**IOT BASED BANK SECURITY SYSTEM**

**A MAJOR PROJECT REPORT**

***Submitted By***

# DANY ROY

# KOUSALYA T

# SWETHA M

***In partial fulfillment for the award of the degree of***

# BACHELOR OF ENGINEERING

**IN**

ELECTRONICS AND COMMUNICATION ENGINEEING



# SRM INSTITUTE OF SCIENCE & TECHNOLOGY RAMAPURAM: CHENNAI 600089

**JUNE-2020**

SRM INSTITUTE OF SCIENCE & TECHNOLOGY RAMAPURAM

# BONAFIDE CERTIFICATE

Certified that this project report **“IOT BASED BANK SECURITY SYSTEM**

**”** is the bonafide work of **“DANY ROY , KOUSALYA T , SWETHA M”** who carried out the project work under by supervision

**SIGNATURE SIGNATURE**

**R.M.RANI (HOD) S.ABINAYAA(SUPERVISOR)**

**Professor & Head PROJECT SUPERVISOR**

Electronics and communication engineering Information Technology Srm Institute of Science & technology, Srm Institute of Science &

Technology,

Ramapuram Campus, Ramapuram Campus,

Chennai-600089 Chennai-600089

# ACKNOWLEDGEMENT

We take the privilege to extend our hearty thanks to the head of the department, **R.M.RANI**, Professor, for his suggestions, support and encouragement towards the completion of this project with perfection.

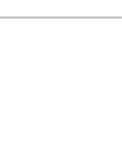
We thank our project coordinator **Mrs. KAVITHA** ,Asst professor, and our internal guide **Ms. ABINAYAA**, Project supervisor, for his/her timely help and guidance throughout the overall process of the project.

We would like to express our sincere thanks to all of our staff members of the department of Information Technology who gave many suggestions for time to time that made our project work better and finished well.

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTER** | **TITLE** | **PAGE NO.** |
|  | **ABSTRACT** | **6** |
|  | **LIST OF FIGURES** | **7** |
| **1** | **INTRODUCTION** | **8-11** |
|  | 1.1 GOALS AND OBJECTIVES | **8** |
|  | 1.1.1 METHOD | **8-9** |
|  | 1.2 WORKING PRINCIPLES | **10** |
|  | 1.3 EXISTING MODELS | **10-11** |
|  | 1.4 SOFTWARE TOOLS | **11** |
| **2** | **SOFTWARE REQUIREMENTS** | **12-16** |
|  | 2.1 EMBEDDED C | **12-13** |
|  | 2.2 MPLAB / PIC CCS | **13-15** |
|  | 2.3 DOTNET | **15-16** |
| **3** | **ALGORITHM** | **17-18** |
|  | 3.1 ARDUINO BASED ALGORITHM | **17,18** |

|  |  |  |
| --- | --- | --- |
| **4** | **MODULES** | **19-25** |
|  | 4.1 DATA COLLECTION | **19-23** |
|  | 4.2 DATA PREPROCESSING | **24-25** |
| **5** | **IMPLEMENTATION** | **26-51** |
|  | 5.1 RESULT ANALYSIS | **26-27** |
|  | 5.2 CONCLUSION | **27** |
|  | 5.3 FUTURE ENHANCEMENTS | **28** |
|  | APPENDIX 1 | **29-49** |
|  | APPENDIX 2 | **50-51** |



**ABSTRACT**

Today security is a main issue for protecting the resources. Security is

important because risk of intrusion and theft has become increasing. Security is also necessary for protecting banks from fire and other abnormal activities. Many people are using various types of security systems. We have found that most of the security systems are developed only for alarm using microcontrollers; in our research we use Microcontrollers with different sensors (PIR, Smoke or Fire, IR and Gas) as observatory to detect or identify intruder or abnormal activities inside the bank and ATM. The main aim of this research is to design a system for alerting theft and to auto arrest the thief in bank or ATM itself from centralized monitoring unit. The purpose of the system is to design a smart and centralized monitoring and control system using IOT technologies.

Bank Security System, IOT based security system, monitoring system, intruder detection system, web monitoring system.

# LIST OF FIGURES

FIG 1 BLOCK DIAGRAM OF MC

FIG 2 FRAME WORK CLASS LIBRARY(FCL)

FIG 3 Arduino based Bank Security System by releasing Knock- Out Gas

FIG 4 PIN DESCRIPTION OF 8051 FIG 5 WIRELESS CAMERA

FIG 6 PIR SENSOR FIG 7 FIRE SENSOR

FIG 8 OUTLOOK OF THE PROJECT

# CHAPTER 1 INTRODUCTION

* 1. **GOALS AND OBJECTIVES**

For a common individual the bank infers a spot which addresses a best component of security. Reliably we are drawn in with banking trade. To confirm our exorbitant pearls, basic reports or cash, we use to use bank locker rooms. It has transformed into a basic bit of our life. To get by in this forceful world and for a predictable improvement, the budgetary business needs to give an abnormal state of security. Because of the open interest every day new branches are opening. The more number of branches required more noteworthy security.

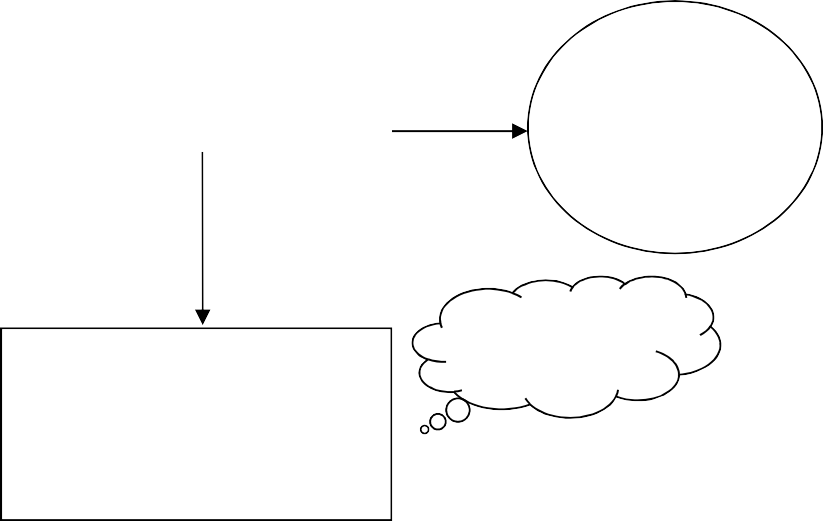
* + 1. **METHOD**

Current structures and organizations are ending up to a regularly expanding degree independent and the monetary region isn't unreasonably far from it. Video observation in moving domains has transformed into a present topic of energy for PC vision development. You can see all the branches are under the perception of CCTV cameras, alert systems, emergency gets, etc. The CCTV cameras are used to screen the unapproved activity. It ought to be watched reliably by an individual which is troublesome work; especially in night times.

