

LAMPIRAN

Lampiran 1. Program Kontroller Keseluruhan

```
#include <SPI.h>
#include <SD.h> //Konfigurasi Untuk MICROSD CARD
#include <Wire.h> //Konfigurasi Untuk I2C
#include <RTClib.h> //Konfigurasi Untuk RTC
RTC_DS1307 rtc;

#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
#include <SD.h>
const int chipSelect = 10;
#include <stdint.h>
#include <Adafruit_MLX90614.h>
#include <Servo.h> //Konfigurasi Untuk Library Motor Servo
Servo servo;

#define trigPin 8
#define echoPin 9
long duration, distance;

const int pinBuzzer = 2;
const int ukursuhu = 4;
const int kirimdata = 5;
const int suhutinggi = 6;
const int sdggl = 7;

const int relay1 = A1;
const int relay = A0;
```

```

Adafruit_MLX90614 mlx = Adafruit_MLX90614()
void (*reset) (void) =0;
void setup() {

while (!Serial)
{
; // MENUNGGU PORT SERIAL TERHUBUNG. Diperlukan hanya
untuk port USB asli
}

/**PROGRAM JIKA PADA POSISI AWAL, MICROSD TIDAK
ADA/RUSAK/TIDAK TERBACA***/
if (!SD.begin(chipSelect)) {
digitalWrite(sdggl, HIGH);
digitalWrite(pinBuzzer, HIGH);
delay(1000);
digitalWrite(sdggl, LOW);
digitalWrite(pinBuzzer, LOW);
delay(10);
lcd.setCursor(0,0);
lcd.print(" MICROSD CARD ");
lcd.setCursor(0,1);
lcd.print(" GAGAL TERBACA ");
delay(1000);

while (1); //PEMBACAAN HANYA 1 KALI, JIKA TIDAK TERBACA,
MAKA TIDAK MELANJUTKAN PROGRAM
}

pinMode(1, OUTPUT);
//Serial.println("Adafruit MLX90614 test");
mlx.begin();
lcd.begin();

```

```

rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));

servo.attach(3);      // motor servo dihubungkan ke pin digital 3
servo.write(10);     // set motor servo ke posisi 0 derajat
delay(1000);        // tunggu hingga posisi 0 derajat
}

/****PROGRAM JIKA SUHU KURANG DARI 38 DERAJAT****/
if (mlx.readObjectTempC() <38)
{
  led_ukursuhu();

/****MENGIRIM DATA KE MICROSD CARD****/
if (dataFile)
{
  dataFile.println("");
  dataFile.print(now.day(), DEC); dataFile.print("-");
dataFile.print(now.month(), DEC); dataFile.print("-");
dataFile.print(now.year(), DEC);
  dataFile.print(" | ");
  dataFile.print(now.hour(), DEC); dataFile.print(":");
dataFile.print(now.minute(), DEC); dataFile.print(":");
dataFile.print(now.second(), DEC);

  dataFile.print(" | ");
  dataFile.print(mlx.readObjectTempC());
  dataFile.print(" | ");
  dataFile.print(mlx.readAmbientTempC());
  dataFile.print(" | SUHU AMAN");
  dataFile.close();

  led_simpan();
}

```

```

else {
    gagal_tersimpan();
    resett();
}
}

```

```

/**PROGRAM POSISI SENSOR HCSR-04 TIDAK MENDETEKSI
APAPUN***/

```

```

else
{
    lcd_awal();
    resett();
}
}

```

```

/**PROGRAM VOID***/

```

```

/**INFORMASI AWAL LCD***/

```

```

void lcd_awal()
{
    lcd.setCursor(0,0);
    lcd.print(" SILAHKAN ");
    lcd.setCursor(0,1);
    lcd.print(" CEK SUHU ANDA ");
    delay(1000);

    lcd.setCursor(0,0);
    lcd.print(" BERDIRI ");
    lcd.setCursor(0,1);
    lcd.print("DI DEPAN SENSOR ");
}

```



```
delay(1000);  
servo.write(10);  
delay(1000);  
}
```

```
/**INDIKATOR PENGUKURAN SUHU**/  
void led_ukursuhu()
```

```
{  
  digitalWrite(ukursuhu, HIGH);  
  digitalWrite(pinBuzzer, HIGH);  
  delay(400);  
  digitalWrite(ukursuhu, LOW);  
  digitalWrite(pinBuzzer, LOW);  
  delay(10);  
}
```

```
/**INDIKATOR BAHWA SUHU TINGGI**/  
void suhu_tinggi()
```

```
{  
  digitalWrite(pinBuzzer, HIGH);  
  digitalWrite(suhutinggi, HIGH);  
  delay(400);  
  digitalWrite(pinBuzzer, LOW);  
  digitalWrite(suhutinggi, LOW);  
  delay(200);  
  digitalWrite(pinBuzzer, HIGH);  
  digitalWrite(suhutinggi, HIGH);  
  delay(400);  
  digitalWrite(pinBuzzer, LOW);  
  digitalWrite(suhutinggi, LOW);  
  delay(200);  
  digitalWrite(pinBuzzer, HIGH);
```



```

digitalWrite(suhutinggi, HIGH);
delay(400);
digitalWrite(pinBuzzer, LOW);
digitalWrite(suhutinggi, LOW);
delay(200);
}

