An Experimental Study on Voice controlled Smart Locking System

Course Title : Project And Thesis

Course Code : ME 400

Supervised By:

Mr. Shuvo Biswas Topu
Lecturer
Department of Mechanical Engineering
Sonargaon University (SU)

Submitted By:

Md. Hosan Ali
Id No : BME1602009025

Belal Hossain
Id No : BME1602009030

Mir Md. Ariful Islam
Id No : BME1602009032

Md. Rafiqul Islam
Id No : BME1602009033

Md. Mir Kashem
Id No : BME1601008129

SONARGAON UNIVERSITY

147/1 , Green Road, Tejgaon , Dhaka – 1215

February 2020

DECLARATION

We hereby, declare that the work presented in this project is the outcome of the investigation and research work performed by us under the supervision of **Mr. Shuvo Biswas Topu, Lecturer, Department of Mechanical Engineering, Sonargaon university (SU).** We also declare that no part of this project there of has been or is being submitted elsewhere for the award of any degree

----

Mr. Shuvo Biswas Topu

Lecturer

Department of Mechanical Engineering

Sonargaon University (su)

APPROVAL

This is to certify that the project on **"An Experimental Study on Voice Controlled Smart Locking System"** by (Md. Hosan Ali. Id No- BME1602009025, Belal Hossain. Id No- BME1602009030, Mir Md. Ariful Islam. Id No- BME1602009032, Md. Rafiqul Islam. Id No- BME1602009033, Md. Mir Kashem. Id No- BME1601008129

Mr. Shuvo Biswas Topu

Lecturer

Department of Mechanical Engineering

Sonargaon University (su)

ABSTRACT

Voice is utilized in this project for control doors or gates or any other security system. The reason for selecting a voice is because it's simply being reproduced by a human. Besides that, usage of voice provides a smart system that will be effective and convenient to be used. It allows the consumer to control his home security & it facilitates several conditions. It involves the management and automation of security. The voice controlled smart locking system has the ability to control the locking system like gate or door using a Smartphone Application with Bluetooth Wireless Technology , Microcontroller & Servo motor. Smartphone application and Bluetooth wireless technology take input voice data from humans, then this data sends to microcontroller. Microcontroller analyzing the audible signal to determine whether it matches the voiceprint stored in memory. The microprocessor executes the command instruction to control the function of the device if a match has been found. The servo motor is used to lock/ unlock the security system with the help of a microcontroller command. The design is a low cost, flexible and using modern technology. The system has been built and operated successfully

ACKNOWLEDGEMENTS

Firstly we give thanks to Almighty Allah from the deep of my hearts. we would like to express our sincere gratitude to our honorable supervisor, **Mr. Shuvo Biswas Topu , Lecturer, Department of Mechanical Engineering, Sonargaon University**, who inspired us in every moment. We are thankful to him for his continuous encouragement, kind co-operation, and scholastic guidance all along with the project work. He has always been extremely generous with his time, knowledge and ideas and allowed us great freedom in this research. we want to convey our thankfulness to **Md. Mostofa Hossain, Associate Professor & Head of The Department of Mechanical Engineering (SU)**, for his help, support, & constant encouragement.We are also convey our thankfulness to the **Vice Chancellor** of Sonargaon University, for his overall support, help & constant encouragement. We are express my humble gratitude to all teachers of the Department of Mechanical Engineering for their support in numerous ways throughout this project work. We are also grateful to the authors whose valuable research papers and books we have considered as a reference in this project paper. Apart from that, we would like to thank our entire friends for sharing knowledge; information and helping to make this project a success. Also thanks for lending us some tools and equipment. Finally, We would like to thanks our parents who have given me tremendous inspirations and supports. Without their mental and financial supports, We would not able to complete my project.

"The Authors"

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

A voice controlled smart locking system permits controlling house or office or industry appliances like the door, gate, box, etc. It conjointly provides an associate emergency system and residential security. It allows the consumer to control his home security & it facilitates several conditions. As an example, if the consumer is permitted to controlling door, gates or any other security system, if he got a smart security system, therefore, several manual actions are replaced by a voice controlled smart locking system that reduces human efforts and time-saving. During this paper, a new style and totally different security appliances are given. Many different electric, mechanical applications are controlled by using the arduino nano microcontroller Based Voice Controlled Smart Locking System. This system is used to lock or unlock the security system using a wireless technology & microcontroller with servo motor

1.2 OBJECTIVES

- The objective of this project is to develop and construct a voice controlled smart locking system for home automation
- To design and implement a secure locking system

1.3 CONCEPT OF THE PROJECT

This system is used to control all the appliances connected to the microcontroller. The methodology of the proposed system is mainly divided into three steps. In the initial step an Android application interfaces with the Bluetooth module. After that within the second step, the microcontroller receives the signal that has been causing by the Bluetooth module. Then the microcontroller sends the activation signal to the servo motor. In the last step servo motor activate the security system

CHAPTER 2

LITERATURE REVIEW

2.1 BACKGROUND

The automated Security system performs tasks to control appliance or home security. Nowadays, the appliance available in the market is getting increased as days passed. Thus, the Voice-controlled smart locking system getting more and more attention. From a long time ago mostly the controlling is done manually such as walking to the door and lock or unlock it. But as time passed the arrival or remote control that give the user an alternative way to control such an appliance without the need for the user to walk to the appliance. From the above, there are some more advancements in the controlling method that been on research. The advancement mention is using the voice to act as the controlling medium to initiate or to control the security. Taking an example such as a security door is only can be activated with the voice of the authorized personal only if then the door will be unlocked. In this project, the voice is also used as the medium to perform the controlling of the security system in the home. This concept is more like a comparison between the source and the data stored in memory (the voice stored during the training process). The way of this concept function is when a user speaks out some command, with then the voice is captured through voice application and Bluetooth module as the input. Once the voice is captured, the usage of a decoding system that will convert the analog (voice) to digital (binary signal). Later, the input voice is compared with the data stored in the memory early before the testing. The output of the comparison is the voice matched with any of the command trained and a certain signal is produced as the input for the controlling system. There are a few projects done locally by other students and researchers regarding the voice-related research. Those are an automated voice-based home navigation system, voice-activated wheelchair, voice-controlled smart house, voice-controlled lighting system. From all the above of the discussion on the previous project that has been done by other students. In this project, the application of the concept and theory used in the above project is applied. Thus this leads to a project that has the capability to produce a system that has the application of voice as the controlling method for controlling the security system at the home. In this system, voice is used as the primary input to the system. It was designed for smart security system where lock/unlock done by voice

2.2 ADVANTAGES

- It is very easy to use and time saving
- All the control would be in your hands by using this voice controlled smart locking system.
- This project will give the ability to observe all the security system within the communication limit through Bluetooth.
- It is very good for physically disabled people

2.3 APPLICATIONS

- Applications are uses as remote control.
- This project used as a security system in applications like homes, hostels, industries.
- We are used to be lazy and creative, the automation system works for us.
- Children do not get any electric shock.
- Protection from all kinds of human hazards.

2.4 SCOPES

- Voice controlled smart locking system makes houses or offices or industry security system secured and time saving
- Portable Android application to use it frequently in hand.
- Bluetooth technology which might be utilized in wireless affiliation for cellular phones and residential appliances. Bluetooth technology offers an associate economical methodology for dominant home security. It's an occasional value and a secured technology

2.5 LIMITATIONS

- Bluetooth technology utilizes in this home automation system, which has a rage ten to twenty meters so the control cannot be achieved from outside this compass.
- The Arduino application may be connected and disconnected the Bluetooth connection frequently
- If we use voice technology off times, we could expertise some physical discomfort and vocal issues.

CHAPTER 3

METHODOLOGY OF THE PROJECT

3.1 SYSTEM REVIEWS :

3.1.1 GENERAL BLOCK DIAGRAM

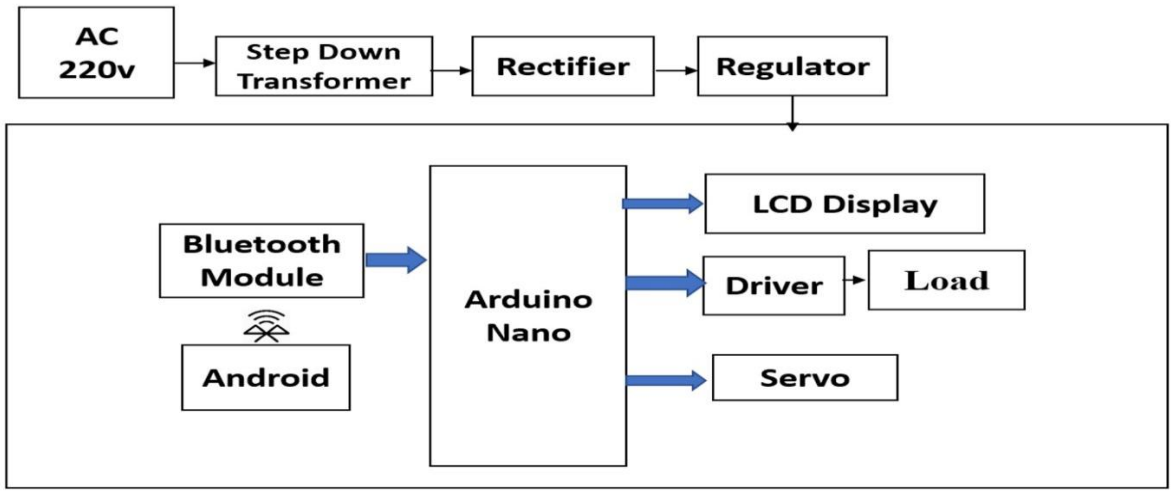

Fig. 2.1 General Block Diagram

3.1.2 BLOCK DIAGRAM DESCRIPTION

The block diagram of the Voice Controlled Smart Locking System as shown in fig.2.1, is using the Arduino Nano, we can operate the Bluetooth module, relay module. When the dedicated command is available, then the Bluetooth module (HC-05) sends a digital output to the control unit. At this time the control units (Arduino Nano) will send commands to servo motor for door lock/unlock