The alert emergency get moreover needs to be pressed physically. This conventional structure requires some portion of work. A structure can be made which will customized recognize unapproved development and instruct to the security experts concerning the banks by different ways with no need of a person.

The Microcontroller Based Bank Security System fulfills all these necessities. A model of this security structure has been arranged in the composition to extend the component of security in bank locker rooms

enough. The development area will be done through camera itself and the hardware related with it will give unmistakable ways to deal with light up the security experts for instance using alert system a notice message and the image which has recognized the development will be normally exchanged on page which can be downloaded from wherever. For illuminating a GSM module will be utilized. The fundamental point of this examination is to structure a framework for alarming burglary and to auto capture the criminal in bank or ATM itself from brought together checking unit. The motivation behind the framework is to plan a savvy and concentrated checking and control framework utilizing IOT advances. The basic objectives of bank security system are following the bank locker room locales, acknowledgment of development and making the principal control move. The further portions will depict that how these objectives have been practiced.



FIRE SENSOR

DRIVER

CIRCUIT

PIR SENSOR

MICRO CONTROLLER

Arduino Uno

**MOTOR (LOCKER**

**)**

IR SENSOR

BUZZER

FIGURE 1 : BLOCK DIAGRAM OF MC

**SMS**

**IOT (PC)**

9

* 1. **WORKING PRINCIPLES**

This kit sends the SMS to the registered user whenever any unauthorized intrusion occurs in bank premises.

It is a GSM kit attached with passive infrared (PIR) sensor. The PIR sensor senses the infrared heat radiation radiated by the human body. This triggers the alarm as well as directs the GSM modem attached to the circuit to send SMS to the concerned registered authorities.

When PIR sensor senses the radiated heat, it sends the signal to the micro controller. The micro controller receives the signal and triggers the alarm attached to the circuit as well as sends the SMS to the concerned authorities.

* 1. **EXISTING MODELS**

In Existing Methods Commercially available anti-theft burglar systems are used and its very expensive and open type even thief can disable it. Also an human security was appointed to monitor the bank. This makes the system costlier. Current camera surveillance systems can be used for monitoring but they require a huge amount of data storage due to continuous video recording. However, our system only monitors the area when motion is detected and there is a possibility of certain activity. Our system

also sends a notification, in case of suspicious activity as it is not possible to continuously keep a watch on such activities.

In our proposed research we use Microcontrollers with different sensors (PIR, Smoke or Fire, IR and Gas) as observatory to detect or identify intruder or abnormal activities inside the bank and ATM. The main aim of this research is to design a system for alerting theft and to auto arrest the thief in bank or ATM itself from centralized monitoring unit. The purpose of the system is to design a smart and centralized monitoring and control system using IOT technologies.

* 1. **SOFTWARE TOOLS**
* EMBEDDED C
* MPLAB / PIC CCS
* DOTNET

# CHAPTER 2 SOFTWARE REQUIREMENTS

* 1. **EMBEDDED C**

C is a high level programming language intended for system programming. Embedded C is an extension that provides support for developing efficient programs for embedded devices. Yet, it is not a part of the C language.

Embedded C programming language is an extension to the traditional C programming language, that is used in embedded systems. The embedded C programming language uses the same syntax and semantics as the C programming language.

The only extension in the Embedded C language from normal C Programming Language is the I/O Hardware Addressing, fixed- point arithmetic operations, accessing address spaces, etc.

Local Declaration Section:

These variables are declared in the respective functions and cannot be used outside the main function.

Let’s look into the Main function section.

Main Function Section:

Every C programs need to have the main function. So does an embedded C program. Each main function contains 2 parts. A declaration part and an Execution part. The declaration part is the part where all the variables are declared. The execution part begins with the curly brackets and ends with the curly close bracket. Both the declaration and execution part are inside the curly braces.

**Function Definition Section :**

In this section, the function is defined.

This is the basic structure of the embedded c program.

With this, we come to an end of this “Embedded C Programming” article. I hope you have understood the basic structure.

Now that you have understood the basics of Programming in C, check out the training provided by Edureka on many technologies like Java, Spring and many more, a trusted online learning company with a network of more than 250,000 satisfied learners spread across the globe.

* 1. **MP LAB :**

MPLAB is a proprietary freeware integrated development environment for the development of embedded applications on PIC and dsPIC microcontrollers, and is developed by Microchip Technology.

MPLAB X is the latest edition of MPLAB, and is developed on the NetBeans platform. MPLAB and MPLAB X support project management, code editing, debugging and programming of Microchip 8-bit PIC and AVR (including ATMEGA) microcontrollers, 16-bit PIC24 and dsPIC microcontrollers, as

well as 32-bit SAM (ARM) and PIC32 (MIPS) microcontrollers. MPLAB is designed to work with MPLAB-certified devices such as the MPLAB ICD 3 and MPLAB REAL ICE, for programming and debugging PIC microcontrollers

using a personal computer. PICKit programmers are also supported by MPLAB. MPLAB X supports automatic code generation with the MPLAB Code Configurator and the MPLAB Harmony Configurator plugins.

MPLAB X is the latest version of the MPLAB IDE built by Microchip Technology, and is based on the open- source NetBeans platform. MPLAB X supports editing, debugging and programming of Microchip 8-bit, 16-bit and 32- bit PIC microcontrollers.

MPLAB X is the first version of the IDE to include cross-platform support for macOS and Linux operating systems, in addition to Microsoft Windows.

