

E-BOOK ESP8266

RFID
Bluetooth HC-05
LCD I2C 16x2
RTC
Dot Matrix
Database
Thingspeak
Telegram
Cayenne

www.anakkendali.com

Daftar Isi

Daftar Isi.....	2
Perkenalan	5
ESP8266 ESP-01.....	5
ESP8266 NodeMCU.....	5
ESP8266 Wemos D1.....	6
BAB 1.....	7
Dasar Python.....	7
Instalasi Python di Windows.....	7
Contoh Program dengan MicroPython.....	8
BAB 2	11
Arduino IDE ESP8266	11
Instalasi Paket Board ESP8266 pada Arduino IDE.....	11
BAB 3	12
GPIO	12
Program Led Blink ESP8266 NodeMCU / Wemos D1	13
Program Led dan Button ESP8266 NodeMCU / Wemos D1	14
BAB 4	15
I2C	15
Program Mengakses LCD 16x2 I2C.....	15
Program Mengakses LCD OLED dan RTC DS3231	16
BAB 5	27
SPI.....	27
Program Mengakses Dotmatrix 8x8 Max7219.....	27
Program Mengakses RFID RC522	28
BAB 6	31
SERIAL.....	31
Program Mengakses Bluetooth HC-05.....	31
Program Komunikasi NodeMCU / Wemos D1 dengan Arduino Uno.....	32
BAB 7	34
Wifi.....	34
Konfigurasi ESP8266 ESP-01 dengan Arduino Uno	34

Program Mengirim Data ke Thingspeak ESP8266 ESP-01.....	36
Program Mengirim Data ke Database MySQL ESP8266 ESP-01	39
Kontrol Relay/LED Arduino Uno dan ESP8266 ESP-01 Alikasi Android.....	56
Kontrol Relay/LED Arduino Uno dan ESP8266 ESP-01 dengan Telegram.....	60
Menghubungkan ESP8266 ESP-01 ke Cayenne	65
Program NodeMCU / Wemos D1 Mini Kirim Data DHT11 ke Thingspeak	69
Program NodeMCU / Wemos D1 Mini Kirim Data DHT11 ke Database MySQL.....	71
Program NodeMCU / Wemos D1 Kontrol LED Aplikasi Android.....	75
Program NodeMCU / Wemos D1 Kontrol LED dengan Cayenne.....	77

Terimakasih Kepada...

- ALLAH SWT. Alhamdulillah segala puji bagi tuhan semesta alam, ALLAH Subhanahu Wata'ala, yang telah memberikan kita nikmat serta kesempatan untuk mempelajari Ciptaanya. Sehingga dapat lebih bersyukur atas apa yang sudah kita terima. karena rahmatnya lah kita memiliki apa yang ada sekarang, dan sejatinya hanya miliknya semua yang ada di dunia dan seisinya. Tidak cukup dengan perkataan terimakasih atas syukur yang telah tuhan berikan kepada kita. Karena itu sebisa mungkin kita dapat memanfaatkan apa yang telah diberikannya dengan cara menjadikan pemberianya bermanfaat bagi diri kita maupun orang-orang sekitar.
- Nabi Muhammad SAW. Sholawat serta salam kepada Nabi Muhammad Sollallahu Alaihi Wassallam, karenanya kita dapat mengenal tuhan dan mengerti jalan yang baik dan jalan yang buruk.
- Kedua Orang Tua. Atas Izin Allah subhanahu wata'ala, orang tua menjadi perantara saya terlahir didunia ini, serta yang selalu mendukung keinginan saya untuk terus belajar sampai ke perguruan tinggi meskipun pekerjaan sebagai seorang Petani.
- Pesantren Al-Ishlah AL-Ishlah Tajug Indramayu. Disinilah tempat saya dididik karakter serta akhlak, semua yang telah saya lalui sewaktu belajar di tempat ini menjadi bermanfaat setelah saya menjadi alumni dan menghadapi kehidupan bermasyarakat.
- dan semua pihak yang berperan dalam kehidupan saya, yang tidak bisa saya sebutkan satu persatu. Hanya dapat mendoakan semoga Allah membala semua jasa-jasanya.

saya memohon maaf bila di dalam ebook ini ada berupa kesalahan, baik tulisan maupun program. Silahkan bisa di diskusikan dengan saya sendiri melalui kontak yang sudah tertera pada halaman tentang penulis.

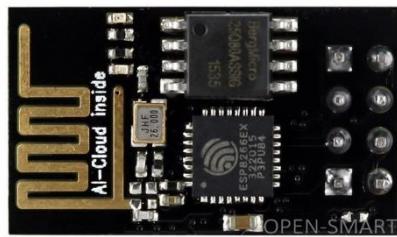
Perkenalan

ESP8266 merupakan modul Wifi yang sangat familiar bagi pecinta mikrokontroller baik mereka yang masih di tingkat dasar maupun tingkat lanjut, dengan modul ini kita dapat mengirim ataupun menerima data melalui jaringan lokal wifi saja ataupun jaringan internet. Pada era revolusi Industri 4.0 keberadaan modul ESP8266 sangatlah bermanfaat sebagai alat bantu untuk mewujudkan semua sistem agar dapat terintegrasi dengan Internet yang kita sebut sebagai *Internet of Things*, baik karena harganya yang terjangkau juga kualitas nyasangat memadai dalam menyediakan layanan untuk kebutuhan user.

Modul ESP8266 banyak tipenya walupun semuanya bisa dikatakan sama, dan berikut adalah penjelasan masing masing dari tipe modul ESP8266.

ESP8266 ESP-01

ESP-01 adalah modul ESP8266 yang bentuknya paling kecil diantara saudaranya yang lain. dan juga hanya memiliki dua buah pin GPIO meskipun bisa langsung di isi program. namun dalam buku ini hanya di jelaskan cara komunikasi ESP-01 dengan Arduino Uno sehingga Arduino Uno dapat mengirim dan menerima data melalui jaringan Wifi Local ataupun Internet.



Gambar 1. ESP-01 [5]

ESP8266 NodeMCU

NodeMCU adalah Modul ESP8266 yang paling familiar diantara saudaranya yang lain. pada modul ini sudah terdapat IC CH340 sehingga sudah bisa langsung di isi program dari komputer menggunakan Port Serial, didalam buku ini akan dijelaskan bagaimana cara memprogram modul NodeMCU menggunakan bahasa pemrograman Python (micropython) dasar, dan bahasa yang digunakan oleh Arduino IDE.