3.1.3 CIRCUIT DIAGRAM

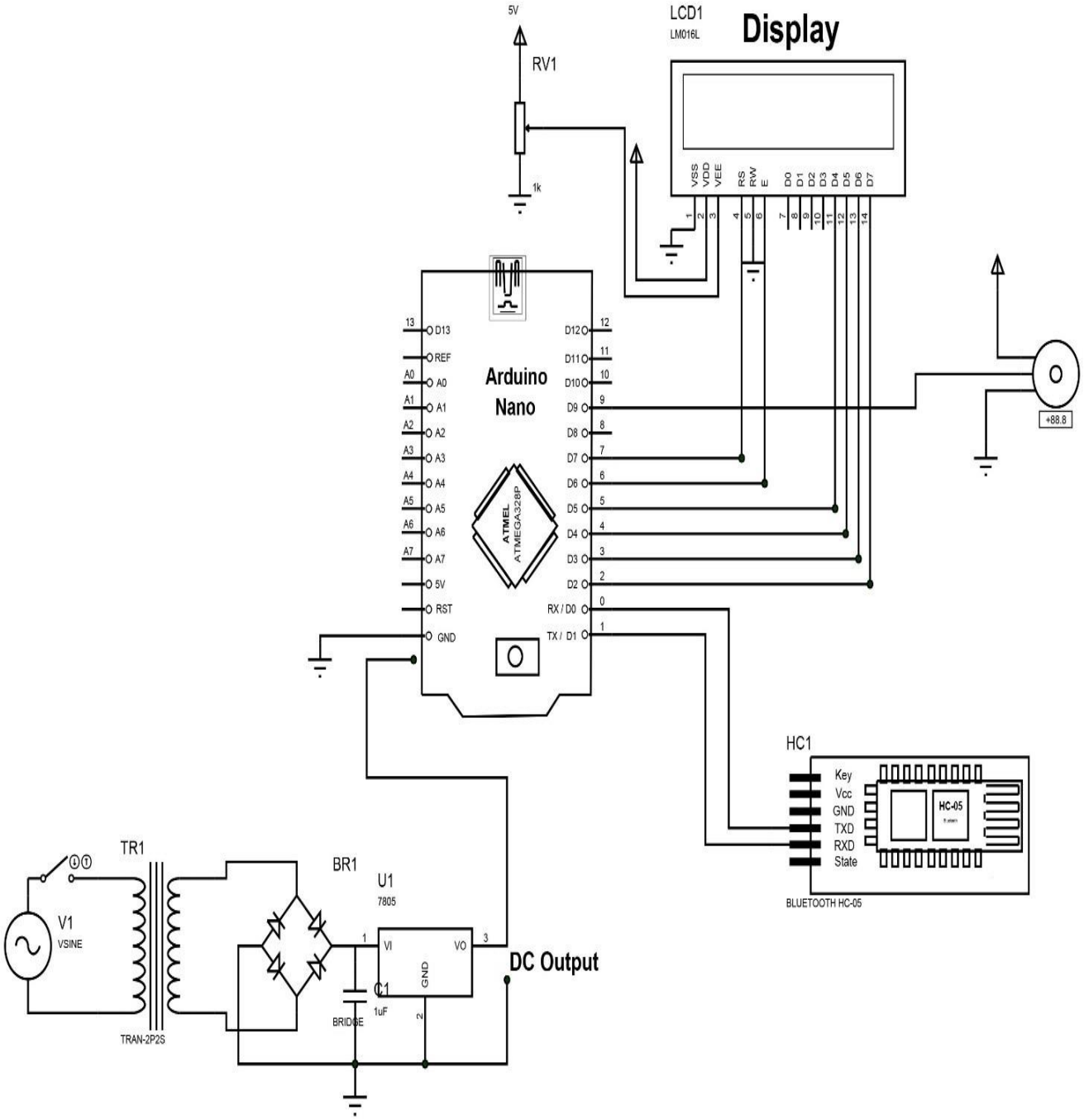

Fig. 2.2 Circuit Diagram

3.1.4 WORKING PROCESS OF OUR CIRCUIT

The power supply section of this circuit consists of a 220V from the mains supply which is step down by the transformer to 12V and is fed to a rectifier. The output is then fed to a filter to remove any AC components present even after rectification. The voltage is regulated using a voltage regulator (LM7805 regulator) to 5V which supplies power to the rest of the circuit. The Arduino is connected to the HC05 Bluetooth Module, which to connect Arduino with a voice command Apps. In the programing section, we implement some voice command, so when we give instruction via apps, the command will analysis by google then it will be sent to microcontroller, Microcontroller analyzing the audible signal to determine whether it matches the voiceprint stored in memory. The microprocessor executes the command instruction to control the function of the device if a match has been found. The servo motor is used to lock/ unlock the security system with the help of microcontroller command

3.1.5 LIST OF COMPONENTS USED IN CIRCUIT

NO	COMPONENT NAME	QUANTITY	USED
01	Bluetooth Module (HC-05)	**01**	To direct communication.
02	Microcontroller-ATmega328P(Arduino NANO)	**01**	To Control the System.
03	Capacitor ,Resistor,Voltage regulator	**04**	Safety
04	Arduino compiler	**01**	To compile code.
05	Arduino voice control app	**01**	To giving command.
06	Step Down Transformer	**01**	Step down the voltage from 220v to 6v 1A
07	LCD Display (16*2)	**01**	Showing Text
08	Bridge Rectifier	**01**	
09	Servo Motor (SG-90)	**01**	Lock and unlock.
10	PCB	**01**	
11	Others		Design &Decoration

Table 2.1 list of the component we used

3.2 COMPONENT DESCRIPTION :

3.2.1 DESCRIPTION OF BLUETOOTH MODULE (HC-05)

Nowadays, everything goes wireless. Phones, computers, game controllers, everything. Wireless technology has allowed us to use electronic devices with AN unbound freedom that simply is not possible with awkward cables hanging off our device. There are several kinds of wireless properties like Wi-FI, Cellular Data, Zigbee, however, one among the foremost in style and widely used wireless protocols is Bluetooth. Whilst the Bluetooth HC-5 standard was declared earlier this year (2018), we'll be taking a look at using the Bluetooth 4.2 standard because it is most typically on the market at the instant.

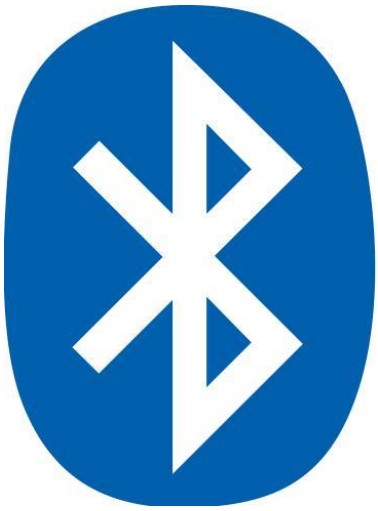

Fig 3.1 Logo of Bluetooth.

HC-05 module is a simple to utilize Bluetooth SPP (Serial Port Protocol) module, intended for straightforward remote sequential association setup. The HC-05 Bluetooth Module can be utilized in a Master or Slave setup, making it an incredible answer for remote correspondence. This sequential port Bluetooth module is a completely qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with a complete 2.4GHz radio handset and baseband. It utilizes CSR Blue-center 04-External single chip Bluetooth framework with CMOS innovation and with AFH (Adaptive Frequency Hopping Feature). The Bluetooth module HC-05 is a MASTER/SLAVE module. As a matter, of course, the processing plant setting is SLAVE. The Role of the module (Master or Slave) can be designed just by AT COMMANDS. The slave modules can't start an association with another Bluetooth gadget, however, they can acknowledge associations. The ace module can start a connection with different gadgets.

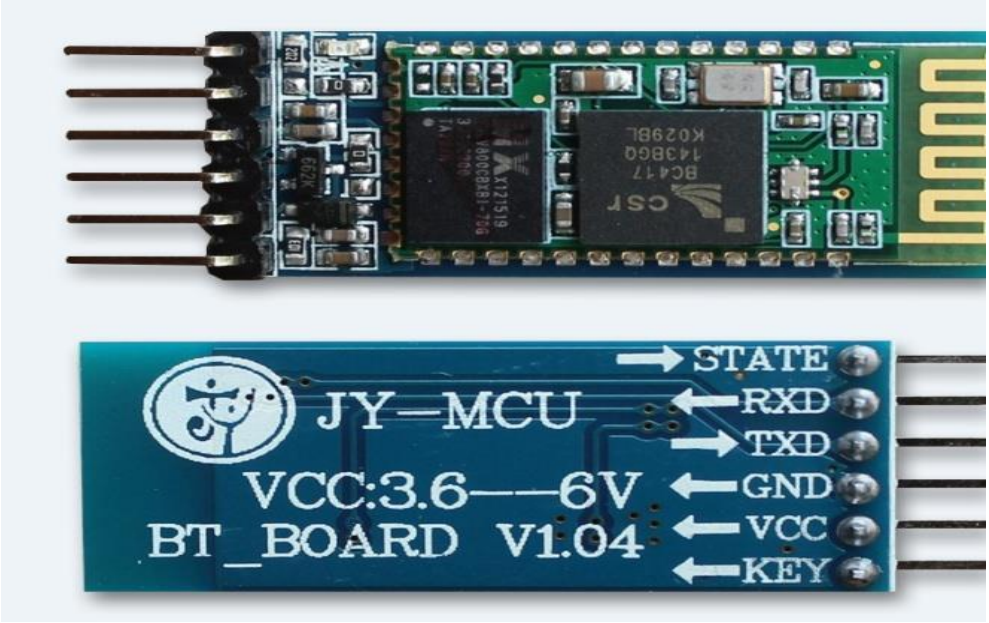

Fig. 3.1.1 Bluetooth Module (HC-05)

3.2.2 FEATURES OF BLUETOOTH MODULE (HC-05)

- Typical -80dBm sensitivity.
- Up to +4dBm RF transmit power.
- 3.3 to 5 V Input/Output.
- PIO (Programmable Input/Output) control.
- With integrated antenna.
- With edge connector.
- DefaultData bits 8, Stop bit 1,Baud rate 9500, Parity No parity.
- Automatically connect to the last device when power on.
- Automatically connect with paired device
- Default PINCODE is “1234” or auto-pairing.

3.2.3 PIN DESCRIPTION OF HC-05

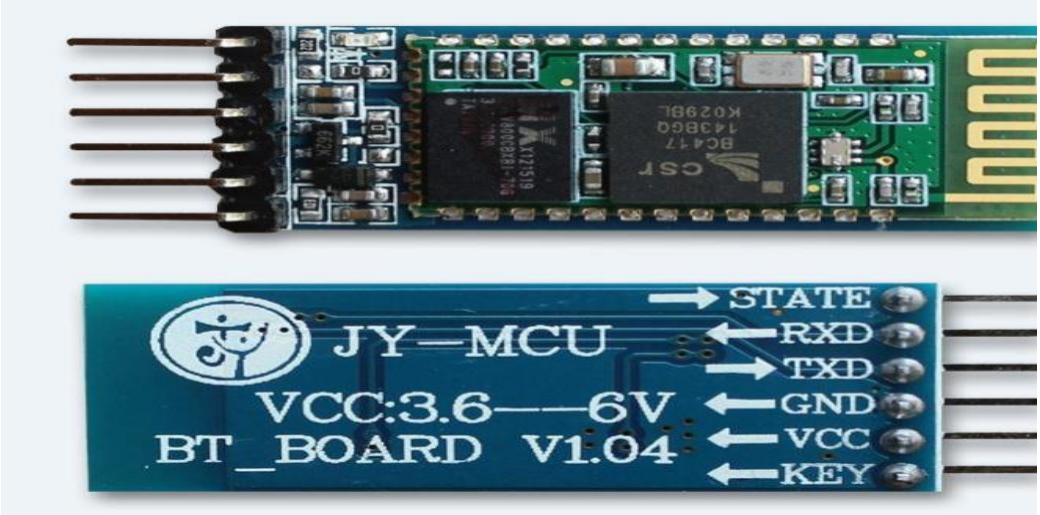

Fig 3.1.2 Pins of HC-05.

SL.	PIN NAME	DESCRIPTION
1	+5V or 3.3V	+5V supply or Power Pin.
2	TX	The Data/Command to be transmitted is sent through this pin.
3	RX	The Received data is read from this pin.
4	KEY/EN	Input pin, which alters module between the Data mode and the AT Command mode.
5	STATE	Output pin, the state of the module is indicated through this pin.
6	GROUND	0V / GND or Power Pin

Table 3.1 Pins Description of Bluetooth Module

3.2.4 BASIC DESCRIPTION OF CONTROLLER UNIT

In the controller unit, we are using the Arduino equipment board (with AVR microcontroller). With the assistance of the Arduino 1.6.8 software system, we are able to simply program AVR IC, as our requirement. Arduino is an open-source electronics prototyping platform supported versatile, easy-to-use hardware and software system. It's meant for artists, designers, hobbyists, and anyone fascinated by making interactive objects or environments. The microcontroller on the board is programmed exploitation the Arduino artificial language (based on Wiring) and also the Arduino development setting (based on Processing). The Arduino NANO could be a microcontroller board that supported the ATmega-328. Fourteen digital I/O pins in this device (which half-dozen will be utilized as Pulse width modulation outputs), Six analog inputs, A 16 MHz ceramic resonator, a USB port, an impact jack, an ICSP header, and a reset push. It contains everything required to support the microcontroller, easily connect it to a laptop with a USB link or power it with an AC-to-DC adapter or battery to begin.

