Implementasi sistem kolimator rontgen thorax berbasis Arduino Mega 2560 dengan fitur utama: pengendalian motor DC, tampilan TFT LCD, touchscreen, dan validasi proses. Setiap bagian kode telah diberikan komentar ilmiah dan humanis untuk memudahkan pemahaman dan pelaporan.

#include <SPI.h>

#include <Adafruit\_GFX.h>

#include <Adafruit\_ST7796S.h>

#include <Fonts/FreeSansBold18pt7b.h>

#include <Fonts/FreeSansBold9pt7b.h>

#include <Fonts/FreeSansBold12pt7b.h>

#include <Fonts/FreeSans9pt7b.h>

#include <Fonts/TomThumb.h>

#include <Adafruit\_ILI9341.h>

#include <XPT2046\_Touchscreen.h>

PIN CONFIGURATION

Deklarasi pin-pin yang digunakan untuk koneksi ke TFT, touchscreen, motor, dan tombol

#define TFT\_CS    41

#define TFT\_RST   43

#define TFT\_DC    45

#define TOUCH\_CS  47

#define TOUCH\_IRQ 49

#define resolusiX 480

#define resolusiY 320

#define TFTR\_CS   31

#define TFTR\_RST  37

#define TFTR\_DC   35

#define TOUCHR\_CS 39

#define TOUCH\_IRQR 33

#define lampu     2

#define XLEAF1A   10

#define XLEAF1B   9

#define XLEAF2A   8

#define XLEAF2B   7

#define YLEAF1A   6

#define YLEAF1B   5

#define YLEAF2A   4

#define YLEAF2B   3

#define XREAD1    A0

#define XREAD2    A1

#define YREAD1    A2

#define YREAD2    A3

#define toleransi 50

#define OPEN\_X1  900

#define CLOSE\_X1  100

#define OPEN\_X2  900

#define CLOSE\_X2  100

#define OPEN\_Y1  900

#define CLOSE\_Y1  100

#define OPEN\_Y2  900

#define CLOSE\_Y2  100

*// Push Button Pins (Opsional)*

*// Pin untuk tombol fisik sebagai alternatif input selain touchscreen*

#define BTN\_KOLIMASI    22

#define BTN\_ILUMINASI   23

#define BTN\_TERCAPAI    24

#define BTN\_TIDAK       25

#define BTN\_KEMBALI     26

Adafruit\_ST7796S layar(TFT\_CS, TFT\_DC, TFT\_RST);

Adafruit\_ILI9341 remot(TFTR\_CS, TFTR\_DC, TFTR\_RST);

XPT2046\_Touchscreen touch(TOUCH\_CS);

unsigned long lastMillis = 0;

int mode = 0;

int ukuran = 0;

const char\* labelUkuran[] = {"18x24", "24x30", "30x40", "35x43"};

String serialCommand = "";

bool menungguPilihanUkuran = false;

bool sedangKolimasi = false;

bool menungguValidasiIluminasi = false;

bool menungguAksiSetelahKolimasi = false;

bool statusKolimasiBerhasil = false;

void setup() {

Fungsi setup() dijalankan sekali saat perangkat dinyalakan. Melakukan inisialisasi komunikasi serial, layar, touchscreen, dan pin I/O yang diperlukan untuk operasi sistem kolimator.

 Serial.begin(115200);

  layar.init(resolusiY, resolusiX, 0, 0, ST7796S\_BGR);

  layar.setRotation(1);

  layar.invertDisplay(true);

  remot.begin();

  touch.begin();

  touch.setRotation(1);

Konfigurasi pin tombol dengan mode INPUT\_PULLUP untuk mencegah floating dan memastikan sinyal yang stabil

 pinMode(BTN\_KOLIMASI, INPUT\_PULLUP);

  pinMode(BTN\_ILUMINASI, INPUT\_PULLUP);

  pinMode(BTN\_TERCAPAI, INPUT\_PULLUP);

  pinMode(BTN\_TIDAK, INPUT\_PULLUP);

  pinMode(BTN\_KEMBALI, INPUT\_PULLUP);

Inisialisasi semua pin motor sebagai OUTPUT dan matikan lampu untuk kondisi awal yang aman

for (int i = 3; i <= 10; i++) pinMode(i, OUTPUT);

  pinMode(lampu, OUTPUT);

  digitalWrite(lampu, LOW);

Menampilkan layar inisialisasi dan menu utama untuk memberikan informasi identitas proyek

  inisialisasi();

  tampilMenu();

}

void loop() {

Fungsi utama yang berjalan terus-menerus. Berisi logika utama untuk menangani input touchscreen, serial, dan tombol fisik dengan sistem state machine yang terorganisir.

