**Smart Locks System using IOT**

**1. INTRODUCTION**

**What is IOT**

The Internet of things (IoT) is the inter-networking of physical devices, vehicles (also referred to as "connected devices" and "smart devices"), buildings, and other item embedded with electronics, software, sensors, actuators, and network connectivity that enable these objects to collect and exchange data.

The IoT allows objects to be sensed or controlled remotely across existing network infrastructure, creating opportunities for more direct integration of the physical world into computer-based systems, and resulting in improved efficiency, accuracy and economic benefit in addition to reduced human intervention.

**1.1 General:**

We will be relying even more often on technology and less on mechanical devices. Conventional locks are not excluded, as they will be destined to leave their place to smart locks. A smart lock is an electromechanical lock which is designed to perform locking and unlocking operations on a door when it receives such instructions from an authorized device using a wireless protocol and a cryptographic key to execute the authorization process. It also monitors access and sends alerts for the different events it monitors and some other critical events related to the status of the device. Smart locks can be considered part of a smart home.

**1.2 Statement of problem:**

Mechanical locks can withstand only seconds-to-minutes of attack before defeat. Unlocked combination dials left unchanged leaves safe open. Needs manual re-locking every time. Burglars with experience can eventually break into combination locks with practice, although it takes time and effort.

Electronic Locks are Battery powered and therefore prone to power outage and automatic lock-outs. If an electrical failure happens, you can never again access the safe’s content.

If an alternative key unlocking mechanism isn’t provided for, the only way to open a permanently sealed electronic safe is to drill into it.

Limited lifespan due to the use of electronic circuits that eventually die over time.

**1.3 Related work:**

There are some factors must be considered when designing a smart home system. These factors to be used in various applications. Now-a-days smart home system playing a major task with high and low price, less and more security, less and more efficient. This system enters into a smart home technology by using some controlling mechanisms like arduino, Short message Service, Internet, and using some microchips. We used android mobile device that allows the security system to be interface with it. The OTP is used to make a communication between the mobile device and security system. The commands for locking door from mobile phone are sent properly to the door. This smart Lock is the secure, simple, and easy to manage your home’s lock. This lock needs no keys and the lock is attached inside the door and you can control it from outside the door. As the lock is inside the door there is no way to break the door by thief. You can find some Smartphone control lock which required to replace you entire lock system but it will made the thing using your traditional old lock.

**2. SYSTEM ANALYSIS**

**1.1Existing System:**

In existing systems, Even the best technology has its limitations, and smart locks are no exception. Smart locks are designed to be more secure than traditional door locks. In large apartment complexes or even for an owner having many keys for each and every apartment, car, or gate he owns, maintaining entry to authorized personnel only is a problem. Besides the costs involved in fabrication, duplication, and distribution of keys, there are security problems in case of lost keys

**1.2 Drawbacks of existing system:**

Electronic Locks are work based on power, If an electrical failure happens, you can never again access the safe’s content.

By providing access to myriad methods of communication, battery life becomes a big problem. Early on Lockitron handled this by implementing a "knock to turn on" feature for wifi access. The process of arriving at your door, knocking to turn the lock on, then authenticating, was frustrating to some of their early adopters. This was limited to wifi, but the knock sensor had issues of its own as some users found their knocks not being registered by the Lockitron.

The Bluetooth support allows for more devices, but Lockitron's Bluetooth/NFC/wifi package allows for any device - you can even unlock it via a text message.

**1.3 Proposed System:**

We everybody are concerned about the security of our home. World becoming smarter and smarter everyday and we want to control and secured our thing in smarter way.. The OTP based door lock security system is proposed to complement the drawbacks of the different security systems such as a digital door-lock and mechanical door-lock based system. The proposed method does not need user’s help to get access to the facility but the user must have the registered mobile phone to get the OTP. Then the OTP will be generated and sent to the user’s mobile phone when the user requests to access facility. And by entering the OTP through keypad on the door the door will open. In case if the mobile is not available or off then the option to open the door is to answer the security question ask by system.

**1.4 Advantages of proposed system:**

Today people are facing more problems about security in all over world, nowadays security is the most important issue in the world; So security gaining more and more importance in recent years

* A highly convenient yet highly secure smart lock.
* It provides unprecedented ease of use and control.

These devices allow to completely forget about your house keys, which then can no longer be lost in left somewhere

The password is valid once and for a short time.