3.2.5 TECHNICAL SPECIFICATION OF ARDUINO NANO

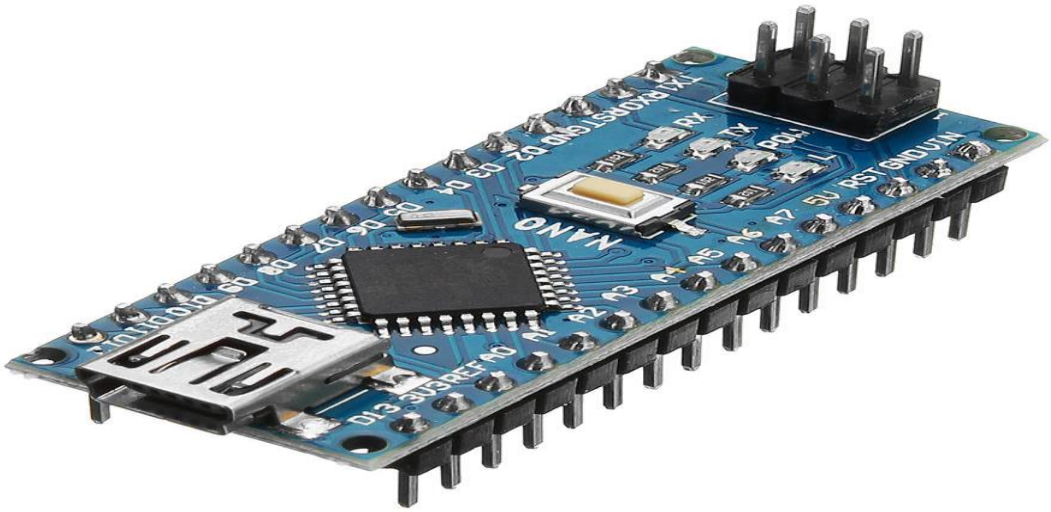

Fig 3.3(a) ArduinoNANO

.

Microcontroller	ATmega328p
Operating Voltage	5V
Supply Voltage (recommended)	7-12V
Max Supply Voltage (not recommended)	20V
Digital INPUT/OUTPUT Pins	Fourteen (of which six provide PW output)
Analog Input Pins	Six
DC Current per INPUT/OUTPUT Pin	40 Ma
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328p) of which 0.5 utilized by boot loader
SRAM	2 KB (ATmega328p)

Table 3.2 Technical specification of ArduinoNANO

3.2.6 DESCRIPTION OF MICROCONTROLLER ATmega328p

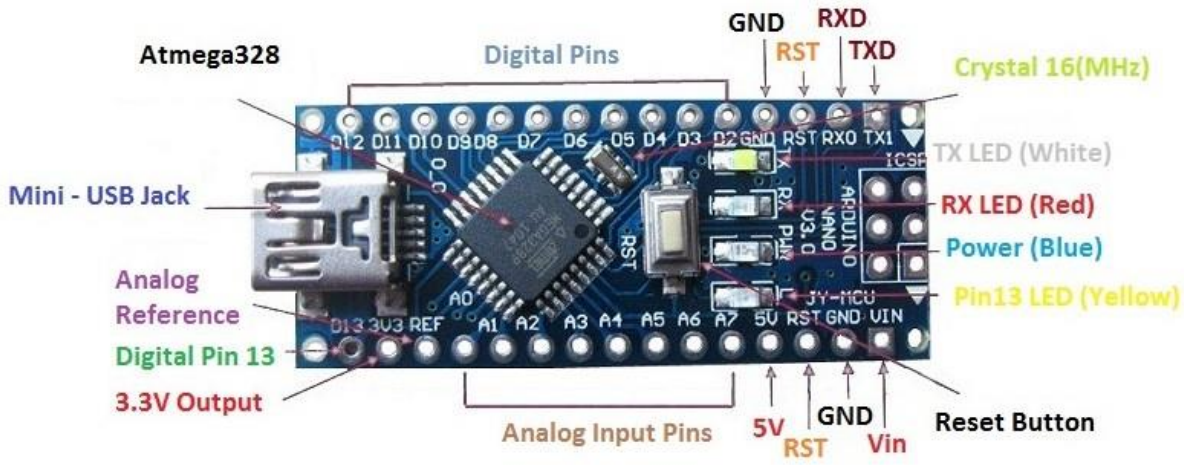

Fig 3.3(b) Arduino atmega328

The Arduino NANO could be a microcontroller board that supported the ATmega-328p (Datasheet). Fourteen digital I/O pins in this device (which half-dozen will be utilized as Pulse width modulation outputs), Six analog inputs, A 16 MHz ceramic resonator, a USB port, an impact jack, an ICSP header, and a reset push. It contains everything required to support the microcontroller, easily connect it to a laptop with a USB link or power it with an AC-to-DC adapter or battery to begin. The NANO varies from every previous board in that it doesn't utilize the FTDI USB-to-serial driver chip. Rather, it includes the Atmega328p modified as a USB-to-sequential converter. "NANO" means "One" in Italian and is called to mark the upcoming release of Arduino 1.0. The NANO and version 1.0 are going to be the reference form of Arduino, moving forward. The NANO is that the latest in an exceedingly series of USB Arduino boards and also the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino board.

3.2.7 BLOCK DIAGRAM OF MICROCONTROLLER – (ATmega328p)

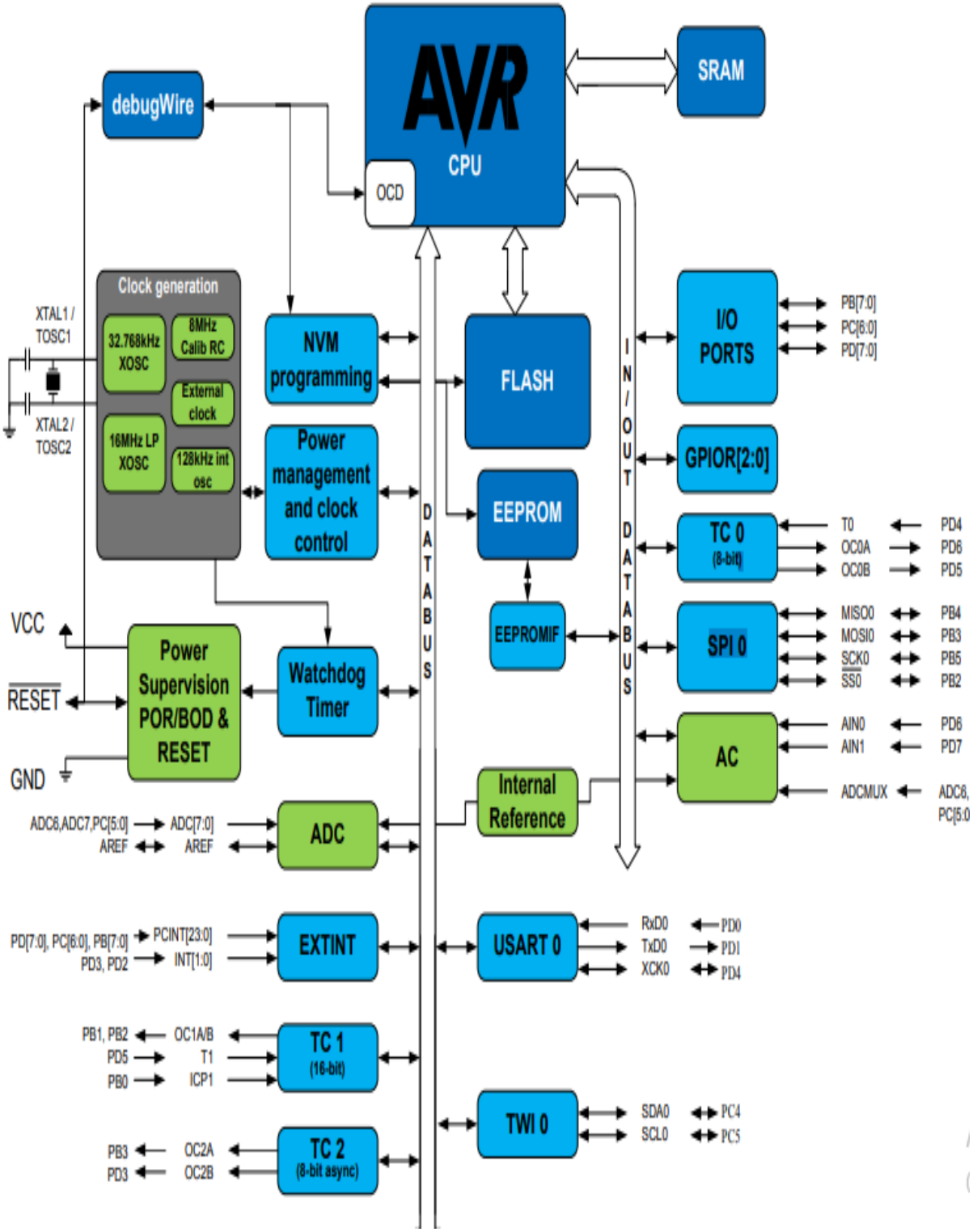

Fig. 3.3.1 Block Diagram of Microcontroller – (Atmega328p)

3.2.8 PIN CONFIGURATIONS OF MICROCONTROLLER (ATmega328p)

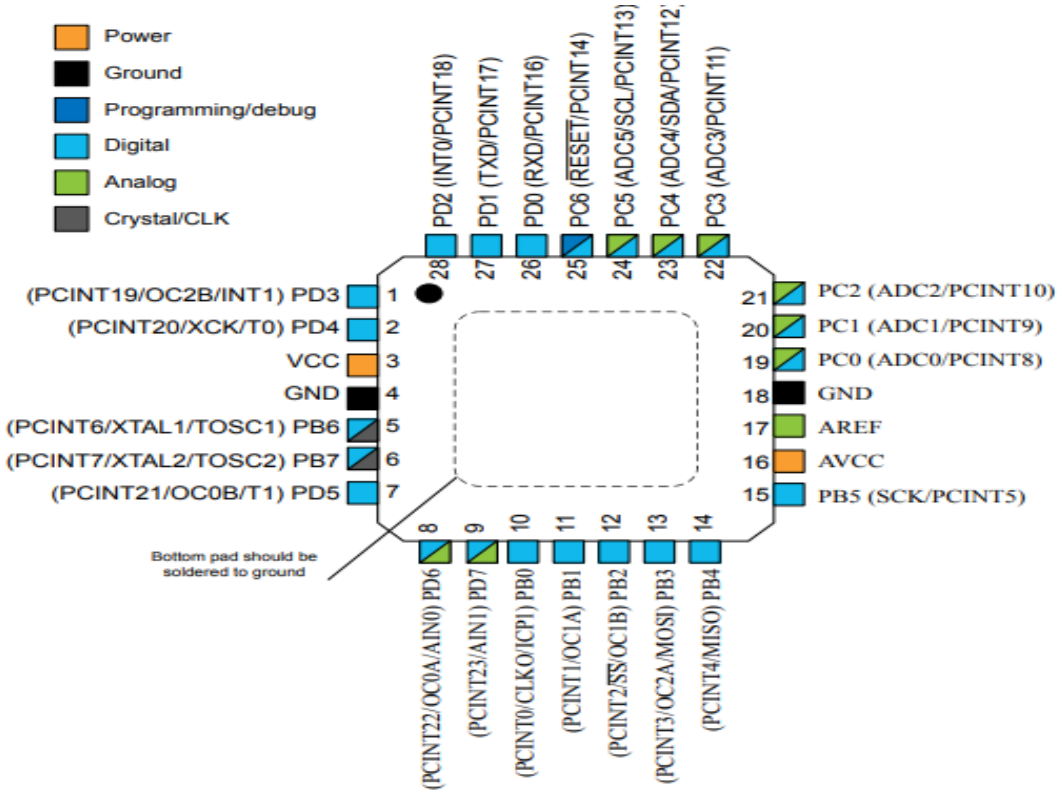

Fig. 3.3.2 Pin Configurations of Microcontroller – (Atmega328)

Pin description

The underneath gives a depiction for every one of the pins, alongside their capacity. The four pins are as below:

VIN: The input voltage to the Arduino board when it's utilizing an external power supply (as against five volts from the USB association or another regulated power source). We are able to provide voltage through this pin, or, if supply voltage via thepower jack, access it through this pin.

5V: The regulated power supply utilize to power the microcontroller and different elements on the board. This will come back either from VIN via an on-board regulator, or be provided by USB or another regulated 5V supply.

3.3V:A 3.3V supply generated by the on-board regulator. Highest current draw is 50 mA

GND (Ground pins)

Input & Output of Arduino NANO

Each of the fourteen digital pins on the Mega can be utilized as an input or output, using pin Mode, digital Write, and digital Read functions. They operate at 5 volts. Every pin will be provided or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 ohms. Additionally, some pins have specialized functions.

