U.S.S.R

Final Report

Submitted to
The Faculty of Operation Catapult LXXXVIII
Rose-Hulman Institute of Technology
Terre Haute, Indiana

By
Group 25

Eric Dong	Arcadia High School Arcadia, CA
Tyler Montague	Heritage High School Frisco,TX
Gabe Doyle	Lincoln Way West High School New Lenox, IL
Julian Flores	St.Joseph Catholic Academy Kenosha,WI

July 27, 2017

Abstract:

The safe is 8x8x12 inches. The safe utilizes fingerprint and keypad verification methods to ensure the it's security. It can be used for storing any items that can fit. The goal of this project was to make a safe that would be hard to break into. Our project uses a microcontroller to manage all of the components in order to make sure the safe does not open unless the owner allows it to be. It'ss security features were made using microcontroller technology. The basic goals of microcontroller technology are to learn more about the technology and to make items that can react to different circumstances in the environment.

To accomplish these objectives, an LCD screen and fingerprint scanner were used to make multiple layers of security in our safe. Two servos were also connected to make our locking mechanism. The microcontroller manages the components to make sure everything runs smoothly. When the user enters in their pin , the fingerprint scanner becomes usable. After the fingerprint is registered, the fingerprint can be used to unlock the safe. This combination of components creates a secure safe that can keep items secure from intruders.

Introduction:

A safe is a box that is used to store items securely. The purpose of the box is to make sure items cannot be retrieved by an intruder. We make sure the box is secure by adding two layers of security to the box. It can be unlocked by entering a pin into the LCD touch screen and by putting an authorized fingerprint into the safe while at the same time learning more about how to use microcontroller technology. To make this board, multiple circuits were built to be controlled by software that was loaded onto the microcontroller.

The fingerprint scanner that we use uses a method that relies on capturing an optical image and using algorithms to detect unique patterns on the surface, such as ridges or unique marks, by analysing the lightest and darkest areas of the picture. Basically, our scanner relies on takes pictures of the fingerprint and remembers it is the correct one that will unlock the lock. The fingerprint scanner will say if it recognizes the fingerprint or not. Depending on the answer, the LCD touch screen will turn on and be able to be used.

The LCD touch screen screen uses what is known as "resistive touch screen technology". These types of screens literally resist your touch. There are two parts of the screen. One part is resist and the other is conductive. They are separated by a graph of tiny dots called spacers to keep the two layers apart until you touch it. An electrical current runs through those layers at all times, but when your finger hits the screen, the two are pressed together and the electrical current changes at the point of contact. The software recognizes a change in the current at these coordinates and carries out the function that corresponds with that spot.

While creating projects like this one, it’s common to run into problems, or bugs, in the software that has to be fixed. Solving issues like this, or debugging, takes a lot of time to do completely. These issues tend to be extremely tedious and hard to find. The amount of problems that occurred depended on how well we made the software and hardware. Mistakes in the coding caused problems that needed to be fixed in order for the safe to work as intended.

This safe was created to help increase our understanding of microcontroller technology and to give people a secure way to store their valuable. We wanted something that would be easy for the consumer to use but also offered extreme security. Our design is superior to all keylock-safes because the only keys that can work only exist in the consumer’s mind-the pin code. The other hides in plain site- the finger. Both alone can hold their own against any other method of security, but together they offer an extreme reliability for the consumer.

Materials and Methods:

The materials we used were a Digilent Chipkit Max32 Microcontroller Board with Mega R3 Headers, an LCD touch screen, a computer with Arduino coding software, a perforated board, a 100 Ω resistor, a 260 Ω resistor, the ADH Technology GT-511C3 fingerprint scanner, four LED lights, wires, a soldering iron, solder, a digital multimeter, a rectangular laser cut box, a servo, a metal bar, and various screws. These materials were all provided by the Rose-Hulman Institute of Technology.