**3. SYSTEM REQUIREMENTS**

Arduino is an open- source electronics prototyping platform based on flexible, easy-to-use hardware and software. It’s intended for artists, designers, hobbyists and anyone interested in creating iterative objects or environments. Arduino can sense the environment by receiving the input from a variety of sensors and can effect its surroundings by controlling rights

**3.2 Hardware** **Requirements:**

* Arduino UNO Board
* GSM Module
* Servo Motor
* Keypad
* LCD display
* Bread Board

**3.1Software Requirements:**

* OTP (One Time Password)
* Arduino IDE for developing code in C/C++
* Embedded C

**4. SYSTEM DESIGN SPECIFICATIONS**

**4.1 Arduino UNO Board:**

Arduino is an open source, computer hardware and software company, project, and user community that designs and manufactures microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world. It is an open-source electronics platform based on easy-to-use hardware and software. It's intended for anyone making interactive projects. The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers.

Most of the Arduino boards consist of an Atmel 8-bit AVR microcontroller (ATmega8, ATmega168, ATmega328, ATmega1280, ATmega2560) with varying amounts of flash memory, pins, and features. The 32-bit Arduino Due, based on the Atmel SAM3X8E was introduced in 2012. The boards use single or double-row pins or female headers that facilitate connections for programming and incorporation into other circuits.

A program for Arduino may be written in any programming language for a compiler that produces binary machine code for the target processor. Atmel provides a development environment for their microcontrollers, AVR Studio and the newer Atmel Studio. The Arduino project provides the Arduino integrated development environment (IDE), which is a cross-platform application written in the programming language Java. It originated from the IDE for the languages Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus.

A program written with the IDE for Arduino is called a sketch. Sketches are saved on the development computer as text files with the file extension .ino. Arduino Software (IDE) pre-1.0 saved sketches with the extension .pde. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU tool chain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.



**ARDUINO**

**ARDUINO FEATURES**

* ATmega328 microcontroller
* Input voltage - 7-12V
* 14 Digital I/O Pins (6 PWM outputs)
* 6 Analog Inputs
* 32k Flash Memory
* 16Mhz Clock Speed

**4.2 GSM:**

**4.2.1 INTRODUCTION:**

GSM Modem Product provides full functional capability to Serial devices to send SMS and Data over GSM Network. The GSM Modem supports popular "AT" command set so that users can develop applications quickly. The product has SIM Card holder to which activated SIM card is inserted for normal use. The power to this unit can be given from UPS to provide uninterrupted operation. This product provides great feasibility for Devices in remote location to stay connected which otherwise would not have been possible where telephone lines do not exist.

**4.2.3 INTERFACE DESCRIPTION:**

The TMAS GSM Terminal provides the following connectors for power supply, interfacing and antenna.

* 2.1mm DC power connector (centre/inner pin is positive)
* SMA connector for antenna (radio interface)
* SIM card holder



**4.2.4 GSM MODULE BENEFITS:**

* Reduce costs
* Increase revenues
* Increase reliability
* Reduce downtime
* Improve customer service
* Ease maintenance

**4.3 LCD display:**

To establish a good communication between human world and machine world, display units play an important role. And so they are an important part of embedded systems. The 16x1 display unit will have 16 characters and are in one line. The 16x2 LCD will have 32 characters in total 16 in 1st line and another 16 in 2nd line.

A Liquid Crystal Display commonly abbreviated as LCD is basically a display unit built using Liquid Crystal technology. The most basic form of electronic display available is 7 Segment displays– which have its own limitations. The next best available option is Liquid Crystal Display which comes in different size specifications. Out of all available LCD modules in market, the most commonly used one is 16×2 LCD Module which can display 32 ASCII characters in 2 lines (16 characters in 1 line)

**Interfacing 16×2 LCD to Arduino Uno:**

LCD modules form a very important part in many arduino based embedded system designs. So the knowledge on interfacing LCD module to arduino is very essential in designing embedded systems. This section of the article is about interfacing an Arduino to 16×2 LCD. Here we are using the LCD module in 4-bit mode. First, I will show you how to display a plain text messages on the LCD module using arduino and then I have designed a useful project using LCD and arduino

The ARDUINO IDE allows the user to use LCD in 4 bit mode. This type of communication enables the user to decrease the pin usage on ARDUINO, unlike other the ARDUINO need not to be programmed separately for using it in 4 it mode because by default the ARDUINO is set up to communicate in 4 bit mode. In the circuit you can see we have used 4bit communication (D4-D7)



LCD Display

**4.4 Servo Motor:**

Servomotors is nothing but a simple electric motor, controlled with the help of servo mechanism. If the motor as controlled device ,associated with servo mechanism.

A tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware



**Servo motor**

A servomotor consumes power as it rotates to the commanded position but then the servomotor rests. Stepper motors continue to consume power to lock in and hold the commanded position.

