

LAMPIRAN

1. Listing program keseluruhan Arduino

```
#include <Adafruit_GFX.h>
#include <Adafruit_ST7735.h>
#include <SPI.h>
#include <SoftwareSerial.h>
#include <Adafruit_Fingerprint.h>

char menu

uint8_t id;
uint8_t getFingerprintEnroll();
int getFingerprintIDez();
bool a=0;

SoftwareSerial mySerial(3, 4);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);

#define TFT_CS 10
#define TFT_RST 12
#define TFT_DC 9
#define TFT_SCLK 13
#define TFT_MOSI 11

Adafruit_ST7735 tft = Adafruit_ST7735(TFT_CS, TFT_DC, TFT_RST);

void setup(void)
{
  pinMode(2, OUTPUT);
  digitalWrite(2, LOW);
  pinMode(5, INPUT);
  digitalWrite(5, HIGH);
  pinMode(6, INPUT);
  digitalWrite(6, HIGH);
  pinMode(7, INPUT);
  digitalWrite(7, HIGH);
  delay(1000);
  on();
  tft.initR(INITR_BLACKTAB);
  tft.setTextWrap(true);
  opening();
  tft.fillRoundRect(1, 15, 128, 50, 8, ST7735_YELLOW);
  tft.fillRoundRect(7, 18, 114, 15, 8, ST7735_BLACK);
  tft.setTextColor(ST7735_WHITE);
  tft.setTextSize(1);
  tft.setCursor(12, 21);
  tft.println("Mengecek Perangkat");
  cek_sensor();
  delay(1000);
  tft.fillScreen(ST7735_BLACK);
  Serial.begin(9600);
}
```

```

    finger.begin(57600);
}

void loop()
{
    pilih_menu();
}

void opening()
{
    tft.fillScreen(ST7735_BLACK);
    tft.setTextColor(ST7735_WHITE);
    tft.setTextSize(1);
    tft.setCursor(35, 44);
    tft.println("Tugas Akhir");
    tft.setCursor(31, 64);
    tft.println("Identifikasi");
    tft.setCursor(20, 74);
    tft.println("Bayi Baru Lahir");
    tft.setCursor(25, 84);
    tft.println("Dengan sistem");
    tft.setCursor(31, 94);
    tft.println("Fingerprint");
    tft.setCursor(55, 114);
    tft.println("Oleh");
    tft.setCursor(32, 124);
    tft.println("Syarifuddin");
    delay(5000);
    tft.fillScreen(ST7735_BLACK);
}

void cek_sensor()
{
    baterai();
    if (finger.verifyPassword())
    {
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 36);
        tft.println("Test Sensor:");
        tft.setCursor(1, 46);
        tft.println("Terpasang");
        ok();
    }
    else
    {
        tft.setTextColor(ST7735_RED);
        tft.setTextSize(1);
        tft.setCursor(1, 36);
        tft.println("Test Sensor:");
        tft.setCursor(1, 46);
        tft.println("Tidak terpasang");
        error();
    }
}

```

```

        error();
        while (1);
    }
}

void baterai()
{
    tft.fillRoundRect(70,1,40,14,8, ST7735_BLACK);
    int Batpin = A0;
    float voltage = 0;
    int sensorVal = analogRead(Batpin);
    for (int i=0; i<100; i++)
    {
        voltage+=Batpin;
    }
    voltage/=100;
    voltage = sensorVal * (5.0/1023.0);
    if (voltage >= 3.51)
    {
        tft.setTextColor(ST7735_WHITE);
        tft.setTextSize(1);
        tft.setCursor(50, 1);
        tft.print("Bat: ");
        tft.print(voltage);
        tft.println(" V");
    }
    if (voltage <= 3.5)
    {
        tft.setTextColor(ST7735_RED);
        tft.setTextSize(1);
        tft.setCursor(50, 1);
        tft.print("Bat: ");
        tft.print(voltage);
        tft.println(" V");
        tft.fillRoundRect(5, 69, 117, 20, 8, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(25, 74);
        tft.println("Baterai Habis");
        error();
        delay (5000);
    }
    delay (100);
}

void ok()
{
    digitalWrite(2,HIGH); //pin 2 bernilai high (1)
    delay(100); //tunda 0,1 detik
    digitalWrite(2,LOW); //pin 2 bernilai low (0)
}