Gambar 2. NodeMCU [6]

ESP8266 Wemos D1

Wemos D1 adalah bagian dari modul ESP8266 yang sama dengan NodeMCU bisa langsung di isi program melalui komunikasi serial karena sudah terdapat IC CH340, dalam buku ini Wemos D1 akan di jelaskan Cara Program dengan Python (micropython) dan Arduino IDE.

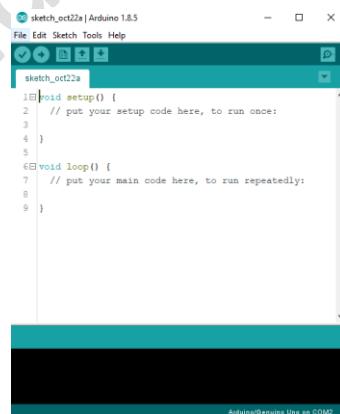


Gambar 3. Wemos D1 [7]



Gambar 4. Wemos D1 Mini[8]

Arduino IDE adalah Software yang sangat sering di gunakan untuk membuat sekaligus mendownload program pada mikrokontroler. Karena selain untuk papan Arduino itu sendiri Arduino IDE bisa untuk mendownload berbagai macam jenis mikrokontroler, seperti **STM32** dan **Modul ESP8266**



Micropython adalah bahasa pemrograman Python yang digunakan untuk memprogram Mikrokontroller. Seperti halnya pada ESP8266, selain itu jika kita mengamati bagaimana Arduino IDE bisa men-Download program pada modul ESP8266 karena terdapat Package ESP8266 yang terinstall di Software Arduino IDE nya. Dan package ESP8266 ini dibuat dari Pemograman Python.

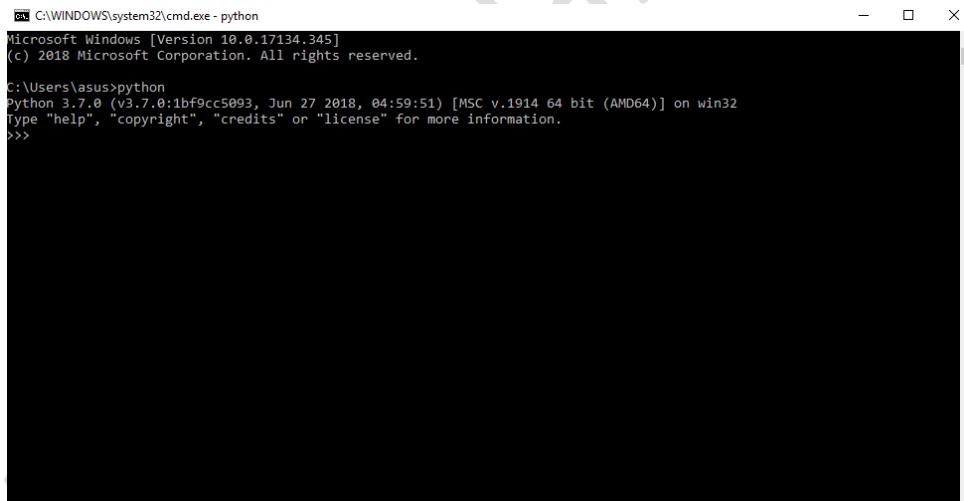
BAB 1

Dasar Python

Instalasi Python di Windows

Windows merupakan Sistem Operasi yang banyak digunakan karena alasan tampilan yang menarik dan mudah mengoperasikanya, sehingga pada buku ini hanya di jelaskan bagaimana cara memasang python di sistem operasi Windows, berikut adalah langkah-langkah meng-install python pada Windows :

1. Kunjungi website resminya di <https://www.python.org/downloads/>
Silahkan download Software nya sesuai Windows yang digunakan
2. Buka hasil download nya dan install compiler python, jangan lupa untuk centang pada kolom “Add Python 3.7 to PATH”
3. Setelah selesai instalasi silahkan buka CMD.exe dengan cara tekan tombol Windows+R lalu ketikan cmd
4. Pada kolom cmd silahkan ketik python dan enter



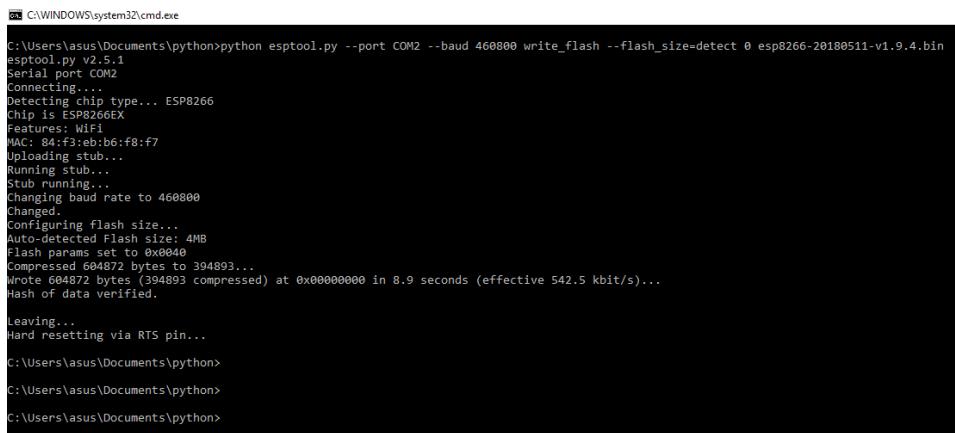
A screenshot of a Windows Command Prompt window titled 'C:\WINDOWS\system32\cmd.exe - python'. The window shows the following text:
Microsoft Windows [Version 10.0.17134.345]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\asus>python
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>

Jika tampilanya seperti gambar di atas berarti python telah berhasil di isntall

5. Setelah itu install beberapa library yang dibutuhkan dengan cara berikut :
python -m pip install numpy
python -m pip install esptool
6. Setelah itu ketikan perintah seperti berikut pada CMD:
cd documents
md python
7. Silahkan pergi ke directori :
C:\Users\asus\AppData\Local\Programs\Python\Python37\Lib\site-packages
Kemudian pindahkan file **esptool.py** ke **C:\Users\asus\Documents\python**
Ketikan perintah
cd python
python esptool.py -port COM2 erase_flash
*COM2 sesuaika dengan port com yang terdeteksi pada komputer

