lower energy consumption, longer lifetime, improved physical robustness, smaller size, and faster switching. Light-emitting diodes are used in applications as diverse as aviation lighting, automotive headlamps, advertising, general lighting, traffic signals, camera flashes, lighted wallpaper and medical devices.[9]

Unlike a laser, the color of light emitted from an LED is neither coherent nor monochromatic, but the spectrum is narrow with respect to human vision, and functionally monochromatic

4.5.1 TYPES

LEDs are made in different packages for different applications. A single or a few LED junctions may be packed in one miniature device for use as an indicator or pilot lamp.

An LED array may include controlling circuits within the same package, which may range from a simple resistor, blinking or color changing control, or an addressable controller for RGB devices. Higher-powered white-emitting devices will be mounted on heat sinks and will be used for illumination. Alphanumeric displays in dot matrix or bar formats are widely available. Special packages permit connection of LEDs to optical fibers for high-speed data communication links.

4.5.2 MINIATURE

These are mostly single-die LEDs used as indicators, and they come in various sizes from 2 mm to 8 mm, through-hole and mount packages.[97] Typical current ratings range from around 1 mA to above 20 mA. Multiple LED dies attached to a flexible backing tape form an LED strip light.

Common package shapes include round, with a domed or flat top, rectangular with a flat top (as used in bar-graph displays), and triangular or square with a flat top. The encapsulation may also be clear or tinted to improve contrast and viewing angle. Infrared devices may have a black tint to block visible light while passing infrared radiation.

Ultra-high-output LEDs are designed for viewing in direct sunlight

5 V and 12 V LEDs are ordinary miniature LEDs that have a series resistor for direct connection to a 5 V or 12 V supply.

4.5.3 HIGH POWER

High-power LEDs (HP-LEDs) or high-output LEDs (HO-LEDs) can be driven at currents from hundreds of mA to more than an ampere, compared with the tens of mA for other LEDs. Some can emit over a thousand lumens. LED power densities up to 300 W/cm^2 have been achieved. Since overheating is destructive, the HP-LEDs must be mounted on a heat sink to allow for heat dissipation. If the heat from an HP-LED is not removed, the device fails in seconds. One HP-LED can often replace an incandescent bulb in a flashlight, or be set in an array to form a powerful LED lamp

4.5.4 AC DRIVEN

LEDs developed by Seoul Semiconductor can operate on AC power without a DC converter. For each half-cycle, part of the LED emits light and part is dark, and this is reversed during the next half-cycle. The efficacy of this type of HP-LED is typically 40 lm/W.[102] A large number of LED elements in series may be able to operate directly from line voltage. In 2009, Seoul Semiconductor released a high DC voltage LED, named as 'Acrich MJT', capable of being driven from AC power with a simple controlling circuit. The low-power dissipation of these LEDs affords them more flexibility than the original AC LED design

4.5.5 Application-specific variations

Flashing

Flashing LEDs are used as attention seeking indicators without requiring external electronics. Flashing LEDs resemble standard LEDs but they contain an integrated multivibrator circuit that causes the LED to flash with a typical period of one second. In diffused lens LEDs, this circuit is visible as a small black dot. Most flashing LEDs emit light of one color, but more sophisticated devices can flash between multiple colors and even fade through a color sequence using RGB color mixing.

Bi-color

Bi-color LEDs contain two different LED emitters in one case. There are two types of these. One type consists of two dies connected to the same two leads antiparallel to each other. Current flow in one direction emits one color, and current in the opposite direction emits the other color. The other type consists of two dies with separate leads for both dies and another lead for common anode or cathode so that they can be controlled independently. The most common bi-color combination is red/traditional green, however, other available combinations include amber/traditional green, red/pure green, red/blue, and blue/pure green.

RGB Tri-color

Tri-color LEDs contain three different LED emitters in one case. Each emitter is connected to a separate lead so they can be controlled independently. A four-lead arrangement is typical with one common lead (anode or cathode) and an additional lead for each color. Others, however, have only two leads (positive and negative) and have a built-in electronic controller.