```

```
void on()
{
    digitalWrite(2,HIGH);
    delay(300);
    digitalWrite(2,LOW);
    delay(50);
    digitalWrite(2,HIGH);
    delay(200);
    digitalWrite(2,LOW);
    delay(50);
    digitalWrite(2,HIGH);
    delay(100);
    digitalWrite(2,LOW);
    delay(50);
}

void error()
{
    digitalWrite(2,HIGH);
    delay(1000);
    digitalWrite(2,LOW);
    delay(100);
    digitalWrite(2,HIGH);
    delay(1000);
    digitalWrite(2,LOW);
    delay(100);
}

void selesai()
{
    digitalWrite(2,HIGH);
    delay(100);
    digitalWrite(2,LOW);
    delay(50);
    digitalWrite(2,HIGH);
    delay(100);
    digitalWrite(2,LOW);
    delay(50);
    digitalWrite(2,HIGH);
    delay(100);
    digitalWrite(2,LOW);
    delay(500);
    digitalWrite(2,HIGH);
    delay(100);
    digitalWrite(2,LOW);
    delay(50);
    digitalWrite(2,HIGH);
    delay(100);
    digitalWrite(2,LOW);
    delay(50);
    digitalWrite(2,HIGH);
    delay(100);
}
}
```

```

void readnumber()
{
  while(digitalRead(5))
  {
    if(digitalRead(6)==0)
    {
      id++;
      tft.fillRoundRect(45, 45, 20, 10, 0, ST7735_BLACK);
      ok();
      delay(300);
    }
    if(digitalRead(7)==0 && id > 0)
    {
      id--;
      tft.fillRoundRect(45, 45, 20, 10, 0, ST7735_BLACK);
      ok();
      delay(300);
    }
    tft.setTextColor(ST7735_WHITE);
    tft.setTextSize(1);
    tft.setCursor(45, 45);
    tft.print(id);
    tft.setCursor(70, 45);
    tft.print("6789");
  }
  delay(500);
}

void pilih_menu()
{
  menu=Serial.read();
  baterai();
  tft.fillRoundRect(5, 29, 117, 20, 8, ST7735_YELLOW);
  tft.fillRoundRect(5, 69, 117, 20, 8, ST7735_BLUE);
  tft.fillRoundRect(5, 109, 117, 20, 8, ST7735_GREEN);
  tft.setTextColor(ST7735_WHITE);
  tft.setTextSize(1);
  tft.setCursor(35, 74);
  tft.println("Pilih Menu");
  delay(2000);
  while (menu=='P')
  {
    ok();
    tft.fillScreen(ST7735_BLACK);
    baterai();
    tft.fillRoundRect(1, 15, 117, 15, 8, ST7735_YELLOW);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(30, 18);
    tft.println("Pendaftaran");
    id=0;
    tft.setTextColor(ST7735_WHITE);
    tft.setTextSize(1);
  }
}

```

```

tft.setTextSize(1);
  tft.setCursor(5, 35);
  tft.println("Tentukan no ID");
  tft.setCursor(5, 45);
  tft.println("No ID:");
  readnumber();
  while (id==0)
  {
    readnumber();
  }
  tft.setCursor(5, 55);
  tft.println("ID telah ditentukan");
  delay(1000);
  tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
  tft.setTextColor(ST7735_BLACK);
  tft.setTextSize(1);
  tft.setCursor(1, 120);
  tft.println("Letakkan jari");
  delay(1000);
  while (! getFingerprintEnroll() );
}
while (menu=='C')
{
  ok();
  tft.fillScreen(ST7735_BLACK);
  baterai();
  menu=a;
  while (menu==a)
  {
    tft.fillRoundRect(5, 64, 117, 20, 8, ST7735_BLACK);
    tft.fillRoundRect(1, 15, 117, 15, 8, ST7735_GREEN);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(50, 18);
    tft.print("Cari");
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
    tft.setCursor(1, 120);
    tft.print("Letakkan Jari Anda");
    getFingerprintID();
  }
}
while (menu=='H')
{
  tft.fillScreen(ST7735_BLACK);
  deleteFingerprint(id);
}
}

uint8_t getFingerprintEnroll() //program registrasi
{
  baterai();
  ok();
  int p;
  tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);

