Design and Implementation of Smart Home Security System with Arduino Based Password Protection

A project Submitted by

1.William Sohag Sarkar.	ID - 2012-1-55-043
2.Farhan Rijvi.	ID - 2012-2-55-038

Under the Supervision of

Mr. Mustafa Mahmud Hussain

Assistant Professor

Department of Electronics and Communications Engineering

East West University

EAST WEST UNIVERSITY

DECLARATION

We hereby declare that this projectis based onoriginal except for source material explicitly acknowledged.We have completed the project on the topic entitled "**Design and Implementation of Smart Home Security System with Arduino Based Password Protection**" as well as we prepared to the Department of Electronic and Communication Engineering of East West University in partial fulfillment of the requirement for the degree of B.Sc. in Electronic and Telecommunication Engineering under the supervision of Mr. Mustafa Mahmud Hussain, Assistant Professor, Department of Electronics and Communication Engineering, East West University. The piece of work or a part of the piece of work has not been submitted for more than one purpose. We also acknowledge that we are aware of University policy and regulations on honesty in academic work, and of the disciplinary guidelines and procedures applicable to breaches of such policy and regulations.

Students Name and Signatures

1. **William Sohag Sarkar**
   2012-1-55-043

----

2. **Farhan Rijvi**
   2012-2-55-038

APPROVAL

This project has been prepared and submitted by William Sohag Sarkar, ID 2012-1-55-043 and Farhan Rijvi, ID 2012-2-55-038. This project report is titled as “Design and Implementation of Smart Home Security System with Arduino Based Password Protection” submitted to the respected members of the Board of Examiners of the Faculty of Engineering, East West University for partial fulfillment of the requirements for the degree of Bachelor of Science in Electronics & Telecommunications Engineering, under complete supervision of the undersigned.

____

Supervisor
Mr. Mustafa Mahmud Hossain
Assistant-Professor,
Department Of ECE,
East West University.

____

Chairperson
Dr. M. Mofazzal Hossain
Chairperson & Professor,
Department of ECE,
East West University.

ACKNOWLEDGEMENT

At first, we are very thankful to Almighty ALLAH, without His help we couldn't complete it. Then gradually come the names of our parents, teachers, family, friends, and mentors. Especially, please allow us to dedicate our acknowledgment of gratitude toward the following significant advisors and contributors: First and foremost, we would like to thanks **Mr. Mustafa Mahmud Hussain** for his most support and encouragement. He offered invaluable detailed advices and showed us proper way how to complete it successfully. He encouraged us and told us how to do it successfully.We would like to extend our thanks to the laboratory staff of ECE Dept. for their fast response and cooperation with us to get some materials we need for our case. Finally, we sincerely thank to my parents, teachers, family, and friends, who provide the advice and financial support. The product of this report would not be possible without all of them.

Table of Contents (Index)

Abstract **07**

Chapter 1...**08**

1.1 "Arduino"what does it mean?... 08
1.2 Why Arduino?.. 09-10

1.2 (a) Getting Started with Arduino and Genuino products.. 10-12

1.3 ArduinoHistory.. 12-13
1.4 Types of Arduino.. 14-17
1.5 Advantage of using Arduino...18-19
1.6 Evaluation 20

Chapter 2(Factors Used in the Design).. **21**

2.1 Introduction 21

2.2 PIR Sensor.. 21

2.2 (a) Operating principles..21-22

2.2 (b) Introduction.. 22

2.2.2How PIR Works.. 23-24

2.2.3 The Sensor Signal..24-25

2.2.4Three-stage architecture for detecting motion..25- 27

2.2.5 Conclusion... 27

2.3.5 5volt 2Ampier power adapter.. 27-28

2.3. (b) Wires (types of)...29-30

2.4 Light Emitting Diode...31

2.4.1 What is LED?.. 31

2.4.2 How Do LED Light Bulbs Works 31-33

2.5 Arduino Writing Sketches...33-34

Chapter 3……… 34

3.1 Introduction... 34

3.2 Components use……………………………………………………………………………………………………… 35

3.3 Internal Circuit Constriction of each components………………………………………………………… 35

3.3.1 Arduino Mega ………………………………………………………………………………………………….…………..35-36

3.3.2 PIR Sensor ……..…………36

3.3.3 Servo Motor ……37

3.3.4 Display …….37

3.3.5 LED Light ………38

3.3.6 Arduino Sketch Software …………………………………………………………………………………………………38

3.4 System Design and Connections ………………………………………………………………………………………39

3.4.1 Block Diagram of PIR Sensor Connection ……………………………………………………………………………39

3.4.2 Block Diagram of Display Connection……………………………………………………………………………..39-40

3.4.3 Block Diagram of LED Connection………………………………………………………………………………………40

3.4.4 Block Diagram of 5V-2A Adapter ………………………………………………………………………………….41

3.5 Block Diagram of Servo Motor ………………………………………………………………………………………..41

3.6 Block Diagram of full Components……………………………………………………………………………………42

Chapter 4 (Implementation and result)…………………………………………………………………………… 43

4.1 Implementation and result ……………………………………………………………………………………………..43

4.1.1 System in Idle mode (Circuit Design)…………………………………………………………………………… 43

4.1.2 System in Active mode (Circuit Design)………………………………………………………………………… 44

4.1.3 Practical Set-up and Result………………………………………………………………………………………… 45

Chapter 5……… 46

FinalProgram Screenshot ……………………………………………………………………………………………………….46

Programming codes………. 46

Code ……….47-52

Chapter 6……… 50

6.1 Discussion………. 53

6.2 Future Work……. 53

References……… 54

ABSTRACT

Home security system is highly preferable not only for home but also anywhere security issues are needed. This security is included for Home/ Bank/ Office and for the purpose of safety;everybody wants to take proper measures to prevent intrusion. Now days, intruder can take advantages to takeany illegal works and can violent any important security issues. Keypad locked system is essential for securing any home or institutions. In this paper, we present the design and development of affordable cost and proper power consumption with password protected home security system.

Chapter-01

1.1 “Arduino” What does it mean?

Arduino is an open-source prototyping platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based onWiring), and the Arduino Software (IDE), based on Processing. Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. Thesoftware, too, is open-source, and it is growing through the contributions of users worldwide.

1.2 Why Arduino?

Thanks to its simple and accessible user experience, Arduino has been used in thousands of different projects and applications. The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics. Designers and architects build interactive prototypes, musicians and artists use it for installations and to experiment with new musical instruments. Makers, of course, use it to build many of the projects exhibited at the Maker Faire, for example. Arduino is a key tool to learn new things. Anyone - children, hobbyists, artists, programmers - can start tinkering just following the step by step instructions of a kit, or sharing ideas online with other members of the Arduinocommunity.There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many others offer similar functionality. All of these tools take the messy details of microcontroller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

- Inexpensive - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50
- Cross-platform - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
- Simple, clear programming environment - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
- Open source and extensible software - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.
- Open source and extensible hardware - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version

of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money.

1.2(a) Getting Started with Arduino and Genuino products:-

*Install the Arduino Software (IDE) on Windows PCs-

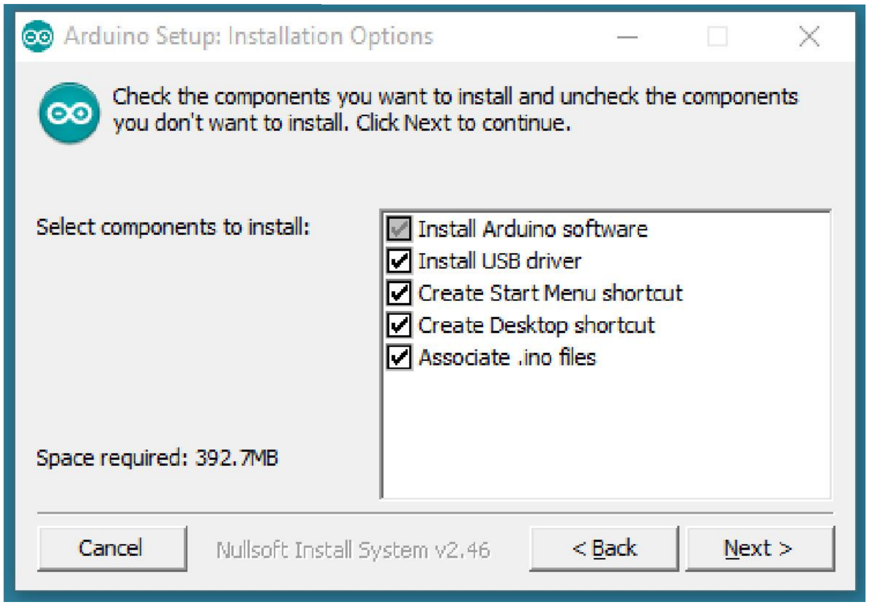
…This document explains how to install the Arduino Software (IDE) on Windows machines.

- ✓ Download the Arduino Software (IDE)
- ✓ Proceed with board specific instructions.

How to Download the Arduino Software (IDE):

Get the latest version from the download page. You can choose between the Installer (.exe) and the Zip packages. We suggest you use the first one that installs directly everything you need to use the Arduino Software (IDE), including the drivers. With the Zip package you need to install the drivers manually. The Zip file is also useful if you want to create a portable installation.

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



Arduino Setup: Installation Options

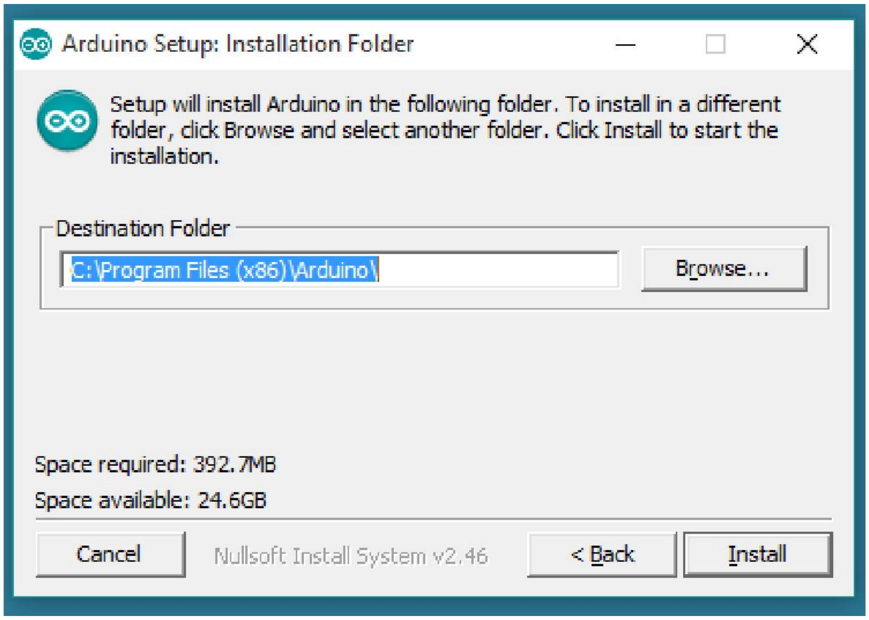
Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

Select components to install:

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

Space required: 392.7MB

Choose the components to install



Arduino Setup: Installation Folder

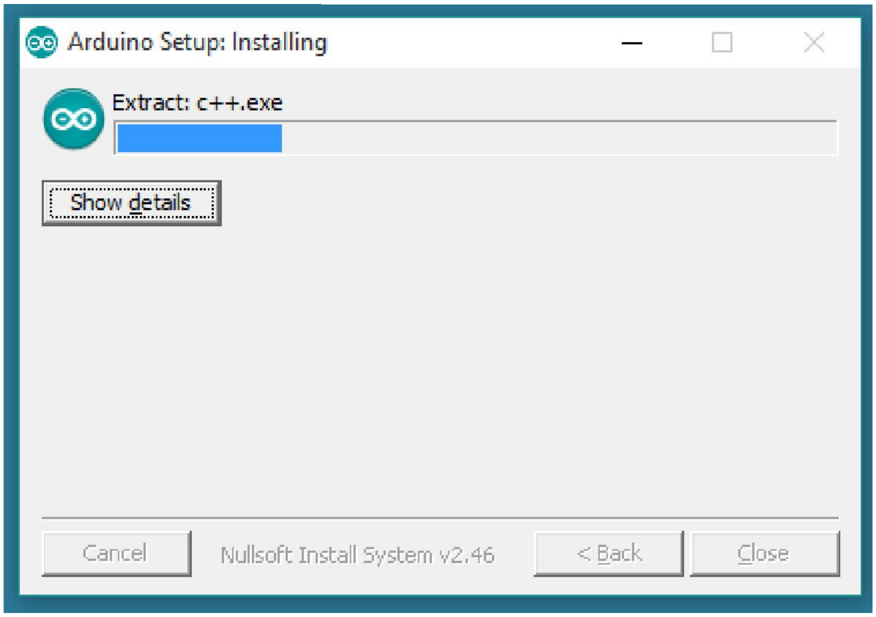
Setup will install Arduino in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.