8. Download firmware untuk ESP8266 di
<http://micropython.org/resources/firmware/esp8266-20180718-v1.9.4-272-g46091b8a.bin>
9. Pindahkan file yang tadi di download ke *C:\Users\asus\Documents\python*
10. Ketikan perintah seperti berikut :
`python esptool.py --port COM2 --baud 460800 write_flash --flash_size=detect 0 esp8266-20180511-v1.9.4.bin`



```

C:\WINDOWS\system32\cmd.exe
C:\Users\asus\Documents\python>python esptool.py --port COM2 --baud 460800 write_flash --flash_size=detect 0 esp8266-20180511-v1.9.4.bin
esptool.py v2.5.1
Serial port COM2
Connecting...
Detecting chip type... ESP8266
Chip is ESP8266EX
Features: WiFi
MAC: 84:f3:eb:b6:f8:f7
Uploading stub...
Running stub...
Stub running...
Changing baud rate to 460800
Changed.
Configuring flash size...
Auto-detected Flash size: 4MB
Flash params set to 0xe0040
Compressed 604872 bytes to 394893...
Wrote 604872 bytes (394893 compressed) at 0x00000000 in 8.9 seconds (effective 542.5 kbit/s)...
Hash of data verified.

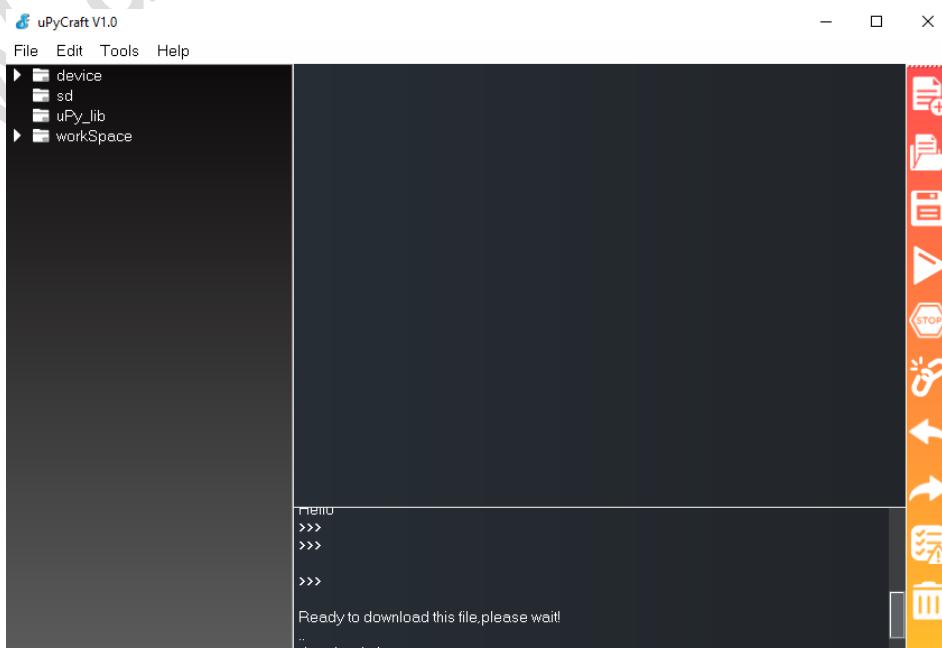
Leaving...
Hard resetting via RTS pin...
C:\Users\asus\Documents\python>
C:\Users\asus\Documents\python>
C:\Users\asus\Documents\python>
```

Gambar 7. Melakukan Flash pada NodeMCU

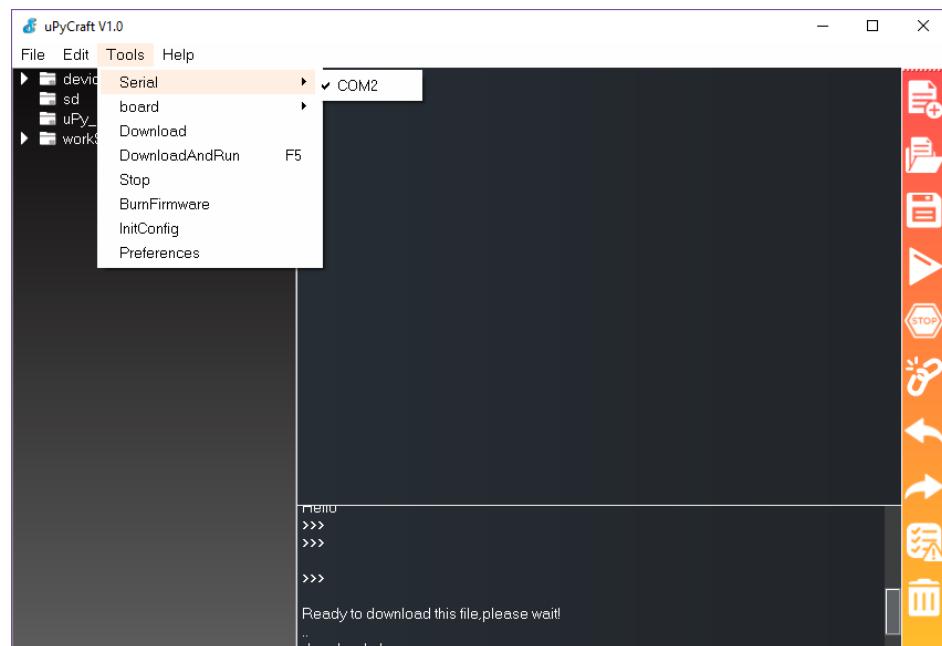
11. Proses instalasi sudah selesai sekarang sudah bisa memprogram langsung Modul ESP8266 menggunakan terminal yang terinstall pada windows, contohnya bisa menggunakan Serial Monitor Arduino atau bisa menggunakan .

Contoh Program dengan MicroPython

1. Pada buku ini akan dijelaskan cara membuat program micropython dengan Software uPyCraft V1.0, Software ini bisa di download pada link
<https://randomnerdtutorials.com/uPyCraftWindows>
2. Install aplikasi uPyCraft V1.0 kemudian buka aplikasi nya.