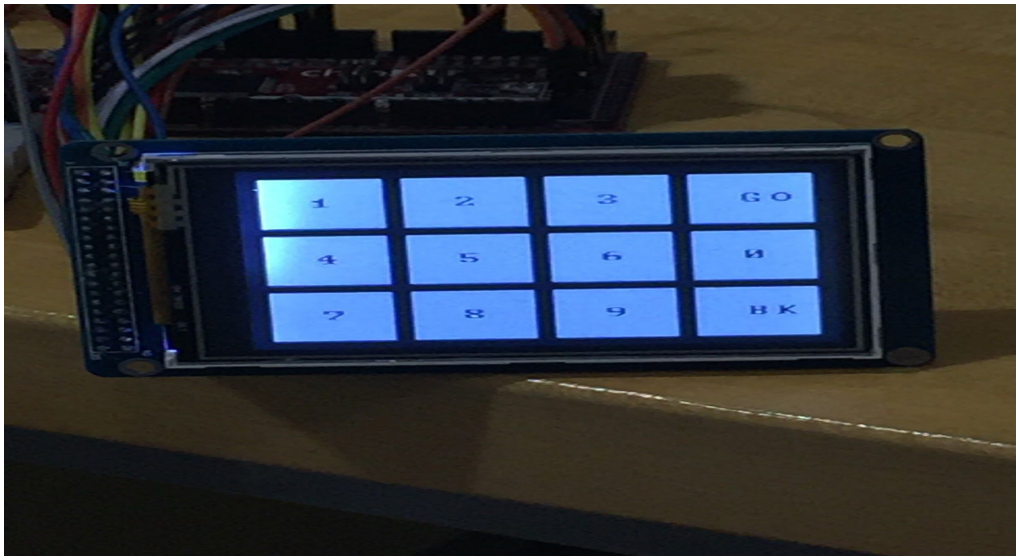
The safe was made using Laser-Cut technology to make an acrylic box. We then mounted our components on our box to keep them from moving. All of the components were managed by the microcontroller to make sure it ran uniformly. The microcontroller made sure that when we put in our fingerprint, the LCD screen would turn on and allow us to enter in the pin code. While we did this, the LED lights would turn on and turn different colors depending on how many digits were entered, if the pin entered was correct, or if the pin entered was wrong.

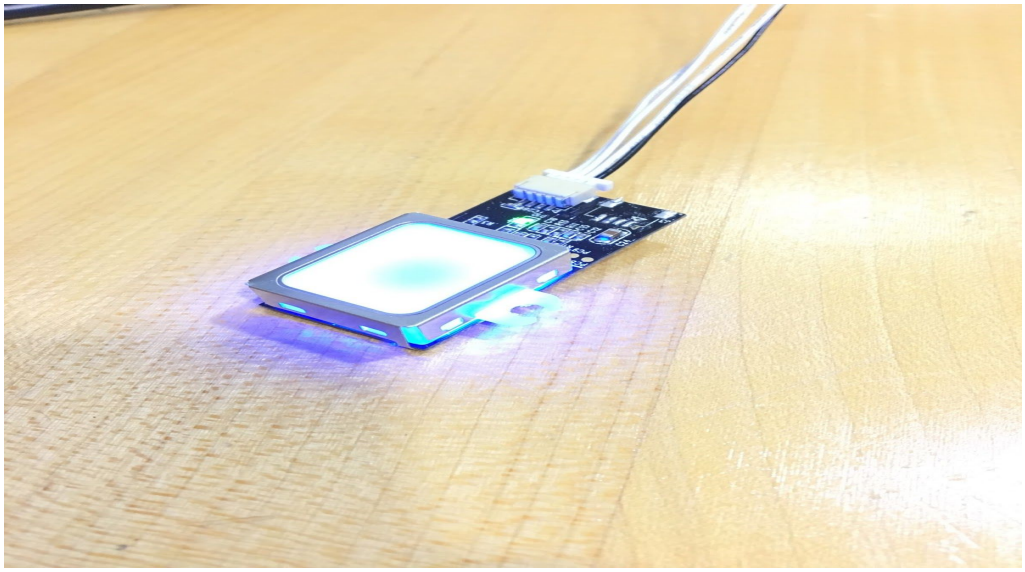
User's Guide:

For new users, they can register their fingerprints when they first turn on the safe. The safe will automatically register the first fingerprints that successfully register with it. The fingerprints can be recent only be going into the program and deleting the registered fingerprint. The LCD touch screen will then turn on and the first 4 digits entered will be the password. These can also be changed by going into the program and editing the password.