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

Space required: 392.7MB

Space available: 24.6GB

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



Arduino Setup: Installing

Extract: c++.exe

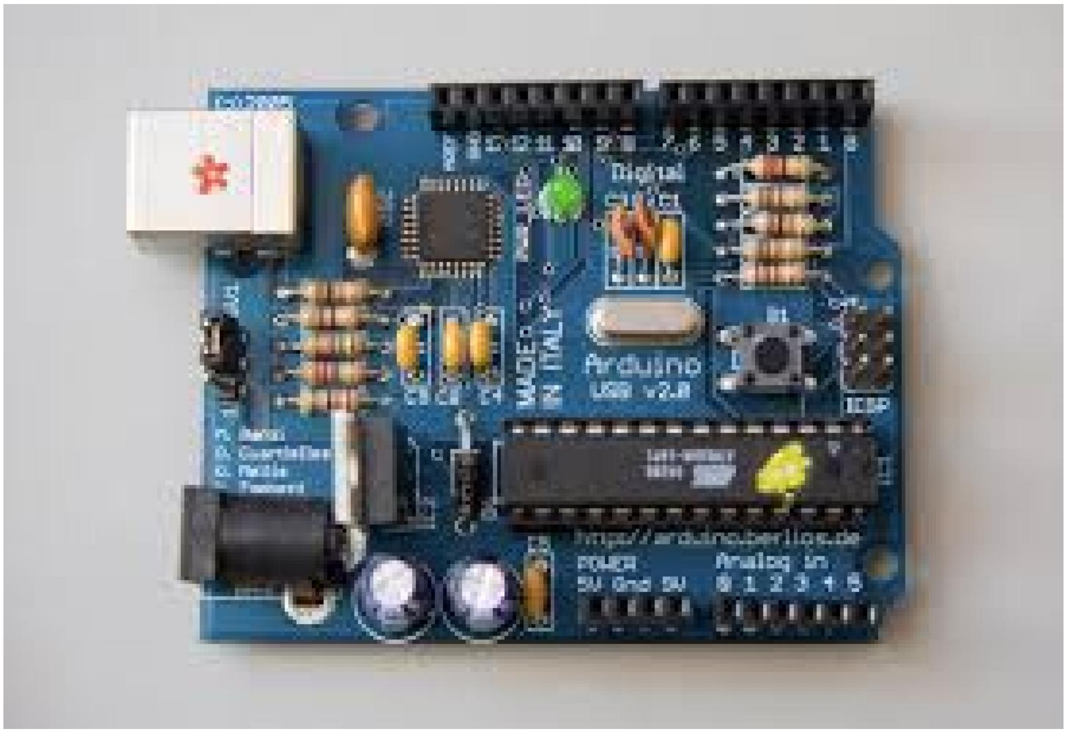
The process will extract and install all the required files to execute properly the Arduino Software (IDE)

#<u>Proceed with board specific instructions</u>:

When the Arduino Software (IDE) is properly installed you can go back to the Getting Started Home and choose your board from the list on the right of the page.

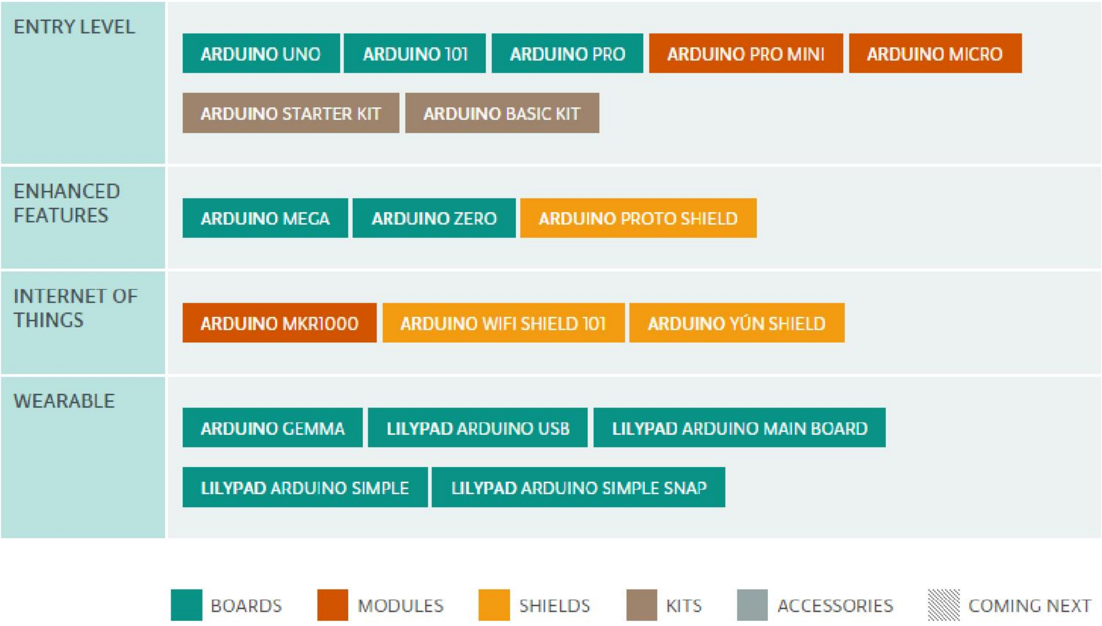
1.3ArduinoHISTORY:-

Colombian student Hernando Barraging created the development platform Wiring as his Master's thesis project in 2004 at the Interaction Design Institute Ivrea (IDII) in Ivrea, Italy. Massimo Banzi and Casey Reas (known for his work on Processing) were supervisors for his thesis. The goal was to create low cost, simple tools for non-engineers to create digital projects. The Wiring platform consisted of a hardware PCB with an ATmega128 microcontroller, an IDE based on Processing and library functions to easily program the microcontroller. In 2005, Massimo Banzi, with David Mellis (then an IDII student) and David Cuartielles, added support for the cheaper ATmega8 microcontroller to Wiring. But instead of continuing the work on Wiring, they forked (or copied) the Wiring source code and started running it as a separate project, called Arduino. The Arduino's initial core team consisted of Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis. The name Arduino comes from a bar in Ivrea, where some of the founders of the project used to meet. The bar was named after Arduin of Ivrea, who was the margrave of the March of Ivrea and King of Italy from 1002 to 1014.Following the completion of the Wiring platform, its lighter, lower cost versions were created and made available to the open-source community. Associated researchers, including David Cuartielles, promoted the idea.The first prototype board, made in 2005, was a simple design, and it wasn't called Arduino. Massimo Banzi would coin the name later that year.



1.4. TYPES OF ARDUINO

There are mainly four types of Arduino ;-



ENTRY LEVEL	🟩 ARDUINO UNO 🟩 ARDUINO 101 🟩 ARDUINO PRO 🟥 ARDUINO PRO MINI 🟥 ARDUINO MICRO 🟫 ARDUINO STARTER KIT 🟫 ARDUINO BASIC KIT
ENHANCED FEATURES	🟩 ARDUINO MEGA 🟩 ARDUINO ZERO 🟧 ARDUINO PROTO SHIELD
INTERNET OF THINGS	🟥 ARDUINO MKR1000 🟧 ARDUINO WIFI SHIELD 101 🟧 ARDUINO YÚN SHIELD
WEARABLE	🟩 ARDUINO GEMMA 🟩 LILYPAD ARDUINO USB 🟩 LILYPAD ARDUINO MAIN BOARD 🟩 LILYPAD ARDUINO SIMPLE 🟩 LILYPAD ARDUINO SIMPLE SNAP

🟩 BOARDS 🟥 MODULES 🟧 SHIELDS 🟫 KITS ⬜ ACCESSORIES ▨ COMING NEXT

Major types are discussed below:

Entry Level

Get started with Arduino using Entry Level products: easy to use and ready to power your first creative projects. These boards and modules are the best to start learning and tinkering with electronics and coding. The StarterKit includes a book with 15 tutorials that will walk you through the basics up to complex projects.

Enhanced Features

Experience the excitement of more complex projects choosing one of the boards with advanced functionalities, or faster performances.

Internet of Things

Make connected devices easily with one of these IoT products and open your creativity with the opportunities of the World Wide Web.

Wearable

Add smartness to your soft projects and discover the magic of sewing the power of electronics directly to textiles.

Retired

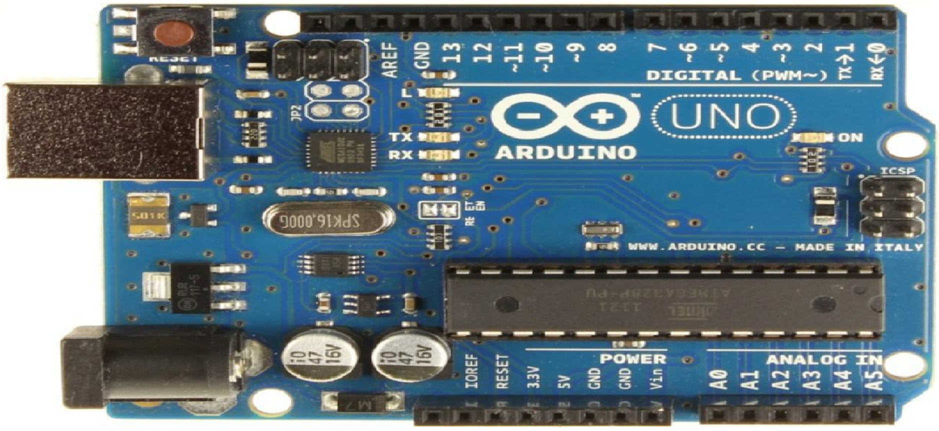
Explore the history of Arduino with a journey through all the boards, accessories, shield, kits and documentation released since 2006.

A Basic Table:

Features	Arduino Uno	Arduino Due	Arduino Mega	Arduino Leonardo
Processor	**16Mhz ATmega328**	**84MHz AT91SAM3X8E**	**16MHz ATmega2560**	**16MHz ATmega32u4**
Memory	**2KB SRAM, 32KB flash**	**96KB SRAM, 512KB flash**	**8KB SRAM, 256KB flash**	**2.5KB SRAM, 32KB flash**
Digital I/O	**14**	**54**	**54**	**20**
Analogue I/O	**6 input, 0 output**	**12 input, 2 output**	**16 input, 0 output**	**12 input, 0 output**

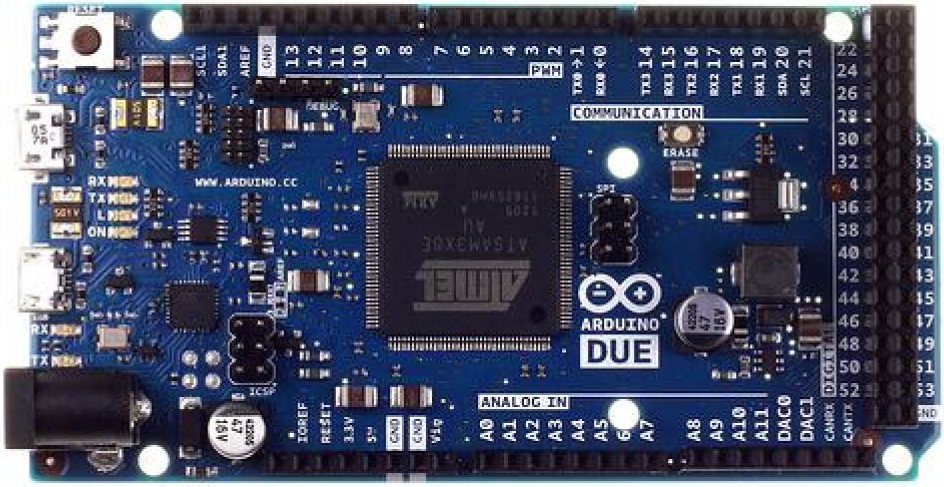
1. The Arduino Uno

The Uno is the most common board and the one labeled as the classic Arduino. This board comes with everything new users need to learn about the electronics and programming requiresstarting this hobby. It is compatible with most available Arduino shields.



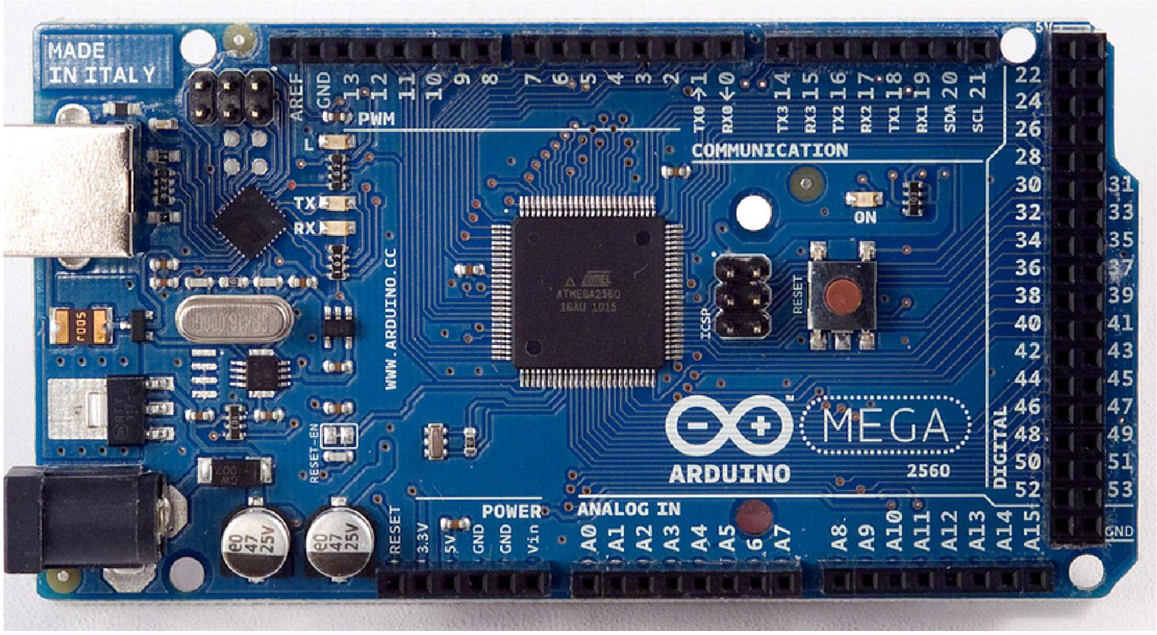
2. <u>The Arduino Due</u>

The Arduino Due is the second iteration of the classic Arduino and offers more features for advanced users. The Due's processor is faster, has more memory, and more I/O ports. It does not support many shields. Because of the faster CPU, the Arduino Due runs on a lower voltage: 3.3V over the Uno's 5V. This means it cannot always support the same devices.