Servomotors are generally used as a high-performance alternative to the stepper motor. Stepper motors have some inherent ability to control position, as they have built-in output steps. This often allows them to be used as an open-loop position control, without any feedback encoder, as their drive signal specifies the number of steps of movement to rotate, but for this the controller needs to 'know' the position of the stepper motor on power up. Therefore, on first power up, the controller will have to activate the stepper motor and turn it to a known position.

**Specifications:**

* Weight: 9 g
* Dimension: 22.2 x 11.8 x 31 mm approx.
* Stall torque: 1.8 kgf·cm
* Operating speed: 0.1 s/60 degree
* Operating voltage: 4.8 V (~5V)
* Dead band width: 10 µs
* Temperature range: 0 ºC – 55 ºC

**4.5 Keypad:**

A keypad is a set of buttons arranged in a block or "pad" which bear digits, symbols or alphabetical letters. Pads mostly containing numbers are called a numeric keypad. Numeric keypads are found on alphanumeric keyboards and on other devices which require mainly numeric input such as calculators, push-button telephones, vending machines, ATMs, Point of Sale devices, combination locks, and digital door locks. Many devices follow the E.161 standard for their arrangement.

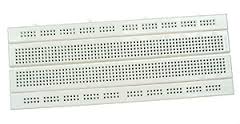
Keypads for the entry of PINs and for product selection appear on many devices including ATMs, vending machines, Point of Sale payment devices, time clocks, combination locks and digital door locks.



**Keypad**

**4.6 Bread Board :**

Bread Board is a board for making an experimental model of an electric circuit. A breadboard is a construction base for prototyping of electronics. the solderless breadboard does not require soldering, it is reusable. This makes it easy to use for creating temporary prototypes and experimenting with circuit design.



**Bread Board**

If you are working with a development board such as an Arduino, then you can simply pull power from the Arduino’s female headers. The Arduino has multiple power and ground pins that you can connect to the power rails or other rows on a breadboard.

Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate. The breadboard has strips of metal underneath the board and connect the holes on the top of the board.

**4.7 OTP(One Time Password):**

A one-time password (OTP) is a password that is valid for only one login session or transaction, on a computer system or other digital device. OTPs avoid a number of shortcomings that are associated with traditional (static) password-based authentication

The most important advantage that is addressed by OTPs is that, in contrast to static passwords, they are not vulnerable to replay attacks. A second major advantage is that a user who uses the same password for multiple systems, is not made vulnerable on all of them, if the password for one of these is gained by an attacker. A number of OTP systems also aim to ensure that a session cannot easily be intercepted or impersonated without knowledge of unpredictable data created during the previous session, thus reducing the attack surface further.

OTP generation algorithms typically make use of pseudorandomness or randomness, making prediction of successor OTPs by an attacker difficult, and also hash functions, which can be used to derive a value but are hard to reverse and therefore difficult for an attacker to obtain the data that was used for the hash. This is necessary because otherwise it would be easy to predict future OTPs by observing previous ones. Concrete OTP algorithms vary greatly in their details. Various approaches for the generation of OTPs are listed below:

* Based on time-synchronization between the authentication server and the client providing the password (OTPs are valid only for a short period of time)
* Using a mathematical algorithm to generate a new password based on the previous password (OTPs are effectively a chain and must be used in a predefined order).
* Using a mathematical algorithm where the new password is based on a chall

**4.8 Embedded C**

Embedded C is a set of language extensions for the C Programming language by the C Standards committee to address commonality issues that exist between C extensions for different embedded systems. Historically, embedded C programming requires nonstandard extensions to the C language in order to support exotic features such as fixed-point arithmetic, multiple distinct memory banks, and basic I/O operations.

Embedded C uses most of the syntax and semantics of standard C, e.g., main() function, variable definition, datatype declaration, conditional statements (if, switch case), loops (while, for), functions, arrays and strings, structures and union, bit operations, macros, etc.

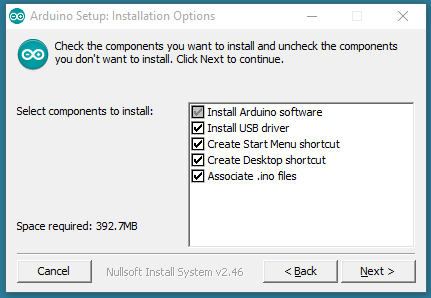
**5. SYSTEM IMPLEMENTATION**

Arduino microcontrollers come in a variety of types. The most common is the Arduino UNO, but there are specialized variations. Before you begin building, do a little research to figure out which version will be the most appropriate for your project.