Fig 4.9 RGB-SMD-LED

RGB LEDs consist of one red, one green, and one blue LED.[104] By independently adjusting each of the three, RGB LEDs are capable of producing a wide color gamut. Unlike dedicated-color LEDs, however, these do not produce pure wavelengths. Modules may not be optimized for smooth color mixing.

Decorative-multicolor

Decorative-multicolor LEDs incorporate several emitters of different colors supplied by only two lead-out wires. Colors are switched internally by varying the supply voltage.

Alphanumeric

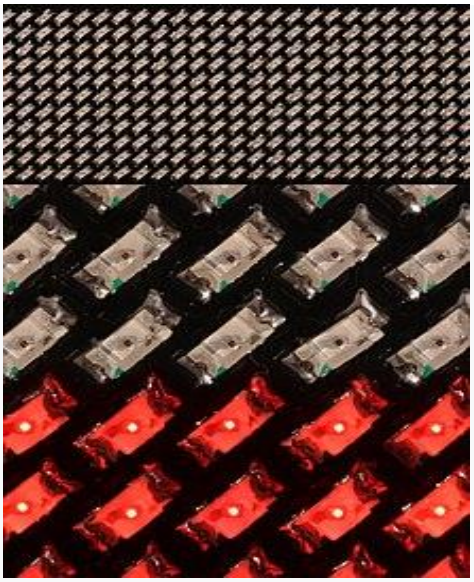


Fig 4.10 Alphanumeric

Composite image of a 11x44 LED matrix lapel name tag display using 1608/0603-type SMD LEDs. Top: A little over half of the 21x86 mm display. Center: Close-up of LEDs in ambient light. Bottom: LEDs in their own red light.

Alphanumeric LEDs are available in seven-segment, starburst, and dot-matrix format. Seven-segment displays handle all numbers and a limited set of letters. Starburst displays can display all letters. Dot-matrix displays typically use 5x7 pixels per character. Seven-segment LED displays were in widespread use in the 1970s and 1980s, but rising use of liquid crystal displays, with their lower power needs and greater display flexibility, has reduced the popularity of numeric and alphanumeric LED displays.

Digital RGB

Digital RGB addressable LEDs contain their own "smart" control electronics. In addition to power and ground, these provide connections for data-in, data-out, and sometimes a clock or strobe signal. These are connected in a daisy chain. Data sent to the first LED of the chain can control the brightness and color of each LED independently of the others. They are used where a combination of maximum control and minimum visible electronics are needed such as strings for Christmas and LED matrices. Some even have refresh rates in the kHz range, allowing for basic video applications. These devices are known by their part number (WS2812 being common) or a brand name such as NeoPixel

Filament

An LED filament consists of multiple LED chips connected in series on a common longitudinal substrate that forms a thin rod reminiscent of a traditional incandescent filament.[105] These are being used as a low-cost decorative alternative for traditional light bulbs that are being phased out in many countries. The filaments use a rather high voltage, allowing them to work efficiently with mains voltages. Often a simple rectifier and capacitive current limiting are employed to create a low-cost replacement for a traditional light bulb without the complexity of the low voltage, high current converter that single die LEDs need.[106] Usually, they are packaged in bulb similar to the lamps they were designed to replace, and filled with inert gas to remove heat efficiently.

Chip-on-board arrays

Surface-mounted LEDs are frequently produced in chip on board (COB) arrays, allowing better heat dissipation than with a single LED of comparable luminous output. The LEDs can be arranged around a cylinder, and are called "corn cob lights" because of the rows of yellow LEDs.

CHAPTER-5
SOFTWARE DESCRIPTIONS

5.1 ARDUINO IDE:

The Arduino IDE is incredibly minimalistic, yet it provides a near-complete environment for most Arduino-based projects. The top menu bar has the standard options, including “File” (new, load save, etc.), “Edit” (font, copy, paste, etc.), “Sketch” (for compiling and programming), “Tools” (useful options for testing projects), and “Help”. The middle section of the IDE is a simple text editor that where you can enter the program code. The bottom section of the IDE is dedicated to an output window that is used to see the status of the compilation, how much memory has been used, any errors that were found in the program, and various other useful messages.