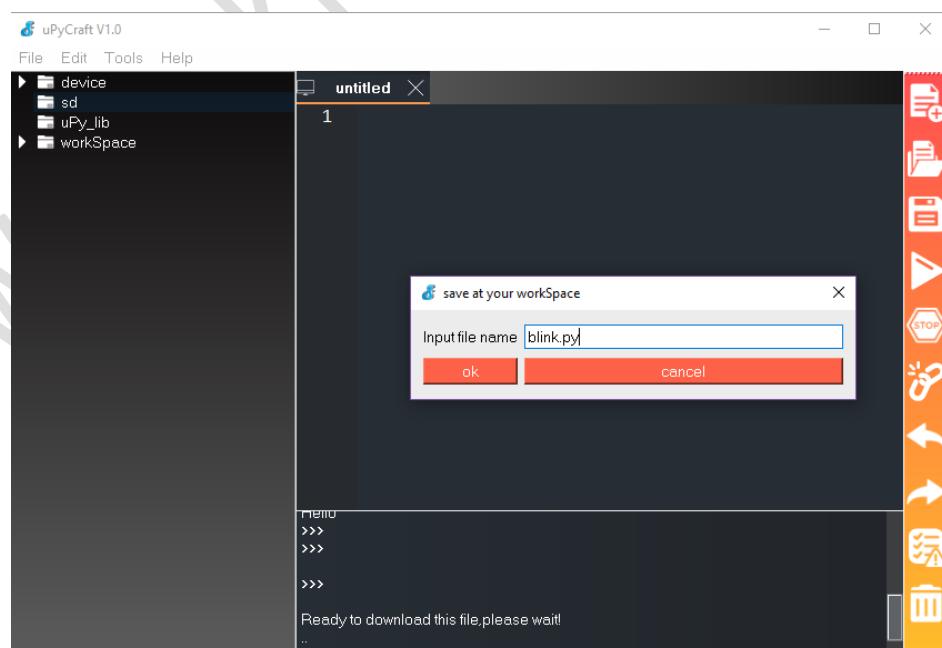


Pilih Tool > Serial > pilih Port Com sesuai yang terdeteksi oleh komputer



Gambar 8.b. uPyCraft V1.0

3. Pilih File > New > setelah keluar jendela editor silahkan save terlebih dahulu, dan berikan nama “blink.py”



Gambar 8.c. uPyCraft V1.0

4. Buatlah program seperti berikut kemudian jalankan dengan cara menekan tombol **F5** atau bisa juga dengan cara pilih Tools > Download And Run

```
from machine import Pin
```

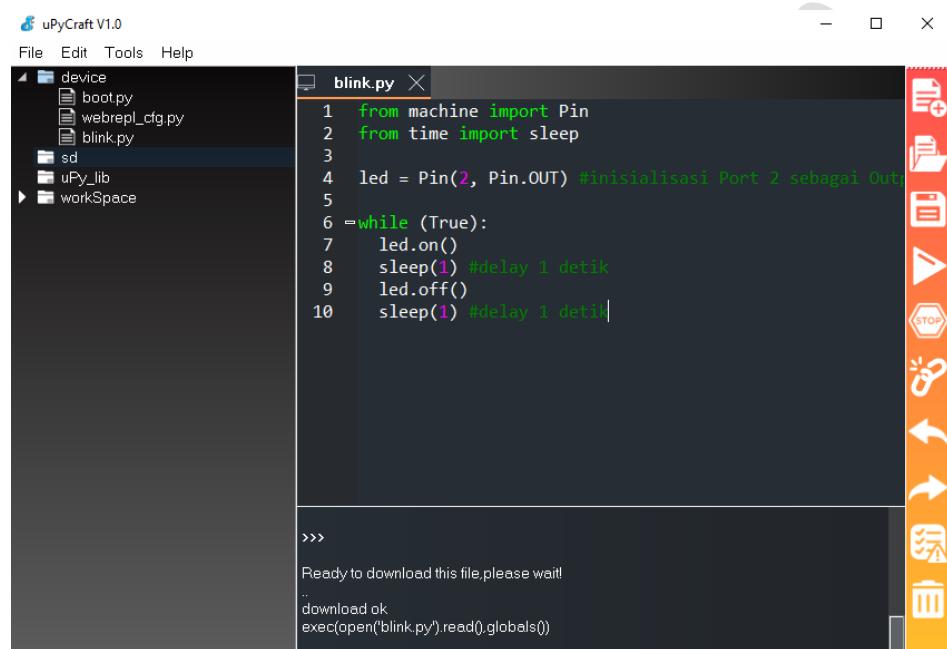
```

from time import sleep

led = Pin(2, Pin.OUT) #inisialisasi Port 2 sebagai Output

while (True):
    led.on()
    sleep(1) #delay 1 detik
    led.off()
    sleep(1) #delay 1 detik

```



Gambar 8.d. uPyCraft V1.0

Demikian tutorial dasar cara memprogram ESP8266 [1] menggunakan MicroPython untuk selebihnya bisa di kembangkan sendiri, karena dibuku ini lebih fokus pada tutorial Cara memprogram ESP8266 dengan Arduino IDE, karena menggunakan cara di atas program yang kita buat tidak di simpan pada memori ESP8266 sehingga ketika kabel USB di cabut maka program akan hilang.