MPLAB X supports the following compilers:

* MPLAB XC8 — C compiler for 8-bit PIC and AVR devices
* MPLAB XC16 — C compiler for 16-bit PIC devices
* MPLAB XC32 — C/C++ compiler for 32-bit MIPS-based PIC32 and ARM-based SAM devices
* HI-TECH C — C compiler for 8-bit PIC devices (discontinued)
* SDCC — open-source C compiler

MPLAB 8.x is the last version of the legacy MPLAB IDE technology, custom built by Microchip Technology in Microsoft Visual C++. MPLAB supports project management, editing, debugging and programming of Microchip 8-bit, 16-bit and 32-bit PIC microcontrollers. MPLAB only works

on Microsoft Windows. MPLAB is still available from Microchip's archives, but is not recommended for new projects.[18]

MPLAB supports the following compilers:

* MPLAB MPASM Assembler
* MPLAB ASM30 Assembler
* MPLAB C Compiler for PIC18
* MPLAB C Compiler for PIC24 and dsPIC DSCs
* MPLAB C Compiler for PIC32
* HI-TECH C

# DOTNET

.NET is a software framework which is designed and developed by Microsoft. The first version of the .Net framework was 1.0 which came in the year 2002. In easy words, it is a virtual machine for compiling and executing programs written in different languages like C#, VB.Net etc. It is used to develop Form-based applications, Web-based applications, and Web services. There is a variety of programming languages available on the .Net platform, VB.Net and C# being the most common ones. It is used to build applications for Windows, phone, web, etc. It provides a lot of functionalities and also supports industry standards.

.NET Framework supports more than 60 programming languages in which 11 programming languages are designed and developed by Microsoft. The remaining Non-Microsoft Languages which are supported by .NET Framework but not designed and developed by Microsoft.

Main Components of .NET Framework

**Common Language Runtime(CLR):** CLR is the basic and Virtual Machine component of the .NET Framework. It is the run-time environment in the .NET Framework that runs the

codes and helps in making the development process easier by providing the various services such as remoting, thread management, type-safety, memory management, robustness, etc..

Basically, it is responsible for managing the execution of .NET programs regardless of any .NET programming

language. It also helps in the management of code, as code that targets the runtime is known as the Managed Code and code doesn’t target to runtime is known as Unmanaged code.

**Framework Class Library(FCL):** It is the collection of reusable, object-oriented class libraries and methods, etc that can be integrated with CLR. Also called the Assemblies. It is just like the header files in C/C++ and packages in the java.

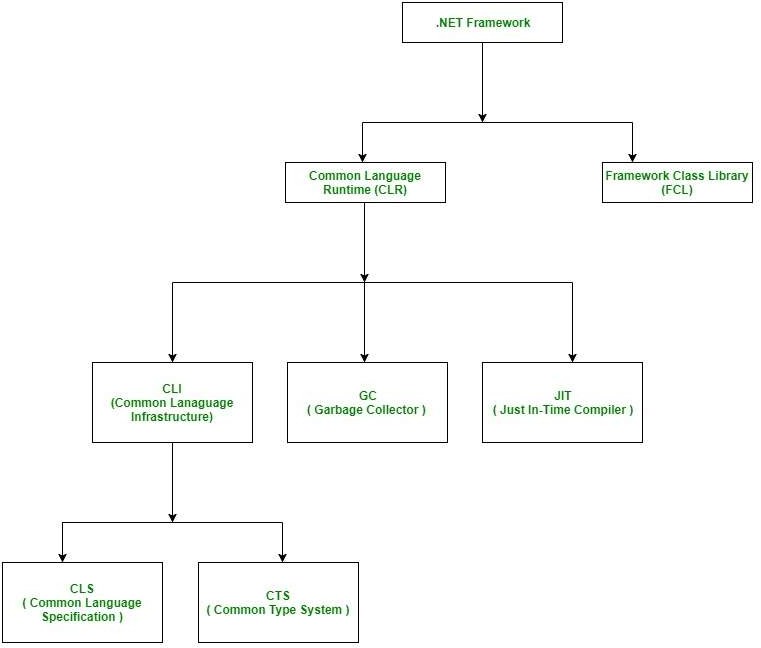
Installing .NET framework basically is the installation of CLR and FCL into the system. Below is the overview of .NET Framework

FIG 2 FRAMEWORK CLASS LIBRARY(FCL)

# CHAPTER 3 ALGORITHM

**3.1 ARSUINO BASED ALGORITHM**

The figure 3 shows the implementation of proposed bank security system by releasing knock-out gas.



Fig. 3 Arduino based Bank Security System by releasing Knock-Out Gas

The Arduino Uno board is programmed to read the PIR sensor data and it sends the signal to bank manager’s mobile through GSM. It also sends the signal to the gas releasing equipment to spray the knock-out gas towards robber. When the robber comes near the locker, he falls down due to release of knock-out gas. Once the bank manager receives the signal, he informs police immediately and catches the robber.