 if (touch.touched()) {

Deteksi sentuhan pada layar touchscreen dengan mapping koordinat yang tepat

TS\_Point p = touch.getPoint();

    int x = map(p.x, 200, 3800, 0, 320);

    int y = map(p.y, 200, 3800, 0, 480);

    if (!menungguPilihanUkuran && !sedangKolimasi && !menungguValidasiIluminasi && !menungguAksiSetelahKolimasi) {

Menu utama: pilihan antara mode kolimasi atau iluminasi berdasarkan area sentuhan

 if (y > 35 && y < 135) {

        kolimasiMode();

      } else if (y > 200 && y < 300) {

        iluminasiMode();

      }

    } else if (menungguPilihanUkuran && !sedangKolimasi) {

Mode pemilihan ukuran film menggunakan layout grid 2x2 dengan deteksi area sentuhan yang presisi

 int boxW = 120, boxH = 80, marginX = 40, marginY = 40, gapX = 40, gapY = 20;

      for (int i = 0; i < 4; i++) {

        int row = i / 2;

        int col = i % 2;

        int x0 = marginX + col \* (boxW + gapX);

        int y0 = marginY + row \* (boxH + gapY);

        if (x > x0 && x < x0 + boxW && y > y0 && y < y0 + boxH) {

          ukuran = i;

          selesaiKolimasi();

        }

      }

Tombol kembali ke menu utama dengan area sentuhan yang jelas

 if (x > 40 && x < 280 && y > 260 && y < 310) {

        tampilMenu();

        menungguPilihanUkuran = false;

      }

    } else if (menungguAksiSetelahKolimasi) {

Setelah kolimasi selesai, hanya tombol kembali yang aktif dengan area sentuhan (60, 260, 200x50)

 if (x > 60 && x < 260 && y > 260 && y < 310) {

Selalu tutup pintu dan matikan lampu sebelum kembali ke pemilihan ukuran untuk keamanan operasi

 digitalWrite(lampu, LOW);

        digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, HIGH);

        digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, HIGH);

        digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, HIGH);

        digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, HIGH);

        delay(2000);

        digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

        digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

        digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

        digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

        kolimasiMode();

        menungguAksiSetelahKolimasi = false;

      }

    } else if (menungguValidasiIluminasi) {

Validasi hasil iluminasi dengan tiga pilihan: tercapai, tidak tercapai, atau kembali ke menu

if (y > 150 && y < 210) { *// Tercapai button area*

        validasiIluminasi(true);

      }

      else if (y > 230 && y < 290) { *// Tidak button area*

        validasiIluminasi(false);

      } else if (y > 310 && y < 370) { *// Kembali button area*

Tutup pintu dan matikan lampu sebelum kembali ke menu utama untuk keamanan

  digitalWrite(lampu, LOW);

        digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, HIGH);

        digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, HIGH);

        digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, HIGH);

        digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, HIGH);

        delay(2000);

        digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

        digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

        digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

        digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

        tampilMenu();

        menungguValidasiIluminasi = false;

      }

    }

  }

 if (Serial.available()) {

Penanganan perintah dari Serial Monitor untuk debugging atau remote control dengan parsing command yang efisien

 char c = Serial.read();

    if (c == '\n') {

      serialCommand.trim();

      if (serialCommand.equalsIgnoreCase("kolimasi")) {

        kolimasiMode();

      } else if (serialCommand.equalsIgnoreCase("iluminasi")) {

        iluminasiMode();

      } else if ((serialCommand == "0" || serialCommand == "1" || serialCommand == "2" || serialCommand == "3") && menungguPilihanUkuran && !sedangKolimasi) {

        ukuran = serialCommand.toInt();

        selesaiKolimasi();

      } else if (serialCommand.equalsIgnoreCase("kembali") && menungguPilihanUkuran && !sedangKolimasi) {

        tampilMenu();

        menungguPilihanUkuran = false;

      } else if (menungguAksiSetelahKolimasi) {

        if (serialCommand.equalsIgnoreCase("kembali")) {

          digitalWrite(lampu, LOW);

          digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, HIGH);

          digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, HIGH);

          digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, HIGH);

          digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, HIGH);

          delay(2000);

          digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

          digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

          digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

          digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

          kolimasiMode();

          menungguAksiSetelahKolimasi = false;

        }

      } else if (menungguValidasiIluminasi) {

        if (serialCommand.equalsIgnoreCase("tercapai")) {

          validasiIluminasi(true);

        } else if (serialCommand.equalsIgnoreCase("tidak")) {

          validasiIluminasi(false);

        } else if (serialCommand.equalsIgnoreCase("kembali")) {

          digitalWrite(lampu, LOW);

          digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, HIGH);

          digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, HIGH);

          digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, HIGH);

          digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, HIGH);

          delay(2000);

          digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

          digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

          digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

          digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

          tampilMenu();

          menungguValidasiIluminasi = false;