Results:

The safe was tested and we had some trouble getting the code to merge. It was difficult but we were able to finally get it to work and we were able to get the program to run smoothly. Our biggest problem was that the servo would not lock or unlock when told to. At the end, it worked as intended and did not have many problems.





Discussion:

The purpose of this project was to build a safe while improving our understanding of microcontroller technology. This goal was reached without many issues. The initial testing showed that we had a few software problems but they were all eventually resolved. The group learned a lot about microcontroller technology as well as working together as a team to get things done.

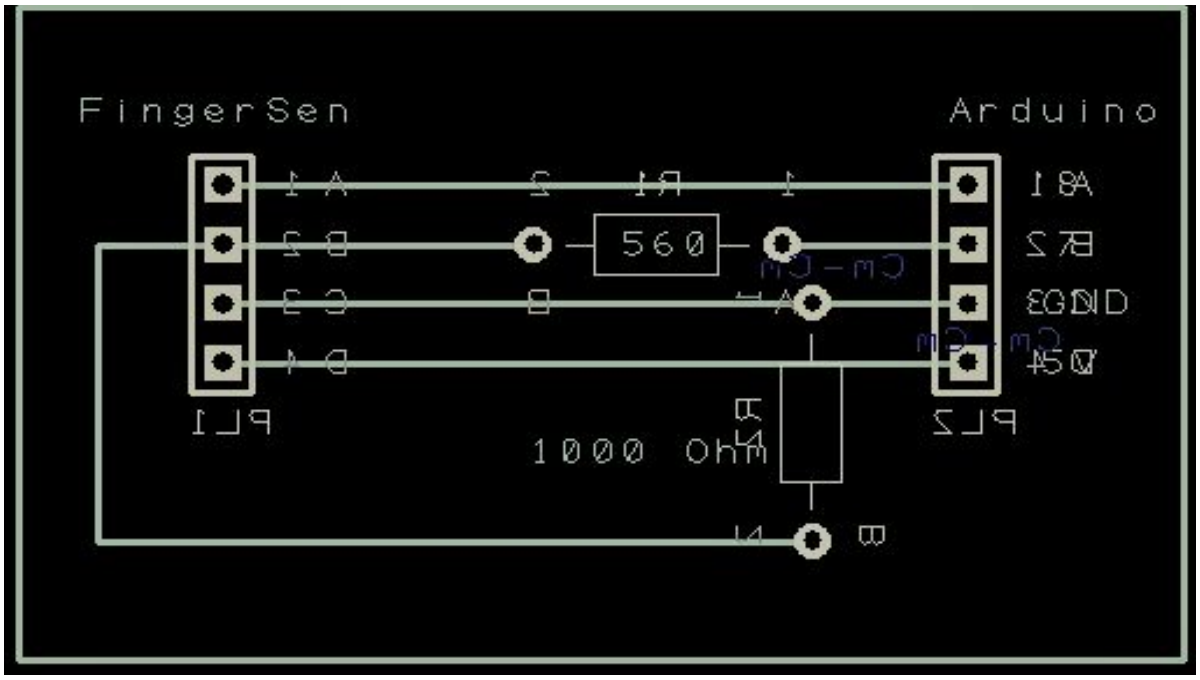
Limited problems were encountered while trying to build a secure safe. The coding problems caused many bumps during our journey, but we managed to smooth out all of the problems which resulted in a fully functioning safe. All of the problems we encountered with coding had to do with the LCD touch screen. We could not get the LCD touch screen to function as intended. During the process, we ended up overheating the LCD screen and we burnt out two LED lights. Although this was very problematic, we were able to resolve the issue.

Another problem that we faced was connecting the fingerprint scanner to the microcontroller. We could not find good instructions that would tell us how to accurately connect the fingerprint scanner so that it could be managed by the microcontroller. We solved this problem and were finally able to find a good code to use on the fingerprint scanner to have it function as intended. The fingerprint scanner, which uses optical scanners to take pictures of a fingerprint to remember it, was finally working.