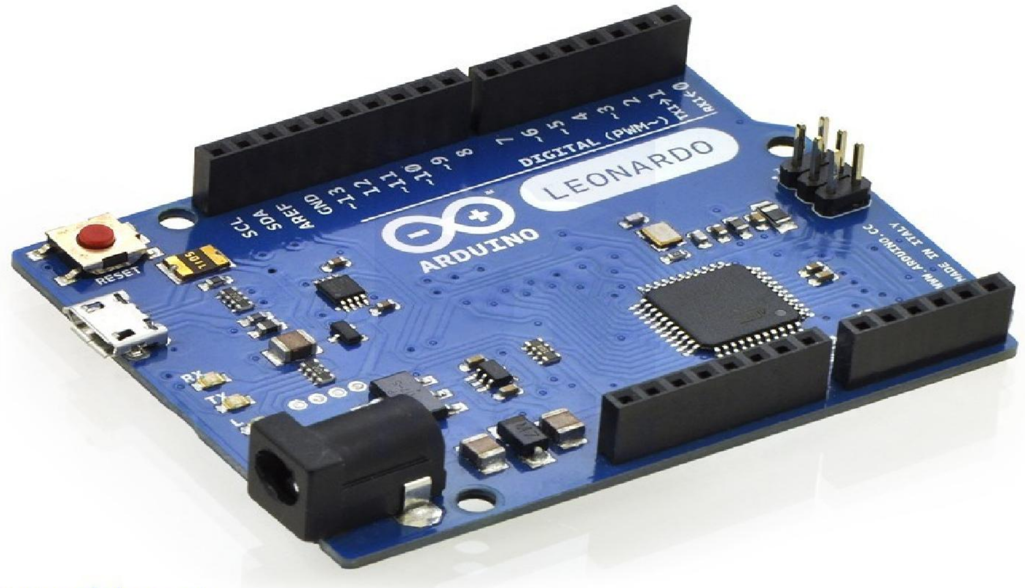
3. The Arduino Mega

The Arduino Mega comes in two types, the Mega 2560 and the MEGA ADK. The ADK is similar to the 2560; however, it also has a programmable USB host chip installed. It uses the same 5V power supply as the Uno, so many of the Arduino shields are also compatible with the Mega; however, because of the placement of some of the pins, not all of them are usable.



4. The Arduino Leonardo

The Leonardo is not a common board, but has similar features to the Uno, including the 5V power supply and the processing power. It is a good board for those who need more input and output ports than the Arduino Uno, but do not need the horsepower or size of the Due. It uses a micro-USB adapter instead of the Uno's full-size USB port.



www.pololu.com

1.5 Advantages of Using Arduino

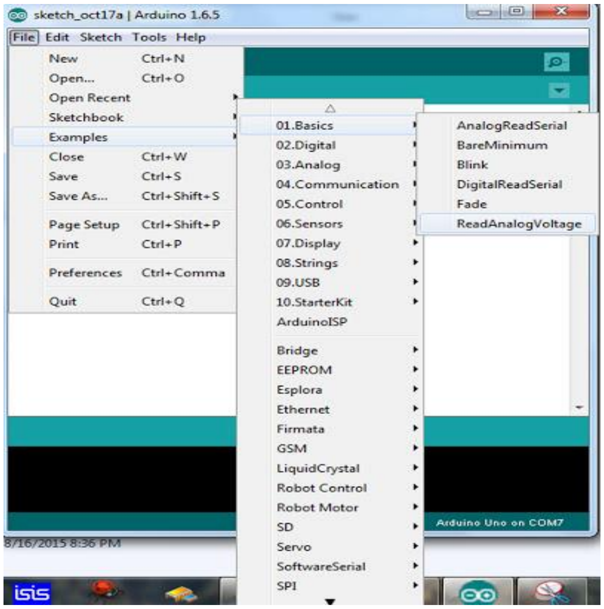
Using an Arduino simplifies the amount of hardware and software development you need to do in order to get a system running. The Arduino hardware platform already has the power and reset circuitry setup as well as circuitry to program and communicate with the microcontroller over USB. In addition, the I/O pins of the microcontroller are typically already fed out to sockets/headers for easy access (This may vary a bit with the specific model).On the software side, Arduino provides a number of libraries to make programming the microcontroller easier. The simplest of these are functions to control and read the I/O pins rather than having to fiddle with the bus/bit masks normally used to interface with the Atmega I/O (This is a fairly minor inconvenience). More useful are things such as being able to set I/O pins to PWM at a certain duty cycle using a single command or doing Serial communication. On the other hand, if you want to measure the voltage using Arduino.

1- Ready to Use: The biggest advantage of Arduino is its ready to use structure. As Arduino comes in a complete package form which includes the 5V regulator, a burner, an oscillator, a micro-controller, serial communication interface, LED and headers for the connections. You don't have to think about programmer connections for programming or any other interface. Just plug it into USB port of your computer and that's it. Your revolutionary idea is going to change the world after just few words of coding.

2- Examples of codes: Another big advantage of Arduino is its library of examples present inside the software of Arduino. I'll explain this advantage using an example ofvoltage measurement. For example if you want to measure voltage using ATmega8 micro-controller and want to display the output on computer screen then you have to go through the whole

process. The process will start from learning the ADC's of micro-controller for measurement, went through the learning of serial communication for display and will end at USB - Serial converters. If you want to check this whole process click on the link below.
DC voltage measurement using Atmel AVR micro-controller.

On the other hand, if you want to measure the voltage using Arduino. Just plug in your Arduino and open the ReadAnalogVoltage example as shown in the figure 1.



3- ***Effortless functions:*** During coding of Arduino, you will notice some functions which make the life so easy. Another advantage of Arduino is its automatic unit conversion capability. You can say that during debugging you don't have to worry about the units conversions. Just use your all force on the main parts of your projects. You don't have to worry about side problems.

4- ***Large community:*** There are many forums present on the internet in which people are talking about the Arduino. Engineers, hobbyists and professionals are making their projects through Arduino. You can easily find help about everything. Moreover the Arduino website itself explains each and every functions of Arduino.

So, We should conclude the advantage of Arduino by saying that during working on different projects you just have to worry about your innovative idea. The remaining will handle by Arduino itself.

1.6 Development:-

It was in the year 2005 that the first ever Arduino board was born in the classrooms of the Interactive Design Institute in Ivrea, Italy. Well, if you are not very familiar with the term, an Arduino is an Open Source microcontroller based development board that has opened the doors of electronics to a number of designers and creative engineers.It was in the Interactive Design Institute that a hardware thesis was contributed for a wiring design by a Colombian student named Hernando Barragan. The title of the thesis was “Arduino–La rivoluzionedell’open hardware” (“Arduino – The Revolution of Open Hardware”). Yes, it sounded a little different from the usual thesis but none would have imagined that it would carve a niche in the field of electronics. A team of five developers worked on this thesis and when the new wiring platform was complete, they worked to make it much lighter, less expensive, and available to the open source community.Arduino is an open-source hardware. The hardware reference designs are distributed under a Creative Commons Attribution Share-Alike 2.5 license and are available on the Arduino website. Layout and production files for some versions of the hardware are also available. The source code for the IDE is released under the GNU General Public License, version 2.Although the hardware and software designs are freely available under copy left licenses, the developers have requested that the name "Arduino" be exclusive to the official product and not be used for derived works without permission. The official policy document on use of the Arduino name emphasizes that the project is open to incorporating work by others into the official product. Several Arduino-compatible products commercially released have avoided the Arduino name by using -duino name variants.

1.7 Evaluation:-

The PIC microcontroller board, introduced in 1985 by General Instruments, became one of the most popular tools for electronics enthusiasts (before the Arduino) for several reasons. Other popular boards for hobbyists include the BASIC Stamp (Parallax Inc., 1990), and Wiring both of which share the benefits of simplicity of programming, and a resulting ease of rapid-prototyping. The Arduino project grew largely out of the “DIY” climate created by the burgeoning popularity of rapid- prototyping boards like PIC. In fact, the immediate precursor to the Arduino was a custom made Wiring microcontroller created by the artist/ designer Hernando Barragan in 2004 for his. In 2005, the Arduino team was formed in Ivrea, Italy, consisting of Barragan, Massimo Banzi, David Cuartielles, Dave Mellis, Gianluca Marino, and Nicholas Zambetti. As a result, the Arduino incorporated the following characteristic, a programming environment based on processing language (a programming language conceived by Ben Fry and Casey Reas, also conceived for artists/ designers), the ability to program the board via a standard USB connection, and a low price point. The Arduino achieved rapid success even within its first two years of existence, selling in a quantity of more than 50,000 boards. By 2009, it had spawned over 13 different incarnations, each specialized for different applications. Today, the Arduino microcontroller has become one of the most popular prototyping platforms in the world, and is a prime example of how hardware and software technologies originally created for military, business, and scientific ways.

Chapter-02

Factors Used in the Design

2.1Introduction:

The designed system consists of 4*4 keypad, Arduino mega 2560, servo motor, breadboard, LED 5mm, wires(male to male and male to female), 220 ohm resistance (20 pice),5 volt 2A adopter, PIR sensor and POT. In this chapter, we are going to discuss briefly about the functions and characteristics of different component in the system.

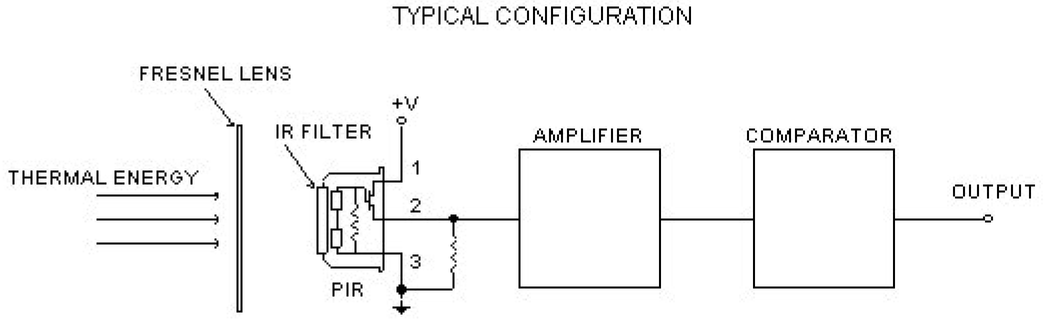
2.2 PIR SENSOR:

- With Variable
- Without Variable

Operating principles:

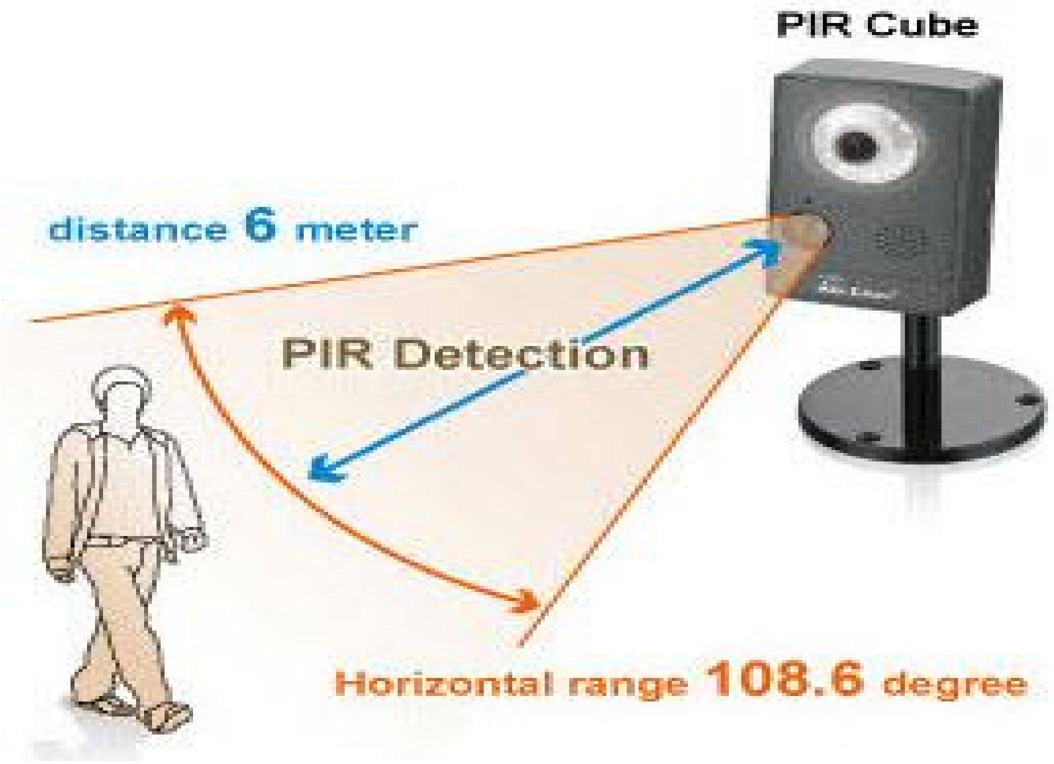
All objects with a temperature above absolute zero emit heat energy in the form of radiation. Usually this radiation isn't visible to the human eye because it radiates at infrared wavelengths, but it can be detected by electronic devices designed for such a purpose.

The term passive in this instance refers to the fact that PIR devices do not generate or radiate any energy for detection purposes. They work entirely by detecting the energy given off by other objects. PIR sensors don't detect or measure "heat"; instead they detect the infrared radiation emitted or reflected from an object.