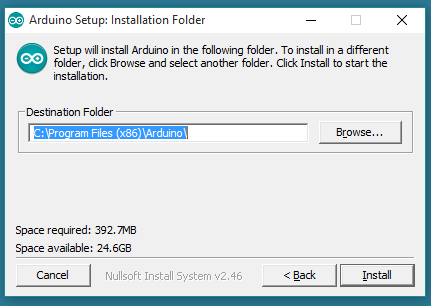
**Download the Arduino Software (IDE):**

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.



Choose the components to install



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

**STEP 1**:

To begin, you'll need to [install the Arduino Programmer](http://arduino.cc/en/main/software), aka the integrated development environment (IDE).

**STEP 2:**

Connect your Arduino to the USB port of your computer. This may require a specific USB cable. Every Arduino has a different virtual serial-port address, so you 'll need to reconfigure the port if you're using different Arduinos.

**STEP 3:**

Set the board type and the serial port in the Arduino Programmer.

**STEP 4**:

Test the microcontroller by using one of the preloaded programs, called sketches, in the Arduino Programmer. Open one of the example sketches, and press the upload button to load it. The Arduino should begin responding to the program: If you've set it to blink an LED light, for example, the light should start blinking.

**STEP 5**:

To upload new code to the Arduino, either you'll need to have access to code you can paste into the programmer, or you'll have to write it yourself, using the Arduino programming language to create your own sketch. An Arduino sketch usually has five parts: a header describing the sketch and its author; a section defining variables; a setup routine that sets the initial conditions of variables and runs preliminary code; a loop routine, which is where you add the main code that will execute repeatedly until you stop running the sketch; and a section where you can list other functions that activate during the setup and loop routines. All sketches must include the setup and loop routines.

**STEP6:**  
Once you've uploaded the new sketch to your Arduino, disconnect it from your computer and integrate it into your project as directed.

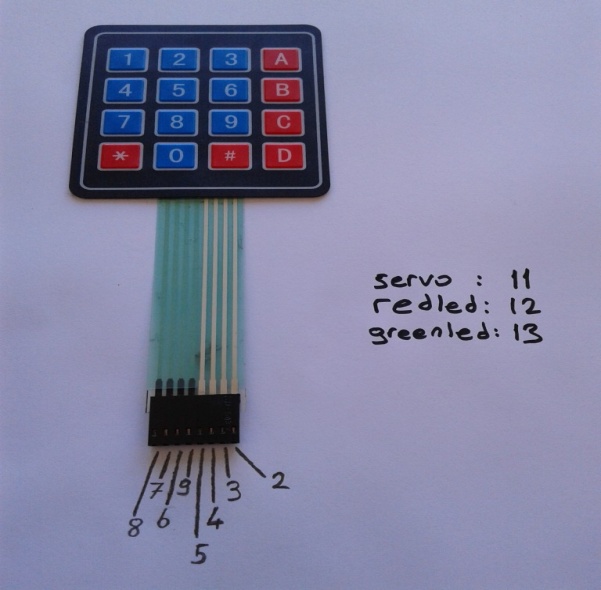
In programming part of this Project first of all in programming we includes library for Arduino and their components

The complete circuit operation is due to the software program embedded inside micro controller. All the functionalities are implemented using software logic and the program. The program performs all the functionalities like

* Takes user inputs through keypad
* Displays messages on LCD
* Give indications on 2 LEDs of password correct or incorrect system locked or unlocked
* Give alert message for different events

The program is written in C language. It is compiled using KEIL (IDE) cross compiler tool. It is compiled for generic 8051 micro controller platforms so it can be used for all different micro controllers of MCS51 family like 89C51 / 89C52 / 89S52 etc. after compiling the program the HEX file is generated. That HEX file is loaded into the internal FLASH (EEPROM) of micro controller using any suitable EEPROM programmer.

**Takes user inputs through keypad:**

****

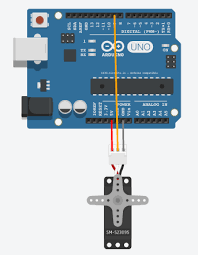
It’s a 4\*4 keypad is used to give the input by connecting with a pin number 8,7,6,9,5,4,3,2 by connecting to Arduino .In this first 4 pins are

**SERVO MOTOR:**

Most servo motors have the following three connections:

* Black/Brown ground wire.
* Red power wire (around 5V).
* Yellow or White PWM wire.

In this experiment, we will connect the power and ground pins directly to the Arduino 5V and GND pins. The PWM input will be connected to one of the Arduino's digital output pins.

