LAMPIRAN A

Progam Antrian Suhu

```
#include <WiFi.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <DFPlayer_Mini_Mp3.h>
#include <SoftwareSerial.h>

#define      LED0      33
#define      call     18
#define      recal     4
#define      prefius     19

#define belas              12 //belas.mp3
#define puluh              13 //puluh.mp3
#define seratus            14 //seratus.mp3
#define ratus              15 //ratus.mp3
#define seribu             16 //seribu.mp3
#define ribu               17 //ribu.mp3
#define koma               18 //Koma.mp3
#define antrianNomor       101

int          ButtonState;
int          LastButtonState   = LOW;
int          LastDebounceTime  = 0;
int          DebounceDelay     = 50;
const String  ClientType       = "Led P10";
int          LEDState          = LOW;
unsigned long CurrMillis       = 0;
unsigned long PrevMillis       = 0;
unsigned long Interval         = 1000;
char*        ESPssid;
char*        ESPpassword;
int          ESPServerPort   = 9001;
long count, proveus;

LiquidCrystal_I2C  lcd(0x27,  16,  2);
IPAddress      ESPServer(192,  168,  4, 1);
WiFiClientESPClient;
```

```
SoftwareSerial dfplayer;

void setup()
{
 Serial.begin(115200);
 dfplayer.begin(9600, SWSERIAL_8N1, 16, 17, false);
 mp3_set_serial (dfplayer);
 mp3_set_volume(30);
 lcd.init();
 lcd.backlight();
 pinMode(LED0, OUTPUT);
 pinMode(call, INPUT_PULLUP);
 pinMode(recal, INPUT_PULLUP);
 pinMode(prefius, INPUT_PULLUP);
 digitalWrite(LED0, !LOW);
 Serial.println("\nI/O Pins Modes Set .... Done");

 lcd.clear();
 lcd.setCursor(0, 0);
 lcd.print("Connecting to...");
 lcd.setCursor(0, 1);
 lcd.print("Server");
 WiFi.mode(WIFI_STA);
 WiFi.begin("Mesin Antrian", "mesinantrian");
 while (WiFi.status() == WL_CONNECTED)
 {
  WiFi.disconnect();
  WiFi.mode(WIFI_OFF);
  lcd.setCursor(15, 1);
  lcd.print(">");
  delay(500);
  lcd.setCursor(15, 1);
  lcd.print(" ");
  delay(500);
  lcd.setCursor(15, 1);
  lcd.print(">");
 }

 WiFi.mode(WIFI_STA);
 WiFi.begin("Mesin Antrian", "mesinantrian");
 CheckWiFiConnectivity();
```

```
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Connected.");
  delay(2000);
  lcd.clear();

  Serial.println("!---  Connecting To " + WiFi.SSID()  + " ---!");
  digitalWrite(LED0,  !HIGH);
  Serial.println("!--  Client Device Connected --!");

  Serial.println("Connected To      : " + String(WiFi.SSID()));
  Serial.println("Signal  Strenght  : " + String(WiFi.RSSI())  + " dBm");
  Serial.print  ("Server IP Address : ");
  Serial.println(ESPServer);
  Serial.print  ("Server Port Num   : ");
  Serial.println(ESPServerPort);
  Serial.print  ("Device MC Address : ");
  Serial.println(String(WiFi.macAddress()));
  Serial.print  ("Device IP Address : ");
  Serial.println(WiFi.localIP());

  ESPRequest();
}

void loop()
{
  CheckWiFiConnectivity();
  ReadButton();
}

void ReadButton()
{
  int caller = digitalRead(call);
  int back = digitalRead(prefius);
  int callback = digitalRead(recal);
  Serial.println(String(caller) + String("  ") + String(back) + String("  ") +
String(callback));
  if ((millis()  - PrevMillis)  > 2000)  {
    lcd.clear();
    PrevMillis  = millis();
  }
  lcd.setCursor(0, 0);
```

```