Serial: Serial 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega328p USB-to-TTL Serial chip.

External Interrupts:2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.

PWM:3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analog Write() function.

SPI : 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

LED: There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

I 2C 20 (SDA) and 21 (SCL): The ATmega328 additionally supports I2C (TWI) and SPI communication. The Arduino Software (IDE) includes a Wire library to simplify the use of the I2C bus. There are a couple of other pins on the board.

AREF: The reference voltage for the simple sources of info (analog Inputs). Utilized with simple Reference.

Reset: Bring this line low to reset the microcontroller. Normally used to add a reset button to shields which block the one on the board.

Memory: The ATmega328 has 32 KB of flash memory for storing code (of which 0.5 KB is utilized for the boot loader), 2 KB of SRAM and 1 KB of EEPROM (which will be read and written with the EEPROM library).

3.2.9 TRANSFORMER

A transformer is an electrical device used to change the value of an alternating voltage. Transformers are widely used in electrical work. They are encountered daily, in industrial, commercial and domestic situations. They vary in size from miniature units used in electronics to huge units used in power stations. The efficient transmission and distribution of electricity throughout the country would be impossible without the use of power transformers.Transformers are also used for safety reasons on construction sites when using power tools and in domestic bathroom situations in shaver units. They are used in doorbell operation and also to power electronic equipment, battery chargers, televisions, computers, alarm systems, etc. Transformers vary considerably in construction, size and shape depending on their application. All transformers rely on the principle of mutual inductance for their operation. Mutual inductance was discussed in detail in Unit 2.1.6 Magnetism, Electromagnetism, and Electromagnetic Induction. See Figure 3.4

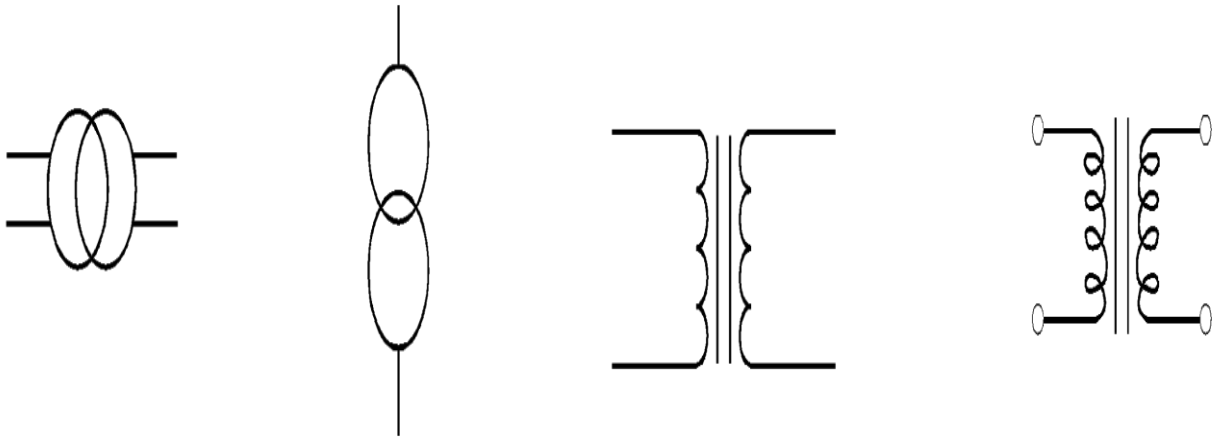

Figure 3.4 : Transformer Symbols

3.2.10 TRANSFORMER CONSTRUCTION

A transformer consists of two coils of wire called windings, which are wound onto a common iron core. The wire used in the two windings, primary and secondary, is coated with an insulating varnish. Both coils are wound onto, but insulated from the iron core. See Figure 3.4.1

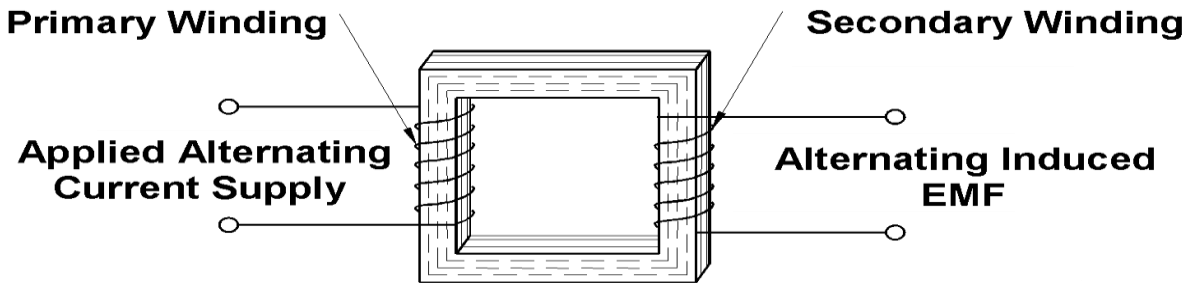

Fig 3.4.1 Construction of Transformer

Transformer Principle

When a conductor or coil is moved in a stationary magnetic field it cuts the lines of magnetic flux and an EMF is induced in the conductor or coil. This same principle also applies when a conductor is held stationary and the magnetic flux is made to change or vary. Now consider an alternating current applied to a stationary coil. A magnetic field will build up and collapse in the coil, continually rising and falling in harmony with the applied AC current as shown in Figure 3.4..2.

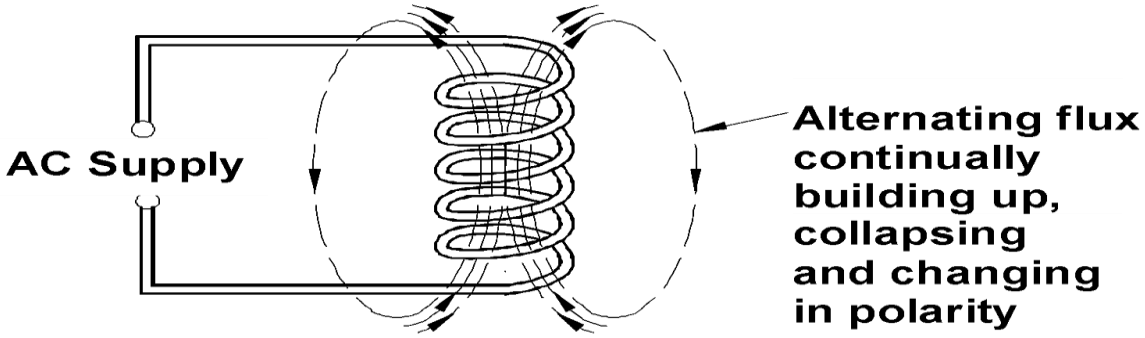

Fig3.4.2 Transformer Primary Coil..

If coil 1 and coil 2 are mounted on an iron core the magnetic flux around both coils will be concentrated. This arrangement of coils and an iron core form the complete device known as a transformer. See Figure 3.4.3

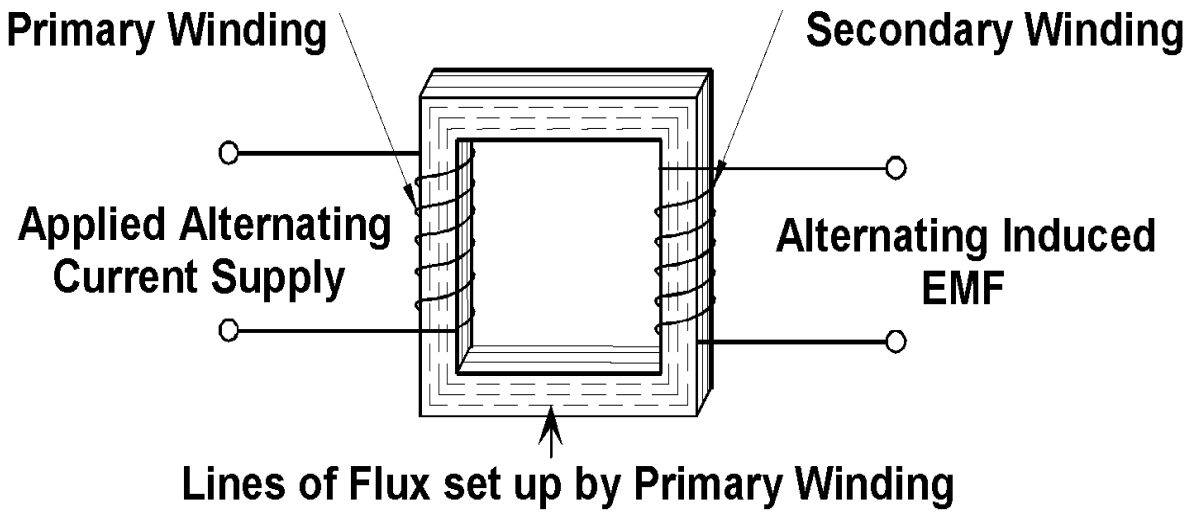

Fig 3.4.3 Lines Of Flux set up by primary Winding.

The input coil of a transformer is fed from the AC supply and is called the primary winding. The output coil, to which the load is connected, is called the secondary winding. It is important to remember that there is no electrical connection between the primary winding and the secondary winding of a transformer. The only common link between the two windings is the magnetic field.

3.2.11 FULL WAVE RECTIFIER

The first building block in the dc power supply is the full wave rectifier. The purpose of the full wave rectifier (FWR) is to create a rectified ac output from a sinusoidal ac input signal. It does this by using the nonlinear conductivity characteristics of diodes to direct the path of the current.

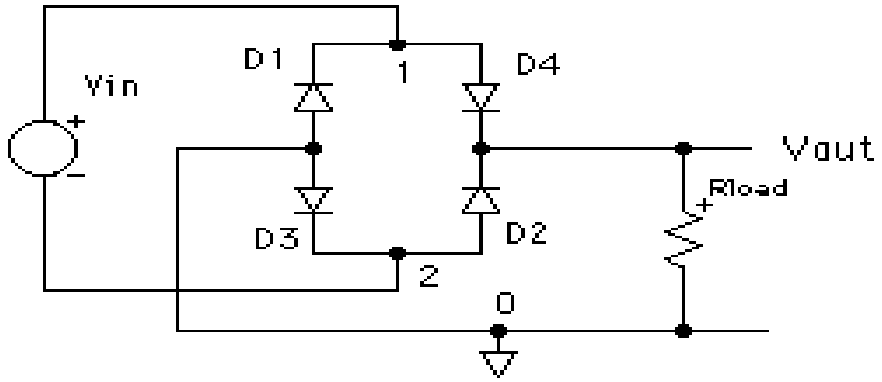

Fig 3.4.4 Common four-diode bridge configuration for the FWR.

Filtered Full Wave Rectifier

The filtered full wave rectifier is created from the FWR by adding a capacitor across the output. See Figure 3.4.5

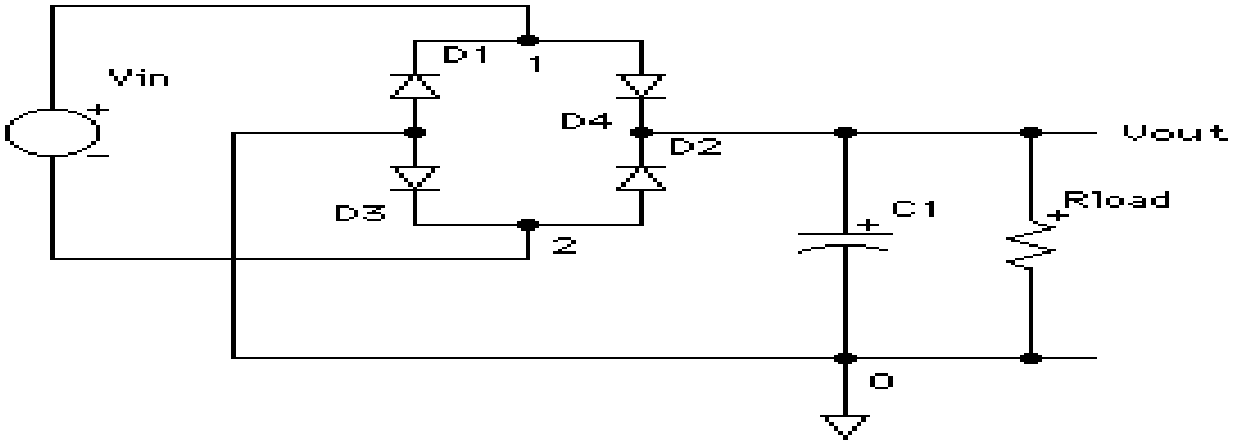

Figure 3.4.5: Filtered full wave rectifier

The result of the addition of a capacitor is a smoothing of the FWR output. The output is now a pulsating dc, with a peak to peak variation called ripple. The magnitude of the ripple depends on the input voltage magnitude and frequency, the filter capacitance, and the load resistance.To describe the source of the voltage ripple, consider the performance of the filtered full wave rectifier above. The input to the rectifier is a sinewave of frequency f. Let Vi be the full wave rectified signal input to the filter stage of the rectifier and Vo be the output. Vi can be approximated as the absolute value of the rectifier input, with frequency 2f. In the time period from T0 to T1, the diode D1 (or D3, depending on the phase of the signal) is forward biased since Vi > VC1 (approximate the forward biased diode as a short circuit). The capacitor C1 charges and the voltage across the load R increases. From T1 to T2, the diodes D1 and D2 are reverse biased (open circuit) because Vcap > Vi, and the capacitor discharges through the load R with a time constant of RC seconds.