        }

      }

      serialCommand = "";

    } else {

      serialCommand += c;

    }

  }

PUSH BUTTON HANDLING

Penanganan tombol fisik sebagai alternatif input selain touchscreen dengan debounce sederhana

 if (!digitalRead(BTN\_KOLIMASI) && !menungguPilihanUkuran && !sedangKolimasi && !menungguValidasiIluminasi && !menungguAksiSetelahKolimasi) {

    delay(200); *// debounce sederhana*

    kolimasiMode();

  }

  if (!digitalRead(BTN\_ILUMINASI) && !menungguPilihanUkuran && !sedangKolimasi && !menungguValidasiIluminasi && !menungguAksiSetelahKolimasi) {

    delay(200);

    iluminasiMode();

  }

  if (!digitalRead(BTN\_TERCAPAI) && menungguValidasiIluminasi) {

    delay(200);

    validasiIluminasi(true);

  }

  if (!digitalRead(BTN\_TIDAK) && menungguValidasiIluminasi) {

    delay(200);

    validasiIluminasi(false);

  }

  if (!digitalRead(BTN\_KEMBALI)) {

    delay(200);

    if (menungguPilihanUkuran && !sedangKolimasi) {

      tampilMenu();

      menungguPilihanUkuran = false;

    }

    else if (menungguAksiSetelahKolimasi) {

      resetPintu();

      kolimasiMode();

      menungguAksiSetelahKolimasi = false;

    }

    else if (menungguValidasiIluminasi) {

      resetPintu();

      tampilMenu();

      menungguValidasiIluminasi = false;

    }

  }

}

void kolimasiMode() {

Fungsi untuk menampilkan dan mengatur mode kolimasi dengan pemilihan ukuran film. Menampilkan grid ukuran film pada layar utama dan remote dengan interface yang user-friendly.

 layar.fillScreen(ST77XX\_WHITE);

  remot.fillScreen(ST77XX\_WHITE);

  layar.setCursor(40, 30);

  remot.setCursor(05, 20);

  layar.setTextColor(ST77XX\_BLACK);

  remot.setTextColor(ST77XX\_BLACK);

  layar.setTextSize(2);

  remot.setTextSize(0);

  layar.print("PILIH UKURAN FILM:");

  remot.setFont(&FreeSans9pt7b);

  remot.setCursor(5, 15); *// Posisi bisa disesuaikan karena font lebih kecil*

  remot.setTextColor(ST77XX\_BLACK);

  remot.print("\n");

  remot.print("PILIH UKURAN FILM:");

  remot.print("\n=====================\n");

  Serial.println("\n=== MODE KOLIMASI ===");

  Serial.println("Pilih Ukuran Film:");

Grid layout 2x2 dengan warna tema gelap yang memberikan kontras yang baik

  const uint16\_t DARK\_YELLOW = 0x9A60; *// Kuning gelap (agak olive)*

    const uint16\_t DARK\_GREEN  = 0x03E0; *// Hijau gelap*

    const uint16\_t MAROON      = 0x8000; *// Merah maroon*

    const uint16\_t NAVY        = 0x000F; *// Biru navy gelap*

  const uint16\_t colors[4] = {DARK\_YELLOW, DARK\_GREEN, MAROON, NAVY};

  int boxW = 200, boxH = 90, marginX = 45, marginY = 40, gapX = 20, gapY = 20;

  int rboxW = 100, rboxH = 40, rmarginX = 10, rmarginY = 70, rgapX = 15, rgapY = 10;

  for (int i = 0; i < 4; i++) {

    int row = i / 2;

    int col = i % 2;

    int x = marginX + col \* (boxW + gapX);

    int y = marginY + row \* (boxH + gapY);

    int rx = rmarginX + col \* (rboxW + rgapX);

    int ry = rmarginY + row \* (rboxH + rgapY);

    layar.fillRoundRect(x, y, boxW, boxH, 10, colors[i]);

    layar.drawRoundRect(x, y, boxW, boxH, 10, ST77XX\_BLACK);

    remot.fillRoundRect(rx, ry, rboxW, rboxH, 5, colors[i]);

    remot.drawRoundRect(rx, ry, rboxW, rboxH, 5, ST77XX\_BLACK);

    layar.setCursor(x + (boxW/2) - (strlen(labelUkuran[i])\*6) - 20, y + boxH/2 + 8);

    remot.setCursor(rx + (rboxW/2) - (strlen(labelUkuran[i])\*3) - 10, ry + rboxH/2 + 4);

    layar.setTextColor(ST77XX\_WHITE);

    remot.setTextColor(ST77XX\_WHITE);

    layar.setTextSize(2);

    remot.setTextSize(1);

    layar.print(labelUkuran[i]);

    remot.print(labelUkuran[i]);