Fig 5.1 Arduino IDE

Projects made using the Arduino are called sketches, and such sketches are usually written in a cut-down version of C++ (a number of C++ features are not

included). Because programming a microcontroller is somewhat different from programming a computer, there are a number of device-specific libraries (e.g., changing pin modes, output data on pins, reading analog values, and timers). This sometimes confuses users who think Arduino is programmed in an “Arduino language.” However, the Arduino is, in fact, programmed in C++. It just uses unique libraries for the device.

The 6 Buttons

While more advanced projects will take advantage of the built-in tools in the IDE, most projects will rely on the six buttons found below the menu bar.

5.1.1 THE BUTTON BARS

1. The **check mark** is used to verify your code. Click this once you have written your code.
2. The **arrow** uploads your code to the Arduino to run.
3. The **dotted paper** will create a new file.
4. The **upward arrow** is used to open an existing Arduino project.
5. The **downward arrow** is used to save the current file.
6. The far-right button is a **serial monitor**, which is useful for sending data from the Arduino to the PC for debugging purposes.

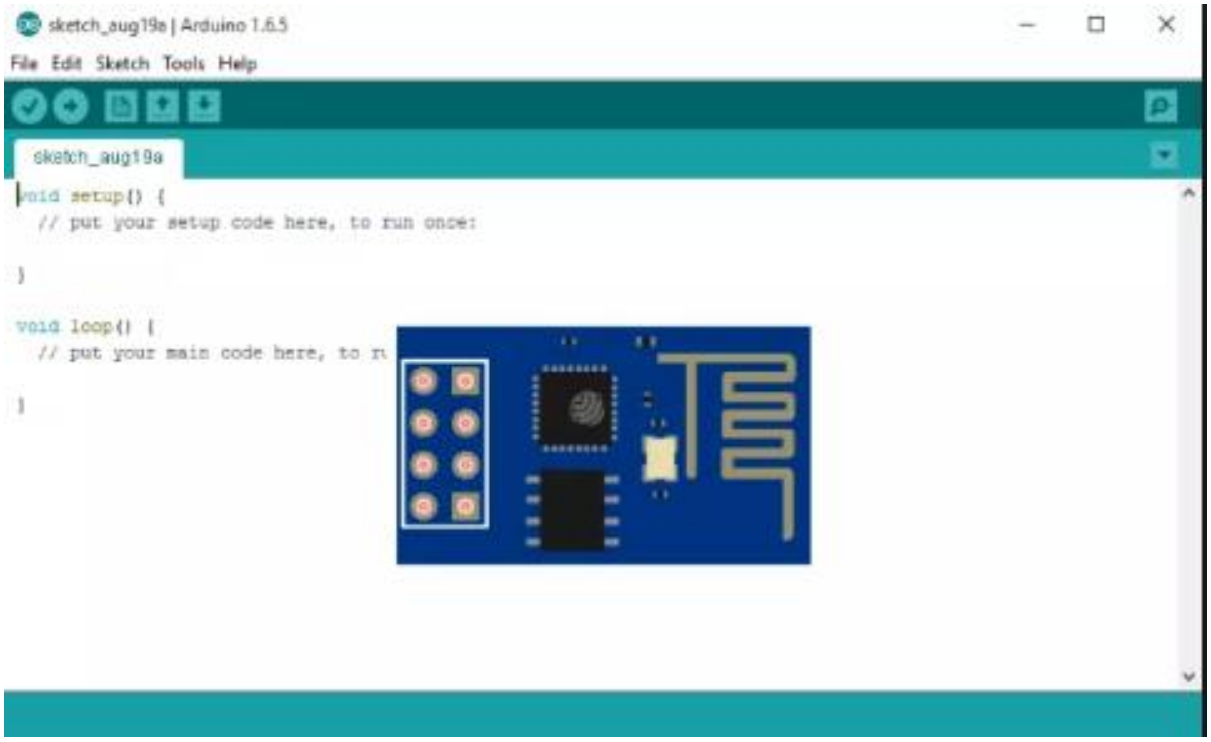


Fig 5.2 Button Bars

The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures.