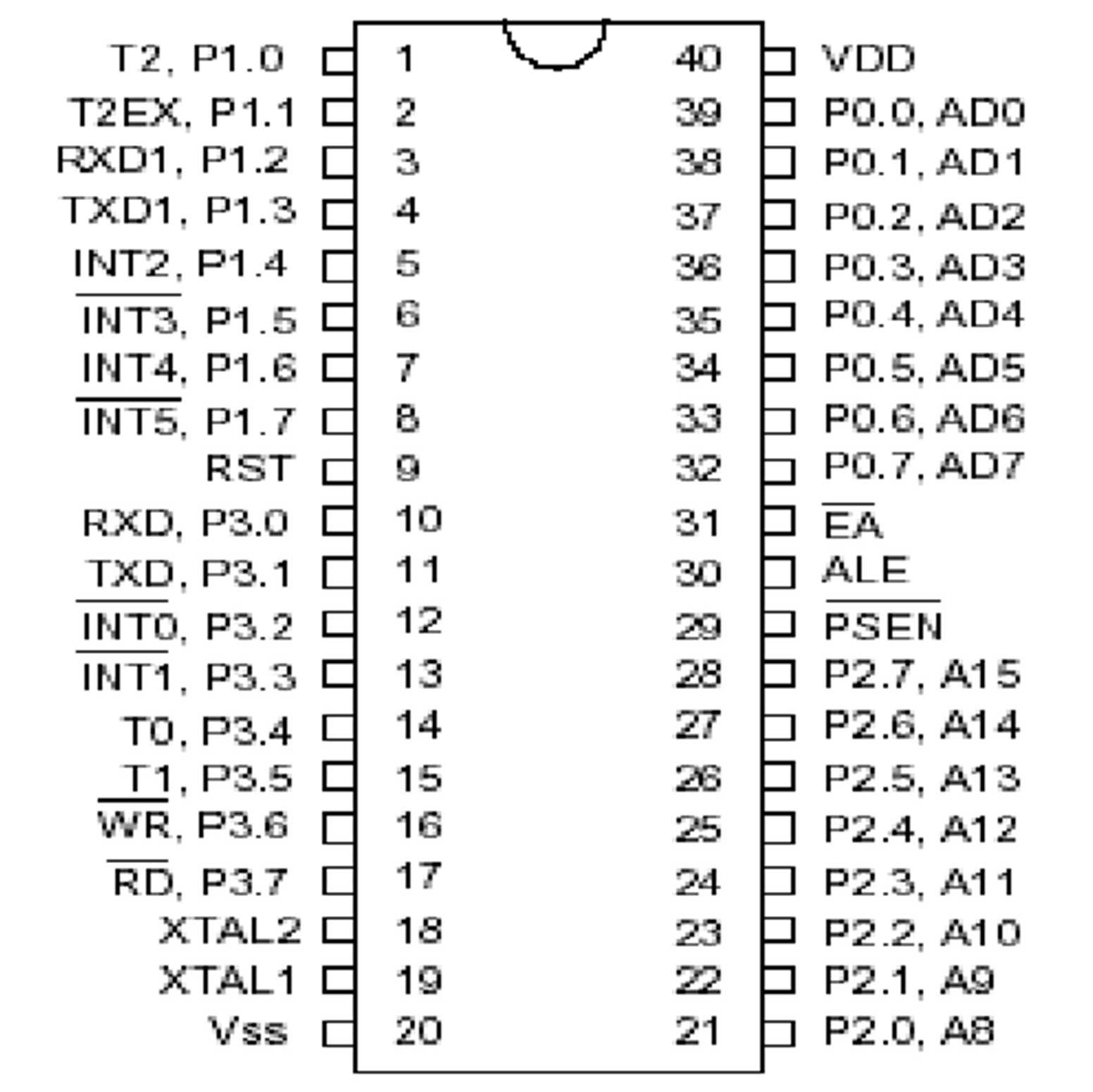
The following are the basic steps involved in performing the bank security system algorithm

1. PIR sensor monitors the entry of robber
2. rduino reads the PIR sensor data
3. The data from Arduino sent to the gas releasing equipment through delay
4. The data from Arduino also sent to the bank manager’s mobile through GSM
5. The gas releasing equipment releases the knock out gas towards robber
6. The knock gas reduces the conscious level of robber and makes him fall down
7. The bank manager will catch the robber easily with the help of police
8. The highly valuable ornaments hard cash and other documents has protected from the robber

# CHAPTER 4 MODULES

* 1. **DATA COLLECTION**

Basically used for control actions.It is used to control the operation of machine using fixed program that is stored in ROM/EPROM and that does not change over the life time.



# FIG 4 PIN DESCRIPTION OF 8051

This zone will layout about the exceptional techniques picked to realize the safety system and the work procedure of the structure. The sector will moreover appear unmistakable program codes used for the utilization of the system.



# FIG 5 WIRELESS CAMERA

**A.CENTRAL PROCESSING UNIT**

A microcontroller (or MCU for microcontroller unit) is a little PC on a lone composed circuit. In present day wording, it resembles, anyway less mind boggling than, a system on a chip or SoC; a SoC may join a microcontroller as one of its fragments. A microcontroller contains somewhere around one CPUs (processor-focuses) close by memory and programmable data/yield peripherals. Program memory as ferroelectric RAM, NOR flash or OTP ROM is moreover routinely included on chip, similarly as a little proportion of RAM. Microcontrollers are

planned for embedded applications, as opposed to the microchips used in PCs or other generally helpful applications containing distinctive discrete chips.



# FIG 6 PIR SENSOR

PIR sensors are more puzzled than an extensive parcel of interchange sensors cleared up in these instructional activities (like photocells, FSRs and tilt switches) in light of the fact that there are different variables that impact the sensors data and yield. The PIR sensor itself has two spaces in it, each opening is made of a phenomenal material that is tricky to IR. The point of convergence used here isn't by and large doing much therefore we see that the two spaces can 'see' out past some partition (basically the affectability of the sensor). Right when the sensor is idle, the two openings recognize a comparative proportion of IR, the encompassing whole radiated from the room or dividers or outside. Right when a warm body like a human or animal travels by, it first gets one segment of the PIR sensor, which causes a positive differential change between the two sections. Right when the warm body leaves the recognizing zone, the modify happens, whereby the sensor makes a negative differential change. These change beats are what is recognized.

# B. SYSTEM SENSING UNIT

It is a system that changes over a straightforward banner into a mechanized banner and it gives an isolated estimation, for instance, an electronic device that changes over a data basic voltage to a propelled number comparing to the span of the voltage. A sensor is a contraption that perceives and responds to a type of commitment from the physical condition. The specific information could be light, heat, development, sogginess, weight, or any of an amazing number of other normal wonders.



**FIG 7 FIRE SENSOR**

This flame sensor circuit misuses the temperature detecting property of a conventional flag diode IN 34 to identify heat from flame. Right now it detects heat; an uproarious caution recreating that of Fire detachment will be delivered. The circuit is excessively delicate and can distinguish an ascent in temperature of 10 degree or more in its region. Common flag diodes like IN 34 and OA 71 shows this property and the interior obstruction of these gadgets will diminish when temperature rises.

# C. SERIAL COMMUNICATION

Comprehensive Asynchronous Receiver/Transmitter is a touch of PC gear that translates data among parallel and successive

structures In media transmission and data transmission, successive correspondence is the route toward sending data one piece at some random minute, sequentially, over a correspondence channel or PC transport. This is rather than parallel correspondence, where a couple of bits are sent all things considered, on an association with a couple of parallel channels.