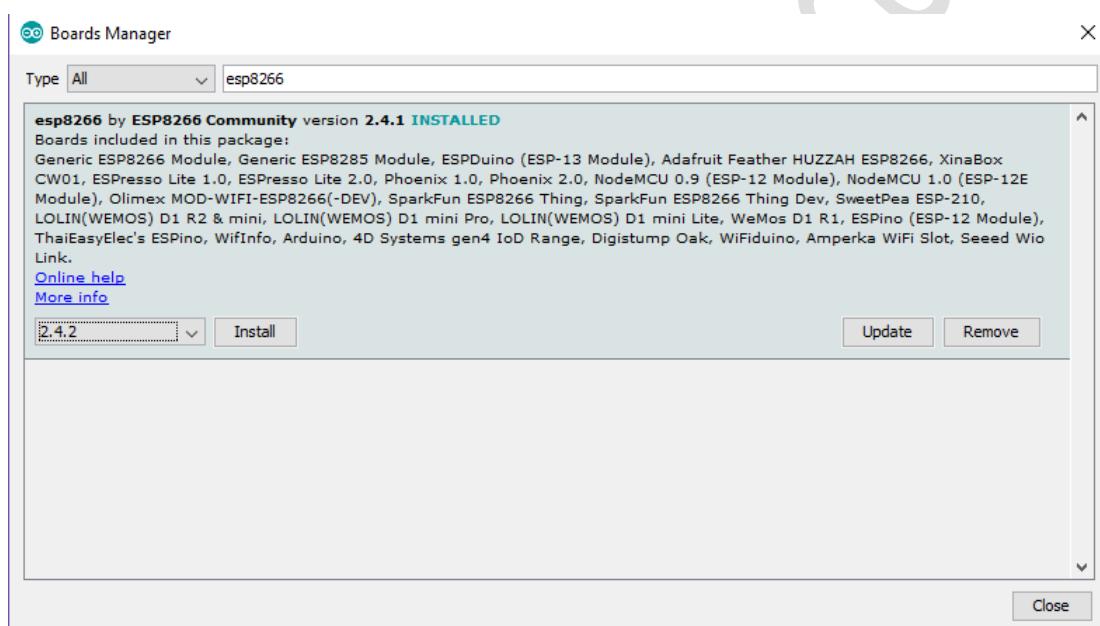
BAB 2

Arduino IDE ESP8266

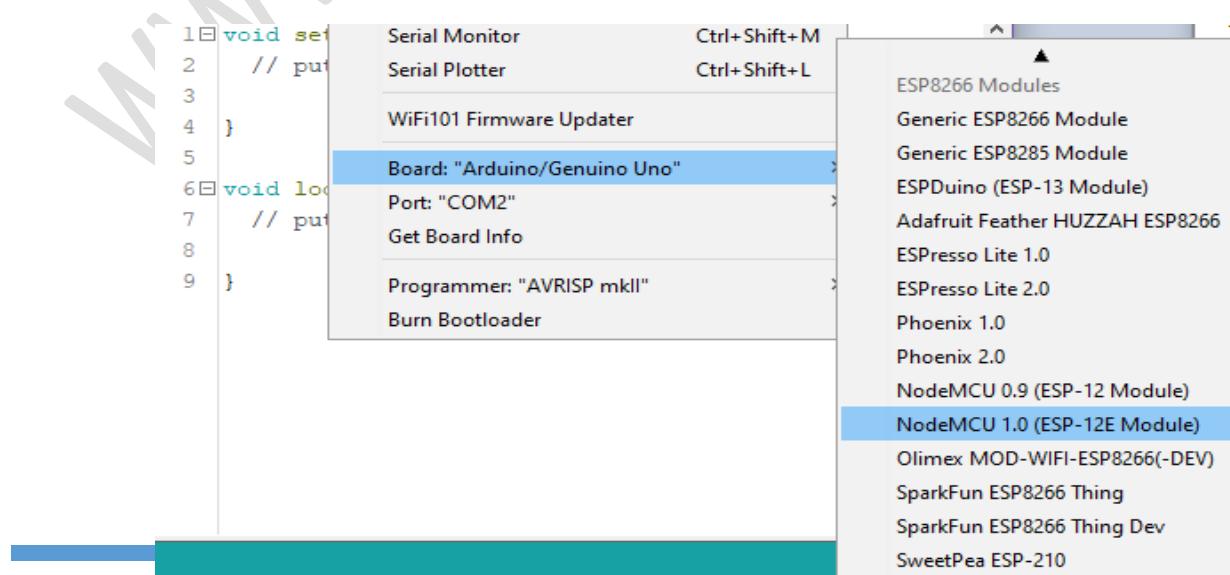
Instalasi Paket Board ESP8266 pada Arduino IDE

Arduino IDE belum memiliki paket untuk modul ESP8266 tapi sudah ada pembuatnya sehingga kita hanya perlu memasang paket itu ke dalam Arduino IDE. Berikut cara nya :

1. Buka Arduino IDE pilih File > Preferences kemudian salin alamat berikut ke kolom Additional Boards Manager URLs
http://arduino.esp8266.com/stable/package_esp8266com_index.json
2. Pilih Tools > Boards > Boards Manager..
Cari ESP8266 Community dan Install versi terbaru nya. Lihat gambar berikut :



3. Jika sudah terinstall maka pada pilihan board akan tersedia pilihan untuk Board ESP8266 seperti gambar berikut :



BAB 3

GPIO

Pada ESP8266 juga sudah terdapat pin GPIO yang bisa di fungsikan sebagai INPUT ataupun OUTPUT, GPIO adalah singkatan dari General Purpose Input Output, fungsinya hampir sama dengan pin Digital IO pada Arduino, tapi sebenarnya memiliki perbedaan antara GPIO dan pin Digital IO.

Dimana letak perbedaan GPIO dan pin Digital IO?

Jika diperhatikan mikrokontroler yang memiliki GPIO bekerja pada daerah tegangan 3.3V sedangkan yang memiliki pin Digital IO bekerja pada daerah tegangan 5V. Contoh yang memiliki GPIO antara lain, Raspberry Pi, STM32, dan ESP8266 dan mereka bekerja pada tegangan 3.3V, sedangkan Arduino menggunakan pin Digital IO dan bekerja pada tegangan 5V.

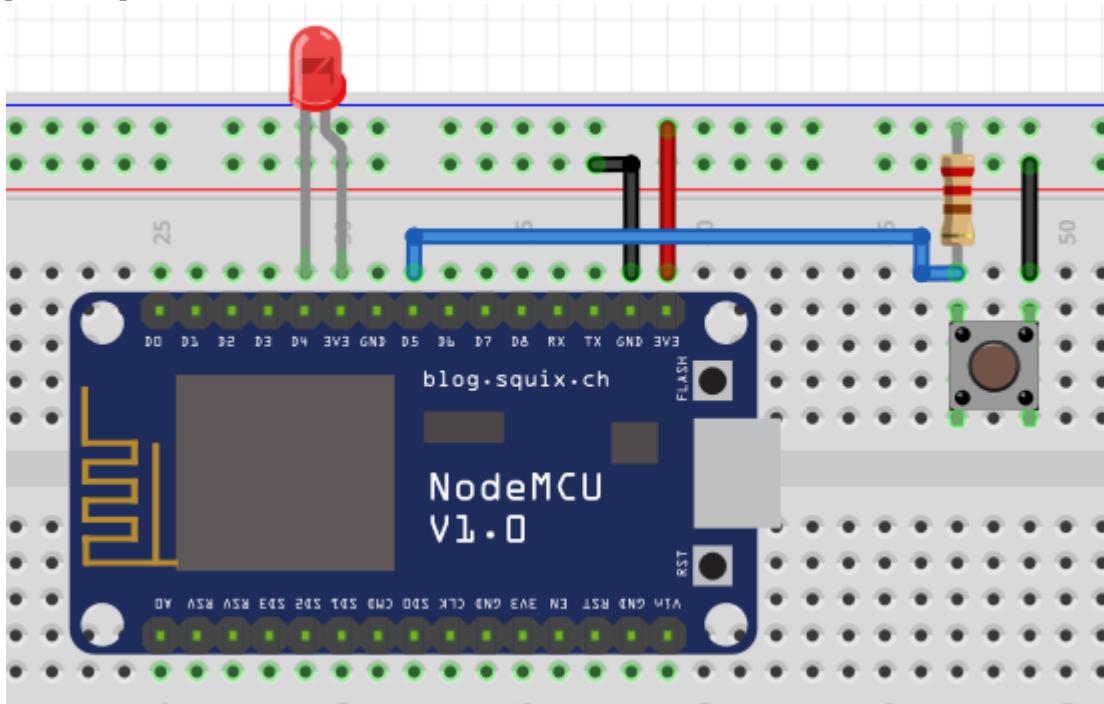
Berikut adalah tabel Pin GPIO pada ESP8266 Wemos D1 dan NodeMCU