User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution.[5] The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware

Projects made using the Arduino are called sketches, and such sketches are usually written in a cut-down version of C++ (a number of C++ features are not included). Because programming a microcontroller is somewhat different from programming a computer, there are a number of device-specific libraries (e.g., changing pin modes, output data on pins, reading analog values, and timers). This sometimes confuses users who think Arduino is programmed in an “Arduino

language.” However, the Arduino is, in fact, programmed in C++. It just uses unique libraries for the device.

The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution.

The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

5.2 EMBEDDED C:

Embedded C Programming is the soul of the processor functioning inside each and every embedded system we come across in our daily life, such as mobile phone, washing machine, and digital camera.
Each processor is associated with an embedded software. The first and foremost thing is the embedded software that decides functioning of the embedded system. Embedded C language is most frequently used to program the microcontroller.

Earlier, many embedded applications were developed using assembly level programming. However, they did not provide portability. This disadvantage was overcome by the advent of various high-level languages like C, Pascal, and COBOL. However, it was the C language that got extensive acceptance for embedded systems, and it continues to do so. The C code written is more reliable, scalable, and portable; and in fact, much easier to understand.

C language was developed by Dennis Ritchie in 1969. It is a collection of one or more functions, and every function is a collection of statements performing a specific task. C language is a middle-level language as it supports high-level applications and low-level applications. Before going into the details of embedded C programming, we should know about RAM memory organization.

5.2.1 Salient features of the language

- C language is a software designed with different keywords, data types, variables, constants, etc.
- Embedded C is a generic term given to a programming language written in C, which is associated with a particular hardware architecture.
- Embedded C is an extension to the C language with some additional header files. These header files may change from controller to controller.
- The microcontroller 8051 #include<reg51.h> is used.

The embedded system designers must know about the hardware architecture to write programs. These programs play prominent role in monitoring and controlling external devices. They also directly operate and use the internal architecture of the microcontroller, such as interrupt handling, timers, serial communication and other available features.

5.2.2 The basic additional features of the embedded software

Data types

The data type refers to an extensive system for declaring variables of different types like integer, character, float, etc. The embedded C software uses four data types that are used to store data in the memory.
The 'char' is used to store any single character; 'int' is used to store integer value, and 'float' is used to store any precision floating point value.
The size and range of different data types on a 32-bit machine is given in the following table. The size and range may vary on machines with different word sizes.

Keywords

There are certain words that are reserved for doing specific tasks. These words are known as keywords. They are standard and predefined in the embedded c. Keywords are always written in lowercase. These keywords must be defined before writing the main program. The basic keywords of an embedded software are given below:

Name	Funtion
sbit	Accessing of single bit
bit	Aceessing of bit addressable memory of RAM
sfr	Accessing of sfr register by another name

Fig 5.3 Keywords

Keywords

sbit: This data type is used in case of accessing a single bit of SFR register.

- Syntax: sbit variable name = SFR bit ;
- Ex: sbit a=P2^1;
- Explanation: If we assign p2.1 as ‘a’ variable, then we can use ‘a’ instead of p2.1 anywhere in the program, which reduces the complexity of the program.

Bit: This data type is used for accessing the bit addressable memory of RAM (20h-2fh).

- Syntax: bit variable name;
- Ex: bit c;
- Explanation: It is a bit sequence setting in a small data area that is used by a program to remember something.

SFR: This data type is used for accessing a SFR register by another name. All the SFR registers must be declared with capital letters.

- Syntax: SFR variable name = SFR address of SFR register;
- Ex: SFR port0=0x80;
- Explanation: If we assign 0x80 as ‘port0’, then we can use 0x80 instead of port0 anywhere in the program, which reduces the complexity of the program.

SFR Register: The SFR stands for ‘Special Function Register’. Microcontroller 8051 has 256 bytes of RAM memory. This RAM is divided into two parts: the first part of 128 bytes is used for data storage, and the other of 128 bytes is used for SFR registers. All peripheral devices like I/O ports, timers and counters are stored in the SFR register, and each element has a unique address.