3.2.12 CAPACITOR

When there is a potential difference across the conductors (e.g., when a capacitor is attached across a battery), an electric field develops across the dielectric, causing positive charge (+Q) to collect on one plate and negative charge (-Q) to collect on the other plate. If a battery has been attached to a capacitor for a sufficient amount of time, no current can flow through the capacitor. However, if an accelerating or alternating voltage is applied across the leads of the capacitor, a displacement current can flow. See Figure 3.4.7

Figure 3.4.7: Capacitor

Capacitors are widely used in electronic circuits for blocking direct current while allowing alternating current to pass. In analog filter networks, they smooth the output of power supplies. In resonant circuits, they tune radios to particular frequencies. In electric power transmission systems, they stabilize voltage and power flow.

3.2.13 RESISTOR

A resistor is a passive two-terminal electrical component that implements electrical resistance as a circuit element. Resistors act to reduce current flow, and, at some time, act to lower voltage levels within circuits. Resistors may have fixed resistances or variable resistances, such as those founding thermostats, visitors, trimmers, photoresistors, hamsters and potentiometers. The current through a resistor is in direct proportion to the voltage across the resistor's terminals. This relationship is represented by Ohm's law. See Figure 3.4.8

Figure 3.4.8: Resistor

Theory of Operation

The behavior of an ideal resistor is dictated by the relationship specified by Ohm ‘slaw:

$$V = I.R$$

Ohm's law states that the voltage (V) across a resistor is proportional to the current(I), where the constant of proportionality is the resistance (R). Equivalently,

Ohm's law can be stated,

$$I = V/R$$

This formulation states that the current (I) is proportional to the voltage (V) and inversely proportional to the resistance (R). This is directly used in practical computations. For example, if a 300-ohm resistor is attached across the terminals of a12 volt battery, then a current of12 / 300 = 0.04 amperes flows through that resistor.

3.2.14 VOLTAGE REGULATOR LM7805

All voltage sources cannot able to give fixed output due to fluctuations in the circuit. For getting constant and steady output, the voltage regulators are implemented. The integrated circuits which are used for the regulation of voltage are termed as voltage regulator ICs.The voltage regulator IC 7805 is actually a member of the 78xx series of voltage regulator ICs. It is a fixed linear voltage regulator. The xx present in 78xx represents the value of the fixed output voltage that the particular IC provides. For 7805 IC, it is +5V DC regulated power supply. This regulator IC also adds a provision for a heat sink. The input voltage to this voltage regulator can be up to 35V, and this IC can give a constant 5V for any value of input less than or equal to 35V which is the threshold limit.

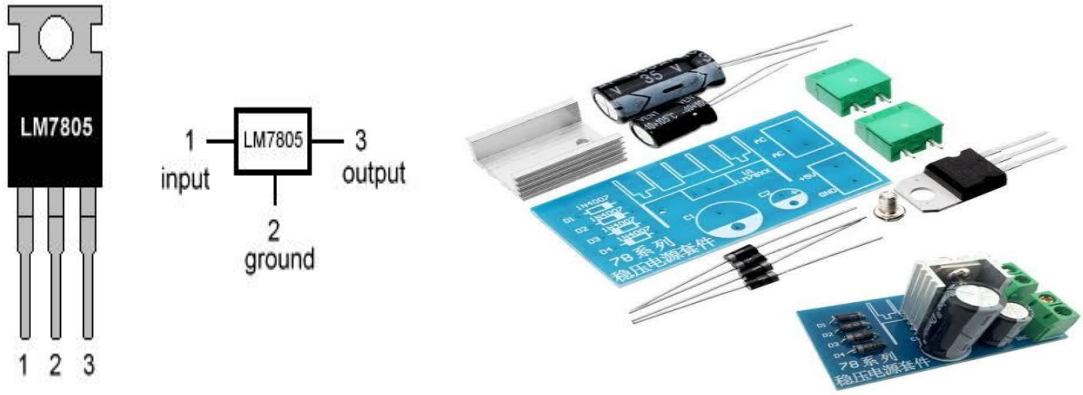

Figure 3.4.9 : Voltage regulator LM7805

3.2.15 DESCRIPTION OF LCD DISPLAY 16X2

A liquid crystal display (LCD) could be a thin, flat display device created from any range of color or monochrome pixels arrayed ahead of a light supply or reflector. every pixel consists of a column of liquid molecules suspended between 2 clear electrodes, and 2 polarizing filters, the axes of the polarity of which are perpendicular to each other. The LCD display is not enabled, data lines are tri-state and they do not interfere with the operation of the microcontroller. Data can be placed at any location on the LCD. For 16×2 LCD, the address locations are

POSITION		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
ADDRESS	LINE1	00	01	02	03	04	05	06	07	40	41	42	43	44	45	46	47

Table 3.6.0 Address locations for a 1x16 line LCD

3.2.16 SHAPE AND SIZES

Even restricted to character-based modules, there's still a good style of shapes and sizes on the market. Line lengths of 8, 16,20,24,32 and 40 characters are all standard, in one, two and four line versions.

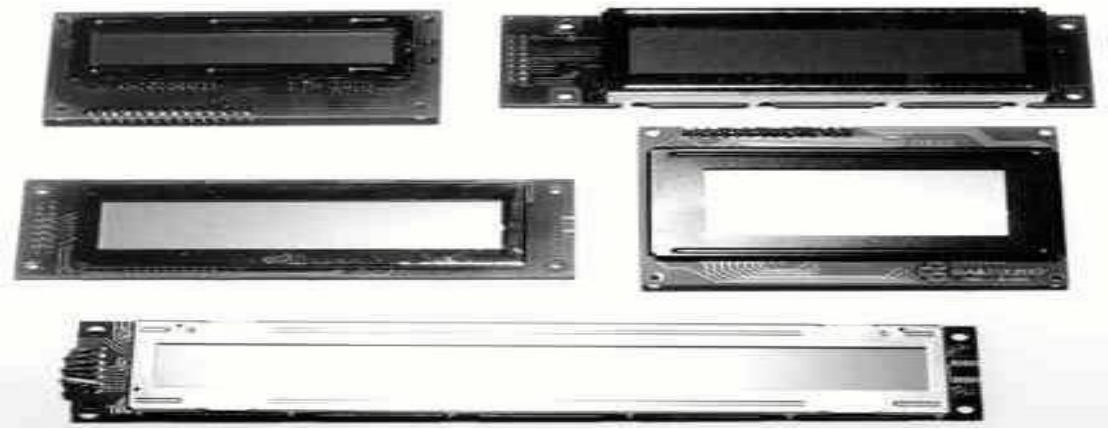

Fig 3.6.1 LCD Different Models.

Totally different LC technologies exist. “Supertwist” types, for example, offer improved contrast and viewing angle over the older “twisted nematic” types. Some modules are on the market with backlighting, in order that they'll be viewed in dimly-lit conditions. The rear lighting is also either “electro-luminescent”, requiring a high voltage electrical converter circuit, or easy LED illumination.

3.2.17 PIN DESCRIPTION

Most of the LCDs with 1 controller has 14 Pins and LCDs with 2 controller has 16 Pins (two extra pins in each for back-light LCD connections).

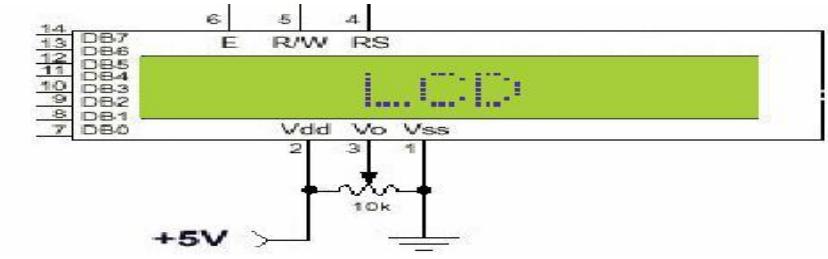

Fig 3.6.2 Pin Diagram of 16x2 line LCD.

PIN	SYMBOL	FUNCTION
1	GND	Power Supply(GND)
2	Vcc	Power Supply(+5V)
3	Vo	Contrast Adjust
4	RS	Instruction/Data Register Select
5	RW	Data Bus Line
6	E	Enable
7-14	DB0-DB7	8-bit data pins
15	A	Power supply for LED (+)
16	K	Power supply for LED (-)

Table 3.4 Pin Description of LCD

3.2.18 CONTROL LINES

EN: Line is named "Enable." This control line is used to inform the LCD that you are sending it information. To send information to the LCD, our program should make sure this line is low (0) and then set the other two control lines and/or place information on the data bus. Once the opposite lines are fully prepared, bring EN high (1) and sit up for the minimum quantity of time needed by the LCD datasheet (this varies from LCD to LCD), and end by bringing it low (0) again.

RS: Line is named the "Register Select" line. When RS is low (0), the data is to be treated as a command or special instruction (such as clear screen, position cursor, etc.). When RS is high (1), the data being sent is text data which should be displayed on the screen. For example, to display the letter "T" on the screen you would set RS high.

RW: Line is named the "Read/Write" control line. When RW is low (0), the information on the data bus is being written to the LCD. When RW is high (1), the program is effectively querying (or reading) the LCD. Only one instruction ("Get LCD status") is a read command. All others are writing commands, so RW will almost always below. Finally, the data bus consists of 4 or 8 lines (depending on the mode of operation selected by the user). In the case of an 8-bit data bus, the lines are referred to as DB0, DB1, DB2, DB3, DB4, DB5, DB6, and DB7.

3.2.19 SERVO MOTOR (SG 90)

A servo motor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity, and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. ... Servomotors are used in applications such as robotics, CNC machinery or automated manufacturing.

Servos are controlled by sending an electrical pulse of variable width or pulse width modulation (PWM), through the control wire. There is a minimum pulse, a maximum pulse, and a repetition rate. A servo motor can usually only turn 90° in either direction for a total of 180° movement.

Fig 3.8 .1: Servo motor

There are lots of servo motors available in the market and each one has its own specialty and applications. The following two paragraphs will help you identify the right type of servo motor for your project/system.

Most of the hobby Servo motors operate from 4.8V to 6.5V, the higher the voltage higher the torque we can achieve, but most commonly they are operated at +5V. Almost all hobby servo motors can rotate only from 0° to 180° due to their gear arrangement so make sure your project can live with the half circle if no, you can prefer for a 0° to 350° motor or modify the motor to make a full circle. The gears in the motors are easily subjected to wear and tear, so if your application requires stronger and long running motors you can go with metal gears or just stick with normal plastic gear.

Next comes the most important parameter, which is the torque at which the motor operates. Again there are many choices here but the commonly available one is the 2.5kg/cm torque which comes with the SG90 Motor. This 2.5kg/cm torque means that the motor can pull a weight of 2.5kg when it is suspended at a distance of 1cm.