    Serial.print(i); Serial.print(". "); Serial.println(labelUkuran[i]);

  }

Tombol Kembali diposisikan dengan baik dan dibuat dengan desain yang konsisten

 layar.fillRoundRect(150, 250, 240, 60, 10, ST77XX\_BLACK); *// X=80 (geser kanan), height=70*

  layar.drawRoundRect(150, 250, 240, 60, 10, ST77XX\_BLACK);

 layar.setCursor(180, 290); *// Y disesuaikan dari 285 ke 290 agar tetap tengah secara vertikal*

  layar.setTextColor(ST77XX\_WHITE);

  layar.print("KEMBALI");

int geserX = 40;

  int geserY = 30;

  remot.fillRoundRect(10 + geserX, 160 + geserY, 120, 30, 5, ST77XX\_BLACK);

  remot.drawRoundRect(10 + geserX, 160 + geserY, 120, 30, 5, ST77XX\_BLACK);

  remot.setCursor(30 + geserX, 180 + geserY);

  remot.print("KEMBALI");

Serial.println("---------------------");

  Serial.println("Ketik 0-3 di Serial Monitor");

  Serial.println("atau sentuh pilihan di LCD");

  Serial.println("Ketik 'kembali' untuk kembali ke menu utama");

  Serial.println("---------------------");

  menungguPilihanUkuran = true;

  sedangKolimasi = false;

  menungguAksiSetelahKolimasi = false;

}

void resetPintu() {

Fungsi untuk menutup semua pintu kolimator secara otomatis dengan prosedur yang aman dan terkontrol

 digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, HIGH);

  digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, HIGH);

  digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, HIGH);

  digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, HIGH);

  delay(2000);

  digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

  digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

  digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

  digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

}

void selesaiKolimasi() {

Fungsi yang dijalankan setelah ukuran film dipilih. Membuka pintu kolimator, membaca sensor posisi potensiometer, dan menampilkan status hasil kolimasi dengan validasi otomatis.

 layar.fillScreen(0xFFFF);

  remot.fillScreen(0xFFFF);

  layar.setCursor(50, 30);

  remot.setCursor(20, 20);

  layar.setTextColor(0x0000);

  remot.setTextColor(0x0000);

  layar.setTextSize(2);

  remot.setTextSize(1);

  layar.print("Ukuran: "); layar.print(labelUkuran[ukuran]);

  remot.print("Ukuran: "); remot.print(labelUkuran[ukuran]);

  Serial.print("Ukuran yang dipilih: "); Serial.println(labelUkuran[ukuran]);

  layar.setCursor(50, 70);

  remot.setCursor(20, 40);

  layar.print("Membuka pintu...");

  remot.print("Membuka pintu...");

  Serial.println("Membuka pintu kolimator...");

Proses pembukaan pintu kolimator dengan kontrol motor DC yang presisi

 digitalWrite(XLEAF1A, HIGH); digitalWrite(XLEAF1B, LOW);

  digitalWrite(XLEAF2A, HIGH); digitalWrite(XLEAF2B, LOW);

  digitalWrite(YLEAF1A, HIGH); digitalWrite(YLEAF1B, LOW);

  digitalWrite(YLEAF2A, HIGH); digitalWrite(YLEAF2B, LOW);

  delay(2000);

  digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

  digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

  digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

  digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

Pembacaan nilai analog dari potensiometer untuk validasi posisi pintu kolimator

int x1 = analogRead(XREAD1);

  int x2 = analogRead(XREAD2);

  int y1 = analogRead(YREAD1);

  int y2 = analogRead(YREAD2);

Menampilkan nilai pembacaan sensor pada layar dan serial monitor untuk monitoring

  layar.setCursor(50, 100);

  remot.setCursor(20, 60);

  layar.print("X1:"); layar.print(x1);

  remot.print("X1:"); remot.print(x1);

  layar.setCursor(50, 140);

  remot.setCursor(20, 80);

  layar.print("X2:"); layar.print(x2);

  remot.print("X2:"); remot.print(x2);

  layar.setCursor(50, 180);

  remot.setCursor(20, 100);

  layar.print("Y1:"); layar.print(y1);

  remot.print("Y1:"); remot.print(y1);

  layar.setCursor(50, 220);

  remot.setCursor(20, 120);

  layar.print("Y2:"); layar.print(y2);

  remot.print("Y2:"); remot.print(y2);

 Serial.print("X1: "); Serial.println(x1);

  Serial.print("X2: "); Serial.println(x2);

  Serial.print("Y1: "); Serial.println(y1);

  Serial.print("Y2: "); Serial.println(y2);