lcd.print("antrian saat ini");
lcd.setCursor(7, 1);
lcd.print(count);

if (caller == 0)
{
  count = count + 1;
  if (count > 200) {
    count = 0;
  }
  else if (count < 0) {
    count = 200;
  }
  delay(500);
  Serial.println    (count);
  ESPClient.println(count);
  ESPClient.flush();
  mp3_play (antrianNomor);
  delay(2000);
  suaraBilangan(count);
  delay(1000);
}
else if (back == 0) {
  count = count - 1;
  if (count > 200) {
    count = 0;
  }
  else if (count < 0) {
    count = 200;
  }
  delay(500);
  Serial.println    (count);
  ESPClient.println(count);
  ESPClient.flush();
  mp3_play (antrianNomor);
  delay(2000);
  suaraBilangan(count);
  delay(1000);
}
else if (callback == 0) {
  Serial.println    (count);
  mp3_play (antrianNomor);
```

```
    delay(2000);
    suaraBilangan(count);
    delay(1000);
    //    ESPClient.println(count);
    //    ESPClient.flush();
   }
}

void CheckWiFiConnectivity()
{
  while (WiFi.status() != WL_CONNECTED)
  {
    for (int i = 0; i < 10; i++)
    {
      digitalWrite(LED0,  !HIGH);
      delay(250);
      digitalWrite(LED0,  !LOW);
      delay(250);
      Serial.print(".");
    }
    Serial.println("");
  }
}

void ESPRequest()
{
  ESPClient.stop();
  if (ESPClient.connect(ESPServer,  ESPServerPort))
  {
    Serial.println     ("<" + ClientType + "- CONNECTED>");
    ESPClient.println  ("<" + ClientType + "- CONNECTED>");
  }
}

void mp3_play_and_wait(uint16_t num) {
  mp3_play (num);
  delay(1000);
}

void suaraBilangan(uint32_t Bilangan)
{
  if (Bilangan < 100)
```

```
  {
    suaraPuluhan(Bilangan);
  }
  else if (Bilangan < 1000)
  {
    suaraRatusan(Bilangan);
  }
  else
  {
    suaraRibuan(Bilangan);
  }
}
void suaraPuluhan(uint8_t Bilangan)
{
  if (Bilangan < 12)
  {
    mp3_play_and_wait(Bilangan);
  }
  else if (Bilangan < 20)
  {
    mp3_play_and_wait(Bilangan - 10);
    mp3_play_and_wait(belas);
  }
  else
  {
    uint8_t puluhan = Bilangan / 10;
    mp3_play_and_wait(puluhan);
    mp3_play_and_wait(puluh);

    puluhan *= 10;
    if (Bilangan - puluhan != 0)
    {
      mp3_play_and_wait((Bilangan - puluhan));
    }
  }
}
void suaraRatusan(uint16_t Bilangan)
{
  uint8_t ratusan = (uint8_t)(Bilangan / 100);
  if (ratusan == 1)
  {
    mp3_play_and_wait(seratus);
```

```
  }
  else
  {
    mp3_play_and_wait(ratusan);
    mp3_play_and_wait(ratus);
  }
  if (Bilangan % 100)
  {
    suaraPuluhan(Bilangan - (ratusan * 100));
  }
}
void suaraRibuan(uint32_t Bilangan)
{
  uint16_t ribuan = (uint16_t)(Bilangan / 1000);
  if (ribuan == 1)
  {
    mp3_play_and_wait(seribu);
  }
  else if (ribuan < 100)
  {
    suaraPuluhan(ribuan);
    mp3_play_and_wait(ribu);
  }
  else
  {
    suaraRatusan(ribuan);
    mp3_play_and_wait(ribu);
  }
  if (Bilangan % 1000)
  {
    suaraRatusan(Bilangan - (ribuan * 1000));
  }
}
```

Progam Counter

```
#include <WiFi.h>
#include <DMD32.h>
#include "fonts/SystemFont5x7.h"
#include "fonts/Arial_black_16.h"
#define DISPLAYS_ACROSS  1
#define DISPLAYS_DOWN  1
```

```
#define    LED0      2
#define    BUTTON    4
#define    MAXSC     10

unsigned long awal = 0, akhir = 1000;
char*      ESPssid;
char*      ESPpassword;
String     Message;
byte b;
String data;