Pin Board	Fungsi	Pin ESP8266
TX	TXD	TXD
RX	RXD	RXD
A0	Analog input, max 3.3V input	A0
D0	IO	GPIO16
D1	IO, SCL	GPIO5
D2	IO, SDA	GPIO4
D3	IO, 10k Pull-up	GPIO0
D4	IO, 10k Pull-up, BUILTIN_LED	GPIO2
D5	IO, SCK	GPIO14
D6	IO, MISO	GPIO12
D7	IO, MOSI	GPIO13
D8	IO, 10k Pull-down, SS	GPIO15
G	Ground	GND
5V	5V	-
3V3	3.3V	3.3V
RST	Reset	RST

Tabel 1. GPIO ESP8266 [2]

Program Led Blink ESP8266 NodeMCU / Wemos D1

Buka Arduino IDE pilih Tools > Boards > NodeMCU V1.0 > Pilih Port COM yang terdeteksi pada komputer



Gambar 9. Rangkaian LED dan Button

Berikut adalah program Led Blink dengan indikator Led bawaam modul ESP8266

```
void setup() {  
    // put your setup code here, to run once:  
    pinMode(2, OUTPUT);  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    digitalWrite(2, 1);  
    delay(1000);  
    digitalWrite(2, 0);  
    delay(1000);  
}
```

```
void setup() {  
    // put your setup code here, to run once:  
    pinMode(D4, OUTPUT);  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    digitalWrite(D4, 1);  
    delay(1000);  
    digitalWrite(D4, 0);  
}
```

```
delay(1000);  
}
```

Kedua program diatas hasil nya sama.

Program Led dan Button ESP8266 NodeMCU / Wemos D1

Buka Arduino IDE pilih Tools > Boards > NodeMCU V1.0 > Pilih Port COM yang terdeteksi pada komputer

```
int state=0;  
void setup() {  
    // put your setup code here, to run once:  
    pinMode(D4, OUTPUT);  
    pinMode(D5, INPUT_PULLUP);  
    Serial.begin(9600);  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    if (digitalRead(D5) == 0){  
        while (digitalRead(D5) == 0){}  
        state = !state;  
        Serial.println(state);  
    }  
    digitalWrite(D4, state);  
}
```

Program diatas jika dijalankan hasilnya toggle

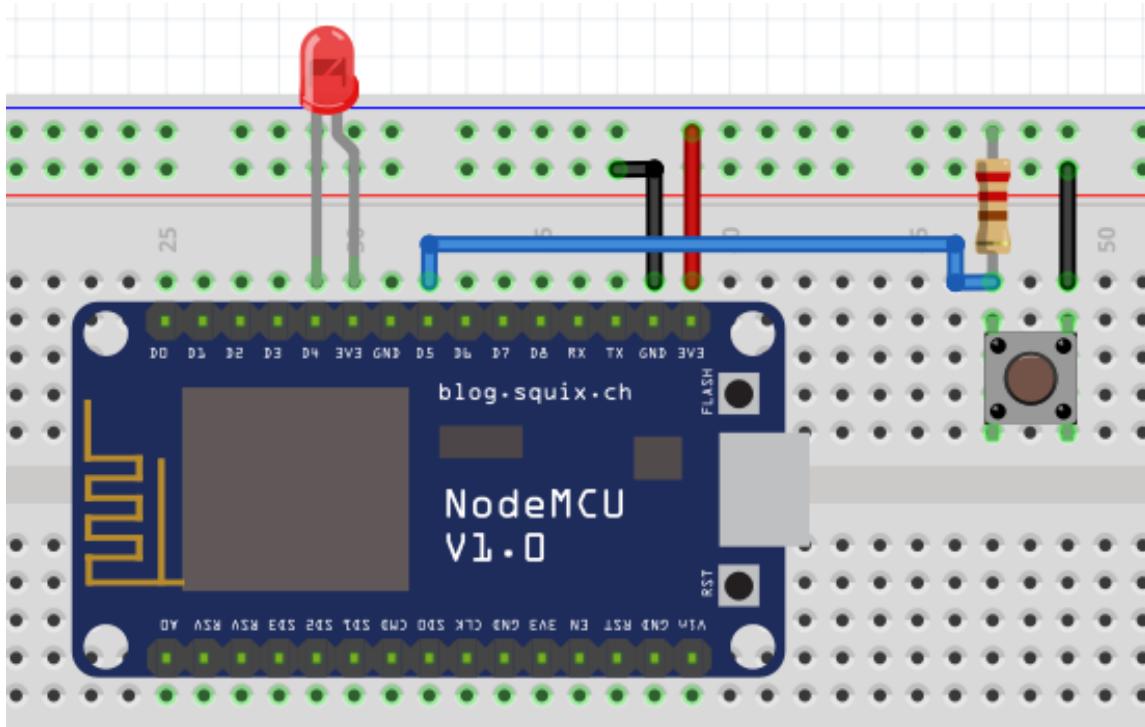
```
uint8_t state=0;  
void setup() {  
    // put your setup code here, to run once:  
    pinMode(D5, INPUT_PULLUP);  
    Serial.begin(9600);  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    if (digitalRead(D5) == 0){  
        while (digitalRead(D5) == 0){}  
        state++;  
        if(state == 255)state=0;  
        Serial.println(state);  
    }  
    analogWrite(D4, state);  
}
```