Validasi otomatis berdasarkan toleransi yang telah ditentukan

bool sesuai = abs(x1 - OPEN\_X1) <= toleransi && abs(x2 - OPEN\_X2) <= toleransi && abs(y1 - OPEN\_Y1) <= toleransi && abs(y2 - OPEN\_Y2) <= toleransi;

  statusKolimasiBerhasil = sesuai;

  layar.setCursor(50, 250);

  remot.setCursor(20, 140);

  layar.print("Status: "); layar.print(sesuai ? "SESUAI" : "TIDAK SESUAI");

  remot.print("Status: "); remot.print(sesuai ? "SESUAI" : "TIDAK SESUAI");

  Serial.print("Status: "); Serial.println(sesuai ? "SESUAI" : "TIDAK SESUAI");

Tombol kembali dengan desain yang konsisten dan ukuran yang memadai

  int btnY = 260;

  int btnW = 250; *// Lebar tombol diperbesar dari 200 ke 250*

  int btnH = 50; *// Tinggi tombol diperbesar dari 50 ke 60*

  int btnX = 130; *// Geser tombol ke kanan (dari 60 ke 100)*

 layar.fillRoundRect(btnX, btnY, btnW, btnH, 10, ST77XX\_BLUE);

  layar.drawRoundRect(btnX, btnY, btnW, btnH, 10, ST77XX\_BLACK);

*// Atur posisi teks di tengah tombol*

  layar.setCursor(btnX + 40, btnY + 35); *// Sesuaikan agar teks berada di tengah secara visual*

  layar.setTextColor(ST77XX\_WHITE);

  layar.print("KEMBALI");

*// Tombol di remot juga bisa disesuaikan (jika ingin disamakan):*

  remot.fillRoundRect(50, 180, 140, 40, 5, ST77XX\_BLUE); *// Geser ke kanan dan perbesar*

  remot.drawRoundRect(50, 180, 140, 40, 5, ST77XX\_BLACK);

  remot.setCursor(80, 200);

  remot.setTextColor(ST77XX\_WHITE);

  remot.print("KEMBALI");

 Serial.println("---------------------");

  Serial.println("Ketik 'kembali' untuk kembali ke pilihan ukuran kolimasi");

  Serial.println("---------------------");

  menungguAksiSetelahKolimasi = true;

  menungguPilihanUkuran = false;

  sedangKolimasi = false;

  menungguValidasiIluminasi = false;

}

void validasiIluminasi(bool *tercapai*) {

Fungsi untuk memvalidasi hasil uji iluminasi berdasarkan input user. Jika tercapai, sistem kembali ke menu utama; jika tidak, pintu dibuka ulang untuk pengujian berulang.

menungguValidasiIluminasi = false;

if (tercapai) {

Jika iluminasi tercapai, matikan lampu dan tutup pintu kemudian kembali ke menu utama

 menungguValidasiIluminasi = false;

  if (tercapai) {

*// Jika tercapai, langsung kembali ke menu utama*

    digitalWrite(lampu, LOW);

*// Tutup pintu*

    digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, HIGH);

    digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, HIGH);

    digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, HIGH);

    digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, HIGH);

    delay(2000);

    digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

    digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

    digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

    digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

    tampilMenu();

  } else {

Jika iluminasi tidak tercapai, tutup pintu sementara kemudian buka kembali untuk pengujian ulang

  digitalWrite(lampu, LOW);

    digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, HIGH);

    digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, HIGH);

    digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, HIGH);

    digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, HIGH);

    delay(2000);

    digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

    digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

    digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

    digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

Buka pintu dan nyalakan lampu kembali untuk pengujian ulang

    digitalWrite(lampu, HIGH);

    digitalWrite(XLEAF1A, HIGH); digitalWrite(XLEAF1B, LOW);

    digitalWrite(XLEAF2A, HIGH); digitalWrite(XLEAF2B, LOW);

    digitalWrite(YLEAF1A, HIGH); digitalWrite(YLEAF1B, LOW);

    digitalWrite(YLEAF2A, HIGH); digitalWrite(YLEAF2B, LOW);

    delay(2000);

    digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

    digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

    digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

    digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

Tetap di halaman uji iluminasi untuk validasi ulang

   menungguValidasiIluminasi = true;

  }

}

void iluminasiMode() {

Fungsi untuk menampilkan mode uji iluminasi pada layar utama dan remote. Menyalakan lampu dengan intensitas 100 lux, membuka pintu, dan menampilkan interface validasi yang user-friendly.

int offsetX = 20; *// geser ke kanan*

  int offsetY = 20; *// geser ke atas*

  layar.fillScreen(ST77XX\_WHITE);

  layar.setTextColor(ST77XX\_BLACK);