```

```
tft.setTextColor(ST7735_BLACK);
tft.setTextSize(1);
tft.setCursor(1, 120);
tft.print("Tunggu Sampai");
tft.setCursor(1, 130);
tft.print("ID sidik jari:");
tft.setCursor(1, 140);
tft.print(id);
tft.setCursor(20, 140);
tft.print("6789 Benar!");
while (p != FINGERPRINT_OK)
{
  p = finger.getImage();
  switch (p)
  {
    case FINGERPRINT_OK:
      tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
      tft.setTextColor(ST7735_BLACK);
      tft.setTextSize(1);
      tft.setCursor(1, 120);
      tft.print("sidik diambil");
      ok();
      break;
    case FINGERPRINT_NOFINGER:
      break;
    case FINGERPRINT_PACKETRECEIVEERR:
      tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
      tft.setTextColor(ST7735_BLACK);
      tft.setTextSize(1);
      tft.setCursor(1, 120);
      tft.print("Kesalahan komunikasi");
      error();
      delay(3000);
      tft.fillScreen(ST7735_BLACK);
      return p;
      break;
    case FINGERPRINT_IMAGEFAIL:
      tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
      tft.setTextColor(ST7735_BLACK);
      tft.setTextSize(1);
      tft.setCursor(1, 120);
      tft.println("Kesalahan sidik jari");
      error();
      delay(3000);
      tft.fillScreen(ST7735_BLACK);
      return p;
      break;
    default:
      tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
      tft.setTextColor(ST7735_BLACK);
      tft.setTextSize(1);
      tft.setCursor(1, 120);
      tft.println("Kesalahan tidak diketahui");
      error();
```

```

        delay(3000);
        tft.fillScreen(ST7735_BLACK);
        return p;
        break;
    }
}
delay(1000);

p = finger.image2Tz(1);
switch (p)
{
    case FINGERPRINT_OK:
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.println("mengkonversi sidik jari");
        ok();
        break;
    case FINGERPRINT_IMAGEMESS:
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.println("sidik jari kacau");
        error();
        delay(3000);
        tft.fillScreen(ST7735_BLACK);
        return p;
    case FINGERPRINT_PACKETRECEIVEERR:
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.println("Kesalahan komunikasi");
        error();
        return p;
    case FINGERPRINT_FEATUREFAIL:
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.println("Fitur sidik jari tidak ditemukan");
        error();
        return p;
    case FINGERPRINT_INVALIDIMAGE:
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.println("Fitur sidik jari tidak ditemukan");
        error();
        return p;
}

```



```

default:
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.println("Kesalahan tidak diketahui");
    error();
    return p;
}

delay(1000);
tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
tft.setTextColor(ST7735_BLACK);
tft.setTextSize(1);
tft.setCursor(1, 120);
tft.print("lepas jari");
ok();
delay(1000);
p = 0;
while (p != FINGERPRINT_NOFINGER)
{
    p = finger.getImage();
}
p = -1;
tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
tft.setTextColor(ST7735_BLACK);
tft.setTextSize(1);
tft.setCursor(1, 120);
tft.print("Letakkan jari lagi");
ok();
while (p != FINGERPRINT_OK)
{
    p = finger.getImage();
    switch (p)
    {
    case FINGERPRINT_OK:
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.print("Sidik jari diambil");
        ok();
        break;
    case FINGERPRINT_NOFINGER:
        break;
    case FINGERPRINT_PACKETRECEIVEERR:
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.println("Kesalahan komunikasi");
        error();
        break;
    }
}

```

```

case FINGERPRINT_IMAGEFAIL:
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.println("Kesalahan sidik jari");
    error();
    break;
default:
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.println("Kesalahan tidak diketahui");
    error();
    break;
}
}
delay(1000);
p = finger.image2Tz(2);
switch (p)
{
case FINGERPRINT_OK:
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("Konversi didik jari");
    ok();
    break;
case FINGERPRINT_IMAGEMESS:
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 110);
    tft.print("Sidik jari kacau");
    error();
    return p;
case FINGERPRINT_PACKETRECIEVEERR:
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 110);
    tft.println("Kesalahan komunikasi");
    error();
    return p;
case FINGERPRINT_FEATUREFAIL:
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 110);
    tft.println("Fitur sidik jari tidak ditemukan");
    error();
    return p;
}
}