# DATA PREPROCESSING

Organize your selected data by formatting, cleaning and sampling from it.

Three common data per-processing steps are:

* **Formatting:** You may not have chosen the details in a format that suits you for working with. The data may also be in in an electronic database and you would like it to be in a spreadsheet, or the information may be in a proprietary file format and you would like it to be in an electronic database or folder.
* **Cleaning:** Cleaning data is the eradication or restoration of unfinished or empty data. There may also be incomplete occurrences of data which do not carry the information that you think you'd like to lever may need to eliminate these occurrences.

In addition, there are attributes which carry sensitive information and that the attributes are likely to be omitted.

* **Sampling**: In sampling, there can be more data set selected than required to work with.

Execution of More data can result in greater computational and memory requirements. A smaller ranked sample of the selected data can be taken for consideration, which will be much quicker to discover and model the solutions before considering the entire data set.

# CHAPTER 5

* 1. **RESULT ANALYSIS**

PIR sensor was fixed in various part of bank and IR sensor was fixed in the entrance of every rooms. Fire and Gas sensor will be fixed in the locker area where all the doors are connected with magnetic locks controlled by relay and burglar alarm connected with the relay circuit. For testing purpose we used single sensors in real time implementation we can proceed with multiple sensors, in future implementation the tested results will give the expected security intimation in real time with proper results. In this proposed work, the PIR and IR sensors did their work accurately with no time delay with the microcontroller, if the signal is valid then microcontroller display the information about the intruder inside the bank to the central processing center. Then the intruder will be arrested with auto arrest system and information will be sent to the Police and bank manager through the central processing centers. We have used some hardware components like sensors for detection of motion, fire, gas. We used motion sensor to detect the motion of the persons in the bank, we detect fire leakage or gas leakage by using fire and gas sensors. We link all sensors in the IOT terminal and by using the LAN connection we set up an web based application to monitor or control the movement’s in the bank.

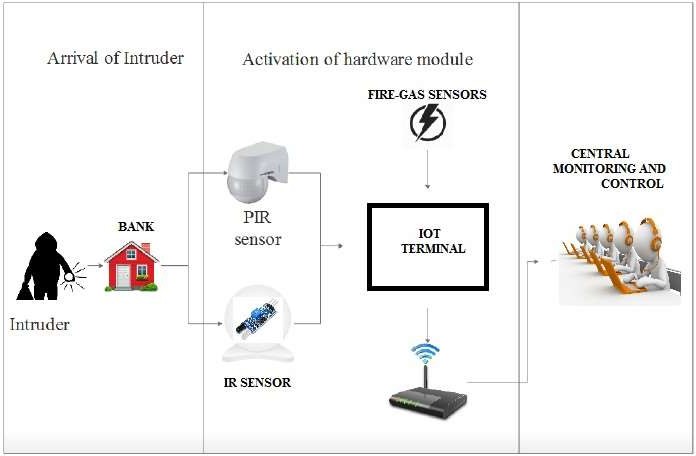


Fig 8 : Outlook of the project

# CONCLUSION

This is a real time application based which tells that there is a need to bring in a revolution in the bank security by making the procedure a little easy and more systematic for the bank officials. It is a low cost, low in power conception, compact in size and standalone system. The microcontroller compares the sensor values. If the intruder was identified the information will be securely processed and monitored through central processing center. By using this method instant monitoring and control was possible. No need to monitor continuously and less human was involved in monitoring section.

# FUTURE ENHANCEMENTS:-

Although we did not meet the target of 100 percent precision in identifying fraud, we ended up building a program with enough time and resources that could come really similar to that goal.The very design of this project allows the incorporation of multiple algorithms as modules, and the combination of their results can increase the accuracy of the end result.

# APPENDIX 1

**CODING:-**

#include <gps.h> #include <HWSerial.h> #include <inetGSM.h> #include <LOG.h> #include <SIM900.h> #include <sms.h> #include <Streaming.h>

#include <WideTextFinder.h> #include <Wire.h>

#include <RTClib.h> #include <Keypad.h>

#include <LiquidCrystal\_I2C.h> #include "FPS\_GT511C3.h" #include "SoftwareSerial.h" #include <String.h>

#include <sms.h>

#include <SoftwareSerial.h> #include "Adafruit\_FONA.h" #define PIN\_RELAY 32

#define PIN\_BUZZER 13

#define PIN\_POWER 35

#define PIN\_PIR 33

#define PIN\_VIB 41

#define PIN\_GSM\_ON 9

#define VIB\_HR 9

#include <call.h>

const byte ROWS = 4; //four rows const byte COLS = 4; //four columns

char hexaKeys[ROWS][COLS] = {

{'1','2','3','A'},

{'4','5','6','B'},

{'7','8','9','C'},

{'S','0','E','D'}

};

byte rowPins[ROWS] = {30, 28, 26, 24}; //connect to the row pinouts of the keypad byte colPins[COLS] = {25, 27, 29, 31}; //connect to the column pinouts of the keypad

Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

SMSGSM sms;

SoftwareSerial mySerial(4, 5);

RTC\_DS1307 RTC;

LiquidCrystal\_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);

FPS\_GT511C3 fps(11, 12);

int hr\_24, hr\_12; String hr\_Status="";

String str\_Password=""; String str\_AdminPW="5647"; String str\_acno="";

int menu\_status=0; int admin\_status=0; int acc\_no=0; int

pir\_val=0; int vib\_val=0;

int buttonState = 0;

int randNumber; String str\_tmprndpw=""; String str\_rndpw="";

unsigned long start, finished, elapsed;

//char smsbuffer[160];

//char n[20];

#define FONA\_RX 2

#define FONA\_TX 10

#define FONA\_RST 4

SoftwareSerial fonaSS = SoftwareSerial(FONA\_TX, FONA\_RX); Adafruit\_FONA fona = Adafruit\_FONA(FONA\_RST); char sendto[21]="+94777211092", message[141]="Anis";

void setup()