Menampilkan judul dengan font yang sesuai dan posisi yang terpusat

layar.setFont(&FreeSansBold12pt7b);

  layar.setTextSize(1);

  int16\_t x1, y1; uint16\_t w1, h1;

  layar.getTextBounds("MODE ILUMINASI", 0, 0, &x1, &y1, &w1, &h1);

  layar.setCursor((resolusiX - w1) / 2 + offsetX, 60 - offsetY);

  layar.print("MODE ILUMINASI");

Menampilkan status lampu dengan intensitas yang jelas

 layar.setFont(&FreeSansBold9pt7b);

  layar.setTextSize(1);

  int16\_t x2, y2; uint16\_t w2, h2;

  layar.getTextBounds("Lampu: ON (100 lux)", 0, 0, &x2, &y2, &w2, &h2);

  layar.setCursor((resolusiX - w2) / 2 + offsetX, 110 - offsetY);

  layar.print("Lampu: ON (100 lux)");

Menampilkan pertanyaan validasi

  int16\_t x3, y3; uint16\_t w3, h3;

  layar.getTextBounds("Cahaya tercapai?", 0, 0, &x3, &y3, &w3, &h3);

  layar.setCursor((resolusiX - w3) / 2 + offsetX, 150 - offsetY);

  layar.print("Cahaya tercapai?");

Konfigurasi tombol dengan dimensi yang optimal

  int btnW = 240, btnH = 38;

  int btnX = (resolusiX - btnW) / 2 + offsetX;

  int btnY1 = 180 - offsetY;

  int btnY2 = btnY1 + btnH + 12;

  int btnY3 = btnY2 + btnH + 12;

Tombol TERCAPAI dengan warna hijau

 layar.fillRoundRect(btnX, btnY1, btnW, btnH, 10, ST77XX\_GREEN);

  layar.drawRoundRect(btnX, btnY1, btnW, btnH, 10, ST77XX\_BLACK);

  layar.setFont();

  layar.setTextSize(2);

  const char\* txtTercapai = "TERCAPAI";

  int16\_t tx1, ty1; uint16\_t tw1, th1;

  layar.getTextBounds(txtTercapai, 0, 0, &tx1, &ty1, &tw1, &th1);

  layar.setCursor(btnX + (btnW - tw1) / 2, btnY1 + (btnH + th1) / 2 - 15);

  layar.setTextColor(ST77XX\_WHITE);

  layar.print(txtTercapai);

Tombol TIDAK dengan warna merah

  layar.fillRoundRect(btnX, btnY2, btnW, btnH, 10, ST77XX\_RED);

  layar.drawRoundRect(btnX, btnY2, btnW, btnH, 10, ST77XX\_BLACK);

  const char\* txtTidak = "TIDAK";

  int16\_t tx2, ty2; uint16\_t tw2, th2;

  layar.getTextBounds(txtTidak, 0, 0, &tx2, &ty2, &tw2, &th2);

  layar.setCursor(btnX + (btnW - tw2) / 2, btnY2 + (btnH + th2) / 2 - 15);

  layar.setTextColor(ST77XX\_WHITE);

  layar.print(txtTidak);

Tombol KEMBALI dengan warna biru

 layar.fillRoundRect(btnX, btnY3, btnW, btnH, 10, ST77XX\_BLUE);

  layar.drawRoundRect(btnX, btnY3, btnW, btnH, 10, ST77XX\_BLACK);

  const char\* txtKembali = "KEMBALI";

  int16\_t tx3, ty3; uint16\_t tw3, th3;

  layar.getTextBounds(txtKembali, 0, 0, &tx3, &ty3, &tw3, &th3);

  layar.setCursor(btnX + (btnW - tw3) / 2, btnY3 + (btnH + th3) / 2 - 15);

  layar.setTextColor(ST77XX\_WHITE);

  layar.print(txtKembali);

Interface untuk layar remote dengan layout yang responsif

  remot.fillScreen(ST77XX\_WHITE);

*// Judul, status, pertanyaan di tengah layar remote*

  int rWidth = 240; *// asumsikan lebar remote*

  int rHeight = 320; *// asumsikan tinggi remote*

  int rBtnW = 160, rBtnH = 50;

  int rBtnX = (rWidth - rBtnW) / 2;

  int rBtnY1 = 120;

  int rBtnY2 = rBtnY1 + rBtnH + 12;

  int rBtnY3 = rBtnY2 + rBtnH + 12;

remot.setTextColor(ST77XX\_BLACK);

  remot.setTextSize(1);

  remot.setFont(&FreeSansBold12pt7b);

  int16\_t rx, ry; uint16\_t rw, rh;

  remot.getTextBounds("MODE ILUMINASI", 0, 0, &rx, &ry, &rw, &rh);

  remot.setCursor((rWidth - rw) / 2, 30 + rh/2);

  remot.print("MODE ILUMINASI");

  remot.setFont(&FreeSansBold9pt7b);

  remot.getTextBounds("Lampu: ON (100 lux)", 0, 0, &rx, &ry, &rw, &rh);

  remot.setCursor((rWidth - rw) / 2, 55 + rh/2);

  remot.print("Lampu: ON (100 lux)");

  remot.getTextBounds("Cahaya tercapai?", 0, 0, &rx, &ry, &rw, &rh);

  remot.setCursor((rWidth - rw) / 2, 80 + rh/2);

  remot.print("Cahaya tercapai?");

  remot.setFont();

  remot.setTextSize(2);