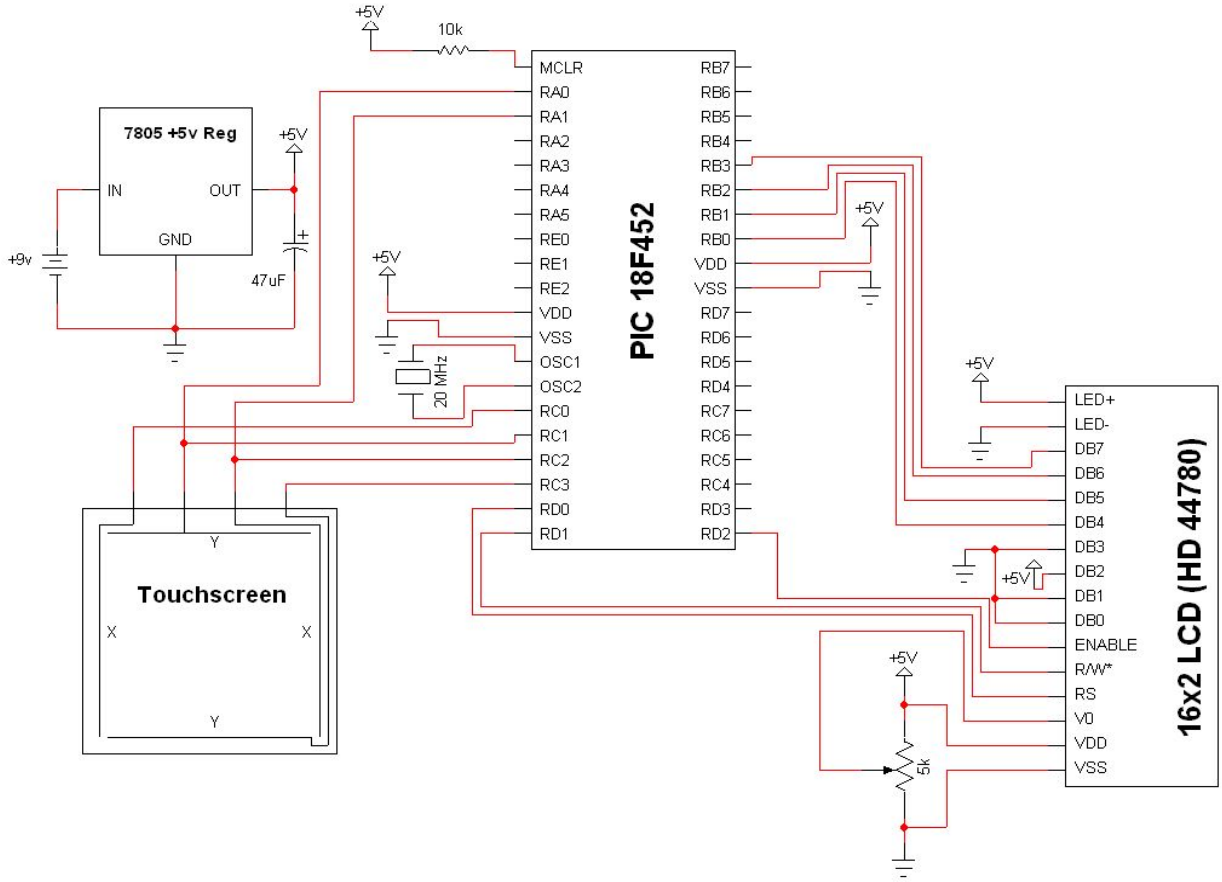
As a group, we learned how difficult it is to design and build a project using a microcontroller. We learned all of the problems that can happen on the way to making our dreams become a reality. We learned to be patient and to persevere until we find a solution to our problems. We realized the importance of working in a team when we first tried to design the safe. Instead of one person coming up with the design, all of us added onto each other's ideas until our final design was what our final project was. We discovered the value in compromise and having thick skin. We understood each other's goals and we knew that every decision made was for the good of the project. We had to set aside our pride so the project could be designed to the best of our ability.