Program diatas jika dijalankan hasilnya akan mengatur kecerahan cahaya LED

BAB 4

I2C

Program Mengakses LCD 16x2 I2C



Gambar 9. Rangkaian LED dan Button

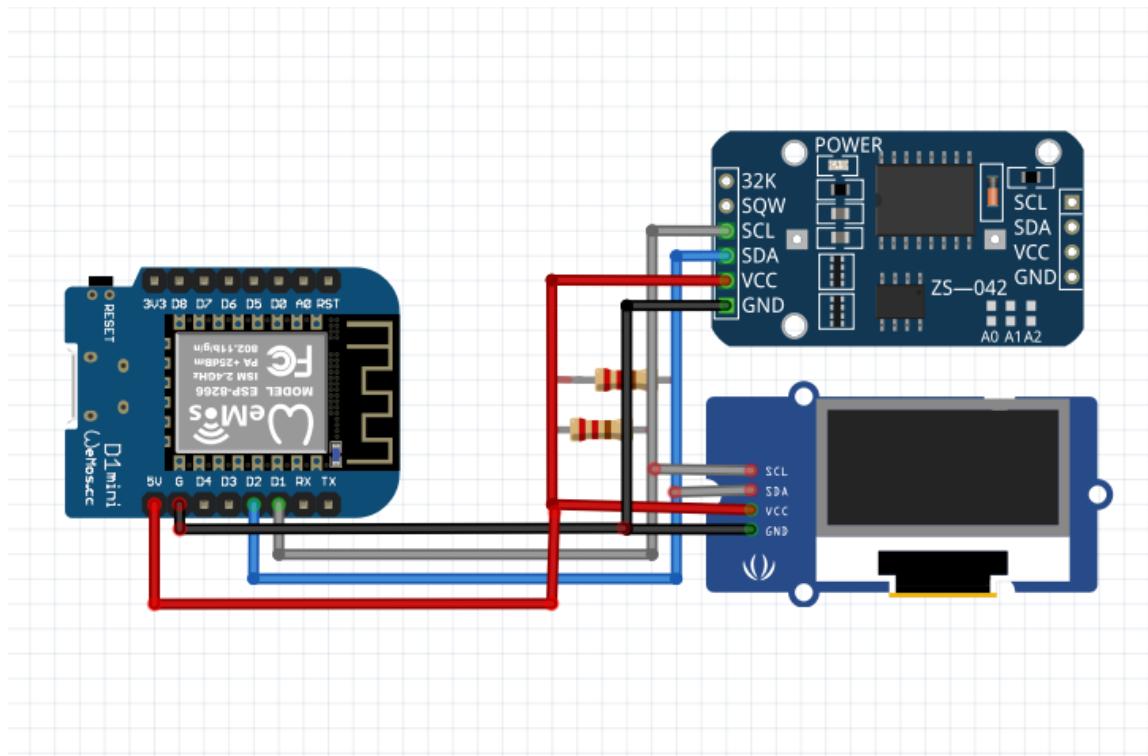
```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);

void setup()
{
    // initialize the LCD
    lcd.begin();
    // Wire.begin(D2, D1);
    // Turn on the blacklight and print a message.
    lcd.print("anakkendali.com");
}

void loop()
{
    // Do nothing here...
}
```

Program Mengakses LCD OLED dan RTC DS3231



Gambar 10. Rangkaian Wemos D1, RTC dan Oled

Sebelum membuat program silahkan Download terlebih dahulu Library OLED di alamat berikut :

https://github.com/adafruit/Adafruit_SSD1306

dan library RTC DS3231

<https://github.com/adafruit/RTCLib>

Program Test LCD OLED

```
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define OLED_RESET 0
Adafruit_SSD1306 display(OLED_RESET);

#define NUMFLAKES 10
#define XPOS 0
#define YPOS 1
#define DELTAY 2

#define LOGO16_GLCD_HEIGHT 16
#define LOGO16_GLCD_WIDTH 16
static const unsigned char PROGMEM logo16_glcd_bmp[] =
{ B00000000, B11000000,
  B00000001, B11000000,
```

```

B00000001, B11000000,
B00000011, B11100000,
B1110011, B11100000,
B1111110, B11111000,
B0111110, B1111111,
B00110011, B10011111,
B00011111, B11111100,
B00001101, B01110000,
B00011011, B10100000,
B00111111, B11100000,
B00111111, B11110000,
B01111100, B11110000,
B01110000, B01110000,
B00000000, B00110000 };

#if (SSD1306_LCDHEIGHT != 64)
#error("Height incorrect, please fix Adafruit_SSD1306.h!");
#endif

void setup() {
  Serial.begin(9600);

  // by default, we'll generate the high voltage from the 3.3v line internally! (neat!)
  display.begin(SSD1306_SWITCHCAPVCC, 0x3D); // initialize with the I2C addr 0x3D
  // (for the 128x64)
  // init done

  // Show image buffer on the display hardware.
  // Since the buffer is initialized with an Adafruit splashscreen
  // internally, this will display the splashscreen.
  display.display();
  delay(2000);

  // Clear the buffer.
  display.clearDisplay();

  // draw a single pixel
  display.drawPixel(10, 10, WHITE);
  // Show the display buffer on the hardware.
  // NOTE: You _must_ call display after making any drawing commands
  // to make them visible on the display hardware!
  display.display();
  delay(2000);
  display.clearDisplay();

  // draw many lines
  testdrawline();
  display.display();
  delay(2000);
  display.clearDisplay();
}

```

```
// draw rectangles
testdrawrect();
display.display();
delay(2000);
display.clearDisplay();

// draw multiple rectangles
testfillrect();
display.display();
delay(2000);
display.clearDisplay();

// draw mulitple circles
testdrawcircle();
display.display();
delay(2000);
display.clearDisplay();

// draw a white circle, 10 pixel radius
display.fillCircle(display.width()/2, display.height()/2, 10, WHITE);
display.display();
delay(2000);
display.clearDisplay();

testdrawroundrect();
delay(2000);
display.clearDisplay();

testfillroundrect();
delay(2000);
display.clearDisplay();

testdrawtriangle();
delay(2000);
display.clearDisplay();

testfilltriangle();
delay(2000);
display.clearDisplay();

// draw the first ~12 characters in the font
testdrawchar();
display.display();
delay(2000);
display.clearDisplay();

// draw scrolling text
testscrolltext();
delay(2000);
display.clearDisplay();
```

```

// text display tests
display.setTextSize(1);
display.setTextColor(WHITE);
display.setCursor(0,0);
display.println("Hello, world!");
display.setTextColor(BLACK, WHITE); // 'inverted' text
display.println(3.141592);
display.setTextSize(2);
display.setTextColor(WHITE);
display.print("0x"); display.println(0xDEADBEEF, HEX);
display.display();
delay(2000);
display.clearDisplay();

// miniature bitmap display
display.drawBitmap(30, 16, logo16_glcd_bmp, 16, 16, 1);
display.display();
delay(1);

// invert the display
display.invertDisplay(true);
delay(1000);
display.invertDisplay(false);
delay(1000);
display.clearDisplay();

// draw a bitmap icon and 'animate' movement
testdrawbitmap(logo16_glcd_bmp, LOGO16_GLCD_HEIGHT, LOGO16_GLCD_WIDTH);
}

void loop() {

}

void testdrawbitmap(const uint8_t *bitmap, uint8_t w, uint8_t h) {
    uint8_t icons[NUMFLAKES][3];

    // initialize
    for (uint8_t f=0; f< NUMFLAKES; f++) {
        icons[f][XPOS] = random(display.width());
        icons[f][YPOS] = 0;
        icons[f][DELTAY] = random(5) + 1;

        Serial.print("x: ");
        Serial.print(icons[f][XPOS], DEC);
        Serial.print(" y: ");
        Serial.print(icons[f][YPOS], DEC);
        Serial.print(" dy: ");
        Serial.println(icons[f][DELTAY], DEC);
    }
}