Implementasi tombol pada layar remote dengan konsistensi desain

*// Tombol TERCAPAI*

  remot.fillRoundRect(rBtnX, rBtnY1, rBtnW, rBtnH, 10, ST77XX\_GREEN);

  remot.drawRoundRect(rBtnX, rBtnY1, rBtnW, rBtnH, 10, ST77XX\_BLACK);

  remot.getTextBounds(txtTercapai, 0, 0, &rx, &ry, &rw, &rh);

  remot.setCursor(rBtnX + (rBtnW - rw) / 2, rBtnY1 + (rBtnH + rh) / 2 - 15);

  remot.setTextColor(ST77XX\_WHITE);

  remot.print(txtTercapai);

*// Tombol TIDAK*

  remot.fillRoundRect(rBtnX, rBtnY2, rBtnW, rBtnH, 10, ST77XX\_RED);

  remot.drawRoundRect(rBtnX, rBtnY2, rBtnW, rBtnH, 10, ST77XX\_BLACK);

  remot.getTextBounds(txtTidak, 0, 0, &rx, &ry, &rw, &rh);

  remot.setCursor(rBtnX + (rBtnW - rw) / 2, rBtnY2 + (rBtnH + rh) / 2 - 15);

  remot.setTextColor(ST77XX\_WHITE);

  remot.print(txtTidak);

*// Tombol KEMBALI*

  remot.fillRoundRect(rBtnX, rBtnY3, rBtnW, rBtnH, 10, ST77XX\_BLUE);

  remot.drawRoundRect(rBtnX, rBtnY3, rBtnW, rBtnH, 10, ST77XX\_BLACK);

  remot.getTextBounds(txtKembali, 0, 0, &rx, &ry, &rw, &rh);

  remot.setCursor(rBtnX + (rBtnW - rw) / 2, rBtnY3 + (rBtnH + rh) / 2 - 15);

  remot.setTextColor(ST77XX\_WHITE);

  remot.print(txtKembali);

Aktivasi sistem iluminasi dengan menyalakan lampu dan membuka pintu

 digitalWrite(lampu, HIGH);

  digitalWrite(XLEAF1A, HIGH); digitalWrite(XLEAF1B, LOW);

  digitalWrite(XLEAF2A, HIGH); digitalWrite(XLEAF2B, LOW);

  digitalWrite(YLEAF1A, HIGH); digitalWrite(YLEAF1B, LOW);

  digitalWrite(YLEAF2A, HIGH); digitalWrite(YLEAF2B, LOW);

  delay(2000);

  digitalWrite(XLEAF1A, LOW); digitalWrite(XLEAF1B, LOW);

  digitalWrite(XLEAF2A, LOW); digitalWrite(XLEAF2B, LOW);

  digitalWrite(YLEAF1A, LOW); digitalWrite(YLEAF1B, LOW);

  digitalWrite(YLEAF2A, LOW); digitalWrite(YLEAF2B, LOW);

  Serial.println("\n=== MODE ILUMINASI ===");

  Serial.println("Lampu ON (100 lux)");

  Serial.println("Status Cahaya:");

  Serial.println("1. Ketik 'tercapai' jika cahaya sesuai");

  Serial.println("2. Ketik 'tidak' jika cahaya tidak sesuai");

  Serial.println("3. Ketik 'kembali' untuk kembali ke menu utama");

  Serial.println("---------------------");

  menungguValidasiIluminasi = true;

}

void inisialisasi() {

Fungsi untuk menampilkan layar pembuka dengan informasi identitas proyek, mahasiswa, dan dosen pembimbing. Berfungsi sebagai splash screen yang informatif dan profesional.

layar.fillScreen(0xFFFFFF);

  remot.fillScreen(0xFFFFFF);

  layar.setFont(&FreeSansBold12pt7b);

  remot.setFont(&FreeSansBold12pt7b);

  layar.setTextColor(0x000000);

  remot.setTextColor(0x000000);

  layar.setTextSize(1);

  remot.setTextSize(1);