```

```

    case FINGERPRINT_INVALIDIMAGE:
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 110);
        tft.println("Fitur sidik jari tidak ditemukan");
        error();
        return p;
    default:
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 110);
        tft.println("Kesaahan tidak diketahui");
        error();
        return p;
}

delay(1000);
tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
tft.setTextColor(ST7735_BLACK);
tft.setTextSize(1);
tft.setCursor(1, 120);
tft.println("ID sidik jari");
tft.setCursor(1, 130);
tft.print(id);
tft.setCursor(20, 130);
tft.println("6789 dibuat");
ok();
delay(1000);

p = finger.createModel();
if (p == FINGERPRINT_OK)
{
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("sidik jari cocok!");
    ok()
}
else if (p == FINGERPRINT_PACKETRECEIVEERR)
{
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("kesalahan komunikasi");
    error();
    return p;
}

```

```

else if (p == FINGERPRINT_ENROLLMISMATCH)
{
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("Sidik jari tidak cocok");
    error();
    return p;
}
else
{
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("kesalahan tidak diketahui");
    error();
    return p;
}

delay(2000);
tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
tft.setTextColor(ST7735_BLACK);
tft.setTextSize(1);
tft.setCursor(1, 120);
tft.print("ID:");
tft.setCursor(20, 120);
tft.print(id);
tft.setCursor(45, 120);
tft.print(6789);
p = finger.storeModel(id);
if (p == FINGERPRINT_OK)
{
    tft.setCursor(10, 130);
    tft.print("Disimpan!");
    Serial.print(id);
    Serial.println("6789P");
    tft.setTextColor(ST7735_WHITE);
    tft.setTextSize(1);
    tft.setCursor(5, 65);
    tft.println("ID berhasil disimpan!");
    selesai();
    delay(5000);
    tft.fillScreen(ST7735_BLACK);
    loop();
}
else if (p == FINGERPRINT_PACKETRECEIVEERR)
{
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("kesalahan komunikasi");
}

```

```

        error();
        return p;
    }
    else if (p == FINGERPRINT_BADLOCATION)
    {
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.print("tidak dapat disimpan pada lokasi tersebut");
        error();
        return p;
    }
    else if (p == FINGERPRINT_FLASHERR)
    {
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.print("kesalahan tuliske flash");
        error();
        return p;
    }
    else
    {
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.print("kesalahan tidak diketahui");
        error();
        return p;
    }
    delay(3000);
}

uint8_t getFingerprintID() //program cari
{
    baterai();
    uint8_t p = finger.getImage();
    if (p != FINGERPRINT_OK)
        return p;
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("Lepas jari");
    delay(1000);

    p = finger.image2Tz();

    p = finger.fingerFastSearch();
}

```

```

    if (p == FINGERPRINT_NOTFOUND)
    {
        tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
        tft.setTextColor(ST7735_BLACK);
        tft.setTextSize(1);
        tft.setCursor(1, 120);
        tft.print("Sidik jari tidak terdaftar");
        error();
        delay(2000);
        return p;
    }

    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_BLUE);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("Sidik jari terdaftar");
    tft.setTextColor(ST7735_WHITE);
    tft.setCursor(5, 35);
    tft.println("ID sidik jari:");
    tft.setCursor(5, 45);
    tft.println(finger.fingerID);
    tft.setCursor(30, 45);
    tft.print("6789");
    Serial.print(finger.fingerID);
    Serial.println("6789C");
    selesai();
    delay(5000);
    tft.fillScreen(ST7735_BLACK);
    loop();
}

uint8_t deleteFingerprint(uint8_t id) //Program hapus
{
    ok();
    int p;
    tft.fillScreen(ST7735_BLACK);
    baterai();
    tft.fillRoundRect(1, 15, 117, 15, 8, ST7735_GREEN);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(50, 18);
    tft.print("Hapus");
    tft.setTextColor(ST7735_WHITE);
    tft.setTextSize(1);
    tft.setCursor(5, 35);
    tft.println("Tentukan no ID");
    tft.setCursor(5, 45);
    tft.println("No ID:");
    readnumber()

    p = finger.deleteModel(id);
    if (p == FINGERPRINT_OK)