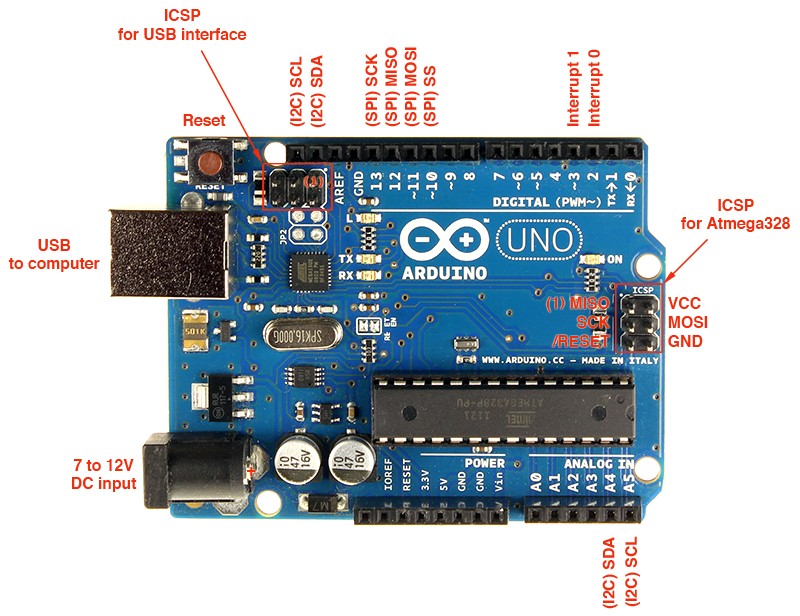
****

**System Testing**

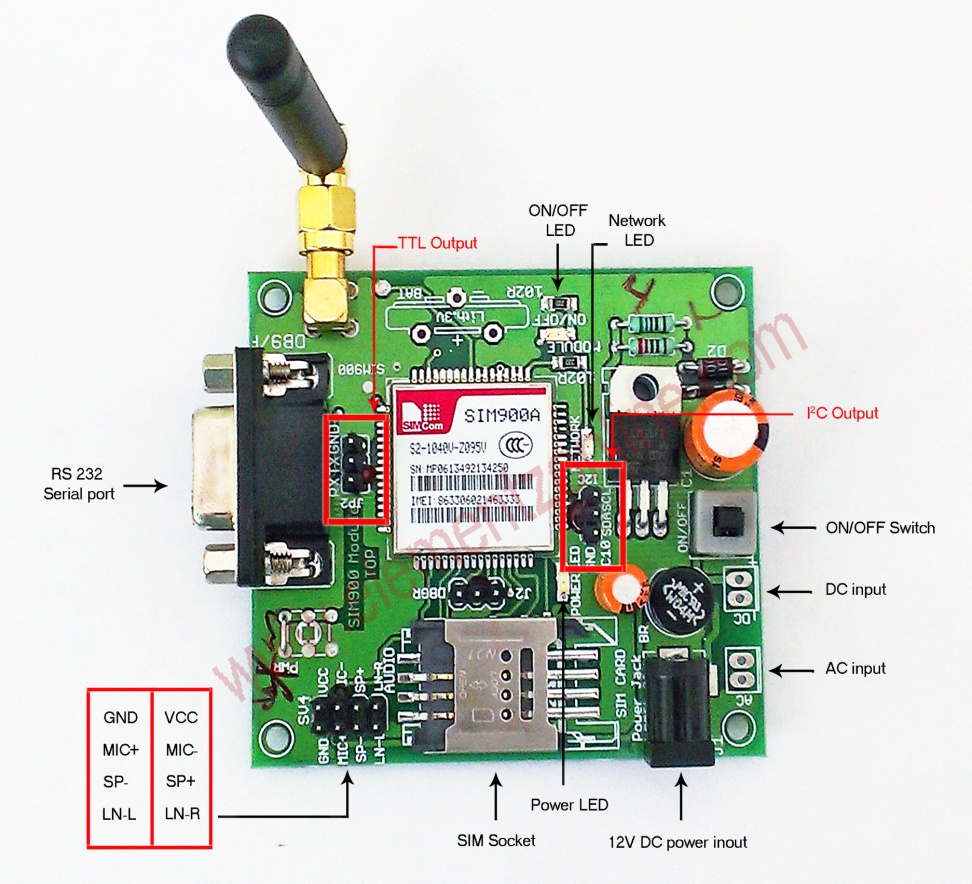
**5.1 Introduction**

After finishing the development of any computer based system the next complicated time consuming process is system testing. During the time of testing only the development company can know that, how far he user requirements have been met out and so on.