{

pinMode(PIN\_RELAY, OUTPUT);

pinMode(PIN\_BUZZER, OUTPUT); pinMode(PIN\_POWER, INPUT); pinMode(PIN\_GSM\_ON, OUTPUT);

pinMode(PIN\_PIR, INPUT);

Serial.begin(9600); lcd.begin(20,4); Wire.begin(); mySerial.begin(9600);

RTC.begin();

RTC.adjust(DateTime( DATE , TIME ));

fps.UseSerialDebug = false; // messages in the serial debug screen fps.Open();

//fps.SetLED(true); // turn on the LED inside the fps

lcd.setCursor(0,0); lcd.print(" Bank Locker "); lcd.setCursor(0,1); lcd.print(" System "); delay(2000); lcd.setCursor(0,0); lcd.print(" Developed By "); lcd.setCursor(0,1); lcd.print(" Sutharshan S"); delay(2000); lcd.clear();

digitalWrite(PIN\_GSM\_ON, HIGH);

beep(200); menu\_status=0; admin\_status=0;

digitalWrite(PIN\_RELAY,HIGH);

fonaSS.begin(4800); // if you're using software serial //Serial1.begin(4800); // if you're using hardware serial

if (! fona.begin(fonaSS)) // can also try fona.begin(Serial1)

{

Serial.println(F("Couldn't find FONA")); while (1);

}

char sendto[21]="+94777211092", message[141]="Welcome to Bank Locker System";

if (!fona.sendSMS(sendto, message)) { Serial.println(F("Failed"));

} else { Serial.println(F("Sent!"));

}

delay(3000);

}

void loop()

{

buttonState = digitalRead(PIN\_POWER);

if(buttonState==HIGH)

{

fps.SetLED(true);

}

else

{

fps.SetLED(false);

}

//Date and Time

if(menu\_status==0)

{

hr\_24=now.hour(); if (hr\_24==0)

{

hr\_12=12;

}

else

{

hr\_12=hr\_24%12;

}

if (hr\_24<12)

{

hr\_Status="AM";

}

else

{

hr\_Status="PM";

}

String v\_date= String(now.year()) + "-" + String(now.month()) + "-" + String(now.day());

String v\_time= String(hr\_12) + ":" + String(now.minute()) + ":" + String(now.second()) + " "

+ hr\_Status;

//Serial.println(v\_date);

//VIB vib\_val = digitalRead(PIN\_VIB); // read input value

if(vib\_val==0)

{

if(now.hour()>=VIB\_HR)

{

Serial.print("Vibration Detected"); lcd.clear();

lcd.setCursor(0,0); lcd.print("Vibration Detected"); lcd.setCursor(0,1); lcd.print("Be Alert"); beep(300); delay(2000); menu\_status=4;

}

}

lcd.setCursor(0,0); lcd.print("Date: " + v\_date); lcd.setCursor(0,1); lcd.print("Time: " + v\_time);

//lcd.setCursor(14,1);

//lcd.print(hr\_Status);

if (fps.IsPressFinger())

{

fps.CaptureFinger(false); int id = fps.Identify1\_N(); if (id <200)

Serial.print("Verified ID:"); Serial.println(id);

lcd.clear(); lcd.setCursor(0,0); lcd.print("Valid Id"); lcd.setCursor(0,1); lcd.print(id);

str\_tmprndpw=""; generatePassword();

str\_tmprndpw="Your PIN Code is:" + String(str\_tmprndpw);

str\_tmprndpw.toCharArray(message,160);

char sendto[21]="+94777211092", message[141]=""; str\_tmprndpw.toCharArray(message,141);

fonaSS.begin(4800); // if you're using software serial

//Serial1.begin(4800); // if you're using hardware serial

if (! fona.begin(fonaSS)) // can also try fona.begin(Serial1)

{

Serial.println(F("Couldn't find FONA")); while (1);

}

Serial.println(F("FONA is OK"));

if (!fona.sendSMS(sendto, message)) {

} else { Serial.println(F("Sent!"));

start=millis(); Serial.print("Start:");

Serial.println(start);

}

lcd.clear(); lcd.setCursor(0,0); lcd.print("SMS Sent");

//lcd.setCursor(0,1);

//lcd.print("Pin Code:" + str\_tmprndpw);

delay(5000);

}

else

{

Serial.println("Finger not found"); lcd.clear();

lcd.setCursor(0,0); lcd.print("Finger not found"); delay(1000);

}

}

else

{

Serial.println("Please press finger");

}

if(digitalRead(PIN\_PIR) == HIGH && now.hour()>22) { beep(200);

}

}

if(menu\_status==4)

{

for(int x=1;x<=5;x++)

{

beep(500);

}

menu\_status=0;

}

char customKey = customKeypad.getKey();

if (customKey)

{

Serial.println(customKey);

if(customKey=='A')//Admin

{

menu\_status=1; beep(200); Serial.println("Admin");

lcd.clear(); lcd.setCursor(0,0); lcd.print("Admin Password:"); lcd.setCursor(0,1); lcd.blink();

if(customKey=='B')//Enroll

{

menu\_status=2; beep(200);

if(admin\_status==1)

{

Serial.println("Enrollment"); lcd.clear();

lcd.setCursor(0,0); lcd.print("E-Enter Ac/No:"); lcd.setCursor(0,1);

lcd.blink();

} else { lcd.clear(); lcd.setCursor(0,0); lcd.print("Log in"); lcd.setCursor(0,1); lcd.print("Required"); delay(2000);

}

}

if(customKey=='C')//Delete

{

menu\_status=3; beep(200);

if(admin\_status==1)

Serial.println("Delete");

lcd.clear(); lcd.setCursor(0,0); lcd.print("D-Enter Ac/No:");

lcd.setCursor(0,1); lcd.blink(); Serial.println(fps.DeleteID(str\_acno.toInt()));

} else { lcd.clear(); lcd.setCursor(0,0); lcd.print("Log in"); lcd.setCursor(0,1); lcd.print("Required"); delay(2000);

}

}

if(customKey=='D')//Process

{

beep(200); lcd.noBlink(); if(menu\_status==1)

{

if(str\_Password==str\_AdminPW)

{

beep(200); admin\_status=1; menu\_status=0;