LAYAR 1: Judul Proyek dengan informasi lengkap tentang sistem yang dikembangkan

  layar.setCursor(30, 130); layar.print("PROTOTIPE KOLIMATOR RONTGEN");

  remot.setCursor(50, 70); remot.print("PROTOTIPE");

  remot.setCursor(47, 100); remot.print("KOLIMATOR");

  remot.setCursor(60, 130); remot.print("RONTGEN");

  layar.setCursor(15, 160); layar.print("THORAX MENGGUNAKAN MOTOR DC");

  remot.setCursor(65, 160); remot.print("THORAX");

  remot.setCursor(20, 190); remot.print("MENGGUNAKAN");

  remot.setCursor(50, 220); remot.print("MOTOR DC");

  layar.setCursor(145, 190); layar.print("BERBASIS MEGA2560");

  remot.setCursor(18, 250); remot.print("BERBASIS MEGA2560");

  delay(1500);

LAYAR 2: Identitas Mahasiswa dengan NIM lengkap

  layar.fillScreen(0xFFFFFF);

  remot.fillScreen(0xFFFFFF);

  layar.setCursor(80, 140); layar.print("MUHAMMAD ARIEF ALHADI");

  remot.setCursor(45, 80); remot.print("MUHAMMAD");

  remot.setCursor(35, 110); remot.print("ARIEF ALHADI");

  layar.setCursor(130, 200); layar.print("P2.20.301.21.032");

  remot.setCursor(25, 190); remot.print("P2.20.301.21.032");

  delay(1500);

LAYAR 3: Dosen Pembimbing I dengan gelar dan NIP lengkap

  layar.fillScreen(0xFFFFFF);

  remot.fillScreen(0xFFFFFF);

  layar.setCursor(80, 130); layar.print("VITA NURDINAWATI, ST, MT");

  remot.setCursor(95, 80); remot.print("VITA");

  remot.setCursor(35, 110); remot.print("NURDINAWATI,");

  remot.setCursor(85, 140); remot.print("ST, MT");

  layar.setCursor(85, 200); layar.print("NIP. 198309032018012001");

  remot.setCursor(100, 200); remot.print("NIP.");

  remot.setCursor(0, 230); remot.print("198309032018012001");

  delay(1500);

LAYAR 4: Dosen Pembimbing II dengan gelar dan NIP lengkap

 layar.fillScreen(0xFFFFFF);

  remot.fillScreen(0xFFFFFF);

  layar.setCursor(90, 130); layar.print("WINDA WIRASA, ST, MT");

  remot.setCursor(85, 80); remot.print("WINDA");

  remot.setCursor(35, 110); remot.print("WIRASA,");

  remot.setCursor(90, 140); remot.print("ST, MT");

  layar.setCursor(85, 200); layar.print("NIP. 196603271990031004");

  remot.setCursor(90, 200); remot.print("NIP.");

  remot.setCursor(0, 230); remot.print("196603271990031004");

  delay(1500);

}

void tampilMenu() {

Fungsi untuk menampilkan menu utama pada layar utama dan remote dengan dua pilihan utama: mode kolimasi dan mode uji iluminasi. Interface dirancang dengan layout yang jelas dan mudah digunakan.

 layar.fillScreen(0xFFFFFF);

  remot.fillScreen(0xFFFFFF);

  layar.setFont();

  remot.setFont();

  layar.setTextColor(0x000000);

  remot.setTextColor(0x000000);

  layar.setTextSize(2);

  remot.setTextSize(2);

Tombol menu kolimasi dengan border dan styling yang konsisten

 layar.drawRect(90, 35, 300, 100, 0x000000);

  remot.drawRect(20, 45, 200, 80, 0x000000);

  layar.setFont(&FreeSansBold9pt7b);

  remot.setFont(&FreeSansBold12pt7b);

  layar.setCursor(160, 100);

  remot.setCursor(60, 95);

   layar.setTextSize(2);

  remot.setTextSize(1);

  layar.print("KOLIMASI");

  remot.print("KOLIMASI");

Tombol menu uji iluminasi dengan desain yang seragam

  layar.setFont(&FreeSansBold9pt7b);

  remot.setFont(&FreeSansBold12pt7b);

  layar.drawRect(90, 200, 300, 100, 0x000000);

  remot.drawRect(20, 200, 200, 80, 0x000000);

  layar.setCursor(215, 240);

  remot.setCursor(100, 230);

  layar.setTextSize(2);

  remot.setTextSize(1);

  layar.print("UJI");

  remot.print("UJI");

  layar.setCursor(145, 280);

  remot.setCursor(55, 260);

  layar.print("ILUMINASI");

  remot.print("ILUMINASI");

Reset semua flag state untuk memastikan kondisi awal yang bersih

  menungguPilihanUkuran = false;

  sedangKolimasi = false;

  menungguAksiSetelahKolimasi = false;

  menungguValidasiIluminasi = false;

}