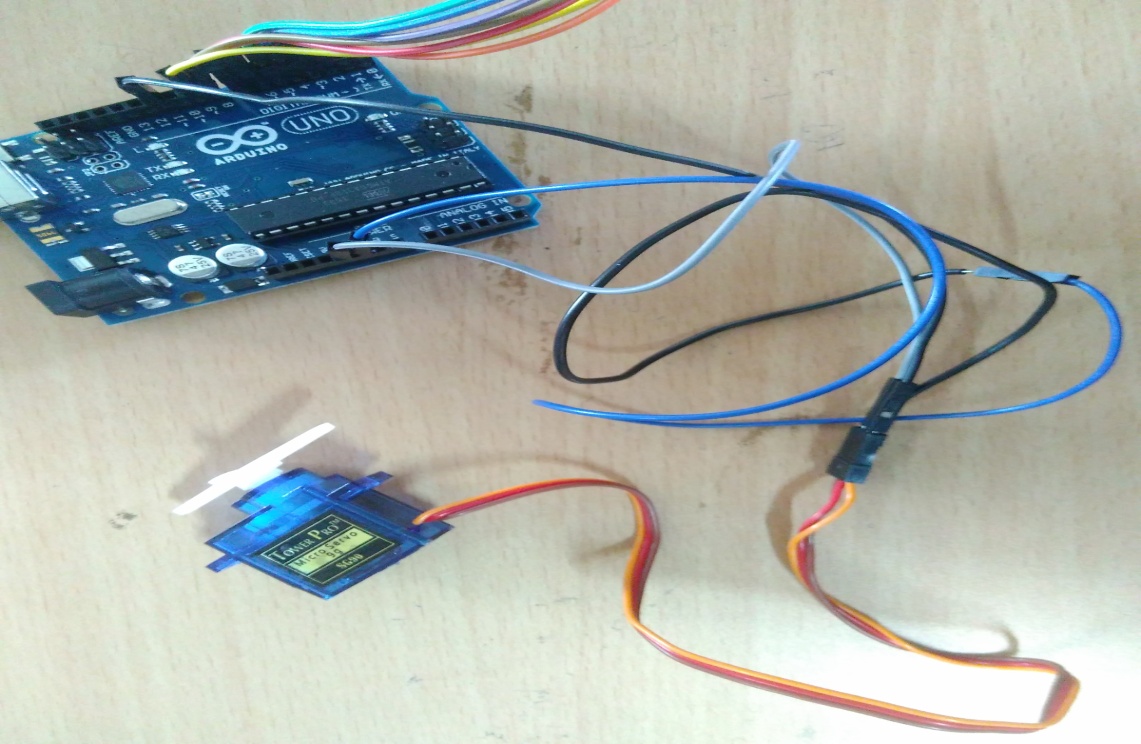
**SCREENSHOTS**

****

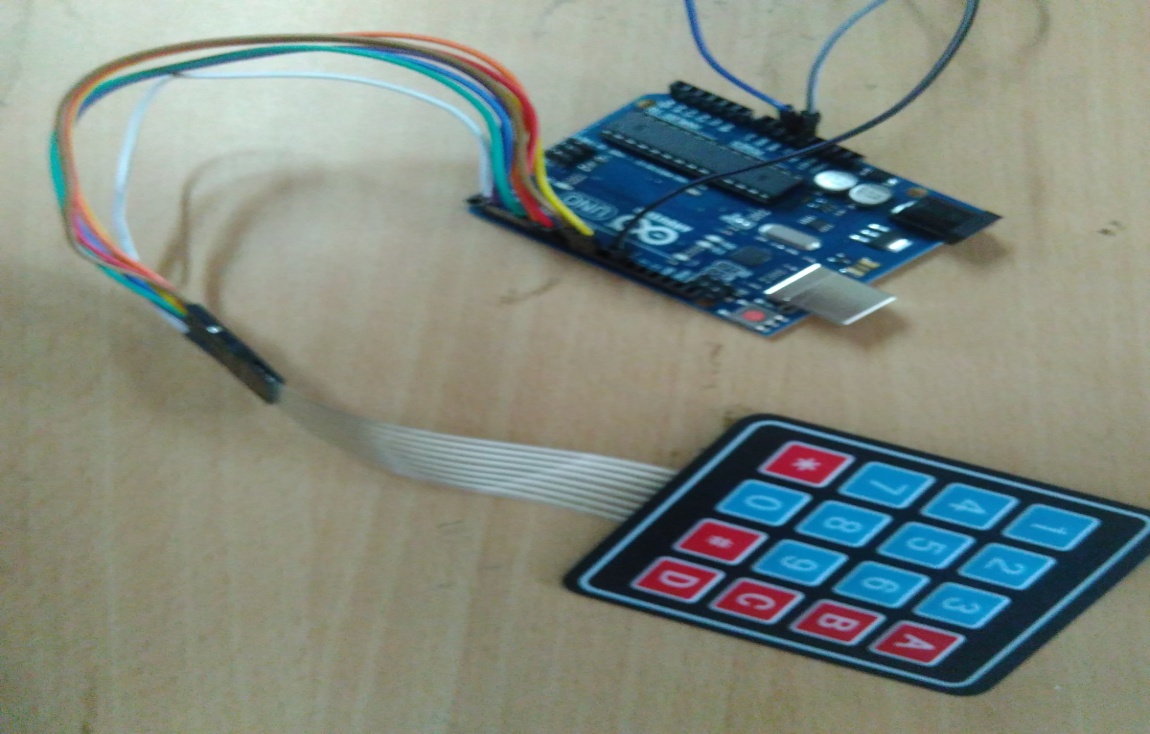
**Arduino pin description**

****

**GSM Module pin description**

****

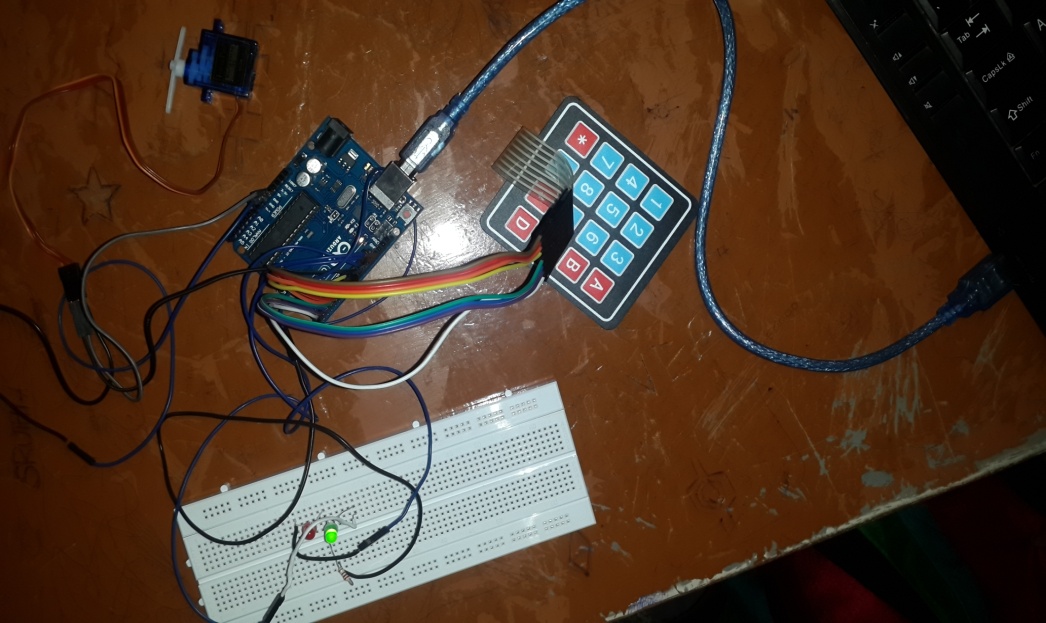
**Arduino connection with servo motor**

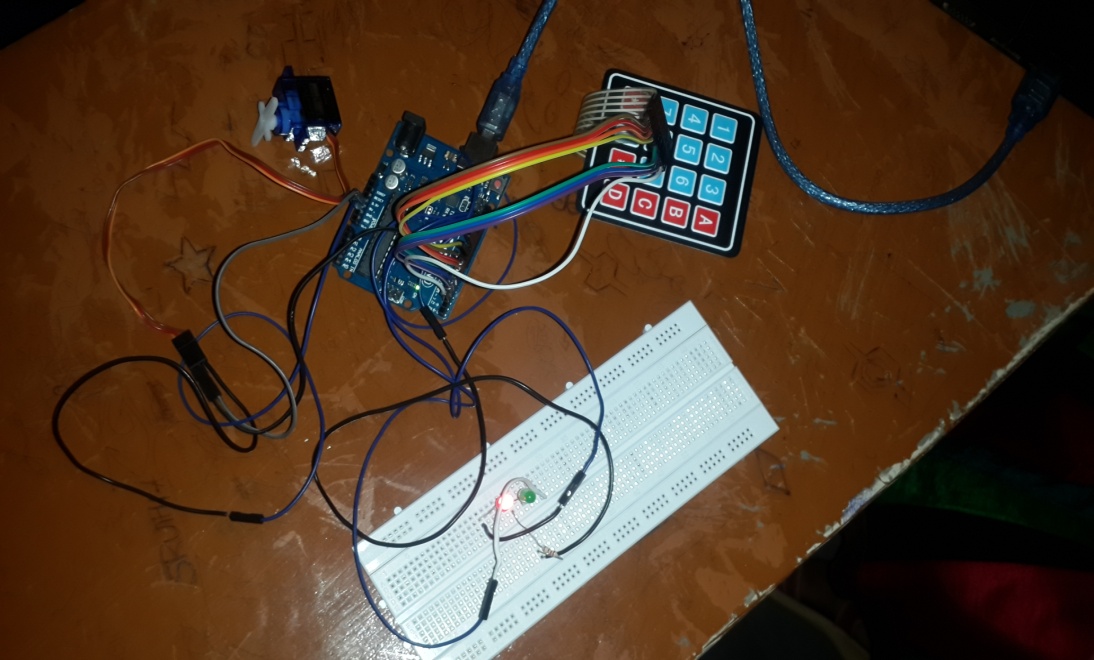
****

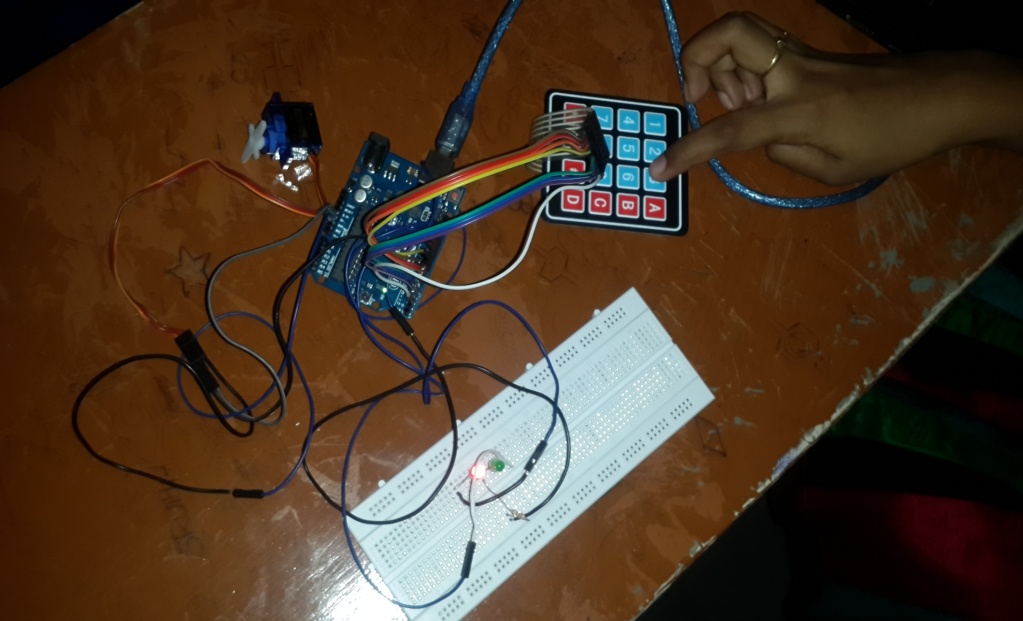
**Arduino connection with the Keypad**

****

**Arduino connection with LCD**

****

****

****

****

****

**CONCLUSION**

With the development of digital technology, many services using OTP have been developed. We have implemented a door lock security system using OT ,gsm sensor. But it observed that this system is most secure than currently available many systems based on card recognition, biometric