} else { lcd.clear(); lcd.setCursor(0,0);

lcd.print("Invalid Password"); beep(500); delay(2000); lcd.clear();

str\_Password=""; admin\_status=0; menu\_status=0;

}

}

if(menu\_status==2)

{

acc\_no=str\_acno.toInt(); Enroll(acc\_no); acc\_no=0; menu\_status=0;

}

if(menu\_status==3)

{

acc\_no=str\_acno.toInt(); fps.DeleteID(acc\_no); lcd.clear();

lcd.setCursor(0,0); lcd.print("Deleted"); beep(500); delay(2000); lcd.clear();

acc\_no=0; menu\_status=0;

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

if(menu\_status==5)

{

finished=millis(); Serial.print("Finished:"); Serial.println(finished);

elapsed=finished-start; Serial.print("Milliseconds Elapsed:"); Serial.println(elapsed);

str\_tmprndpw=str\_tmprndpw.substring(str\_tmprndpw.length()- 6,str\_tmprndpw.length());

if(elapsed<30000)

{

if(str\_rndpw==str\_tmprndpw)

{

//Relay digitalWrite(PIN\_RELAY,LOW);

//beep(200);

//delay(2000);

//digitalWrite(PIN\_RELAY,HIGH); beep(200);

Serial.println("Pin Success");

lcd.clear(); lcd.setCursor(0,0); lcd.print("PIN Success");

delay(2000);

} else

{

Serial.println("Invalid Pin Code"); lcd.clear(); lcd.setCursor(0,0); lcd.print("Invalid PIN"); delay(2000);

}

menu\_status=0;

}

else

{ lcd.clear(); lcd.setCursor(0,0); lcd.print("Time Out");

}

str\_rndpw=""; str\_tmprndpw="";

}

}

if(customKey=='S')//Enroll

{

menu\_status=5;

beep(200); Serial.println("Pin Enter");

lcd.clear(); lcd.setCursor(0,0); lcd.print("Enter Pin:"); lcd.setCursor(0,1);

lcd.blink();

}

if(customKey=='E')//Enroll

{ lcd.clear(); menu\_status=0; admin\_status=0; acc\_no=0; digitalWrite(PIN\_RELAY,HIGH);

}

if (customKey=='1' || customKey=='2' || customKey=='3' || customKey=='4' || customKey=='5' || customKey=='6' || customKey=='7' || customKey=='8' || customKey=='9' || customKey=='0' || customKey=='.')

{

if(menu\_status==1)

{

lcd.print("\*");

str\_Password += customKey; Serial.println(str\_Password);

}

if(menu\_status==2 || menu\_status==3)

{

lcd.print(customKey); str\_acno += customKey;

Serial.println(str\_acno);

}

if(menu\_status==5)

{

lcd.print(customKey); str\_rndpw += customKey;

Serial.println(str\_rndpw);

}

}

delay(100);

}

}

void beep(unsigned char delayms){ analogWrite(PIN\_BUZZER, 20);

delay(delayms); analogWrite(PIN\_BUZZER,0); delay(delayms);

}

//void SendSMSMessage()

//{

// if (!fona.sendSMS("+94777211092", smsbuffer)) {

// Serial.println(F("Failed"));

// } else {

// Serial.println(F("Sent!"));

// }

//}

void generatePassword()

{

//String temp\_pw=""; randomSeed(analogRead(0)); // read from an analog port with nothing connected

for(int i=0; i < 6; i++)

{

randNumber = random(0,9); str\_tmprndpw=str\_tmprndpw

+ String(randNumber);

}

Serial.print(str\_tmprndpw);

}

void Enroll(int en)

{

// Enroll test

// find open enroll id

//int enrollid = 0; bool usedid = true; while (usedid == true)

{

usedid = fps.CheckEnrolled(en); //enrollid++; if (usedid==true)

{

lcd.clear(); lcd.setCursor(0,0); lcd.print("Id Already Exist");

}

}

fps.EnrollStart(en);

// enroll

Serial.print("Press finger to Enroll #"); Serial.println(en);

lcd.clear(); lcd.setCursor(0,0); lcd.print("Place finger to"); lcd.setCursor(0,1); lcd.print("Enroll -1");

while(fps.IsPressFinger() == false) delay(100); bool bret = fps.CaptureFinger(true); int iret =

0; if (bret != false)

{

Serial.println("Remove finger"); lcd.clear();

lcd.setCursor(0,0); lcd.print("Remove Finger");

fps.Enroll1();

while(fps.IsPressFinger() == true) delay(100); Serial.println("Press same finger again"); lcd.clear();

lcd.setCursor(0,0); lcd.print("Place Finger to"); lcd.setCursor(0,1); lcd.print("Enroll -2");

while(fps.IsPressFinger() == false) delay(100); bret = fps.CaptureFinger(true);

if (bret != false)

{

Serial.println("Remove finger");

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Remove Finger");

fps.Enroll2();

while(fps.IsPressFinger() == true) delay(100); Serial.println("Press same finger yet again");

lcd.clear(); lcd.setCursor(0,0); lcd.print("Place Finger to"); lcd.setCursor(0,1); lcd.print("Enroll -2");

false)

while(fps.IsPressFinger() == false) delay(100); bret = fps.CaptureFinger(true); if (bret !=

{

Serial.println("Remove finger");

lcd.clear();

lcd.setCursor(0,0); lcd.print("Remove Finger");

iret = fps.Enroll3(); if (iret == 0)

{

Serial.println("Enrolling Successfull");

lcd.clear();

lcd.setCursor(0,0); lcd.print("Enroll.

Success"); lcd.setCursor(0,1);

lcd.print("Id:" + String(en));

delay(2000); menu\_status=0;

}

else

{

lcd.setCursor(0,0);

lcd.clear();

Serial.print("Enrolling Failed with error code:"); Serial.println(iret);

lcd.print("Enrollment");

lcd.setCursor(0,1); lcd.print(" Failed"); delay(2000);

menu\_status=0;

}

}

else Serial.println("Failed to capture third finger");

}

else Serial.println("Failed to capture second finger");

}

else Serial.println("Failed to capture first finger");

}

# APPENDIX 2

**RESULT :**

The desired outcome of the project was accomplished. Moreover, to adopt new technology as it comes, to provide more security and services, the future work can be done for bank locker system. The future work was analyzed when developing the project to increase the scope further.