Introduction:

PIR sensors allow you to sense motion, almost always used to detect whether a human has moved in or out of the sensors range. They are small, inexpensive, low-power, easy to use and don't wear out. For that reason they are commonly found in appliances and gadgets used in homes or businesses. They are often referred to as PIR, "Passive Infrared", "Pyroelectric", or "IR motion" sensors The IR energy of an object depends on different parameters such as its temperature, color, and texture. This energy is not visible with the human eye, but PIR sensors can detect it.



2.2.2 How PIR Works:

The aim of the PIR sensor is to detect motion in a specific area. As seen in Figure: "PIR sensor", these sensors contain two parts that are sensitive to IR. Since it is motion that is important, the signal delivered by the sensor reflects different amounts of IR detected by each part. If both parts "see" the same amount of IR, the sensor detects nothing. But, if one of the two parts "sees" more or less IR than the other part, the sensor detects something and the output voltage of the sensor varies.

The area of the two rectangles that is IR sensitive is small (≈2 mm² for each rectangle). Clearly, it would be better to increase the IR sensitive area. This can be done in two ways:

- By increasing the number of sensors
- By using a Fresnel lens.

The latter method is highly recommended as it is inexpensive and improves IR reception by the sensor. It multiplies the area visible to the sensor and it focuses reception of IR by the sensor.

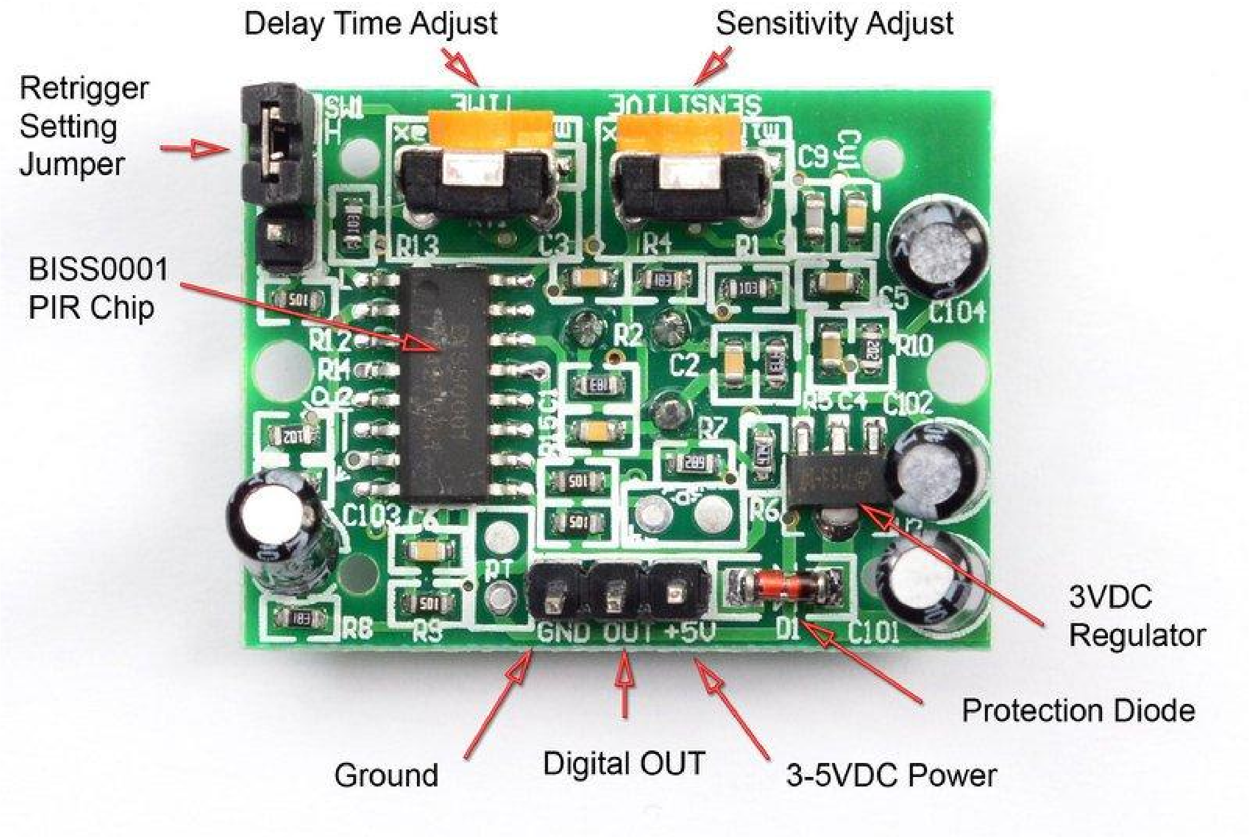


Figure: Principle of the PIR sensor

2.2.3 The sensor signal

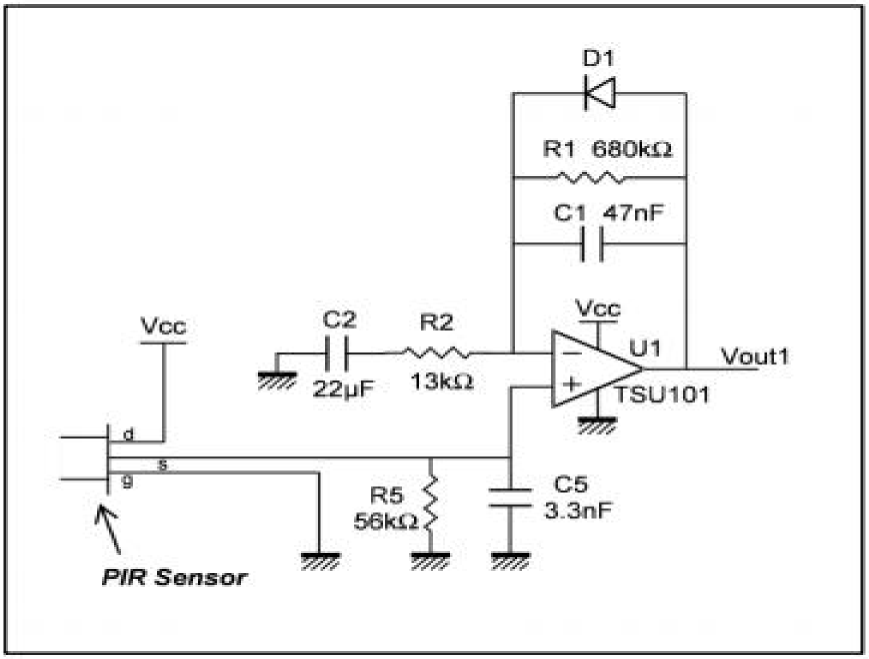
When a body, with a temperature different to ambient, moves into the detection field of a PIR sensor, the sensor provides a small AC signal. This signal is in the range of 1 mVpp. This small voltage is around a DC signal that may significantly vary from one sensor to another. For this reason, it is mandatory to cancel the DC part of the signal and to amplify the AC part. As the signal is disturbed by the environment, a noise filter is useful.

2.2.4 Three-stage architecture for detecting motion

To detect motion, a two-stage architecture based on op-amps is used to condition the signal of a PIR sensor. This architecture is explained below.

Stage 1

The first architectural stage amplifies the signal. It cancels the DC part of the signal and filters the high frequency noise that could lead to false detections

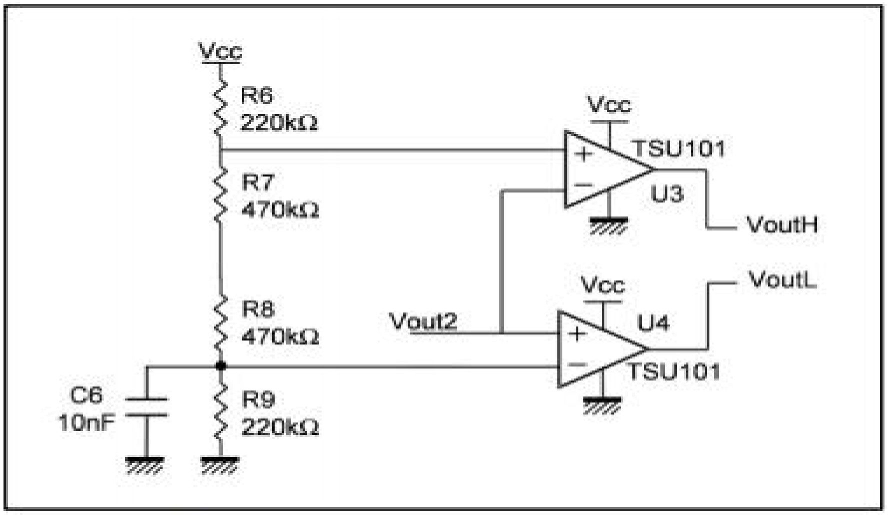


1. C = capacitor, D = diode, R = resistor, U = op-amp

Figure shows that noise is filtered thanks to the components R1 and C1. The cut-off frequency is 5 Hz (fhigh1 = 1/ (2 x π x R1 x C1)). This application does not need to work at higher frequencies because usually we are detecting human motion. The second filter is used to reject the DC part of the signal. R2 and C2 perform a high pass filter that has a cut-off frequency of 0.6 Hz (flow1 = 1/(2 x π x R2 x C2)). Since we do not amplify the DC part of the signal, the Vio, which is the input offset voltage of the op-amp, has no importance in this application. The stage gain is 53.3 (Gain1 = 1 + R1/R2). This amplification allows a usable signal that is higher than the noise level. The DC bias of the first architectural stage is determined by the sensor. To avoid saturation, the gain must not be too big because amplification is made around this common mode voltage (which is not Vcc/2). STMicroelectronics' TSU101 op-amp is a perfect fit for this architectural stage. As well as having very low consumption, the TSU101 operates with a typical current consumption of only 600 nA at Vcc = 3.3 V.

Stage2

The second and last architectural stage allows the user to perform a window comparator. As a consequence, the signal is perfectly conditioned to go to a microcontroller. Figure shows the schematic of this stage.



1. C = capacitor, R = resistor, U = op-amp

When a heat source is detected, the output of the op-amp U3 and/or U4 is in low state. The divider bridge, composed of the resistors R6, R7, R8, and R9, is used to set the voltage reference of these devices. The TSU101 op-amp is an input/output rail to rail op-amp, and there is no constraint on the input common mode voltage. Thus, while the voltage references of U3 and U4 are within the Vcc range the window comparator works. For the following example, the reference of U3 is set to 0.84 * Vcc:

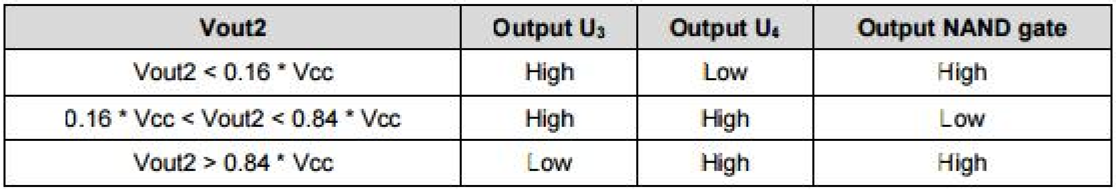
$$\frac{R_7 + R_8 + R_9}{R_6 + R_7 + R_8 + R_9} \times Vcc = 0.84 \times Vcc$$

When the signal (Vout2) is above this reference (2.77 V if Vcc = 3.3 V), the output of U3 is in low state, close to ground. U4 is used to detect when the signal is below a certain reference. In this example, the voltage reference is set to 530 mV thanks to the divider bridge composed of R6, R7, R8, and R9. The voltage reference calculation is:

$$\frac{R_9}{R_6 + R_7 + R_8 + R_9} \times Vcc = 0.16 \times 3.3 = 0.53V$$

Thus, when the signal (Vout2) is lower than 0.53 V, the output voltage of U4 is in low state (see Table). If only one digital output is required, the user may use a NAND gate connected to the outputs of U3 and U4.