int dataint;

hw_timer_t * timer = NULL;
DMD dmd(DISPLAYS_ACROSS,  DISPLAYS_DOWN);
WiFiServer  ESPServer(9001);
WiFiClient  ESPClient[MAXSC];

void IRAM_ATTR triggerScan()
{
  dmd.scanDisplayBySPI();
}

void setup(void)
{
  //inialisasi
  Serial.begin(115200);
  pinMode(LED0,  OUTPUT);
  pinMode(BUTTON,  INPUT_PULLUP);
  Serial.println("I/O  Pins Modes Set .... Done");
  SetWifi("Mesin Antrian", "mesinantrian");
  uint8_t cpuClock = ESP.getCpuFreqMHz();
  timer = timerBegin(0,  cpuClock, true);
  timerAttachInterrupt(timer, &triggerScan, true);
  timerAlarmWrite(timer,  300, true);
  timerAlarmEnable(timer);
  dmd.clearScreen(  true );
}

void loop(void)
{
  //terima kirim  data
  AvailableClients();
```

```
  AvailableMessage();
  dataint = Message.toInt();

  if (dataint > 200)  {
    dataint = 0;
  }

  //tampil nilai display
  dmd.selectFont(Arial_Black_16);
  dmd.dikasiInfoMaseh(dataint);
}

//open wifi akses poin
void SetWifi(char*  Name, char* Password)
{
  WiFi.disconnect();
  WiFi.mode(WIFI_AP_STA);
  Serial.println("WIFI  Mode : AccessPoint Station");
  ESPssid      = Name;
  ESPpassword  = Password;
  WiFi.softAP(ESPssid, ESPpassword);
  Serial.println("WIFI  < " + String(ESPssid)  + " > ... Started");
  delay(500);
  IPAddress IP = WiFi.softAPIP();
  Serial.print("AccessPoint IP : ");
  Serial.println(IP);
  Serial.print("AccessPoint MC : ");
  Serial.println(String(WiFi.softAPmacAddress()));
  ESPServer.begin();
  ESPServer.setNoDelay(true);
  Serial.println("Server  Started");
}

//deteksi client konek
void AvailableClients()
{
  if (ESPServer.hasClient())
  {
    if (digitalRead(LED0)  == HIGH)  digitalWrite(LED0,  LOW);
    for (uint8_t i = 0; i < MAXSC;  i++)
    {
      if (!ESPClient[i]  || !ESPClient[i].connected())
```

```
    {
      if (ESPClient[i])
      {
        ESPClient[i].stop();
      }
      if (ESPClient[i]  = ESPServer.available())
      {
        Serial.println("New  Client: " + String(i + 1));
      }
      continue;
    }
  }
  WiFiClient  ESPClient  = ESPServer.available();
  ESPClient.stop();
 }
 else
 {
  digitalWrite(LED0,  HIGH);
  delay(250);
  digitalWrite(LED0,  LOW);
  delay(250);
 }
}

//deteksi pesan
void AvailableMessage()
{
 for (uint8_t i = 0; i < MAXSC; i++)
 {
  if (ESPClient[i] && ESPClient[i].connected() &&
ESPClient[i].available())
  {
    while (ESPClient[i].available())
    {
      Message = ESPClient[i].readStringUntil('\n');
      ESPClient[i].flush();
      ClientNumber();
      dmd.clearScreen( true );
      Serial.println(Message);
    }
  }
 }
```

```
}

//nomor client
void  ClientNumber()  {
  if (Message == "<Cliente  01-1>")  {
    Serial.println("datamasuk1");
  }
  if (Message == "<Cliente  01-0>")  {
    Serial.println("datamasuk0");
  }
}
```

Progam Display

```
#include <WiFi.h>
#include <DMD32.h>
#include "fonts/SystemFont5x7.h"
#include "fonts/Arial_black_16.h"

#define DISPLAYS_ACROSS  1
#define DISPLAYS_DOWN  1
#define     LED0       2
#define     BUTTON     4
#define     MAXSC      10

unsigned long awal = 0, akhir = 1000;
char*       ESPssid;
char*       ESPpassword;
String      Message;
byte b;
String data;
int dataint;

hw_timer_t  * timer = NULL;
