

LAMPIRAN

```
// Komunikasi Arduino ke NodeMCU menggunakan SoftwareSerial
#include <SoftwareSerial.h>
SoftwareSerial mySerial(6, 7); // Pin 6 sebagai RX dan Pin 7 sebagai TX

// Library untuk Keypad
#include <Keypad.h>
const byte ROWS = 4; // Jumlah baris pada keypad
const byte COLS = 4; // Jumlah kolom pada keypad
char hexaKeys[ROWS][COLS] = { // Tata letak tombol pada keypad
  { '1', '2', '3', 'A' },
  { '4', '5', '6', 'B' },
  { '7', '8', '9', 'C' },
  { '*', '0', '#', 'D' }
};
byte rowPins[ROWS] = { 5, 4, 3, 2 }; // Pin untuk baris keypad
byte colPins[COLS] = { A3, A2, A1, A0 }; // Pin untuk kolom keypad
Keypad customKeypad = Keypad(makeKeymap(hexaKeys), rowPins, colPins,
ROWS, COLS);
char customKey; // Variabel untuk menyimpan tombol yang ditekan

// Sensor Ultrasonik
const int trigPin = 9; // Pin trigger sensor ultrasonik
const int echoPin = 10; // Pin echo sensor ultrasonik
int sensorUltra; // Variabel untuk menyimpan jarak yang diukur

// LCD 16x2
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2); // Inisialisasi LCD I2C dengan alamat 0x27

// Modul DFPlayer Mini
#include "DFRobotDFPlayerMini.h" // Library untuk DFPlayer Mini
SoftwareSerial softwareSerial(PIN_MP3_RX, PIN_MP3_TX); // Serial komunikasi
untuk DFPlayer Mini
DFRobotDFPlayerMini player; // Objek DFPlayer Mini

// Modul Relay
#define selenoid A6 // Pin untuk relay solenoid

// ESP32CAM
#define kamera 8 // Pin untuk kontrol kamera ESP32CAM
```

```

// Variabel Tambahan
bool datang = false; // Status kedatangan paket
String phoneNumber = ""; // Variabel untuk menyimpan nomor telepon
String specialCode = "221022"; // Kode khusus yang diterima

void setup() {
  Serial.begin(9600); // Mulai komunikasi serial dengan baud rate 9600
  mySerial.begin(9600); // Mulai komunikasi serial dengan NodeMCU
  lcd.begin(); // Inisialisasi LCD
  lcd.setCursor(0, 0);
  lcd.print("Kotak Penerima");
  lcd.setCursor(0, 1);
  lcd.print("Paket Pakaian");
  delay(1000); // Tunda 1 detik untuk menampilkan pesan di LCD

  pinMode(kamera, OUTPUT); // Atur pin kamera sebagai OUTPUT
  digitalWrite(kamera, 0); // Matikan kamera
  pinMode(solenoid, OUTPUT); // Atur pin relay sebagai OUTPUT
  digitalWrite(solenoid, 0); // Matikan relay

  pinMode(trigPin, OUTPUT); // Atur pin trigger sensor ultrasonik sebagai OUTPUT
  pinMode(echoPin, INPUT); // Atur pin echo sensor ultrasonik sebagai INPUT

  softwareSerial.begin(9600); // Mulai komunikasi serial dengan DFPlayer Mini
  player.begin(softwareSerial); // Inisialisasi DFPlayer Mini
  player.volume(30); // Set volume maksimum (0 hingga 30)
  delay(1); // Tunda sejenak
}

void loop() {
  sensorUltra = readDistance(trigPin, echoPin); // Membaca jarak dari sensor
  ultrasonik

  if (sensorUltra <= 50) { // Jika jarak kurang dari atau sama dengan 50 cm
    digitalWrite(kamera, 1); // Nyalakan kamera
    datang = true; // Tandai bahwa paket datang
  } else {
    digitalWrite(kamera, 0); // Matikan kamera
    datang = false; // Tandai bahwa paket tidak datang
  }

  if (datang) { // Jika paket datang
    lcd.setCursor(0, 0);
    lcd.print("Silahkan");
  }
}

```

```

    lcd.setCursor(0, 1);
    lcd.print("Masukkan No Hp");
    phoneNumber = getPhoneNumber(); // Ambil nomor telepon dari keypad
    if (phoneNumber.length() >= 10 || phoneNumber == specialCode) { // Jika nomor
valid
    masukkanPaket(); // Masukkan paket
    } else {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Gagal");
    lcd.setCursor(0, 1);
    lcd.print("Masukkan No Hp");
    delay(2000); // Tunda 2 detik
    }
    } else {
    lcd.setCursor(0, 0);
    lcd.print("Kotak Penerima");
    lcd.setCursor(0, 1);
    lcd.print("Paket Pakaian");
    }

    delay(200); // Tunda 200 milidetik
    lcd.clear(); // Bersihkan LCD
}

void masukkanPaket() { // Fungsi untuk memasukkan paket
    digitalWrite(solenoid, 1); // Nyalakan relay solenoid
    Serial.print("No HP: ");
    Serial.println(phoneNumber); // Tampilkan nomor telepon

    // Kirim Data ke NodeMCU
    mySerial.println(phoneNumber); // Kirim nomor telepon ke NodeMCU

    player.play(1); // Putar file audio pertama pada DFPlayer Mini
    delay(4000); // Tunda 4 detik
    digitalWrite(solenoid, 0); // Matikan relay solenoid
}

String getPhoneNumber() { // Fungsi untuk mendapatkan nomor telepon dari keypad
    String number = "";
    char key;
    unsigned long startTime = millis(); // Waktu mulai pengambilan nomor
    lcd.clear();
    lcd.setCursor(0, 0);

```

```

lcd.print("Masukkan No Hp:");

while (true) { // Loop untuk mendapatkan input dari keypad
  key = customKeypad.getKey(); // Ambil tombol yang ditekan
  if (key) {
    startTime = millis(); // Reset timer pada tombol ditekan
    if (key >= '0' && key <= '9') { // Jika tombol angka
      number += key; // Tambah angka ke nomor
      lcd.setCursor(number.length() - 1, 1);
      lcd.print(key);
    } else if (key == '*') { // Jika tombol * (hapus karakter terakhir)
      if (number.length() > 0) {
        number.remove(number.length() - 1);
        lcd.setCursor(number.length(), 1);
        lcd.print(' ');
        lcd.setCursor(number.length(), 1);
      }
    } else if (key == '#') { // Jika tombol # (validasi nomor)
      if (number.length() >= 10 || number == specialCode) {
        break; // Keluar dari loop jika nomor valid
      } else {
        lcd.setCursor(0, 0);
        lcd.print("Gagal: 10 digit");
        delay(2000); // Tunda 2 detik
        lcd.setCursor(0, 0);
        lcd.print("Masukkan No Hp:");
        lcd.setCursor(0, 1);
        lcd.print(number);
      }
    }
  }
}

if (millis() - startTime > 10000) { // Cek timeout (10 detik)
  return ""; // Kembalikan string kosong jika timeout
}

return number; // Kembalikan nomor telepon yang dimasukkan
}

int readDistance(int triggerPin, int echoPin) { // Fungsi untuk membaca jarak dari
sensor ultrasonik
  digitalWrite(triggerPin, LOW);
  delayMicroseconds(2);
  digitalWrite(triggerPin, HIGH);

```

```
delayMicroseconds(10);  
digitalWrite(triggerPin, LOW);  
unsigned long duration = pulseIn(echoPin, HIGH);  
float distance = duration * 0.034 / 2; // Hitung jarak dalam cm  
return distance; // Kembalikan jarak  
}
```