Table 1: NAND truth table



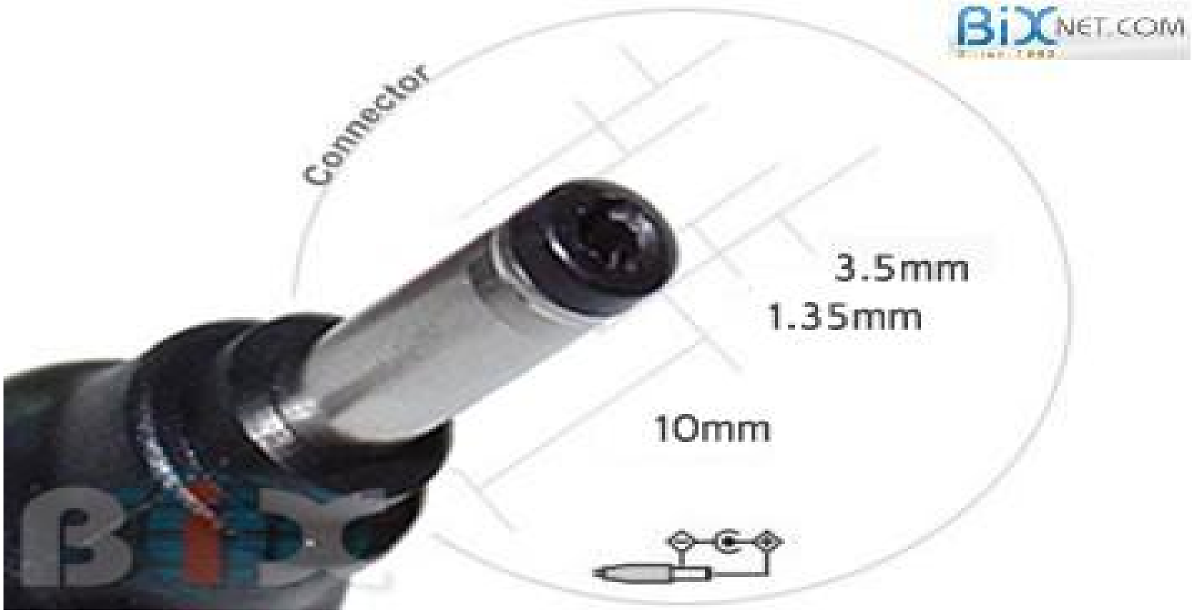
Vout2	Output U_3	Output U_4	Output NAND gate
Vout2 < 0.16 * Vcc	High	Low	High
0.16 * Vcc < Vout2 < 0.84 * Vcc	High	High	Low
Vout2 > 0.84 * Vcc	Low	High	High

2.2.5 Conclusion

PIR sensors are widely used and require op-amps to amplify and filter the signal they generate which is noisy and small in amplitude. Op-amps or comparators can be used to compare the amplified signal with threshold voltages before it goes into the I/O of a microcontroller

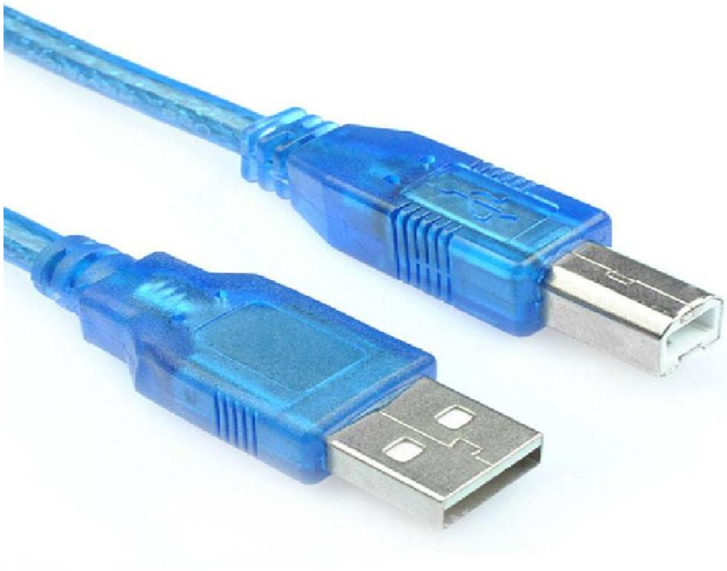
2.3. 5volt 2ampier power adapter

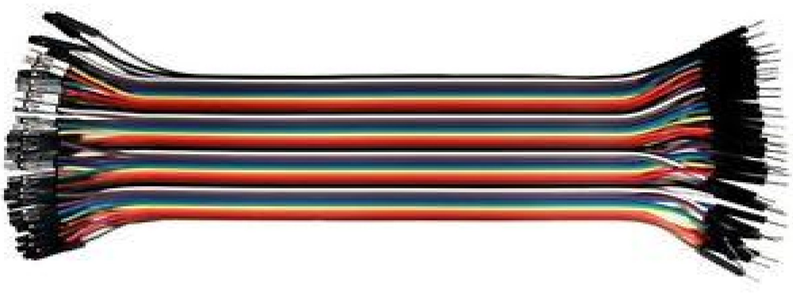
For additional power help we use 5v-2A power adapter. There are various types of power adapter that we found in market.



2.3(b) Wires (types of):

Aurduino Cable
USB 2.0 cable type A/B
Male to Male
Male to Female
Adapter power plug and play wire etc.





<u>USB cable type A/B</u>.

Standard USB 2.0 cable. Use it to connect Genuino Uno, Genuino Mega 2560, Genuino 101 or any board with the USB female A port of your computer. Cable length is approximately 178cm. Cable color and shape may vary slightly from image as our stock rotates.

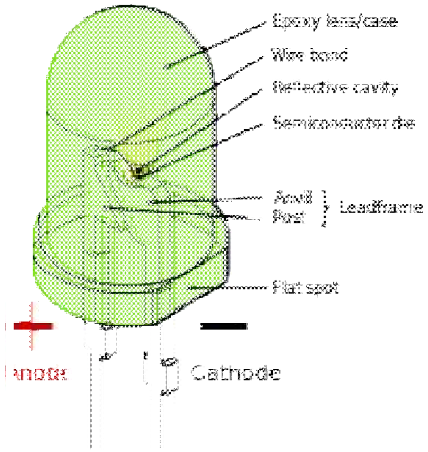
2.4 Light Emitting Diode

2.4.1 What is LED?

A **light-emitting diode** (**LED**) is a two-lead semiconductor light source. It is a p–n junction diode, which emits light when activated.When a suitable voltage is applied to the leads, electrons are able to recombine with electron holes within the device, releasing energy in the form of photons. This effect is called electroluminescence, and the color of the light (corresponding to the energy of the photon) is determined by the energy band gap of the semiconductor.

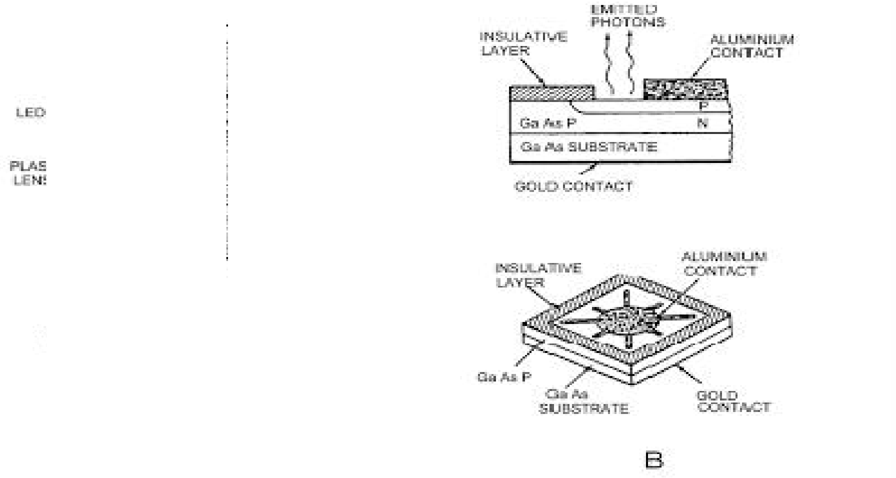
An LED is often small in area (less than 1 mm^2) and integrated optical components may be used to shape its radiation pattern.

Appearing as practical electronic components in 1962, the earliest LEDs emitted low-intensity infrared light. Infrared LEDs are still frequently used as transmitting elements in remote-control circuits, such as those in remote controls for a wide variety of consumer electronics. The first visible-light LEDs were also of low intensity, and limited to red. Modern LEDs are available across the visible, ultraviolet, and infrared wavelengths, with very high brightness.



2.4.2How Do LED Light Bulbs Work:

An LED is a small light bulb that easily fits into an electrical circuit. These light bulbs do not get hot and they are illuminated by the constant movements of the electrons located in the semiconductor material. The average life of an LED is around 50,000 hours, and it surpasses the life of the typical incandescent bulb by several tens of thousands of hours. Nowadays, you can find LED lights in a variety of applications, including LCD TVs. There is a strong reason why people are looking for cost-effective and energy-efficient lighting solutions for their homes: these lighting choices lower the utility bills and also take care of the problem of non-renewable energy sources. In the US alone, almost 90% of the energy relies on non-renewable sources. LEDs are by far the most efficient lights available on the market, and they have become so popular that they can be easily found in home-goods and hardware stores. Unfortunately, these innovative light bulbs are still a mystery to many people, because their mechanism is radically different than the one of standard light bulbs. Here you will find detailed information about how LEDs work and how are they able to generate constant and reliable lights without any maintenance.



The answer lies in the name of the light bulb: LED stands for “Light Emitting Diodes” and it refers to a special type of diode that is especially created to provide light when electricity passes through it. These diodes are basically semiconductors that are used to control the direction of the electricity flow.

The average light emitting diode does this through what is known as a “p-n junction”, which is formed of a p-type semiconductor (which contains positively charged carriers) and an n-type semiconductor (which contains electrons). The connection between these two types of semiconductors is known as a “p-n junction”. These junctions are the building blocks of every semiconductor electronic device.

Light emitting diodes are modified in order to produce a form of light radiation, and the electrons from n-type semiconductors simply connect to the carriers in the p-type semiconductor, and the electricity flow is unidirectional: it is possible only from n to p, not the other way round. When the junction is formed, electrons located in the diode change their state and start to emit photons: electrons change their orbit by moving from a higher orbit to a lower one – when they do that, they automatically lose energy in the form of light. The more energy is lost, the more powerful the photons of light. This is when the color starts to change and light is being emitted by the LED.

The color that is emitted by the LED is somehow influenced by the material of the bulb as well as the current that runs through it. The semiconductor materials can be designed to limit the amount of energy that flows through them, and this will automatically change the color of the light – this is how you obtain a multi-color LED bulb. When the LED starts to produce light, this phenomenon is known as “electroluminescence”, and this is different from any other kind of light emission because it provides no heat. When the traditional light bulbs emit light, that process is called “incandescence” because they emit heat along with the light (thus the name “incandescent light bulbs”).

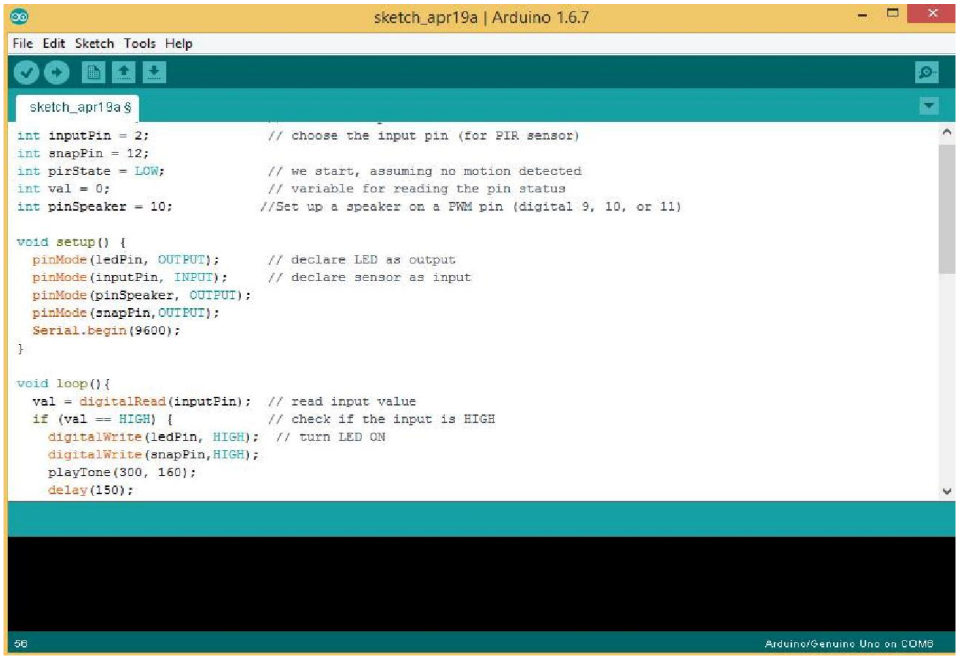
Generally speaking, LED lights are designed to emit one color of light, but over the past years manufacturers have come up with bicolor light bulbs, and the most popular colors at the moment are red, green and blue (although there is a wide variety of colored LED bulbs available on the market). In the case of large LED displays, the manufacturers use a combination of LEDs that aim to give a wide range of colors (these displays use a 3-LED pixel).

The improved intensity and the low cost have broadened the applications. The LED contains a pair of leads along with a plastic body that is encased around the leads and the semiconductor – this way, the light is emitted through the bulb plastic. However, most LED bulbs have a round shape that aims to focus the light and focus it in one particular direction. A regular LED bulb does not depend on the voltage and it requires no maintenance over the years.

2.5 Arduino Writing Sketches

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.


sketch_apr19a | Arduino 1.6.7
File Edit Sketch Tools Help
sketch_apr19a §
56
Arduino/Genuino Uno on COM8

```
int inputPin = 2;               // choose the input pin (for PIR sensor)
int snapPin = 12;
int pirState = LOW;             // we start, assuming no motion detected
int val = 0;                    // variable for reading the pin status
int pinSpeaker = 10;           //Set up a speaker on a PWM pin (digital 9, 10, or 11)

void setup() {
  pinMode(ledPin, OUTPUT);      // declare LED as output
  pinMode(inputPin, INPUT);     // declare sensor as input
  pinMode(pinSpeaker, OUTPUT);
  pinMode(snapPin,OUTPUT);
  Serial.begin(9600);
}

void loop(){
  val = digitalRead(inputPin);  // read input value
  if (val == HIGH) {            // check if the input is HIGH
    digitalWrite(ledPin, HIGH);  // turn LED ON
    digitalWrite(snapPin,HIGH);
    playTone(300, 160);
    delay(150);
```

Chapter-03

3.1 Introduction:

This project provides a guideline to the students who are new in the world of Arduino and help them to understand about PIR sensor,LED,Keypad,servo motor and how to make security system using these components. By this component a full automated system will be active and user can control it as far his/her choose.