DMD dmd(DISPLAYS_ACROSS,   DISPLAYS_DOWN);
WiFiServer   ESPServer(9001);
WiFiClient   ESPClient[MAXSC];

void IRAM_ATTR triggerScan()
{
  dmd.scanDisplayBySPI();
}
```

```
void setup(void)
{
  //inialisasi
  Serial.begin(115200);
  pinMode(LED0,  OUTPUT);
  pinMode(BUTTON,  INPUT_PULLUP);
  Serial.println("I/O  Pins Modes Set .... Done");
  SetWifi("Mesin Antrian", "mesinantrian");
  uint8_t cpuClock = ESP.getCpuFreqMHz();
  timer = timerBegin(0,  cpuClock, true);
  timerAttachInterrupt(timer, &triggerScan, true);
  timerAlarmWrite(timer,  300, true);
  timerAlarmEnable(timer);
  dmd.clearScreen(  true );
}

void loop(void)
{
  //terima kirim  data
  AvailableClients();
  AvailableMessage();
  dataint = Message.toInt();

  if (dataint > 200)  {
    dataint = 0;
  }

  //tampil nilai display
  dmd.selectFont(Arial_Black_16);
  dmd.dikasiInfoMaseh(dataint);
}

//open wifi akses poin
void SetWifi(char*  Name, char* Password)
{
  WiFi.disconnect();
  WiFi.mode(WIFI_AP_STA);
  Serial.println("WIFI  Mode : AccessPoint Station");
  ESPssid      = Name;
  ESPpassword  = Password;
  WiFi.softAP(ESPssid, ESPpassword);
```

```
  Serial.println("WIFI  < " + String(ESPssid)  + " > ...  Started");
  delay(500);
  IPAddress IP = WiFi.softAPIP();
  Serial.print("AccessPoint IP  : ");
  Serial.println(IP);
  Serial.print("AccessPoint MC  : ");
  Serial.println(String(WiFi.softAPmacAddress()));
  ESPServer.begin();
  ESPServer.setNoDelay(true);
  Serial.println("Server  Started");
}

//deteksi client konek
void AvailableClients()
{
  if (ESPServer.hasClient())
  {
    if (digitalRead(LED0)  == HIGH)  digitalWrite(LED0,  LOW);
    for (uint8_t i = 0; i < MAXSC;  i++)
    {
      if (!ESPClient[i]  || !ESPClient[i].connected())
      {
        if (ESPClient[i])
        {
          ESPClient[i].stop();
        }
        if (ESPClient[i]  = ESPServer.available())
        {
          Serial.println("New  Client: " + String(i + 1));
        }
        continue;
      }
    }
    WiFiClient  ESPClient  = ESPServer.available();
    ESPClient.stop();
  }
  else
  {
    digitalWrite(LED0,  HIGH);
    delay(250);
    digitalWrite(LED0,  LOW);
    delay(250);
```

```
  }
}

//deteksi pesan
void AvailableMessage()
{
  for (uint8_t i = 0; i < MAXSC; i++)
  {
    if (ESPClient[i] && ESPClient[i].connected() &&
ESPClient[i].available())
    {
      while (ESPClient[i].available())
      {
        Message = ESPClient[i].readStringUntil('\n');
        ESPClient[i].flush();
        ClientNumber();
        dmd.clearScreen( true );
        Serial.println(Message);
      }
    }
  }
}

//nomor client
void ClientNumber() {
  if (Message == "<Cliente 01-1>") {
    Serial.println("datamasuk1");
  }
  if (Message == "<Cliente 01-0>") {
    Serial.println("datamasuk0");
  }
}
```

BIODATA PENULIS

Nama	: Aldhi Sulthon Fathoni
Tempat /Tanggal lahir	: Cilacap, 24 Juni 2000
Alamat	: Jln. Bharata Rt 04/ Rw 06, Tritih Wetan Kecamatan Jeruklegi, Kabupaten Cilacap.
Telepon/Hp	: 081252333417
Hobi	: Futsal
Email	: Aldysulton2400@gmail.com
Riwayat Pendidikan	:

- SDN 01 Tritih Wetan Cilacap : 2006 - 2012
- SMPN 1 Jeruklegi Cilacap : 2012 - 2015
- SMK Boedi oetomo Cilacap : 2015 - 2018