Along with the bank locker system, the AES128 bit encryption can be enabled for secure communication with implementation of CryptoShield. This adds security layer to the system by processing encryption and decryption. This shield inbuilt with AES-128 encrypted EEPROM that provides both authentication and confidential non-volatile date storage. The system security can be enhanced further with the help of this shield.

Nowadays internet banking plays a major role in human’s life in order to check and transfer payments. The access details of bank locker system can be merged with internet banking with the execution of Ethernet shield. The Ethernet shield can be programmed to act as a web server between Arduino mega microcontroller and the bank’s website. Whenever the access details of the bank locker system is logged it would send to bank’s website through virtual port using Ethernet shield.

Instead of fingerprint biometric authentication, the face recognition

can be used with “onetime password” generation. The face recognition is one of the biometric technology rapidly increasing these days and widely used in various application. With the face recognition bank locker system become more user friendly to all type of users.

The fingerprint image sensing quality can be improved by using 3D image scanner to provide more accuracy. This may leads to result in false acceptance rate and false rejection rate marginally.

The latest cellular shield SM5100B which is compatible with Arduino platform can be deployed further for speedy communication. Use of this cellular shield would take less processing time to send the “one-time password” to user’s phone than current GSM module.

# REFERNCES

1. Gyanendra K Verma, Pawan Tripathi, “A Digital Security System with Door Lock System Using RFID Technology”, International Journal of Computer Applications (IJCA) (0975 – 8887), Volume 5– No.11, August 2010.
2. Ari Juels, “RFID Security & Privacy: A research survey”, IEEE JOURNAL on selected areas in communication, Volume: 24, 28 September 2005, page no. 16-17.
3. R. Ramani, S. Valarmathy, S. Selvaraju, P. Niranjan “Bank Locker Security System based on RFID and GSM Technology, ” International Journal of Computer Applications (0975 – 8887) Volume 57– No.18, November 2012.
4. K Chandrasekar, M Surumbar Khuzhali, “MEMS Accelerometer Based Password Recognition System using GSM" International Journal of Engineering Research and Science & Technology Vol. 3, No. 2, May 2014.
5. Swetha J, “RFID based Automated Bank Locker System, ” International Journal of Research in Engineering and Technology EISSN: 2319-1163 | pISSN: 2321-7308, Volume: 03 Issue: 05 | May- 2014.
6. Mr Abhijeet S. Kale, Prof. Sunpreet Kaur Nanda, “Design of Highly Secured Automatic Teller Machine System by Using Aadhaar Card and Fingerprint” International Journal of Engineering Science Invention ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726 [www.ijesi.org](http://www.ijesi.org/) Volume 3 Issue 5ǁ May 2014 ǁ PP.22-26.
7. ChetnaKoli.R, Nikita.S,Kheratkar.V, Pooja.S, Ganganalli.T & Shirsa.G,2014 ‘Bank Locker System Using Iris’ International Journal

of Advanced Research in Computer Engineering & Technology

(IJARCET) Volume 3, Issue 3.

1. Raghu Ram.Gangi, SubhramanyaSarma&Gollapudi,2013 ‘Locker Opening And Closing System Using RFID, Fingerprint, Password’ International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Volume 2, Issue.
2. Balaji Venkatesh. A.M, Karthik Kalkura & Shriraam A.C, 2013 ‘Student Locker Protection Using RFID Tag & Reader’ International Journal of Engineering and Advanced Technology (IJEAT) ,Volume- 3, Issue-2.
3. Swetha.J, 2011, ‘RFID Based Automated Bank Locker System’, International Journal of Research in Engineering and Technology, Volume-1.
4. Bramhe, 2011 ‘SMS Based Secure Mobile Banking’, International Journal of Engineering and Technology, Volume-3.
5. Joshua Bapu.J & Sirkazi Mohd Arif, 2013 ‘Locker security system using RFID and GSM technology’ International Journal on Advances in Engineering and Technology, Volume 3.
6. Ramesh.S, Soundaria Hariharan & Shruthi Arora, 2012 ‘Monitoring and Controlling of Bank Security System’, International Journal of Advanced Research in Computer Science and Software Engineering, Volume-2.
7. Manoj V, Bramhe ‘SMS Based Secure Mobile Banking Department’ International Journal of Engineering and Technology Vol.3 (6).
8. Abhishek Shukla, Shruti Tyagi & Shweta Gupta,2013 ‘An Ultimate Security to Bank Lockers Using Multi-Model Biometric

Systems’ International Journal of Research Review in Engineering

Science &Technology (IJRREST) Volume-2, Issue-2.

1. Aruna.D.Mane & Sirkazi MohdArif, 2012‘Locker Security System Using RFID and GSM Technology’ International journal of advances in engineering & technology, volume-1.
2. Swetha J, RFID based automated bank locker system, IJRET: International Journal of Research in Engineering and Technology. Volume: 03, Issue: 05, May-2014,eISSN: 2319-1163| pISSN: 2321- 7308.
3. Gaurav Chavan, Sourabh Dabke, Anup Ghandghe, Mrs. K.A.Musale, Bank locker security system using android application, International Research Journal of Engineering and Technology (IRJET) Volume: 02 Issue: 01 | Apr2015, e-ISSN: 2395 -0056, p-ISSN: 2395-0072.
4. Aruna.D.Mane1 and Sirkazi Mohd Arif2, Locker Security System Using RFID and GSM Technologyǁ, International journal of Scientific and Technology Research, Volume 4, Issue 07, JULY 2015.
5. Raghu Ram.Gangi, Subhramanya Sarma.Gollapudi, Locker opening and closing system using RFID, fingerprint, password and GSM, International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Volume 2, Issue 2, March – April 2013 ISSN 2278- 68:56.
6. Sagar S .Palsodkar, Prof S.B. Patil, Bank Lockers Security System using Biometric and GSM Technology, SSRG International Journal of Electronics and Communication Engineering (SSRG-IJECE) – Volume 2, Issue 4–April 2015, ISSN: 2348 – 8549.