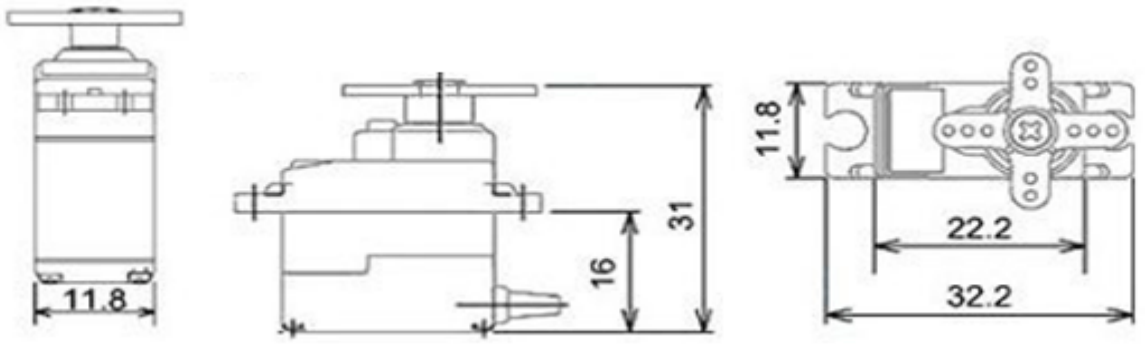

Fig 3.8.2 : Servo motor Dimentions

Servo Motor Features

- Operating Voltage is +5V typically
- Torque: 2.5kg/cm
- Operating speed is 0.1s/50°
- Gear Type: Plastic
- Rotation : 0°-180°
- Weight of motor: 9gm
- Package includes gear horns and screws

Fig 3.8.3 : Servo motor all parts

3.2.20 WIRES

Fig 3.9(a) Female to Male Jumper Wire. Fig 3.9(b) Male to Male jumper wire.

In this project, we need couples of Female to Male and Male to Male jumper wire. That is for connecting the Bluetooth module to the ArduinoNANO board. And we need some 220v carried wire to connect load in the relay module.

Fig 3.9.1 220v carried wire.

3.2.21 POWER SUPPLY

An AC adapter, AC/DC adapter, or AC/DC converter is a type of external power supply, typically enclosed during a case like an AC plug. In the power supply section, we use one step down transformer to step down the voltage from 220-volt ac to 9-volt dc. The output of the transformer is further connected to the two diode circuits. Here two diode work as a full wave rectifier circuit. The output of the full wave rectifier is now filtered by the capacitor. Capacitor converts the pulsating dc into smooth dc with the help of charging and discharging effect. The output of the capacitor is now regulated by the IC 7805 regulator. IC 7805 provides a 5-volt regulation to the circuit and provides regulated 5-volt power supply. The output of the regulator is currently once more filter by the capacitor. In the output of the capacitor, we use one resistor and one led in series to produce a visible indication to the circuit.

Fig 3.10 12V Power supply.

3.2.22 LOCK / UNLOCK SYSTEM

Servo Motor is connected With Door. Here We Show how to control Door lock/unlock. This is not fixed, it can vary in different places and different areas of use.

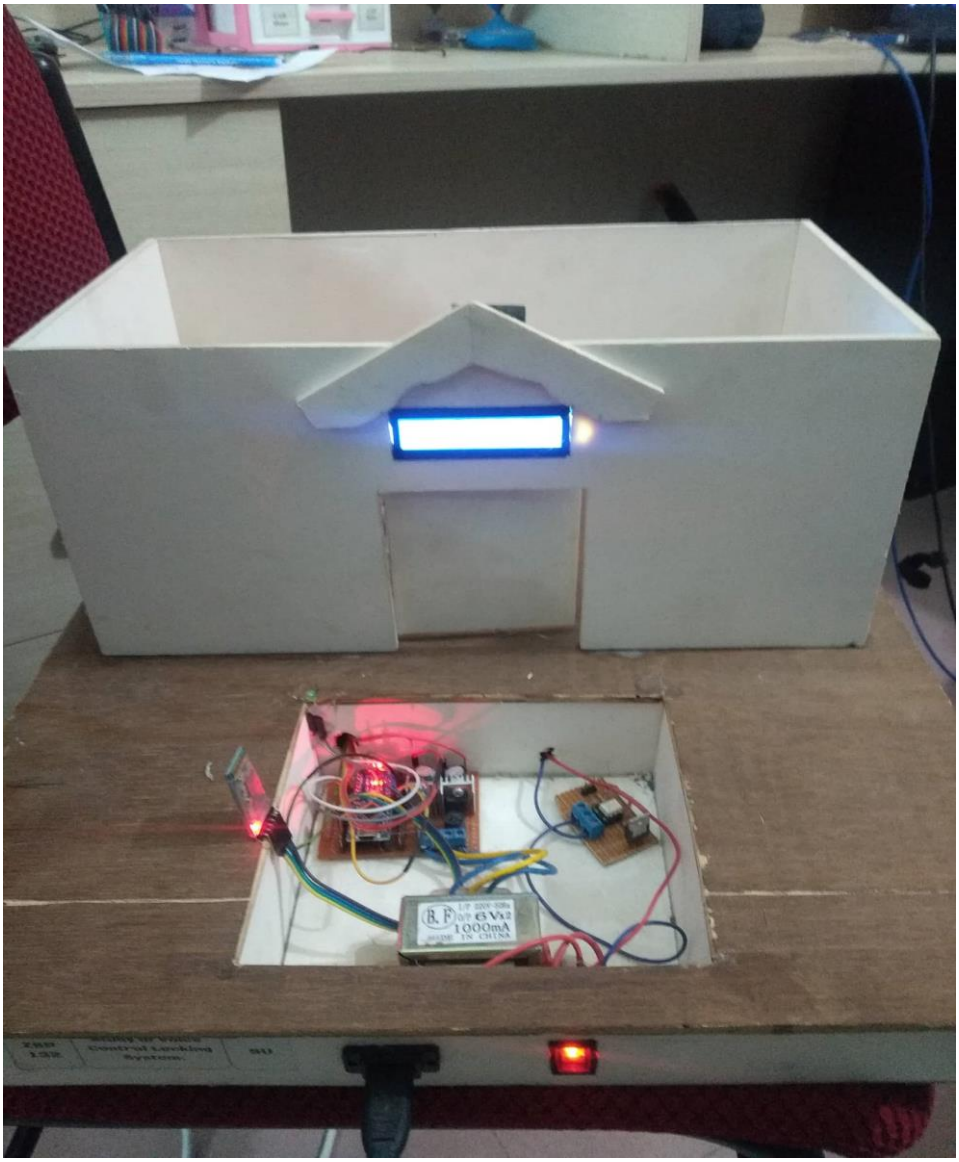

Fig 3.11 lock/unlock System

3.3 APP & SOFTWARE OVERVIEW :

3.3.1 ARDUINO VOICE CONTROL APP

Control your Arduino with voice commands using an Android smartphone! Before we make a voice activated home automation system, we must first learn the basic principles of the experiment. This guide will let you command the Arduino using your Android smartphone and a HC-05 Bluetooth module. See figure 3.2

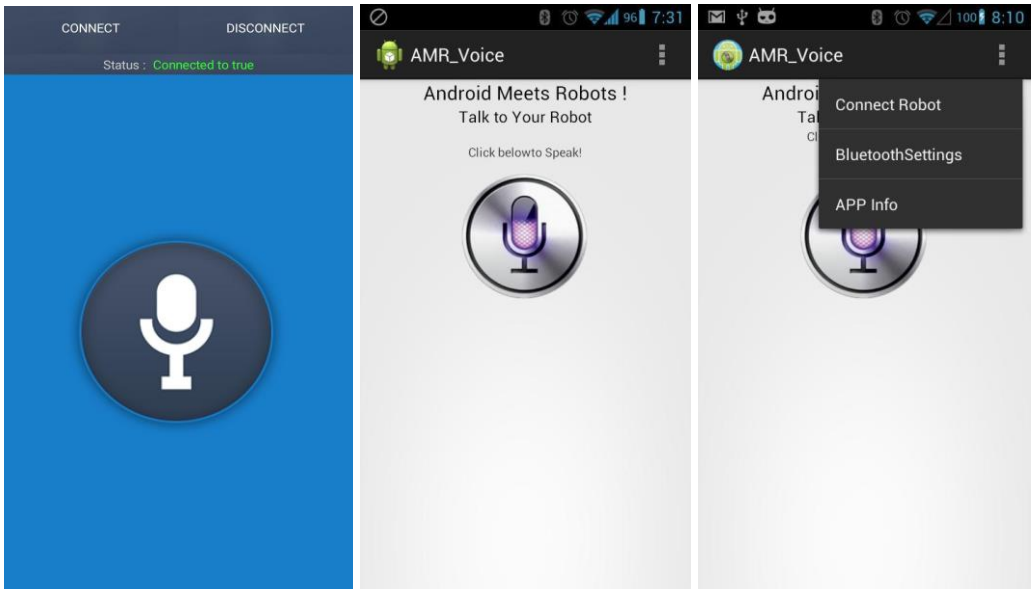

Fig 4.1 Home page of Arduino voice control App.

3.3.2 ARDUINO SOFTWARE (IDE)

Fig 4.2 Arduino Software.

The Arduino software is published as open source tools, available for extension by experienced programmers. The language may be distended through C++ libraries, and people needing to perceive the technical details will build the leap from Arduino to the AVR C programming language on which it's based. Similarly, we can add AVR-C code directly into our Arduino programs that it's primarily based. The Arduino Integrated Development Environment - or Arduino Software package (IDE) contains a text editor for writing code, a message space, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and real hardware to transfer programs and communicate with them. Programs that are written utilized Arduino Software (IDE) are known as sketches. These sketches are written text editor and are saved with the file extension .ino. The editor has options for cutting/pasting and for searching/replacing text. The message area gives feedback whereas saving and exporting and conjointly display errors. The console displays text output by the Arduino software (IDE), together with complete error messages and alternative info. The bottom right-hand corner of the window displays the organized board and port. The toolbar buttons allow us to verify and transfer programs, create, open, and save sketches and open the serial monitor.

The Arduino is open source environment that makes it simple to write, re-write the code and upload it to the INPUT/OUTPUT board. It runs on Windows, Mac OS X, and Linux. The setting is written in Java and based on Processing, Avr-GCC, and different open source software. The screenshot of Arduino1.6.8 is shown below

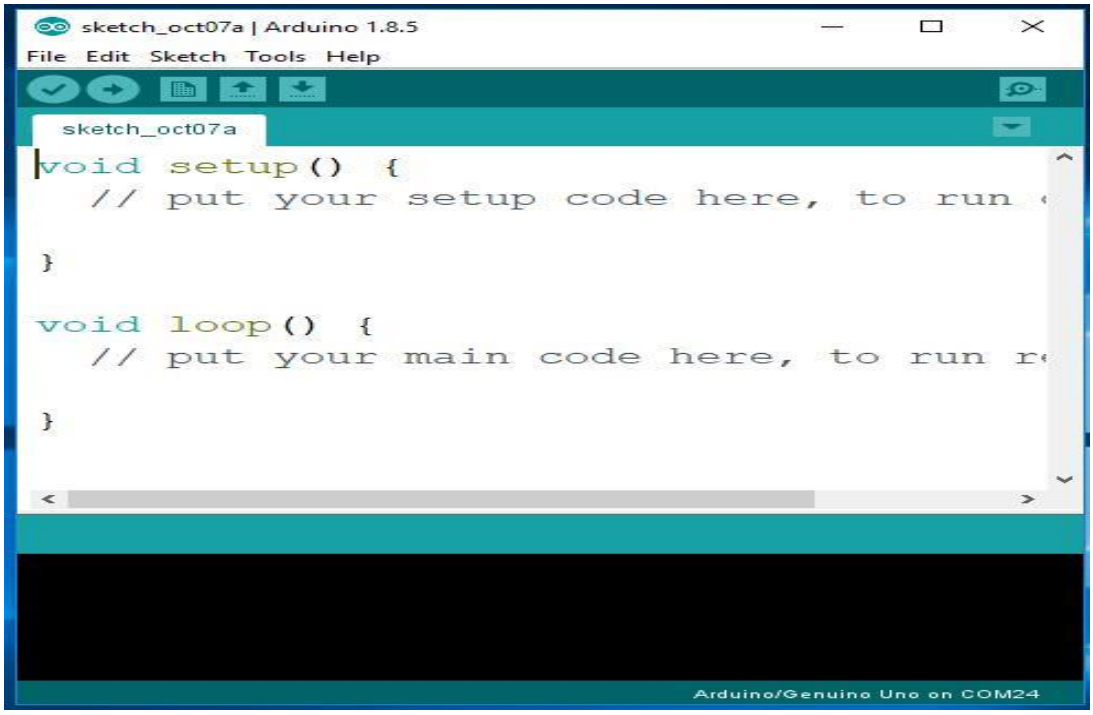

Fig. 4.3 Software Platform

It is conjointly capable of collecting and uploading programs to the board with a single click. There's usually no need to edit build files or run programs on a command-line interface. Although building on command-line is possible if needed with some third-party tools such as Ino. Arduino IDE comes with a C/C++ library known as "Wiring" (from the project of the same name), which makes several common input/output operations much easier. Arduino programs are written in C/C++, although users only needed to define two functions to make a runnable program. Setup() – a function run once at the beginning of a program that can initialize settings loop() – a function known as repeatedly till the board power off.