```

```

{
    tft.setTextColor(ST7735_WHITE);
    tft.setTextSize(1);
    tft.setCursor(5, 55);
    tft.println("Berhasil di hapus!");
    selesai();
    delay(10000);
    tft.fillScreen(ST7735_BLACK);
    loop();
}
else if (p == FINGERPRINT_PACKETRECEIVEERR)
{
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("kesalahan komunikasi");
    error();
    return p;
}
else if (p == FINGERPRINT_BADLOCATION)
{
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("tidak dapat dihapus pada lokasi tersebut");
    error();
    return p;
}
else if (p == FINGERPRINT_FLASHERR)
{
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("kesalahan tuliske flash");
    error();
    return p;
}
else
{
    tft.fillRoundRect(0, 115, 128, 40, 0, ST7735_RED);
    tft.setTextColor(ST7735_BLACK);
    tft.setTextSize(1);
    tft.setCursor(1, 120);
    tft.print("kesalahan tidak diketahui");
    error();
    return p;
}
}

```

2. List Program Keseluruhan Delphi 7

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    Edit1: TEdit;
    Label1: TLabel;
    Edit2: TEdit;
    Label2: TLabel;
    Button1: TButton;
    procedure Button1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

uses Unit2;

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
  if (edit1.text<>'a') then
  begin
    messagedlg('Admin tidak terdaftar', MTWarning, [MBOK],0);
    exit;
  end;
  if (edit2.text<>'a') then
  begin
    messagedlg('Kata Sandi Salah', MTWarning, [MBOK],0);
    exit;
  end;
  showmessage('Selamat datang');

  with TForm2.Create(Self) do
  try
    form1.hide;
    form2.show;
    form2.Panel3.show;
    form2.Panel4.hide;
```



```
form2.Panel5.Hide;
form2.Panel6.hide;
except
end;
end;
end.

unit Unit2;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, ExtCtrls, DBCtrls, ADODB, DB, CPort, Grids,
  DBGrids,STRutils, sSkinManager, ComCtrls;

type
  TForm2 = class(TForm)
    Panel1: TPanel;
    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    Button5: TButton;
    Panel2: TPanel;
    Panel4: TPanel;
    Label10: TLabel;
    Label11: TLabel;
    Label12: TLabel;
    Label13: TLabel;
    Label14: TLabel;
    Label15: TLabel;
    Label16: TLabel;
    Label17: TLabel;
    Label18: TLabel;
    Button11: TButton;
    Button12: TButton;
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit6: TEdit;
    Edit7: TEdit;
    Edit8: TEdit;
    Panel5: TPanel;
    Label19: TLabel;
    Label20: TLabel;
    Label21: TLabel;
    Label22: TLabel;
    Label23: TLabel;
    Label24: TLabel;
    Label25: TLabel;
    Label26: TLabel;
    Label27: TLabel;
  end;
end;
```

```
Edit9: TEdit;
Edit10: TEdit;
Edit11: TEdit;
Edit14: TEdit;
Edit15: TEdit;
Edit16: TEdit;
Button13: TButton;
Button14: TButton;
ComPort1: TComPort;
ADOConnection1: TADOConnection;
ADOTable1: TADOTable;
ADOQuery1: TADOQuery;
DBGrid1: TDBGrid;
DataSource1: TDataSource;
Panel3: TPanel;
Label1: TLabel;
Label2: TLabel;
Label3: TLabel;
Label4: TLabel;
Label5: TLabel;
Label6: TLabel;
Label7: TLabel;
Label8: TLabel;
Label9: TLabel;
DBText1: TDBText;
DBText2: TDBText;
DBText3: TDBText;
DBText4: TDBText;
DBText5: TDBText;
DBText6: TDBText;
DBText7: TDBText;
DBText8: TDBText;
Panel6: TPanel;
Label28: TLabel;
Edit17: TEdit;
Button9: TButton;
ComboBox1: TComboBox;
ComboBox2: TComboBox;
Button10: TButton;
Memo1: TMemo;
Timer1: TTimer;
sSkinManager1: TsSkinManager;
DateTimePicker2: TDateTimePicker;
DateTimePicker3: TDateTimePicker;
Button15: TButton;
Button16: TButton;
Button17: TButton;
Button6: TButton;
procedure Button3Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure Button5Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button11Click(Sender: TObject);
```