```

```

/****INFORMASI DATA GAGAL TERKIRIM****/

```

```

void gagal()

```

```

{
  lcd.setCursor(0,0);
  lcd.print("  DATA  ");
  lcd.setCursor(0,1);
  lcd.print(" GAGAL TERKIRIM ");
  delay(1000);
}

```

```

/****INFORMASI LEPAS MASKER****/

```

```

void masker()

```

```

{
  lcd.setCursor(0,0);
  lcd.print(" SILAHKAN LEPAS ");
  lcd.setCursor(0,1);
  lcd.print(" MASKER ANDA ");
  delay(1000);
}

```

```

/****MENGAKTIFKAN KAMERA****/

```

```

void kamera()

```

```

{
  digitalWrite(relay1, LOW);
  digitalWrite(relay, LOW);
}

```



```

delay(200);
digitalWrite(relay1, HIGH);
delay(1);
digitalWrite(relay, LOW);
delay(5000);
digitalWrite(relay, HIGH);
digitalWrite(relay1, HIGH);
delay(1000);
}

```

```

/**INDIKATOR DATA TERSIMPAN***/

```

```

void led_simpan()
{
digitalWrite(kirimdata, HIGH);
digitalWrite(pinBuzzer, HIGH);

delay(300);
digitalWrite(kirimdata, LOW);
digitalWrite(pinBuzzer, LOW);
delay(10);
}

```

```

/**JIKA GAGAL MENGIRIM KE SDCARD***/

```

```

void gagal_tersimpan()
{
digitalWrite(sdggl, HIGH);

delay(500);
digitalWrite(sdggl, LOW);
delay(10);
}

```



Lampiran 2. Program Kamera

```
#include "esp_camera.h"
#include "Arduino.h"
#include "FS.h"
#include "SD_MMC.h"
#include "soc/soc.h"
#include "soc/rtc_cntl_reg.h"
#include "driver/rtc_io.h"
#include <EEPROM.h>

#define EEPROM_SIZE 1

RTC_DATA_ATTR int bootCount = 0;

// Pin untuk CAMERA_MODEL_AI_THINKER
#define PWDN_GPIO_NUM 32
#define RESET_GPIO_NUM -1
#define XCLK_GPIO_NUM 0
#define SIOD_GPIO_NUM 26
#define SIOC_GPIO_NUM 27
#define Y9_GPIO_NUM 35
#define Y8_GPIO_NUM 34
#define Y7_GPIO_NUM 39
#define Y6_GPIO_NUM 36
#define Y5_GPIO_NUM 21
#define Y4_GPIO_NUM 19
#define Y3_GPIO_NUM 18
#define Y2_GPIO_NUM 5
#define VSYNC_GPIO_NUM 25
#define HREF_GPIO_NUM 23
#define PCLK_GPIO_NUM 22
```



```

int pictureNumber = 0;

void setup() {
    WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);
    Serial.begin(115200);

    Serial.setDebugOutput(true);

    camera_config_t config;
    config.ledc_channel = LEDC_CHANNEL_0;
    config.ledc_timer = LEDC_TIMER_0;
    config.pin_d0 = Y2_GPIO_NUM;
    config.pin_d1 = Y3_GPIO_NUM;
    config.pin_d2 = Y4_GPIO_NUM;
    config.pin_d3 = Y5_GPIO_NUM;
    config.pin_d4 = Y6_GPIO_NUM;
    config.pin_d5 = Y7_GPIO_NUM;
    config.pin_d6 = Y8_GPIO_NUM;
    config.pin_d7 = Y9_GPIO_NUM;
    config.pin_xclk = XCLK_GPIO_NUM;
    config.pin_pclk = PCLK_GPIO_NUM;
    config.pin_vsync = VSYNC_GPIO_NUM;
    config.pin_href = HREF_GPIO_NUM;
    config.pin_sscb_sda = SIOD_GPIO_NUM;
    config.pin_sscb_scl = SIOC_GPIO_NUM;
    config.pin_pwdn = PWDN_GPIO_NUM;
    config.pin_reset = RESET_GPIO_NUM;
    config.xclk_freq_hz = 20000000;
    config.pixel_format = PIXFORMAT_JPEG;

    pinMode(4, INPUT);

```

```
digitalWrite(4, LOW);  
rtc_gpio_hold_dis(GPIO_NUM_4);  
  
if(psramFound()){  
    config.frame_size = FRAMESIZE_UXGA;  
    config.jpeg_quality = 10;  
    config.fb_count = 2;  
} else {  
    config.frame_size = FRAMESIZE_SVGA;  
    config.jpeg_quality = 12;  
    config.fb_count = 1;  
}
```