recognition ,passward based etc. Also the installation cost is low. It is easy to operate and more secure for multi users.The problem of loss or theft which may occurs sometimes in available systems were solved by using OTP value on user’s mobile phone in our proposed system.

**FUTURE ENHANCEMENT**

* We can add fire, wind and LPG sensors so that in case of Accident, the doors will automatically open
* We can add fingerprint sensor so entry will be allowed for the authorized person using their fingerprints

**BIBLIOGRAPHY**

* Santosh.B.Panjagal, M Lakshmipathy , “DESIGN AND IMPLEMENTATION OF ADVANCED SECURITY SYSTEMBASED ON ONE-TIME PASSWORD FOR HIGHLY SECURE ZONES” , International Journal of Electronics and Communication Engineering & Technology (IJECET),ISSN 0976 – 6464(Print), ISSN 0976 – 6472(Online) Volume 4, Issue 4, July-August (2013).
* Ilkyu Ha, “Security and Usability Improvement on a Digital Door Lock System based on Internet of Things” ,International Journal of Security and Its Applications, Vol.9, No.8 (2015).
* SafaHamdare, VarshaNagpurkar, JayashriMittal , “Securing SMS Based One Time Password Technique from Man in the Middle Attack” , International Journal of Engineering Trends and Technology (IJETT) , Volume 11
* [**https://www.arduino.cc/en/Reference/HomePage**](https://www.arduino.cc/en/Reference/HomePage)
* [**https://programmingelectronics.com/how-to-use-and-understand-the-arduino-reference/**](https://programmingelectronics.com/how-to-use-and-understand-the-arduino-reference/)