To improve this project in the future, a better safe could be made by adding more security components to it so it could be harder to break into. Adding facial recognition would be very valuable when trying to keep the safe from intruders. Another thing we feel should be improved is the materials used to make the safe. Since our safe is only laser cut out of a type of plastic, we feel it is not as durable as it could be. Because of this, we feel a more sturdy material such as metal would keep the safe more secure and harder to open using regular tools.

Fingerprint Sensor Schematic:



Touch Screen Schematic:



Appendix II

```
#include <UTFT.h>
#include <URTouch.h>
#include "FPS_GT511C3.h"
#include "SoftwareSerial.h"
#include <Servo.h>
Servo myservo;
int blue1 = 72;
int blue2 = 12;
int blue3 = 11;
int blue4 = 81;
int green = 78;
int redPin = 73;
extern uint8_t BigFont[];
extern uint8_t SmallFont[];
UTFT myGLCD(ITDB32S, 82, 83, 84, 85);
URTouch myTouch(48, 49, 50, 51, 52);
FPS_GT511C3 fps(8, 7);
int x, y;
char stCurrent[20] = "";
int stCurrentLen = 0;
char stLast[20] = "";
String str = "";
String pwd = "";
void waitForIt(int x1, int y1, int x2, int y2) {
  myGLCD.setColor(255, 0, 0);
  myGLCD.drawRoundRect (x1 - 1, y1 - 1, x2 + 1, y2 + 1);
  while (myTouch.dataAvailable())
    myTouch.read();
  myGLCD.setColor(0, 0, 0);
  myGLCD.drawRoundRect (x1 - 1, y1 - 1, x2 + 1, y2 + 1);
}

//void enroll()
//{
//  // Enroll test
//
//  // find open enroll id
//  int enrollid = 0;
//  bool usedid = true;
//  while (usedid == true)
//  {
//    usedid = fps.CheckEnrolled(enrollid);
```

```
//    if (usedid == true) enrollid++;
//  }
//  fps.EnrollStart(enrollid);
//
//  // enroll
//  String b = "Press finger to Enroll #" + enrollid;
//  myGLCD.print(b, 24, 120);
//  while (fps.IsPressFinger() == false) delay(100);
//  bool bret = fps.CaptureFinger(true);
//  int iret = 0;
//  if (bret != false)
//  {
//    Serial.println("Remove finger");
//    fps.Enroll1();
//    while (fps.IsPressFinger() == true) delay(100);
//    Serial.println("Press same finger again");
//    while (fps.IsPressFinger() == false) delay(100);
//    bret = fps.CaptureFinger(true);
//    if (bret != false)
//    {
//      Serial.println("Remove finger");
//      fps.Enroll2();
//      while (fps.IsPressFinger() == true) delay(100);
//      Serial.println("Press same finger yet again");
//      while (fps.IsPressFinger() == false) delay(100);
//      bret = fps.CaptureFinger(true);
//      if (bret != false)
//      {
//        Serial.println("Remove finger");
//        iret = fps.Enroll3();
//        if (iret == 0)
//        {
//          Serial.println("Enrolling Successfull");
//        }
//        else
//        {
//          Serial.print("Enrolling Failed with error code:");
//          Serial.println(iret);
//        }
//      }
//      else Serial.println("Failed to capture third finger");
//    }
//    else Serial.println("Failed to capture second finger");
```

```
//  }
//  else Serial.println("Failed to capture first finger");
//}
void lightUp(String pin)
{

  if (pin.length() == 0)
  {
    digitalWrite(blue1, LOW);
    digitalWrite(blue2, LOW);
    digitalWrite(blue3, LOW);
    digitalWrite(blue4, LOW);