5.3 The Structure of an Embedded C Program

- comments
- preprocessor directives
- global variables
- main() function

{

- local variables
- statements
- ……..
- ……..

}

- fun(1)

{

- local variables
- statements
- ……..
- ……..

}

CHAPTER-6

CONCLUSION

The project has proposed the idea of smart homes that can support a lot of home automation systems. A smart home contains a connection between wireless communication. Smart homes are a huge system that includes multiple technologies and applications that can be used to provide security and control of the home easily.

REFERENCE

[1] Vinay Sagar K N, Kusuma S M., "Home Automation Using Internet Of Things" IRJET Vol. 2, Issue no.3, Jan. 2015

[2] Vlad Bande, S.Pop, Ciascai loan, Dan Pitica, "Real Time Sensor Acquisition interfacing using MATLAB", IEEE, Dec. 2012.

[3] Kishore P Jadhav, Santosh G Bari, "Hand Gesture Based Switching Using MATLAB", IJIREEICE, Vol.4, May 2016.

[4] Sharmad Pasha, "ThinkSpeak Based Sensing and Monitoring System For IoT with MATLAB Analysis", IJNTR, Vol.2, June 2016.

[5] Angel Deborah S., "Home Automation Systems - A Study", IJCA, Vol. 116, April 2015.

[6] Prof. (Dr.) Khanna SamratVivekanand Omprakash., "WIRELESS HOME SECURITY SYSTEM WITH MOBILE", IJAET, Vol. 2, Dec. 2011.

[7] J.Chandramohan, R.Nagarajan , K.Satheeshkumar , N.Ajithkumar , P.A.Gopinath , S.Ranjithkumar, "Intelligent Smart Home Automation and Security System Using Arduino and Wi-fi", IJECS, Vol.6, March 2017.

[8] Surinder Kaur, Rashmi Singh, Neha Khairwal, Pratyk Jain, "HOME AUTOMATION AND SECURITY SYSTEM", ACII, Vol.3, July 2016.

[9] Jayashri Bangali, Arvind Shaligram, "Design and Implementation of Security Systems for Smart Home based on GSM technology", IJSH, Vol.7, 2013.

[10] M. Wu, T. J. Lu, F. Y. Ling, J. Sun, and H. Y. Du, "Research on the architecture of Internet of Things," in Proc. 3rd ICACTE, 2010.

[11] Mr. Ann Abraham Babu, Prof. Satish Kumar Varma, Prof. Rupali Nikhare , "Hand gesture recognition system for human computer interaction using contour analysis" , IJRET, Vol.4 Issue 3,March 2015.

[12] Ms. K. Thakur, Ms. A. Daphal, Prof. S. Ahankari "Monitor and control using virtual switch technology", IJERT, Vol.3 Issue 3, March 2014.

[13] Grant R. McMillan „The Technology and Applications of Gesture Based Control"in 2018.

[14] Raj Kumar Bansal, Ashok Kumar Goyal, Manoj Kumar Sharma „MATLAB and Its Applications In Engineering".

[15] Daniel Thalmann, "Gesture Recognition Motion Capture, Motion Retargeting, and Action Recognition", EPFL – VRlab.

[16] Manoj Harsule, Prashant Ingole, "Home appliances control using hand gesture-based command", conference paper june, 2013.

[17] D. Meisner, C. M. Sadler, L. A. Barroso, W. D. Weber and T. F. Wenisch, "Power management of online data-intensive services," Computer Architecture (ISCA), 2011 38th Annual International Symposium on, San Jose, CA, 2011, pp. 319-330.

[18] R. K. Pradhan and M. A. Gregory, "Access network energy efficient dynamic power scaling," Telecommunication Networks and Applications Conference (ATNAC), 2012Australasian, Brisbane, QLD, 2012, pp. 1-5.

[19] N. Sinha, K. E. Pujitha and J. S. R. Alex, "Xively based sensing and monitoring system for IoT," Computer Communication and Informatics (ICCCI), 2015 International Conference on, Coimbatore, 2015, pp. 1-6.

[20] ThingSpeak – an API and Web Service for the Internet of Things, a published paper 2016 Aug 31.