3.4 HARDWARE IMPLEMENTATION

3.4.1 INTERFACING OF BLUETOOTH (HC-05)

Module HC-05 is connected with Arduino, by attaching its one Data pin D0 and D1 with respectively and Module HC-05 operates on 5 volts.

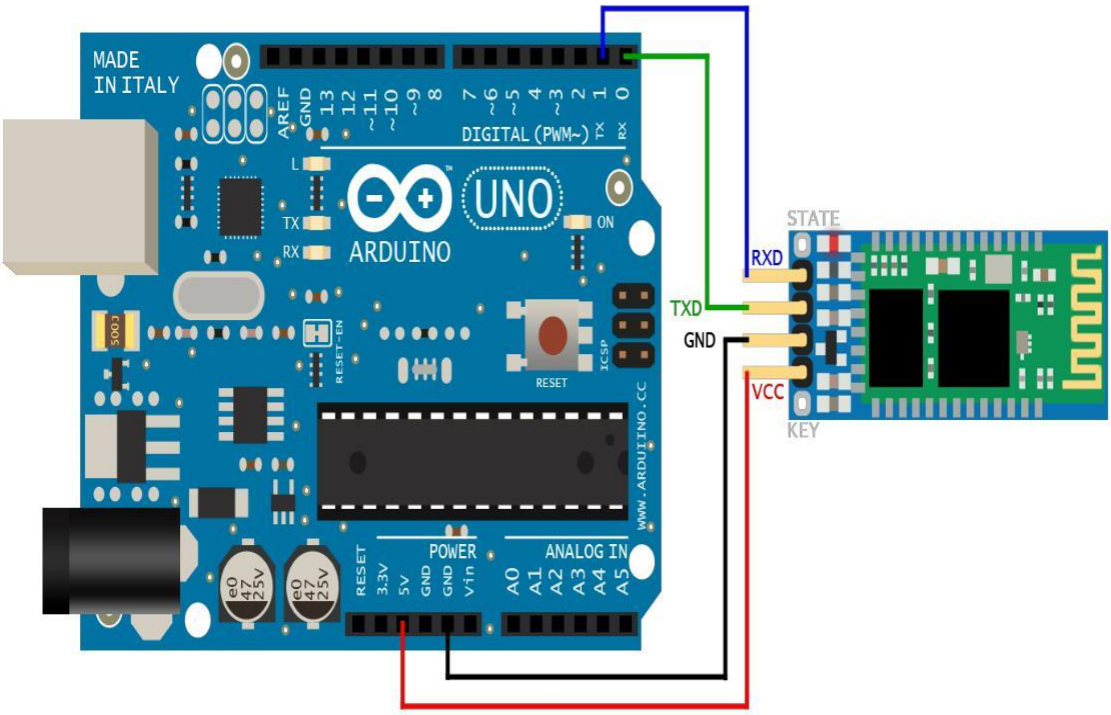

Fig. 5.1 Interfacing Of HC-05.

3.4.2 INTERFACING WITH APPLICATION

At first, we need to unlock your smartphone, The we need to turn ON our Bluetooth in the setting option. Then we need to connect our HC-05 with our smartphone. if any password or pin is asked for pair then we need to dial "1234" as like Fig 5.2(a). Then we need to open our Android application. Go to settings option and press connect robot as like Fig 5.2(a). then we see our device is connected with our Android application. now we are ready to use our device. if we want to connect a new smartphone then we need to disconnect our paired device and then we can connect our new device.

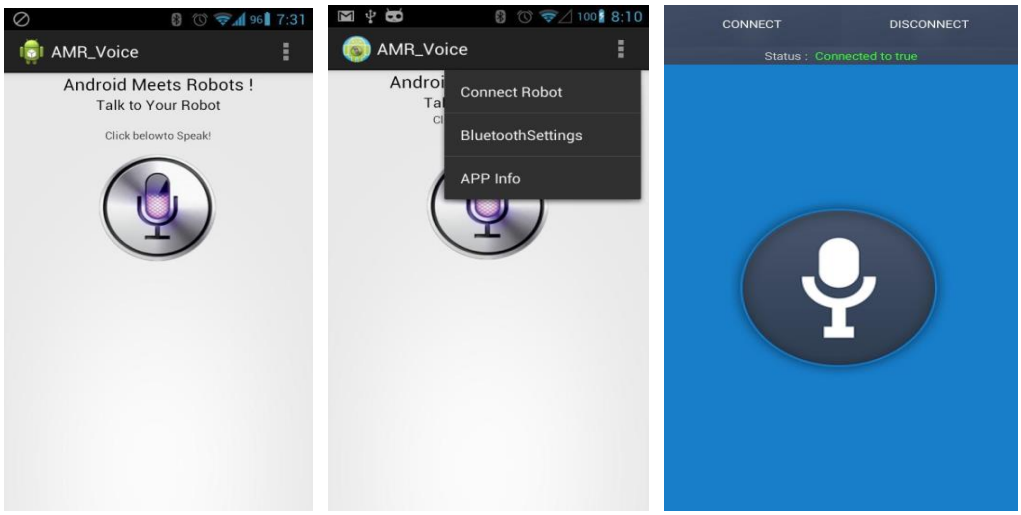

Fig 5.2 Interfacing of Application

After pairing we need to connect the Bluetooth with voice app, we need to give voice command to control the project, as for example Door (Closed/Open)

3.4.3 INTERFACING WITH SERVO MOTOR

The below figure shows how to connect a Servo Motor with the ArduinoNANO

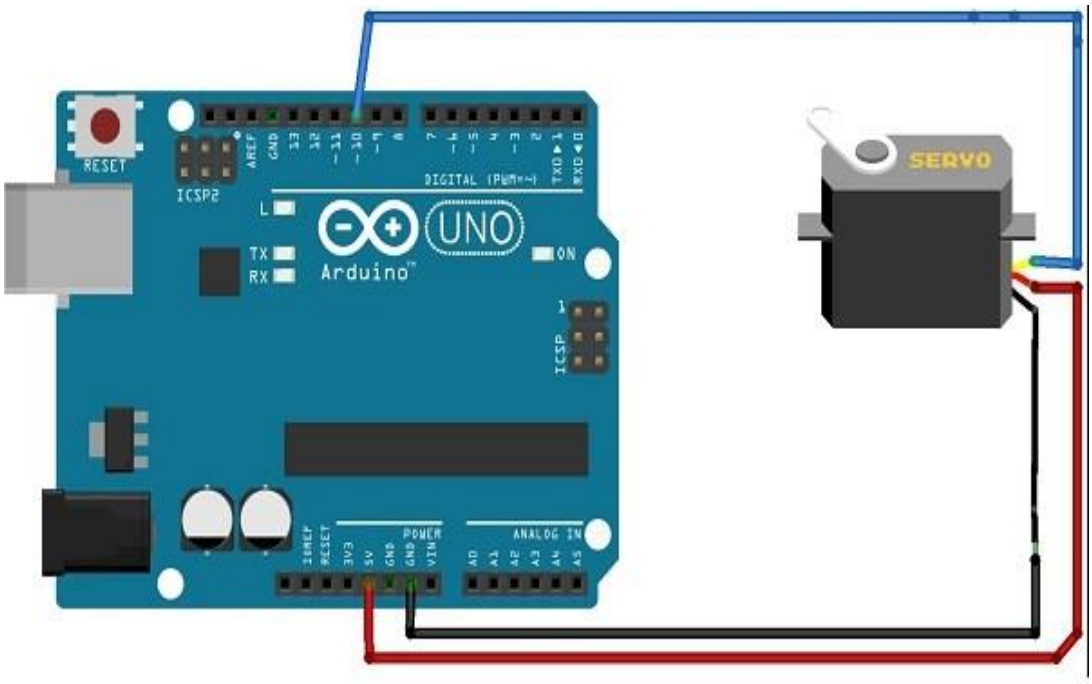

Fig 5.3 Interfacing with Servo motor

A servo motor has everything built-in: a motor, a feedback circuit, and most importantly, a motor driver. It just needs one power line, one ground, and one control pin.

Following are the steps to connect a servo motor to the Arduino:

- The servo motor has a female connector with three pins. The darkest or even black one is usually the ground. Connect this to the Arduino GND.
- Connect the power cable that in all standards should be red to 5V on the Arduino.
- Connect the remaining line on the servo connector to a digital pin on the Arduino.

3.4.4 LCD INTERFACING

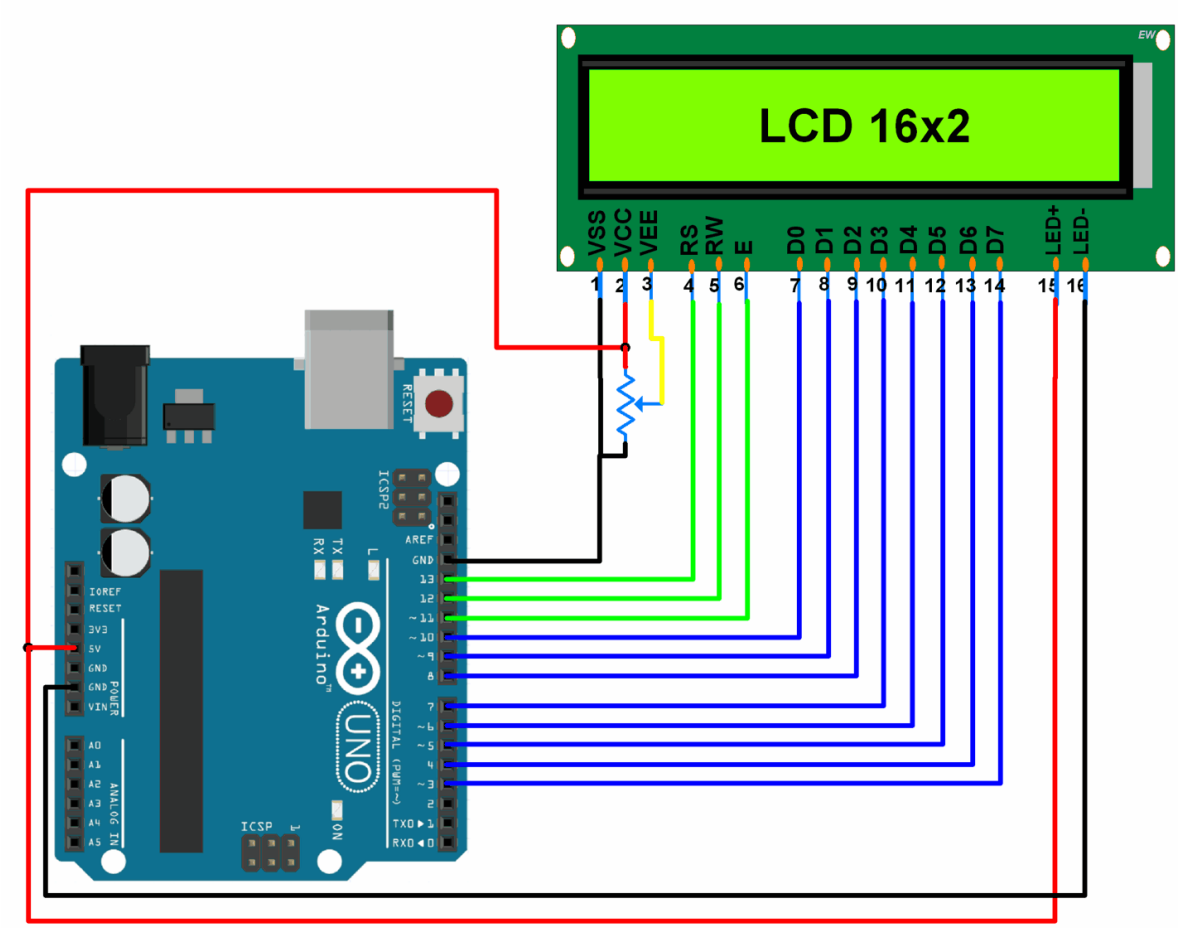

Fig 5.4 Interfacing with Loads.