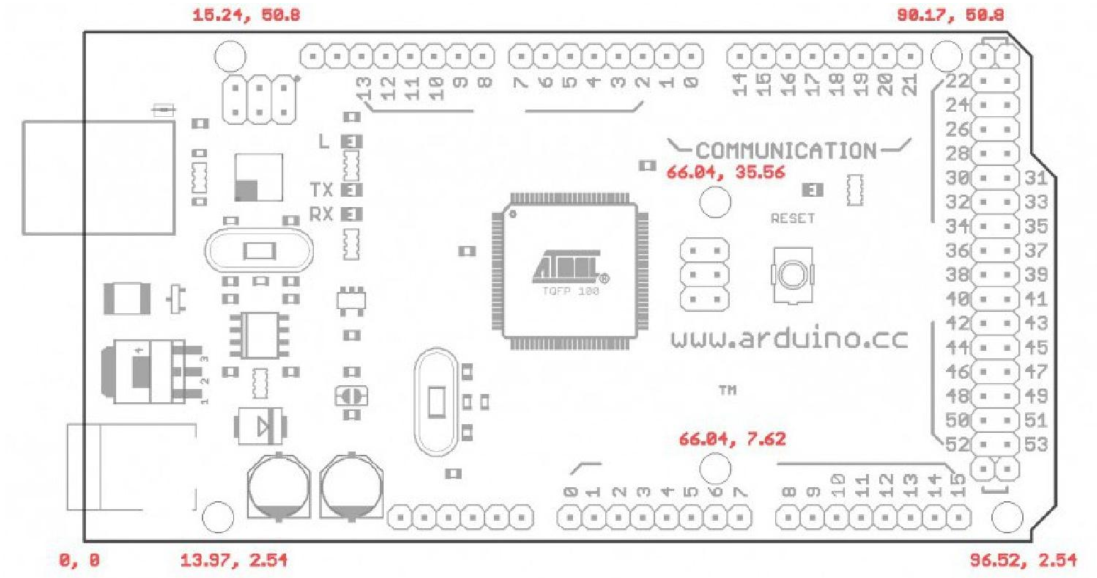
3.2 Components Used:

The designed system consists of the following components-

- Arduino Mega,
- pir sensor,
- 5V 1A Adapter,
- LED Lights,
- Arduino Sketch Software ,
- Display,
- Keypad,
- Servo Motor,
- Male to Male Connectors,
- Male to Female Connectors.
- Potentiometer etc.

3.3 Internal Circuit Construction of each Components

3.3.1 Arduino Mega



3.3.2 PIR sensor

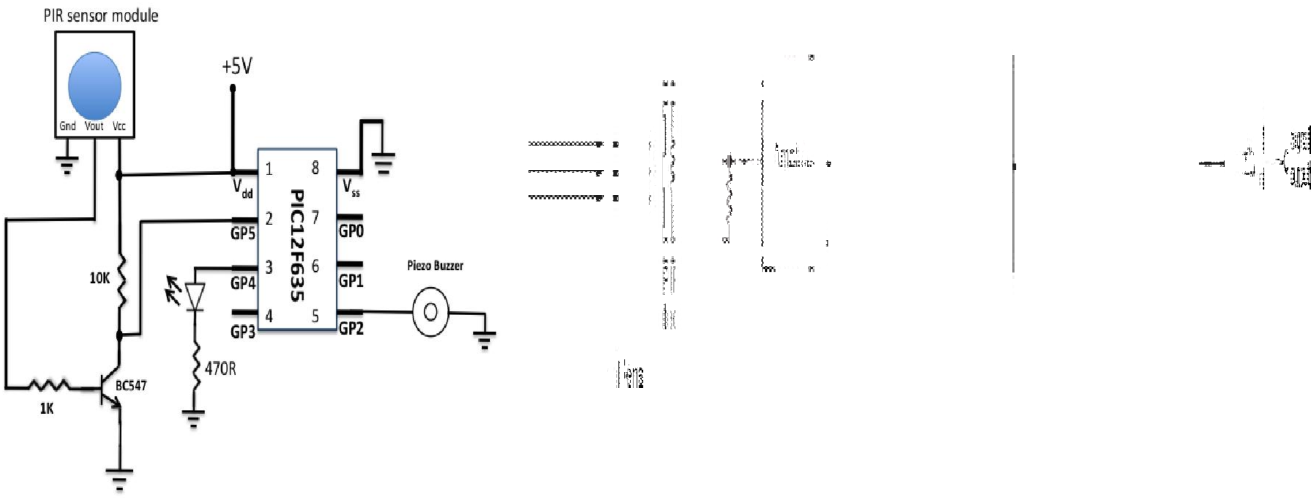


Fig: Circuit diagram for PIR sensor

3.3.3 Servo Motor

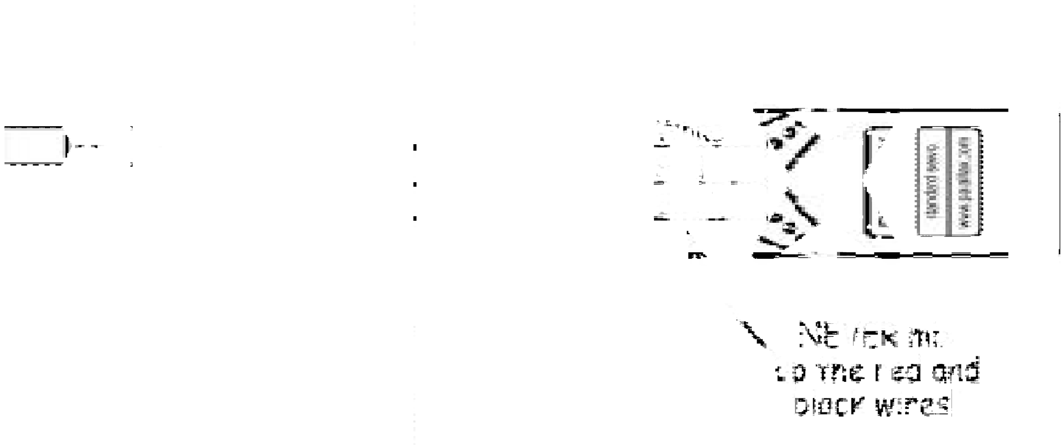


Fig: block diagram for Servo Motor

3.3.4 Display

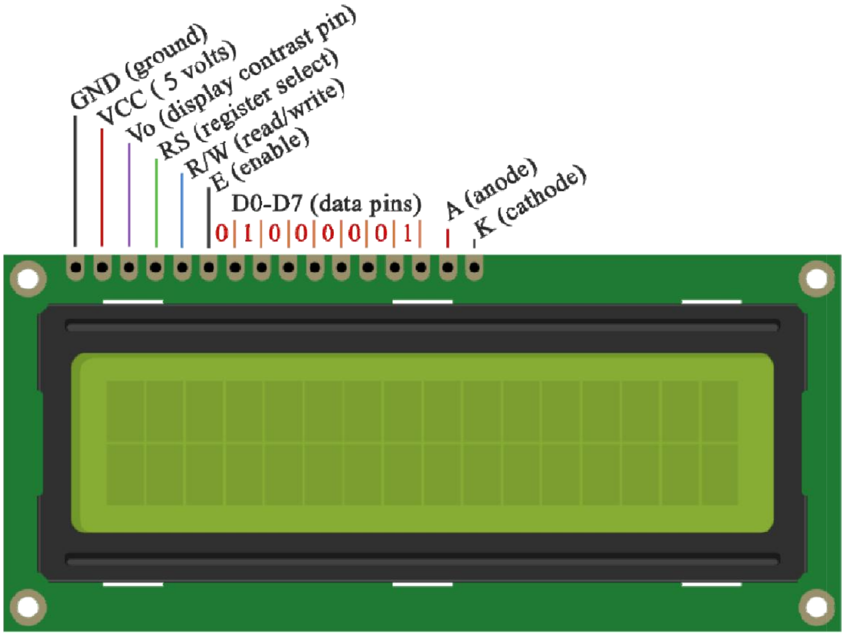


Fig: diagram for display

3.3.5 LED LIGHT

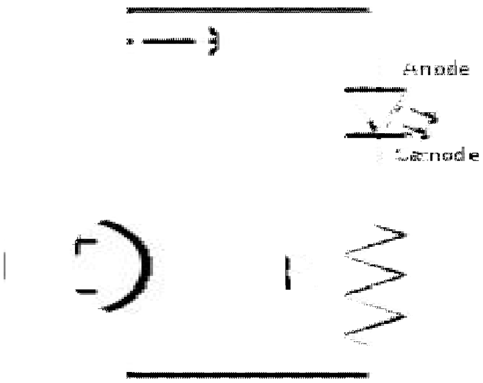
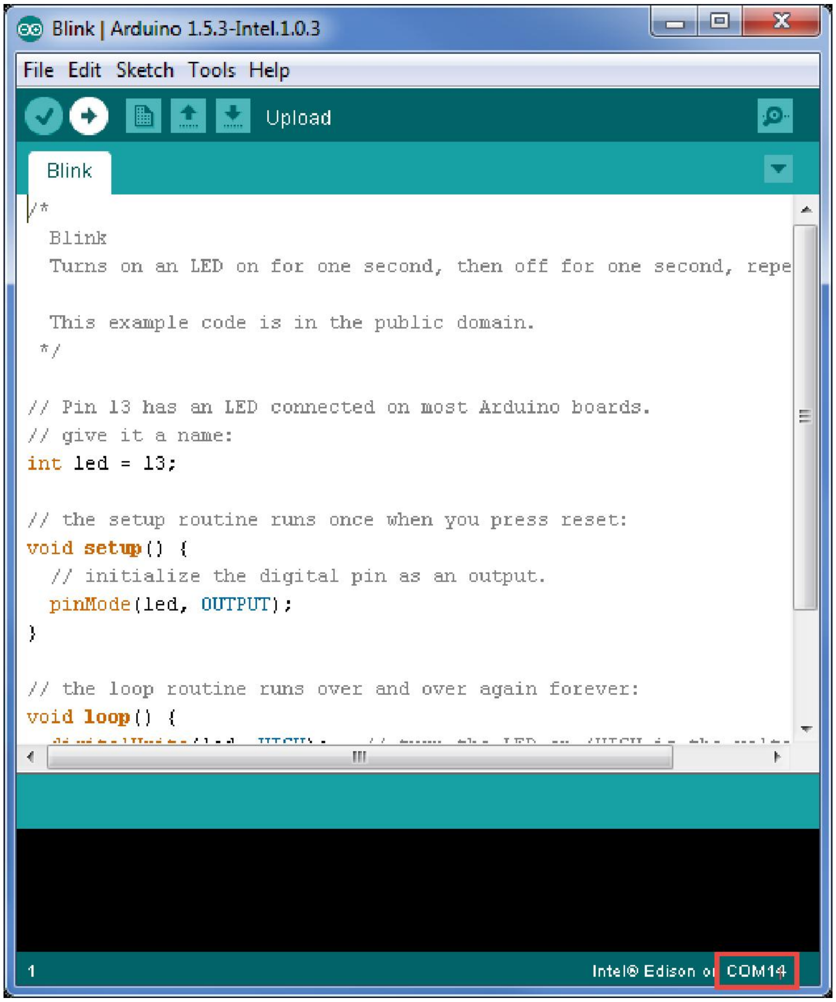


Fig: block diagram for LED Light

3.3.6Arduino Sketch Software


Blink | Arduino 1.5.3-Intel.1.0.3
File Edit Sketch Tools Help
Upload
Blink

```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repe

  This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
```

3.4 System Designand Connections:-

3.4.1 Block diagram of PIR Sensor Connection

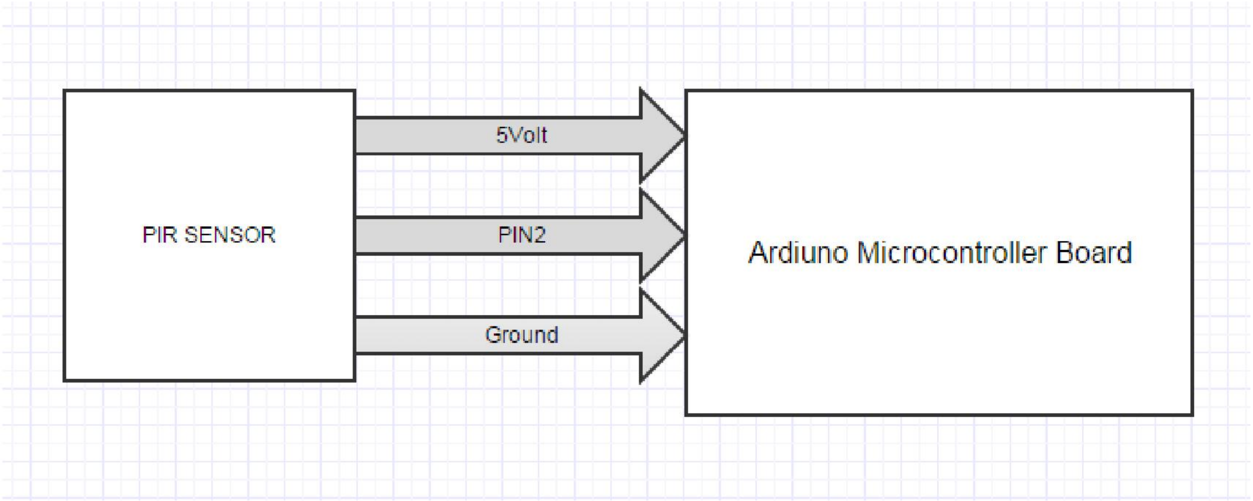


Figure: Block Diagram of PIR Sensor

3.4.2 Block diagram of Display Connections:

In here we used a pir sensor and it is directly connected to the Arduino.In the pir sensor there are 3 pin. First one is connected to the 5 volt of the Arduino Microcontroller second pin is connected to the pin 2 and 3rd one is connected to the ground of the Arduino Microcontroller.