  }
  if (pin.length() == 1)
  {
    digitalWrite(blue1, HIGH);
    digitalWrite(blue2, LOW);
    digitalWrite(blue3, LOW);
    digitalWrite(blue4, LOW);
  }
  if (pin.length() == 2)
  {
    digitalWrite(blue1, HIGH);
    digitalWrite(blue2, HIGH);
    digitalWrite(blue3, LOW);
    digitalWrite(blue4, LOW);
  }
  if (pin.length() == 3)
  {
    digitalWrite(blue1, HIGH);
    digitalWrite(blue2, HIGH);
    digitalWrite(blue3, HIGH);
    digitalWrite(blue4, LOW);
  }
  if (pin.length() == 4)
  {
    digitalWrite(blue1, HIGH);
    digitalWrite(blue2, HIGH);
    digitalWrite(blue3, HIGH);
    digitalWrite(blue4, HIGH);
  }
  delay(100);
}
```

```
void touchFunc(String pin)
{
  if (str.length() < 4)
  {
    str += pin;
    lightUp(str);
  }
}
void flappyBird()
{
  myGLCD.clrScr();
  myGLCD.setFont(BigFont);
}

void lockLCD()
{
  myGLCD.clrScr();
  myGLCD.setFont(BigFont);
  start();
}
void changepwd(String pin)
{
  if (pwd.length() < 4)
  {
    pwd += pin;
    lightUp(pwd);
  }
  else
  newpwd();
}
void newpwd()
{
  pwd = "";
  myGLCD.clrScr();
  myGLCD.setFont(BigFont);
    int n = 0;
  String nums[] = {"1", "4", "7", "2", "5", "8", "3", "6", "9", "GO", "0", "BK"};

  for (int x = 0; x < 4; x++)
  {
    for (int y = 0; y < 3; y++)
    {
      myGLCD.setBackColor(0, 0, 0);
```

```
    myGLCD.setColor(255, 255, 255);
    myGLCD.fillRect(12 + x * 77, 12 + y * 76, 77 + 77 * x, 76 + y * 76);
    myGLCD.setBackColor(255, 255, 255);
    myGLCD.setColor(0, 0, 0);
    if (n != 9 || n != 11)
    {
      myGLCD.print(nums[n], 37 + x * 77, 35 + y * 76);
    }

    else {
      myGLCD.print(nums[n], 32 + x * 77, 32 + y * 76);
    }
    n++;
  }
}
while (true)
{
  x = 0;
  y = 0;
  if (myTouch.dataAvailable())
  {
    myTouch.read();
    x = myTouch.getX();
    y = myTouch.getY();
    if ((y >= 12) && (y <= 76)) // Upper row
    {
      if ((x >= 12) && (x <= 77)) // Button: 1
      {
        waitForIt(12, 12, 77, 76);
        changepwd("1");
      }
      if ((x >= 89) && (x <= 154)) // Button: 1
      {
        waitForIt(89, 12, 154, 76);
        changepwd("2");
      }
      if ((x >= 166) && (x <= 231)) // Button: 1
      {
        waitForIt(166, 12, 231, 76);
        changepwd("3");
      }
      if ((x >= 243) && (x <= 308)) // Button: 1
      {
```

```
    waitForIt(243, 12, 308, 76);
    myGLCD.setFont(BigFont);
    myGLCD.clrScr();
    myGLCD.setBackColor(0, 0, 0);
    myGLCD.setColor(255, 255, 255);
    if(pwd.length() == 4)
    {
    myGLCD.print("YOUR NEW PASSWORD IS", 20, 50);
    myGLCD.print(pwd, 120, 140);
    delay(2000);
    mainMenu();
    }
  }
}
if ((y >= 88) && (y <= 152)) // Center row
{
  if ((x >= 12) && (x <= 77)) // Button: 1
  {
    waitForIt(12, 88, 77, 152);
    changepwd("4");
  }
  if ((x >= 89) && (x <= 154)) // Button: 1
  {
    waitForIt(89, 88, 154, 152);
    changepwd("5");
  }
  if ((x >= 166) && (x <= 231)) // Button: 1
  {
    waitForIt(166, 88, 231, 152);
    changepwd("6");
  }
  if ((x >= 243) && (x <= 308)) // Button: 1
  {
    waitForIt(243, 88, 308, 152);
    changepwd("0");
  }
}
if ((y >= 164) && (y <= 228)) // Center row
{
  if ((x >= 12) && (x <= 77)) // Button: 1
  {
    waitForIt(12, 164, 77, 228);
    changepwd("7");
```

```
      }