```

    procedure Button13Click(Sender: TObject);
    procedure Button9Click(Sender: TObject);
    procedure Button10Click(Sender: TObject);
    procedure MemolChange(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure ComPort1RxChar(Sender: TObject; Count: Integer);
    procedure Button14Click(Sender: TObject);
    procedure Button12Click(Sender: TObject);
    procedure Button15Click(Sender: TObject);
    procedure Button16Click(Sender: TObject);
    procedure Button17Click(Sender: TObject);
    procedure Button6Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form2: TForm2;

implementation

uses Unit3, Unit1;

{$R *.dfm}

procedure TForm2.Button3Click(Sender: TObject);
begin
if button1.Caption='Putuskan' then
begin
comport1.WriteStr('P');

edit1.Clear;
edit2.Clear;
edit3.Clear;
edit6.Clear;
edit7.Clear;
edit8.Clear;

panel3.Hide;
panel4.Show;
panel5.Hide;
panel6.Hide;
end
else if button1.Caption='Koneksikan' then
begin
showmessage('Koneksikan dengan perangkat');
end
end;

```

```

procedure TForm2.Button4Click(Sender: TObject);
begin
if button1.Caption='Putuskan' then
begin
Edit9.Text:=adoquery1.FieldByName('ID').AsString;
Edit10.Text:=adoquery1.FieldByName('Ruangan').AsString;
Edit11.Text:=adoquery1.FieldByName('Nama Ibu').AsString;
Combobox2.Text:=adoquery1.FieldByName('Jenis Kelamin Bayi').AsString;
datetimepicker3.Date:=adoquery1.FieldByName('Tanggal Lahir
Bayi').Asdatetime;
Edit14.Text:=adoquery1.FieldByName('Berat Bayi').AsString;
Edit15.Text:=adoquery1.FieldByName('Panjang Bayi').AsString;
Edit16.Text:=adoquery1.FieldByName('No Rekam Medis').AsString;

panel3.Hide;
panel4.Hide;
panel5.Show;
panel6.Hide;
end

else if button1.Caption='Koneksikan' then
begin
showmessage('Koneksikan dengan perangkat');
end
end;

procedure TForm2.Button5Click(Sender: TObject);
begin
if button1.Caption='Putuskan' then
begin
comport1.WriteStr('C');
edit17.Clear;
panel3.Hide;
panel4.Hide;
panel5.Hide;
panel6.Show;
end
else if button1.Caption='Koneksikan' then
begin
showmessage('Koneksikan dengan perangkat');
end;
end;

procedure TForm2.Button1Click(Sender: TObject);
begin
if button1.Caption='Koneksikan' then
begin
Comport1.Open;
button1.Caption:='Putuskan';
showmessage('Berhasil terkoneksi!');
form2.ADOQuery1.Close;
form2.ADOQuery1.SQL.Clear;
form2.ADOQuery1.SQL.Append('select* from databayi');
form2.ADOQuery1.Open;

```

```

end
else if button1.caption='Putuskan' then
begin
Comport1.Close;
button1.Caption:='Koneksikan';
panel3.Show;
panel4.Hide;
panel5.Hide;
panel6.Hide;
end;
end;

procedure TForm2.Button2Click(Sender: TObject);
begin
comport1.ShowSetupDialog;
end;