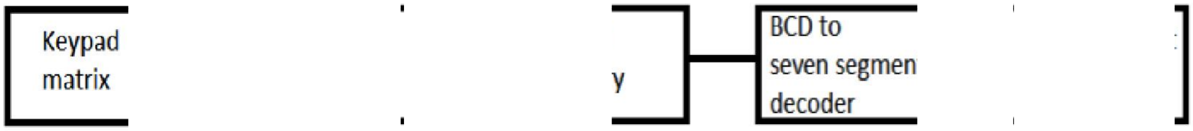


Figure: Block Diagram of Display Connections

We used a display and it is also directly connected to the Arduino.In the display,1 to 5 pins are connected with Aurduino's 11 to 15 pins. First one is connected to the pin 13 of the Arduino Microcontroller and the 2ndone is connected to the ground of the Arduino Microcontroller.

3.4.3 Block diagram of LED Connection:

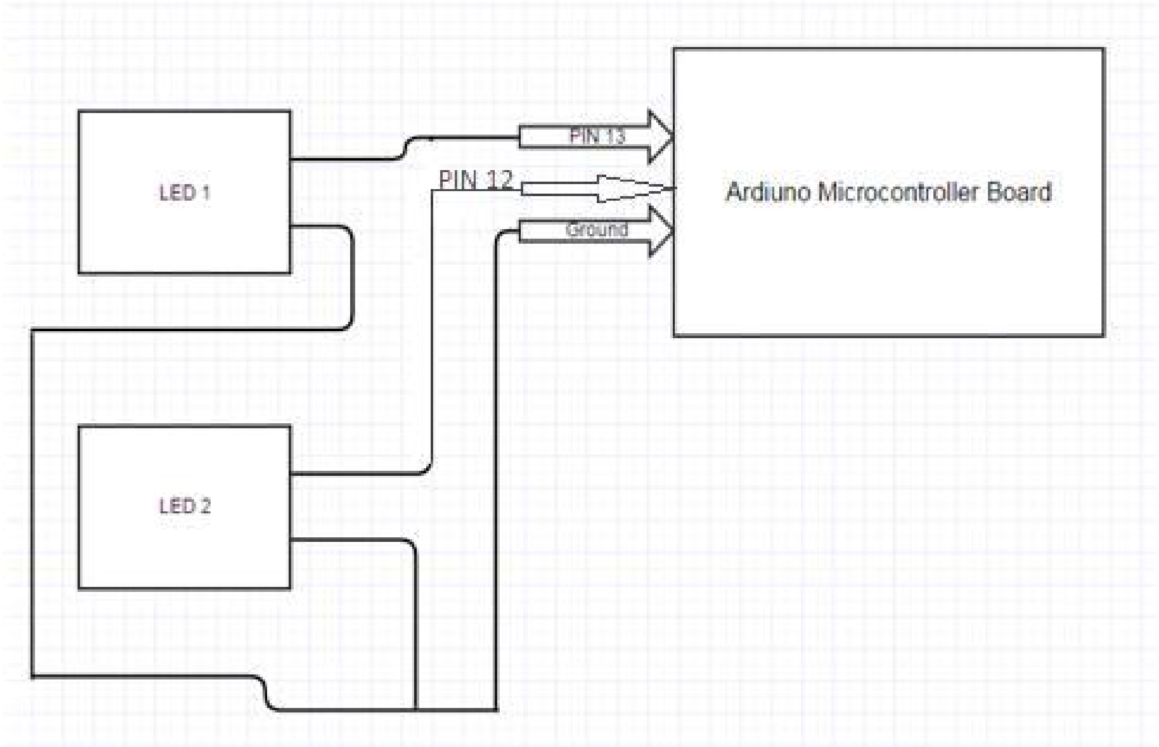


Figure: Block Diagram of 2 LED connected with Arduino

In this connection both of the Leds pin 1 is connected to the pin 13 of the Arduino and both of the Leds pin 2 is connected to ground of the Arduino Microcontroller.

3.4.4 Block diagram of 5V 2A Adapter:

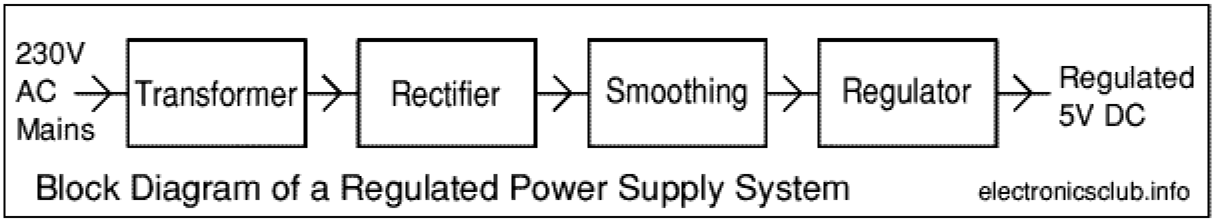


Figure: Block diagram of 5V 1A Adapter

5V 1A adapter is used for extra power supply for Arduino and Display so that the program run and display successfully.

3.5 Block Diagram of Servo Motor:-

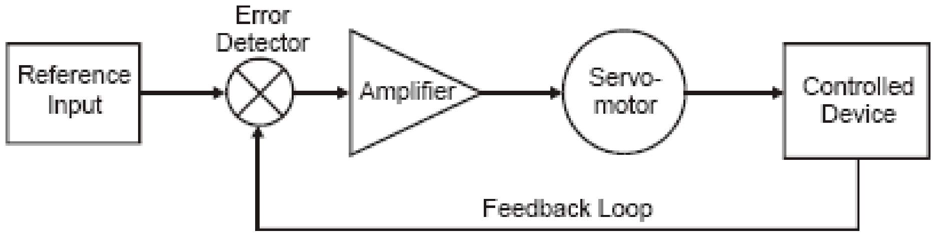


Figure 3-44. Typical servo system block diagram

Figure: Block diagram of Servo Motor

3.6 Block Diagram of Full components:-

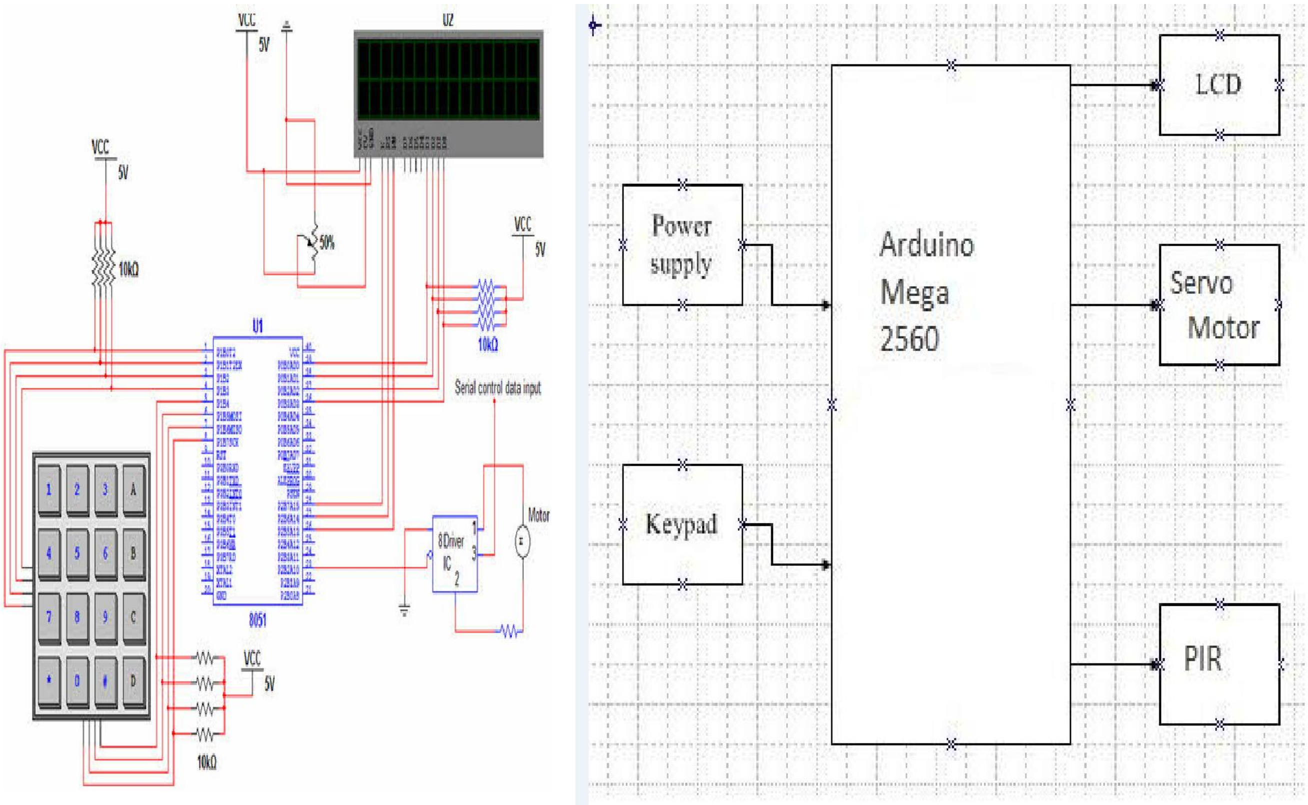


Fig: Block Diagram of full component connection

To make up the system the separate components are combine together. we used a pir sensor and it is directly connected to the Arduino. In the PIR sensor there are 3 pin. First one is connected to the 5 volt of the Arduino Microcontroller second pin is connected to the pin 2 and 3rd one is connected to the ground of the Arduino Microcontroller. A servo motor and it is also directly connected to the Arduino. In the Servo Motor there are 2 pin. First one is connected to the pin 13 of the Arduino Microcontroller and the 2nd one is connected to the ground of the Arduino Microcontroller. We use two LED light, and both of the Leds pin 1 is connected to the pin 13 of the Arduino and both of the Leds pin 2 is connected to ground of the Arduino Microcontroller. In the port of relay NO and pole is connected the selfie stick switch and port A and B is connected to the pin 12 and ground respectively of the Arduino Microcontroller.A selfie stick supported smart phone is use for captured the picture of moving object. It is connected to the Relay. Here we use arduino mega 2560 as a heart of it.

Chapter-04

4.1 Implementation and Result

After hard working, all the parts are connected as circuit design. Then we upload the programming code as we compiled in the Arduino and we get positive result. At last we see that it works properly according to our design.

4.1.1 System in Idle Mode (Circuit design):

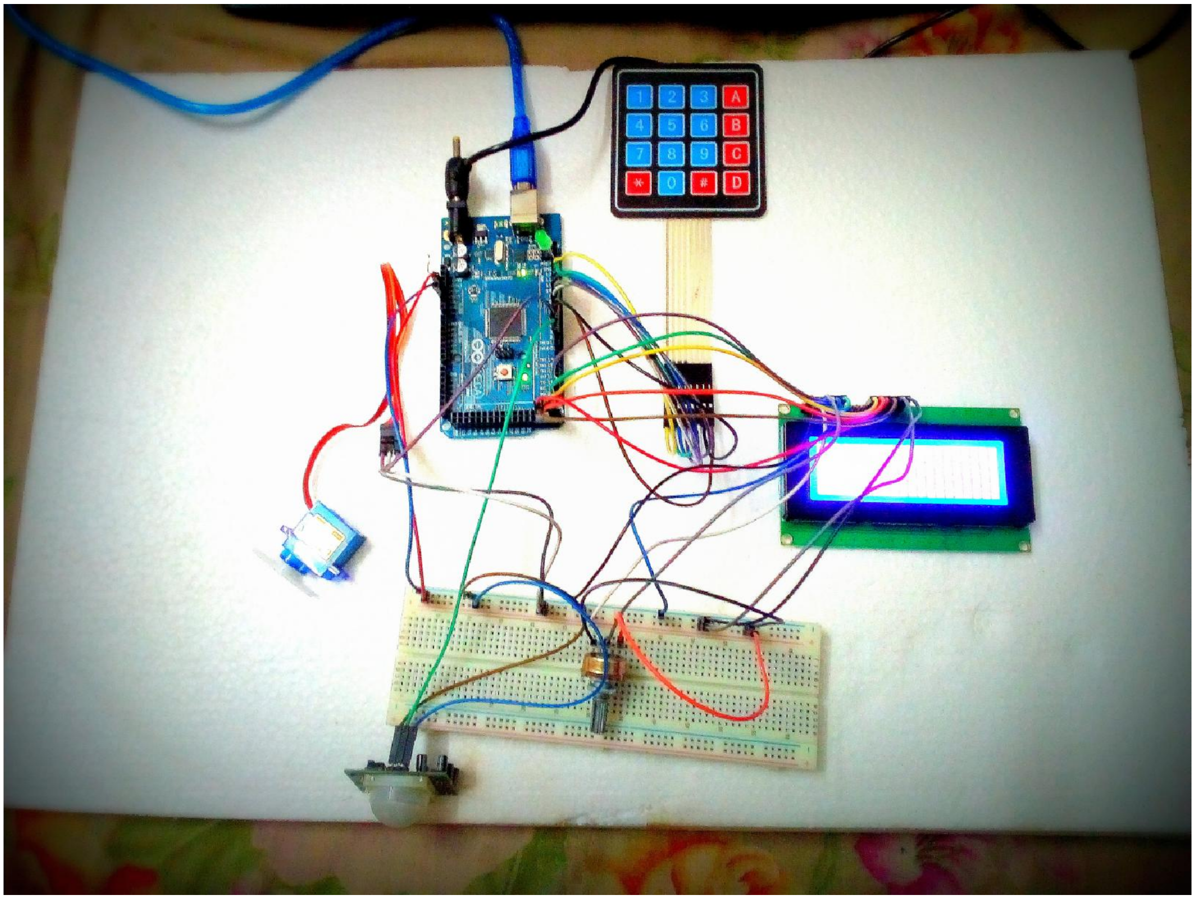


Fig: Ideal mode for circuit design

4.1.2System in Active Mode: (Circuit design):

