LISTING PROGRAM

#include "Adafruit\_Fingerprint.h"

#include "SoftwareSerial.h"

#include <Servo.h>

#include <Arduino.h>

#include <U8g2lib.h>

#ifdef U8X8\_HAVE\_HW\_SPI

#include <SPI.h>

#endif

#ifdef U8X8\_HAVE\_HW\_I2C

#include <Wire.h>

#endif

U8G2\_ST7920\_128X64\_F\_HW\_SPI u8g2(U8G2\_R0, /\* CS=/ 10, / reset=\*/ U8X8\_PIN\_NONE);

SoftwareSerial mySerial(2, 3);

Adafruit\_Fingerprint finger = Adafruit\_Fingerprint( & mySerial);

Servo myservo;

int nilaisensor;

byte irf = 6;

int idFinger = 0;

int pos = 0;

const int bzPin = 7;

void buzz(boolean valid) {

if (valid) {

tone(bzPin, 523, 200);

} else if (!valid) {

tone(bzPin, 123, 600);

}

}

void setup() {

finger.begin(57600);

myservo.attach(4);

pinMode(8, INPUT);

myservo.write(0);

Serial.begin(9600);

pinMode(irf, OUTPUT);

digitalWrite(irf, LOW);

u8g2.begin();

u8g2.clearBuffer();

u8g2.setFont(u8g2\_font\_ncenB08\_tr);

u8g2.sendBuffer();

}

void loop() {

FINGERPRINT();

Serial.println(idFinger);

if (idFinger >= 0) {

u8g2.clearBuffer();

u8g2.drawStr(8, 37, "akses diterima");

u8g2.sendBuffer();

digitalWrite(irf, HIGH);

delay(2000);

for (pos = 0; pos <= 180; pos += 1)

{

myservo.write(pos);

delay(10);

}

delay(8500);

for (pos = 180; pos >= 0; pos -= 1)

{

myservo.write(pos);

delay(5);

}

delay(1500);

digitalWrite(irf, LOW);

} else if (idFinger == -2) {

buzz(true);

u8g2.clearBuffer();

u8g2.drawStr(8, 37, "akses ditolak");

u8g2.sendBuffer();

delay(2000);

}

if (digitalRead(8) == HIGH) {

digitalWrite(irf, HIGH);

u8g2.clearBuffer();

u8g2.drawStr(8, 37, "pintu terbuka");

u8g2.sendBuffer();

delay(2000);

for (pos = 0; pos <= 180; pos += 1)

{

myservo.write(pos);

delay(10);

}

delay(8500);

for (pos = 180; pos >= -25; pos -= 1)

{

myservo.write(pos);

delay(5);

}

delay(1500);

digitalWrite(irf, LOW);

}

u8g2.clearBuffer();

u8g2.drawStr(0, 33, "scan disini");

u8g2.sendBuffer();

delay(50);

}

void FINGERPRINT() {

idFinger = getFingerprintIDez();

delay(50);

}

uint8\_t getFingerprintID() {

uint8\_t p = finger.getImage();

switch (p) {

case FINGERPRINT\_OK:

break;

case FINGERPRINT\_NOFINGER:

return p;

case FINGERPRINT\_PACKETRECIEVEERR:

return p;

case FINGERPRINT\_IMAGEFAIL:

return p;

default:

return p;

}

p = finger.image2Tz();

switch (p) {

case FINGERPRINT\_OK:

break;

case FINGERPRINT\_IMAGEMESS:

return p;

case FINGERPRINT\_PACKETRECIEVEERR:

return p;

case FINGERPRINT\_FEATUREFAIL:

return p;

case FINGERPRINT\_INVALIDIMAGE:

return p;

default:

return p;

}

p = finger.fingerFastSearch();

if (p == FINGERPRINT\_OK) {

} else if (p == FINGERPRINT\_PACKETRECIEVEERR) {

return p;

} else if (p == FINGERPRINT\_NOTFOUND) {

return p;

} else {

return p;

}

return finger.fingerID;

}

int getFingerprintIDez() {

uint8\_t p = finger.getImage();

if (p != FINGERPRINT\_OK) return -1;

p = finger.image2Tz();

if (p != FINGERPRINT\_OK) return -1;

p = finger.fingerFastSearch();

if (p != FINGERPRINT\_OK) return -2;

return finger.fingerID;

}