procedure TForm2.Button11Click(Sender: TObject);
begin
adoquery1.Open;
adoquery1.append;
adoquery1.FieldByName('ID').AsString:=Edit1.Text;
adoquery1.FieldByName('Ruangan').AsString:=Edit2.Text;
adoquery1.FieldByName('Nama Ibu').AsString:=Edit3.Text;
adoquery1.FieldByName('Tanggal
Lahir
Bayi').Asdatetime:=datetimepicker2.date;
adoquery1.FieldByName('Jenis Kelamin Bayi').AsString:=Combobox1.Text;
adoquery1.FieldByName('Berat Bayi').AsString:=Edit6.Text;
adoquery1.FieldByName('Panjang Bayi').AsString:=Edit7.Text;
adoquery1.FieldByName('No Rekam Medis').AsString:=Edit8.Text;
adoquery1.Post;
showmessage('Data Berhasil di Tambahkan');

begin
form2.adoquery1.Close;
form2.adoquery1.SQL.clear;
form2.adoquery1.SQL.Add('select * from databayi');
form2.adoquery1.Open;
panel3.Show;
panel4.Hide;
panel5.Hide;
panel6.Hide;
end;
end;

procedure TForm2.Button13Click(Sender: TObject);
begin
adoquery1.Open;
adoquery1.Edit;
adoquery1.FieldByName('ID').AsString:=Edit9.Text;
adoquery1.FieldByName('Ruangan').AsString:=Edit10.Text;
adoquery1.FieldByName('Nama Ibu').AsString:=Edit11.Text;
adoquery1.FieldByName('Jenis Kelamin Bayi').AsString:=Combobox2.Text;

```

```

adoquery1.FieldName('Tanggal Lahir Bayi').Asdatetime:=Datetime
picker3 .date;
adoquery1.FieldName('Berat Bayi').AsString:=Edit14.Text;
adoquery1.FieldName('Panjang Bayi').AsString:=Edit15.Text;
adoquery1.FieldName('No Rekam Medis').AsString:=Edit16.Text;
adoquery1.Post;
showmessage('Data Berhasil di Edit');

begin
form2.adoquery1.Close;
form2.adoquery1.SQL.clear;
form2.adoquery1.SQL.Add('select * from databayi');
form2.adoquery1.Open;

panel3.Show;
panel4.Hide;
panel5.Hide;
panel6.Hide;
end;
end;

procedure TForm2.Button9Click(Sender: TObject);
begin
adoquery1.Close;
adoquery1.SQL.clear;
adoquery1.SQL.Add('select * from databayi where ID="'+edit17.Text+'"');
adoquery1.Open;

panel3.Show;
panel4.Hide;
panel5.Hide;
panel6.Hide;
end;

procedure TForm2.Button10Click(Sender: TObject);
begin
panel3.Show;
panel4.Hide;
panel5.Hide;
panel6.Hide;
end;

procedure TForm2.Memo1Change(Sender: TObject);
var
data:string;
hitung:real;
begin
data:=memo1.Lines[memo1.lines.count-1];
if rightstr(data,1)='P' then
begin
if midstr(data,5,1)='9' then
begin
edit1.Text:=leftstr(data,5);
hitung:=strtofloat(edit1.Text);

```

```

edit1.Text:=floattostr(hitung);
end

else if midstr(data,5,1)='8' then
begin
edit1.Text:=leftstr(data,6);
hitung:=strtofloat(edit1.Text);
edit1.Text:=floattostr(hitung);
end

else if midstr(data,5,1)='7' then
begin
edit1.Text:=leftstr(data,7);
hitung:=strtofloat(edit1.Text);
edit1.Text:=floattostr(hitung);
end
end

else if rightstr(data,1)='C' then
begin
if midstr(data,5,1)='9' then
begin
edit17.Text:=leftstr(data,5);
hitung:=strtofloat(edit17.Text);
edit17.Text:=floattostr(hitung);
button9.Click;
end

else if midstr(data,5,1)='8' then
begin
edit17.Text:=leftstr(data,6);
hitung:=strtofloat(edit17.Text);
edit17.Text:=floattostr(hitung);
button9.Click;
end

else if midstr(data,5,1)='7' then
begin
edit17.Text:=leftstr(data,7);
hitung:=strtofloat(edit17.Text);
edit17.Text:=floattostr(hitung);
button9.Click;
end
end;
end;

procedure TForm2.Timer1Timer(Sender: TObject);
begin
Unit3.QuickReport3.QRLabel11.Caption:= formatdatetime('"Pada Tanggal:
"dd/mm/yyyy', now)+char(10)+formatdatetime('" Pukul: "hh:nn:ss',now);
Unit3.QuickReport3.QRLabel11.Refresh;
end;

```