      if ((x >= 89) && (x <= 154)) // Button: 1
      {
        waitForIt(89, 164, 154, 228);
        changepwd("8");
      }
      if ((x >= 166) && (x <= 231)) // Button: 1
      {
        waitForIt(166, 164, 231, 228);
        changepwd("9");
      }
      if ((x >= 243) && (x <= 308)) // Button: 1
      {
        waitForIt(243, 164, 308, 228);
        if (pwd.length() > 0)
        {
          pwd = pwd.substring(0, pwd.length() - 1);
          lightUp(pwd);
          delay(100);
        }
        else
        {
          pwd = "";
          lightUp(pwd);
          delay(100);
        }
      }
    }
  }
 }
}
void addFing()
{
  myGLCD.clrScr();
  myGLCD.setFont(SmallFont);
  myGLCD.setBackColor(0, 0, 0);
  myGLCD.setColor(255, 255, 255);
  //enroll();
}

void mainMenu()
{
  int n = 0;
```

```
String nums[] = {"LOCK", "SET PWD", "ADD FING"};
myGLCD.clrScr();
myGLCD.setFont(SmallFont);
for (int x = 0; x < 3; x++)
{
  myGLCD.setBackColor(0, 0, 0);
  myGLCD.setColor(255, 255, 255);
  myGLCD.fillRect(4 + x * 104, 70, 104 + x * 104, 170);
  myGLCD.setBackColor(255, 255, 255);
  myGLCD.setColor(0, 0, 0);
  myGLCD.print(nums[x], 24 + x * 104, 120);
}
while (true)
{
  x = 0;
  y = 0;
  if (myTouch.dataAvailable())
  {
    myTouch.read();
    x = myTouch.getX();
    y = myTouch.getY();
    if ((x > 4 && x < 104 ) || (x > 108 && x < 208) || (x > 212 && x < 312)) {
      break;
    }
  }
}
if (x > 4 && x < 104)
{
  waitForIt(4, 70, 104, 170);
  myservo.write(1);
  delay(2000);
  lockLCD();
}
if (x > 108 && x < 208)
{
  waitForIt(108, 70, 208, 170);
  newpwd();
}
if (x > 212 && x < 312)
{
  waitForIt(212, 70, 312, 170);
  addFing();
}
```

```
}

boolean pwdConfirm(String a)
{

  if (a.equals(pwd))
  {

    digitalWrite(blue1, LOW);
    digitalWrite(blue2, LOW);
    digitalWrite(blue3, LOW);
    digitalWrite(blue4, LOW);
    myservo.write(179);
    mainMenu();
    return true;
  }

  else
  {

    return false;
  }
}

void screenFunc()
{
  int n = 0;
  String nums[] = {"1", "4", "7", "2", "5", "8", "3", "6", "9", "GO", "0", "BK"};

  for (int x = 0; x < 4; x++)
  {
    for (int y = 0; y < 3; y++)
    {
      myGLCD.setBackColor(0, 0, 0);
      myGLCD.setColor(255, 255, 255);
      myGLCD.fillRect(12 + x * 77, 12 + y * 76, 77 + 77 * x, 76 + y * 76);
      myGLCD.setBackColor(255, 255, 255);
      myGLCD.setColor(0, 0, 0);
      if (n != 9 || n != 11)
      {
        myGLCD.print(nums[n], 37 + x * 77, 35 + y * 76);
      }
```

```
    else {
      myGLCD.print(nums[n], 32 + x * 77, 32 + y * 76);
    }
    n++;
  }
}
while (true)
{

  x = 0;
  y = 0;
  if (myTouch.dataAvailable())
  {
    myTouch.read();
    x = myTouch.getX();
    y = myTouch.getY();
    if ((y >= 12) && (y <= 76)) // Upper row
    {
      if ((x >= 12) && (x <= 77)) // Button: 1
      {
        waitForIt(12, 12, 77, 76);
        touchFunc("1");
      }
      if ((x >= 89) && (x <= 154)) // Button: 1
      {
        waitForIt(89, 12, 154, 76);
        touchFunc("2");
      }
      if ((x >= 166) && (x <= 231)) // Button: 1
      {
        waitForIt(166, 12, 231, 76);
        touchFunc("3");
      }
      if ((x >= 243) && (x <= 308)) // Button: 1
      {
        waitForIt(243, 12, 308, 76);
        if (pwdConfirm(str))