LCDs (Liquid Crystal Displays) are used in embedded system applications for displaying various parameters and status of the system.

- LCD 16x2 is a 16-pin device that has 2 rows that can accommodate 16 characters each.
- LCD 16x2 can be used in 4-bit mode or 8-bit mode.
- It is also possible to create custom characters.
- It has 8 data lines and 3 control lines that can be used for control purposes.
- For more information about LCD 16x2 and how to use it, refer to the topic LCD 16x2 module in the sensors and modules section.

3.4.5 SYSTEM OPERATION

The base of the system is ATmega328p. The Android App acts as a remote. In the homework, we often use TV, AC, lock remotes. The apps are also like the remote. The inputs given to the microchip are the outputs of module HC-05. The module receives data from the app. The outputs of the controller are given to the servo motor. The Output of the Bluetooth module is given to the D0, D1 pin of ATmega328. When the microcontroller receives output from the HC-05, then it gives a command to operate the servo motor. After getting command from microcontroller, servo motor activate the security system. The receiving and transmitting pins of the Bluetooth module are connected to the TX and RX pins of ATmega328p respectively.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 RESULT

Voice control Smart Locking System is proposed and when the power is on, we set up our communication with Android phones and given voice command to execute our controlled system by preset data. The overall system was designed and tested by introducing the home security controlled wirelessly and the wireless system is possible by Bluetooth Module. Our project is completed a successful implementation & testing. It can be seen that output clearly. So we said that our system was effective fast response most secure and easy to construct.

4.2 FINAL EXPERIMENTAL SETUP

ATmega328p is the base of the system. The inputs given to the microchip are the outputs of module HC-05. The Output of the Bluetooth module is given to the D0, D1 pin of ATmega328p.

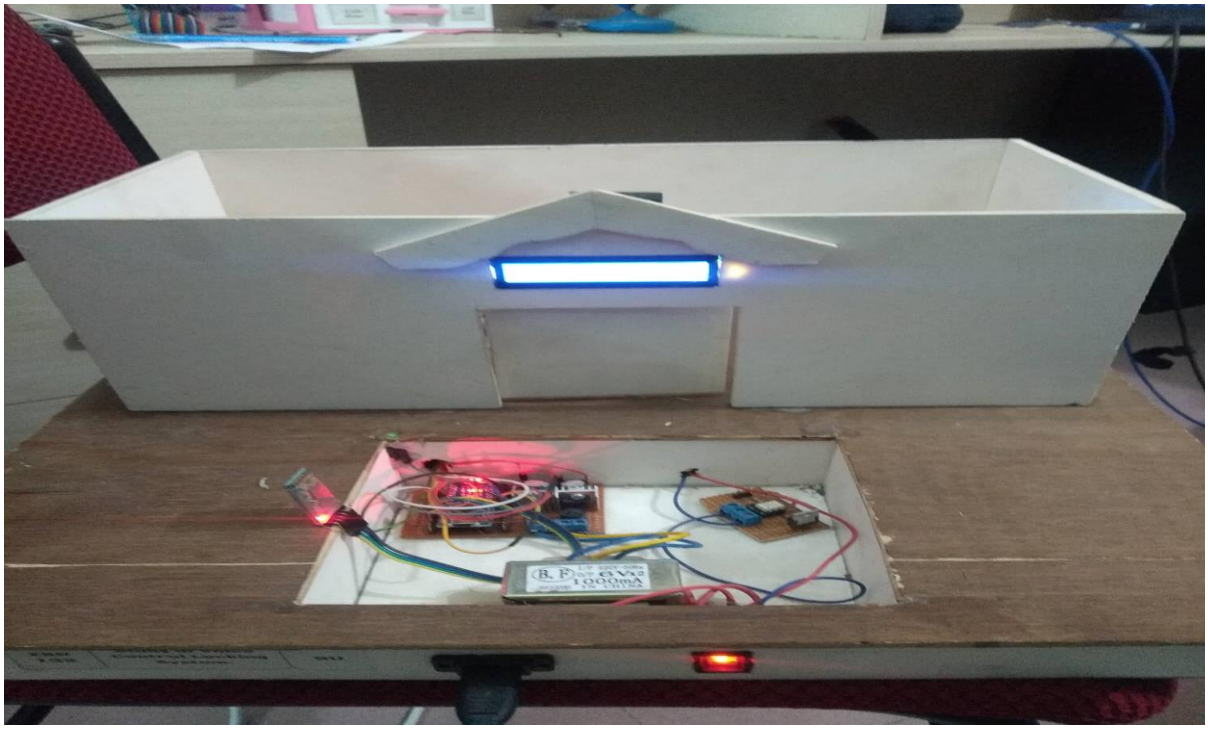

Fig 6.1 Our Project

Fig 6.1.1 Setup of our Voice Controlled Smart Locking system

We will connect out Bluetooth TX pin in RX pin, and RX pin in TX pin at ATmega328p

4.3 FUTURE RECOMMENDATION

- In the future, more sensors can be added with it for better security
- This monitoring system can be increased more by using GSM, Wifi.
- Software & app development can secure this locking system more

4.4 CONCLUSION

This project is a microcontroller based mostly on the project. A Voice Controlled Smart locking system is extremely effective during this time. During this project, we wish to execute the system to be used in varied places like home appliances, industrial automation, etc. A Bluetooth module receives data through an Android app like “Arduino voice control”. The data of the Bluetooth module delivers to the ArduinoNANO board. ArduinoNANO controls the servo motor to lock/unlock the door. We have finally succeeded in creating the “Voice Controlled Smart Locking System” satisfactorily. More information is gained also a lot of experiences are faced and a lot of information is collected ultimately, we've concluded with an excellent pleasure for achieving our aim. We’ve planned to satisfy my technical requirements. The knowledge we've earned with this project really would follow until the end of our career.

REFERENCES

[1] "Accupoint™ Linear Encoders". Epilog Laser. Archived from the original on 2012-10-07.

[2] Lander, Cyril W. (1993). "2. Rectifying Circuits". Power electronics (3rd ed.). London: McGraw-Hill. ISBN 978-0-07-707714-3.

[3] Suk-Hwan Suh; Seong Kyoon Kang; Dae-Hyuk Chung; Ian Stroud (22 August 2008). Theory and Design of CNC Systems. Springer Science & Business Media. pp. 11–. ISBN 978-1-84800-336-1. Archived from the original on 21 March 2017.

[4] "Getting Started: FOUNDATION > Introduction". arduino.cc.

[5] "Arduino's New CEO, Federico Musto, May Have Fabricated His Academic Record". WIRED. Retrieved 2017-12-22.

[6] Introducing new Arduino Pro IDE with advanced features". *Seed Studio*. Retrieved 26 October 2019.

[7] Bedell, Frederick (1942). "History of A-C Wave Form, Its Determination and Standardization". Transactions of the American Institute of Electrical Engineers. 61 (12): 864. doi :10.1109/T-AIEE.1942.5058456.

[8] Understanding Transformer Noise" (PDF). FP. Archived from the original (PDF) on 10 May 2006. Retrieved 30 January 2013.

[9] "LCD Image Persistence". Fujitsu technical support. Fujitsu. Archived from the original on April 23, 2012. Retrieved December 11, 2011.

APPENDIX A

Program Code:

Program is the brain of our project. Arduino is an open-source computer hardware which works on the program's instructions.

```
#include <SoftwareSerial.h>          //// include library
#include <Servo.h>                   //// include library
#include <Wire.h>                    //// include library
#include <LCD.h>                     //// include library
#include <LiquidCrystal_I2C.h>       //// include library

LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);

// Set the LCD I2C address
//0x27 is zero rows and 27 colums
//2 for enable pin, 1 for read and write pin , 0 for register selector pin
//data pins 4,5,6,7, 3 is the back light control pin
//POSITIVE is the back light polarity.

Servo myservo;

// servo: a variable of type Servo
// create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0;          // variable to store the servo position

const int rxPin = 7;      //blutooth module pin
const int txPin = 8;      //blutooth module pin

// difference between int and const int is that int is r/w while const int is r/only.
//int const* int const* is pointer to constant integer. This means that the variable being
declared is a pointer, pointing to a constant integer. Effectively, this implies that the pointer
is pointing to a value that shouldn't be changed.

SoftwareSerial mySerial(rxPin, txPin);
int ac=2;

String data;

///A string is a data type used in programmings, such as an integer and floating-point unit,
but is used to represent text rather than numbers. It is comprised of a set of characters that
can also contain spaces and numbers. For example, the word "hamburger" and the phrase "I
ate 3 hamburgers" are both strings
```

```
void setup()
{
  lcd.begin(16, 2);                // lcd.begin(cols, rows)
   myservo.attach(9);              //pin: the number of the pin that the servo is attached to
   Serial.begin(9500);
    mySerial.begin(9500);
```

//Serial. begin(9500); passes the value 9500 to the speed parameter. This tells the Arduino to get ready to exchange messages with the Serial Monitor at a data rate of 9500 bits per second. ... This tells the Arduino to send a series of binary ones and zeros to the Serial Monitor.

```
lcd.setCursor(0, 0);             //  lcd.setCursor(0, 0); // top left
lcd.print("===WELCOME TO===");
lcd.setCursor(0,1);              //  lcd.setCursor(0, 1); bottom left
lcd.print("===YOUR HOME===");
pinMode(ac, OUTPUT);
myservo.write(180);              // tell servo to go to position in variable 'pos'

digitalWrite(ac,LOW);
```

//The digitalWrite() function is used to write a HIGH or a LOW value to a digital pin. If the pin has been configured as an OUTPUT with pinMode(), its voltage will be set to the corresponding value: 5V (or 3.3V on 3.3V boards) for HIGH, 0V (ground) for LOW.

```
}



void loop()
{
   int i=0;
   char ch=0;    // char is a Data Types
   data="";

   while(1)   {
```

//while(1) or while(any non-zero integer)
// loop runs infinitely if 0 loop does not run

```
    while(mySerial.available()<=0);
    ch = mySerial.read();
    if(ch=='#')              //dec value 35        = #
    break;
    data+=ch;           //  +=    =>a += b     =>a = a+b
   }
```

```
Serial.println(data);

 if(data=="*door closed")
   {
 digitalWrite(ac,LOW);
 lcd.clear();
lcd.setCursor(0, 0);
lcd.print("===Door Lock===");
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("===WELCOME TO===");
lcd.setCursor(0,1);
lcd.print("===YOUR HOME===");
for (pos = 90; pos <= 180; pos += 1)    {   // goes from 0 degrees to 180 degrees
   myservo.write(pos);                      // tell servo to go to position in variable 'pos'
   delay(15);                              // waits 15ms for the servo to reach the position
  }
     Serial.println("ac on");
   }

  else if(data=="*door open")
   {
    digitalWrite(ac,HIGH);
    for (pos = 180; pos >= 90; pos -= 1) {      // goes from 180 degrees to 0 degrees
   myservo.write(pos);                          // tell servo to go to position in variable 'pos'
   delay(15);
   lcd.clear();
   lcd.setCursor(0, 0);
   lcd.print("===Door Open===");
}
    Serial.println("ac off");
   }

}
```