```

procedure TForm2.ComPort1RxChar(Sender: TObject; Count: Integer);
var data:string;
begin
comport1.ReadStr(data,count);
memo1.Text:=memo1.Text+data;
end;

procedure TForm2.Button14Click(Sender: TObject);
begin
panel3.Show;
panel4.Hide;
panel5.Hide;
panel6.Hide;
end;

procedure TForm2.Button12Click(Sender: TObject);
begin
panel3.Show;
panel4.Hide;
panel5.Hide;
panel6.Hide;
end;

procedure TForm2.Button15Click(Sender: TObject);
begin
if button1.Caption='Putuskan' then
begin
comport1.WriteStr('H');
form2.adoquery1.delete;
showmessage('Data Berhasil di Hapus');
end
else if button1.Caption='Koneksikan' then
begin
showmessage('Koneksikan dengan perangkat');
end
end;

procedure TForm2.Button16Click(Sender: TObject);
begin
form2.ADOQuery1.Close;
form2.ADOQuery1.SQL.Clear;
form2.ADOQuery1.SQL.Append('select* from databayi');
form2.ADOQuery1.Open;
end;

procedure TForm2.Button17Click(Sender: TObject);
begin
Unit3.QuickReport3.QRLabel12.Caption:=adoquery1.FieldByName('ID').AsString;
Unit3.QuickReport3.QRLabel13.Caption:=adoquery1.FieldByName('Ruangan').AsString;
Unit3.QuickReport3.QRLabel14.Caption:=adoquery1.FieldByName('Nama Ibu').AsString;

```



```
Unit3.QuickReport3.QRLabel14.Caption:=adoquery1.FieldName('Nama
Ibu').AsString;
Unit3.QuickReport3.QRLabel15.Caption:=adoquery1.FieldName('Jenis
Kelamin Bayi').AsString;
Unit3.QuickReport3.QRLabel16.Caption:=adoquery1.FieldName('Tanggal
Lahir Bayi').AsString;
Unit3.QuickReport3.QRLabel17.Caption:=adoquery1.FieldName('Berat
Bayi').AsString;
Unit3.QuickReport3.QRLabel18.Caption:=adoquery1.FieldName('Panjang
Bayi').AsString;
Unit3.QuickReport3.QRLabel19.Caption:=adoquery1.FieldName('No Rekam
Medis').AsString;
Unit3.QuickReport3.preview;
end;

procedure TForm2.Button6Click(Sender: TObject);
begin
form2.Close;
form1.Close;
comport1.Close;
end;
end.
```

3. Foto Alat



Bagian atas alat



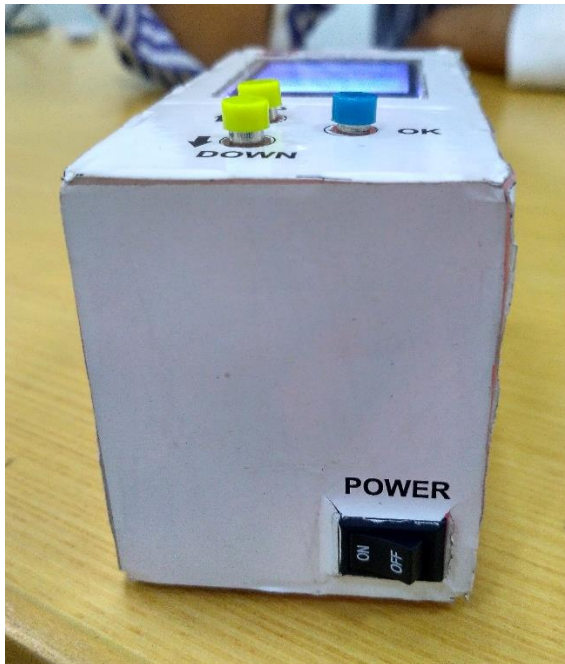
Bagian Kanan Alat



Bagian Samping Kiri Alat



Bagian Depan Alat



Bagian Belakang Alat