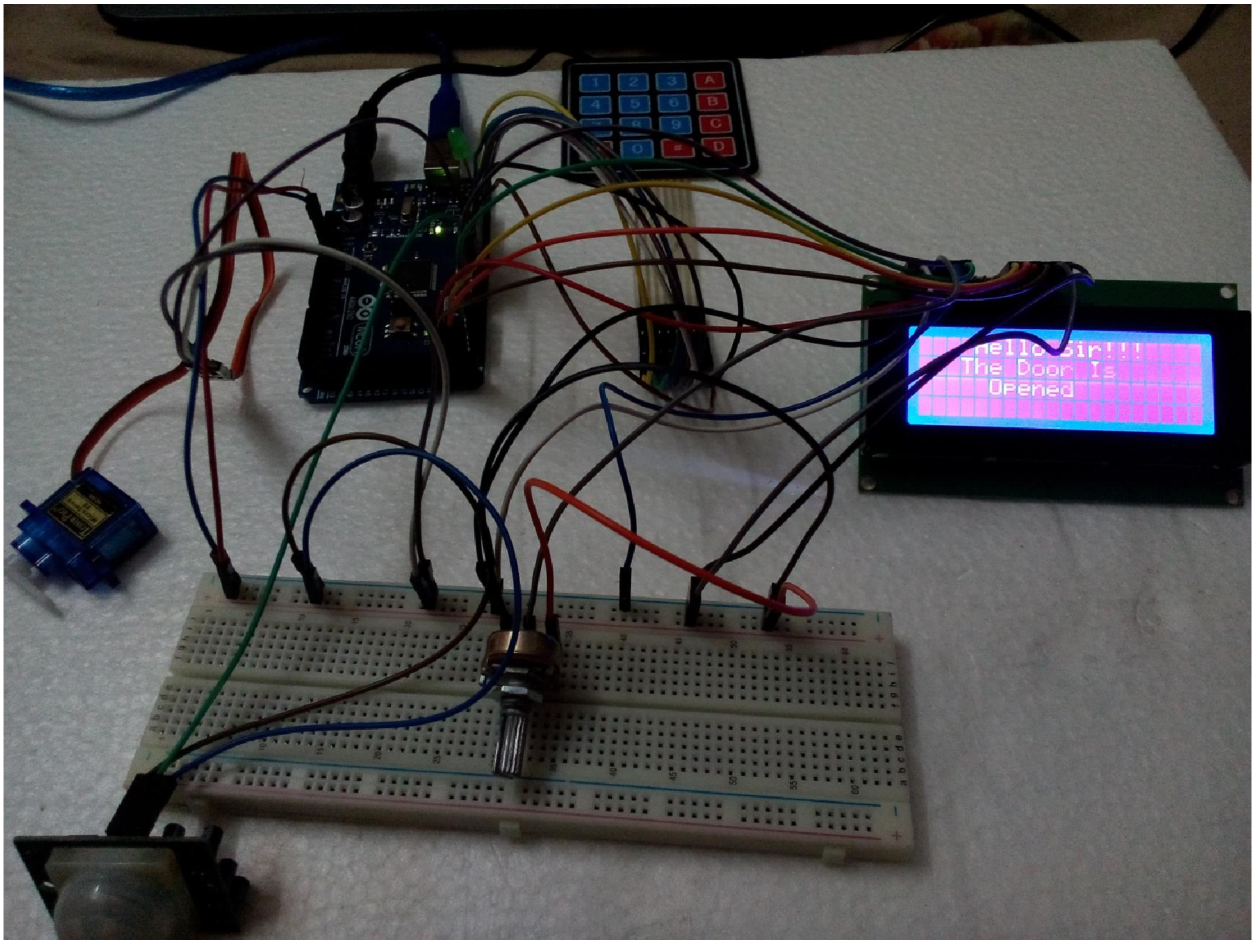


Fig: Active mode for our project (circuit design)

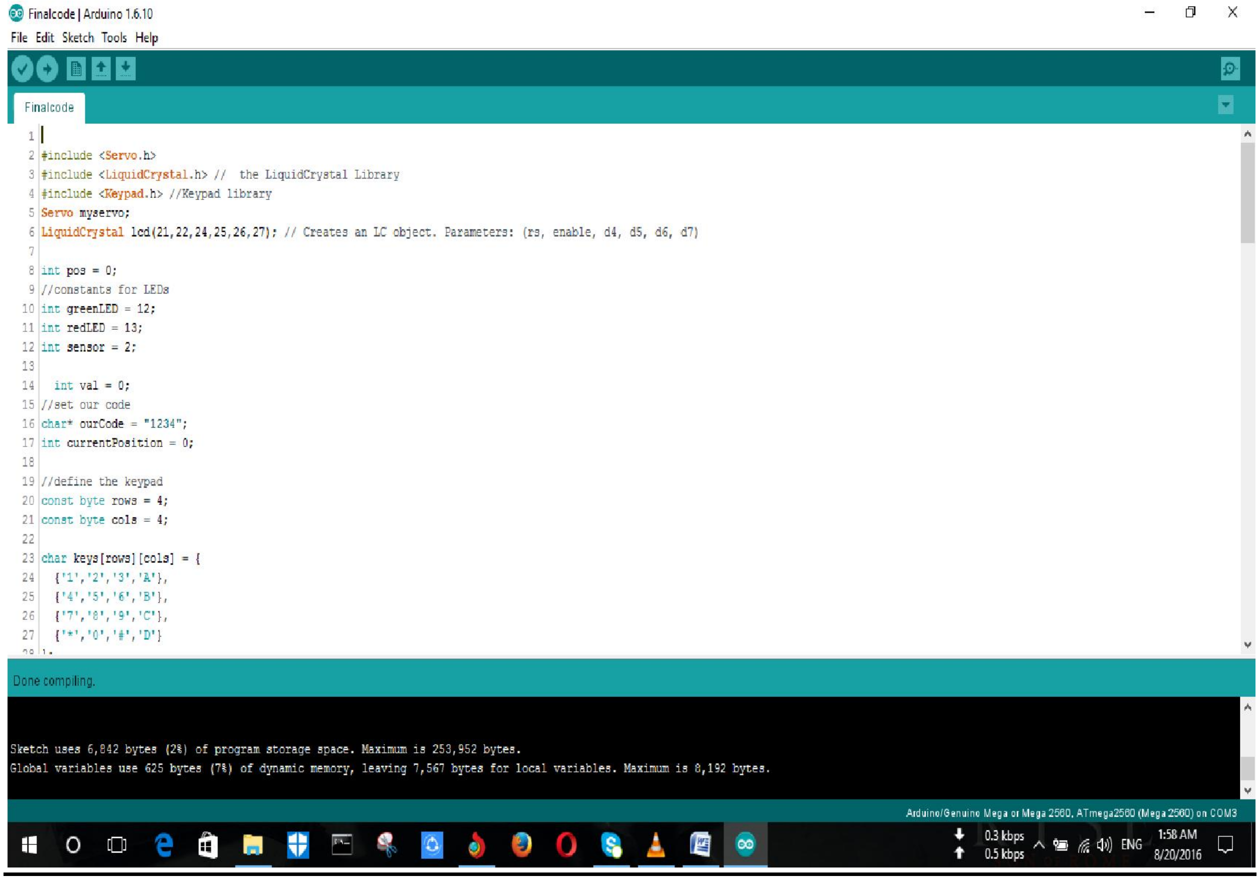
4.1.3Practical set-up and Result:

After designing the circuit we still had a lot of challenges ahead. In the above picture it is shown that in a home there is the system is applied for security purpose. When a moving object is moving in the sensor range it take that as system input signal. When getting the input signal the Led lights are lilt and an alarm is generated as well as the phone camera captured that moving object automatically. In the 1st picture there is no moving object and system is in idle mode, but in the 2nd picture a moving object is passing through the system and the leds are lits. As a result alarm is active. So user can easily find what happen on the home. So it has a great impact in the case of security.

Chapter –05

Programming Codes

Final Code Screenshot:


Finalcode | Arduino 1.6.10
File Edit Sketch Tools Help
Finalcode
Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM3
0.3 kbps
0.5 kbps
ENG
1:58 AM
8/20/2016

```

#include <Servo.h>
#include <LiquidCrystal.h> //  the LiquidCrystal Library
#include <Keypad.h> //Keypad library
Servo myservo;
LiquidCrystal lcd(21,22,24,25,26,27); // Creates an LC object. Parameters: (rs, enable, d4, d5, d6, d7)

int pos = 0;
//constants for LEDs
int greenLED = 12;
int redLED = 13;
int sensor = 2;

  int val = 0;
//set our code
char* ourCode = "1234";
int currentPosition = 0;

//define the keypad
const byte rows = 4;
const byte cols = 4;

char keys[rows][cols] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
```

Done compiling.

```
Sketch uses 6,842 bytes (2%) of program storage space. Maximum is 253,952 bytes.
Global variables use 625 bytes (7%) of dynamic memory, leaving 7,567 bytes for local variables. Maximum is 8,192 bytes.
```

CODES:

```
#include <Servo.h>
#include <LiquidCrystal.h> //  theLiquidCrystal Library
#include <Keypad.h> //Keypad library
Servo myservo;
LiquidCrystallcd(21,22,24,25,26,27); // Creates an LC object. Parameters: (rs, enable, d4, d5, d6, d7)

intpos = 0;
//constants for LEDs
intgreenLED = 12;
intredLED = 13;
int sensor = 2;

intval = 0;
//set our code
char* ourCode = "1234";
intcurrentPosition = 0;

//define the keypad
const byte rows = 4;
const byte cols = 4;
```

```
char keys[rows][cols] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};

byterowPins[rows] = {11,10,9,8};
bytecolPins[cols] = {7,6,5,4};

Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins, rows, cols);




void setup()
{

  //setup and turn off both LEDs
pinMode(redLED, OUTPUT);
pinMode(greenLED, OUTPUT);
digitalWrite(redLED, LOW);
digitalWrite(greenLED, LOW);
```

```
myservo.attach(3);

Serial.begin(9600);

lcd.begin(16,4);
lcd.clear();
}

void loop()
{
val = digitalRead(sensor);   // read sensor value
Serial.println(val);

lcd.clear();
lcd.setCursor(6,0);
lcd.print("WELCOME!");

int l;

char key = keypad.getKey();

if (int(key) != 0) {
```

```
for (l=0; l<=currentPosition; ++l)
  {
   //lcd.print("*");
  }

if (key == ourCode[currentPosition])
   {
    ++currentPosition;
if (currentPosition == 4)
    {
unlockDoor();
currentPosition = 0;
    }
   } else {
invalidCode();
currentPosition = 0;
   }


 }
}
```

```
voidinvalidCode()
{
digitalWrite(redLED, HIGH);
lcd.clear();
Serial.println("Unauthorized");
lcd.setCursor(3,0);
lcd.print("Unauthorized!!!");
lcd.setCursor(3,1);
lcd.print("Access Denied");
delay(5000);
digitalWrite(redLED, LOW);


}

voidunlockDoor()
{
Serial.println("Door open");
lcd.clear();
lcd.setCursor(4,0);
lcd.print("Hello Sir!!!");
lcd.setCursor(3,1);
lcd.print("The Door Is");
lcd.setCursor(9,2);

lcd.print("Opened");

if (val == HIGH) {

unlockDoor();

digitalWrite(greenLED, HIGH);

}

  //add any code to unlock the door here

delay(5000);

digitalWrite(greenLED, LOW);


for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees

    // in steps of 1 degree

myservo.write(pos);              // tell servo to go to position in variable 'pos'

delay(15);                       // waits 15ms for the servo to reach the position

  }

for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0 degrees

myservo.write(pos);              // tell servo to go to position in variable 'pos'

delay(15);                       // waits 15ms for the servo to reach the position

  }


}
```

------------Code Ended------------

Chapter- 06

6.1 Discussion:-

Technology now a day has taken on a whole new meaning. The Intelligent Home Security System using Keypad Lock has been designed and developed for making our life more easy and secured. We use 5v Arduino board and 5v-2A power adapter power supply for running the machine. We also use PIR sensor for detecting object receiving. Finally, we have designed and developed the whole control system and tested using keypad. We fix all the problems encountered during the design and testing of the system. The production cost is low. Hence, our product is suitable for commercial use.

6.2 Future Work:

With home security as the most powerful driver in the adoption of smart home systems, it's no wonder that the world of smart locks are growing. The proposed system of Keypad lock and PIR sensor based home security system to detect the theft or any moving object. The future work of this system will include a centralized control for the group of particular house for current response for any kind unauthorized entry or illegal issues of system.

References

1. Book: Introduction to Arduino: A piece of cake! by Alan G. Smith
2. Book: Beginning Arduino by MICHAEL MCROBERTS
3. https://www.arduino.cc/
4. http://www.circuitstoday.com/review-of-best-arduino-books
5. Book: ArduinoFor Dummies By John Nussey.
6. http://www.engineerexperiences.com/
7. https://en.wikipedia.org/wiki/Passive_infrared_sensor
8. https://en.wikipedia.org/wiki/Servomotor
9. http://www.instructables.com/id/ARDUINO-WIRELESS-HOME-SECURITY-SYSTEM/
10. http://www.academia.edu/7481382/SMART_HOME_SCEQURITY_SYSTEM
11. http://www.techhive.com/article/2894258/how-to-keep-your-connected-home-safe-7-steps-you-can-take-to-boost-home-sec
12. https://www.indiegogo.com/projects/canary-the-first-smart-home-security-device-for-everyone#/
13. http://www.instructables.com/id/ARDUINO-WIRELESS-HOME-SECURITY-SYSTEM/
14. http://lifehacker.com/build-a-wireless-home-security-system-with-an-arduino-1619435213
15. http://lifehacker.com/build-a-wireless-home-security-system-with-an-arduino-1619435213