        {
          digitalWrite(green, HIGH);
          delay(1000);
          digitalWrite(green, LOW);
          //Serial.println("ACCESS GRANTED");
```

```
      str = "";
      delay(100);
      break;
    }
    else
    {
      digitalWrite(redPin, HIGH);
      delay(1000);
      digitalWrite(redPin, LOW);
      str = "";
      digitalWrite(blue1, LOW);
      digitalWrite(blue2, LOW);
      digitalWrite(blue3, LOW);
      digitalWrite(blue4, LOW);
      //Serial.println("ACCESS DENIED");
      delay(100);
    }

  }
}
if ((y >= 88) && (y <= 152)) // Center row
{
  if ((x >= 12) && (x <= 77)) // Button: 1
  {
    waitForIt(12, 88, 77, 152);
    touchFunc("4");
  }
  if ((x >= 89) && (x <= 154)) // Button: 1
  {
    waitForIt(89, 88, 154, 152);
    touchFunc("5");
  }
  if ((x >= 166) && (x <= 231)) // Button: 1
  {
    waitForIt(166, 88, 231, 152);
    touchFunc("6");
  }
  if ((x >= 243) && (x <= 308)) // Button: 1
  {
    waitForIt(243, 88, 308, 152);
    touchFunc("0");
  }
}
```

```
    if ((y >= 164) && (y <= 228)) // Center row
    {
      if ((x >= 12) && (x <= 77)) // Button: 1
      {
        waitForIt(12, 164, 77, 228);
        touchFunc("7");
      }
      if ((x >= 89) && (x <= 154)) // Button: 1
      {
        waitForIt(89, 164, 154, 228);
        touchFunc("8");
      }
      if ((x >= 166) && (x <= 231)) // Button: 1
      {
        waitForIt(166, 164, 231, 228);
        touchFunc("9");
      }
      if ((x >= 243) && (x <= 308)) // Button: 1
      {
        waitForIt(243, 164, 308, 228);
        if (str.length() > 0)
        {
          str = str.substring(0, str.length() - 1);
          lightUp(str);
          delay(100);
        }
        else
        {
          str = "";
          lightUp(str);
          delay(100);
        }
      }
    }
   }
  }
}
void setup()
{
  Serial.begin(9600);
  myservo.attach(44);
  pinMode(blue1, OUTPUT);
  pinMode(blue2, OUTPUT);
```

```
  pinMode(blue3, OUTPUT);
  pinMode(blue4, OUTPUT);
  pinMode(green, OUTPUT);
  pinMode(redPin, OUTPUT);
  pinMode(45, OUTPUT);
  start();
}
void start() {
  // put your setup code here, to run once:

  delay(100);
  fps.Open();
  fps.SetLED(true);
  digitalWrite(redPin, LOW);
  digitalWrite(green, LOW);
  digitalWrite(blue1, LOW);
  digitalWrite(blue2, LOW);
  digitalWrite(blue3, LOW);
  digitalWrite(blue4, LOW);
  digitalWrite(45, HIGH);
  myservo.write(1);
  checkFinger();
}

void checkFinger() {
  // put your main code here, to run repeatedly:
  // Identify fingerprint test
  while(true)
  {
  if (fps.IsPressFinger())
  {

    fps.CaptureFinger(false);
    int id = fps.Identify1_N();
    if (id < 200)
    {
      fps.SetLED(false);
      digitalWrite(45, LOW);
      myGLCD.InitLCD();
      myGLCD.setFont(BigFont);
      myGLCD.clrScr();
      myTouch.InitTouch();
      myTouch.setPrecision(PREC_HI);
```

```
      screenFunc();
      //    digitalWrite(blue1, HIGH);
      //    digitalWrite(blue2, HIGH);
      //    digitalWrite(blue3, HIGH);
      //    digitalWrite(blue4, HIGH);
    }
  }
  else
  {
    digitalWrite(blue1, LOW);
    digitalWrite(blue2, LOW);
    digitalWrite(blue3, LOW);
    digitalWrite(blue4, LOW);
  }
  delay(100);
}
}
void loop()
{
  delay(100);
}
```