```

```

}

while (1) {
    // draw each icon
    for (uint8_t f=0; f< NUMFLAKES; f++) {
        display.drawBitmap(Icons[f][XPOS], Icons[f][YPOS], bitmap, w, h, WHITE);
    }
    display.display();
    delay(200);

    // then erase it + move it
    for (uint8_t f=0; f< NUMFLAKES; f++) {
        display.drawBitmap(Icons[f][XPOS], Icons[f][YPOS], bitmap, w, h, BLACK);
        // move it
        Icons[f][YPOS] += Icons[f][DELTAY];
        // if its gone, reinit
        if (Icons[f][YPOS] > display.height()) {
            Icons[f][XPOS] = random(display.width());
            Icons[f][YPOS] = 0;
            Icons[f][DELTAY] = random(5) + 1;
        }
    }
}

void testdrawchar(void) {
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.setCursor(0,0);

    for (uint8_t i=0; i < 168; i++) {
        if (i == '\n') continue;
        display.write(i);
        if ((i > 0) && (i % 21 == 0))
            display.println();
    }
    display.display();
    delay(1);
}

void testdrawcircle(void) {
    for (int16_t i=0; i<display.height(); i+=2) {
        display.drawCircle(display.width()/2, display.height()/2, i, WHITE);
        display.display();
        delay(1);
    }
}

void testfillrect(void) {
    uint8_t color = 1;

```

```

for (int16_t i=0; i<display.height()/2; i+=3) {
    // alternate colors
    display.fillRect(i, i, display.width()-i*2, display.height()-i*2, color%2);
    display.display();
    delay(1);
    color++;
}
}

void testdrawtriangle(void) {
for (int16_t i=0; i<min(display.width(),display.height())/2; i+=5) {
    display.drawTriangle(display.width()/2, display.height()/2-i,
        display.width()/2-i, display.height()/2+i,
        display.width()/2+i, display.height()/2+i, WHITE);
    display.display();
    delay(1);
}
}

void testfilltriangle(void) {
uint8_t color = WHITE;
for (int16_t i=min(display.width(),display.height())/2; i>0; i-=5) {
    display.fillTriangle(display.width()/2, display.height()/2-i,
        display.width()/2-i, display.height()/2+i,
        display.width()/2+i, display.height()/2+i, WHITE);
    if (color == WHITE) color = BLACK;
    else color = WHITE;
    display.display();
    delay(1);
}
}

void testdrawroundrect(void) {
for (int16_t i=0; i<display.height()/2-2; i+=2) {
    display.drawRoundRect(i, i, display.width()-2*i, display.height()-2*i,
        display.height()/4, WHITE);
    display.display();
    delay(1);
}
}

void testfillroundrect(void) {
uint8_t color = WHITE;
for (int16_t i=0; i<display.height()/2-2; i+=2) {
    display.fillRoundRect(i, i, display.width()-2*i, display.height()-2*i,
        display.height()/4, color);
    if (color == WHITE) color = BLACK;
    else color = WHITE;
    display.display();
    delay(1);
}
}

```

```

}

void testdrawrect(void) {
    for (int16_t i=0; i<display.height()/2; i+=2) {
        display.drawRect(i, i, display.width()-2*i, display.height()-2*i, WHITE);
        display.display();
        delay(1);
    }
}

void testdrawline() {
    for (int16_t i=0; i<display.width(); i+=4) {
        display.drawLine(0, 0, i, display.height()-1, WHITE);
        display.display();
        delay(1);
    }
    for (int16_t i=0; i<display.height(); i+=4) {
        display.drawLine(0, 0, display.width()-1, i, WHITE);
        display.display();
        delay(1);
    }
    delay(250);

    display.clearDisplay();
    for (int16_t i=0; i<display.width(); i+=4) {
        display.drawLine(0, display.height()-1, i, 0, WHITE);
        display.display();
        delay(1);
    }
    for (int16_t i=display.height()-1; i>=0; i-=4) {
        display.drawLine(0, display.height()-1, display.width()-1, i, WHITE);
        display.display();
        delay(1);
    }
    delay(250);

    display.clearDisplay();
    for (int16_t i=display.width()-1; i>=0; i-=4) {
        display.drawLine(display.width()-1, display.height()-1, i, 0, WHITE);
        display.display();
        delay(1);
    }
    for (int16_t i=display.height()-1; i>=0; i-=4) {
        display.drawLine(display.width()-1, display.height()-1, 0, i, WHITE);
        display.display();
        delay(1);
    }
    delay(250);

    display.clearDisplay();
    for (int16_t i=0; i<display.height(); i+=4) {
}

```

```

        display.drawLine(display.width()-1, 0, 0, i, WHITE);
        display.display();
        delay(1);
    }
    for (int16_t i=0; i<display.width(); i+=4) {
        display.drawLine(display.width()-1, 0, i, display.height()-1, WHITE);
        display.display();
        delay(1);
    }
    delay(250);
}

void testscrolltext(void) {
    display.setTextSize(2);
    display.setTextColor(WHITE);
    display.setCursor(10,0);
    display.clearDisplay();
    display.println("scroll");
    display.display();
    delay(1);

    display.startscrollright(0x00, 0x0F);
    delay(2000);
    display.stopscroll();
    delay(1000);
    display.startscrollleft(0x00, 0x0F);
    delay(2000);
    display.stopscroll();
    delay(1000);
    display.startscrolldiagright(0x00, 0x07);
    delay(2000);
    display.startscrolldiagleft(0x00, 0x07);
    delay(2000);
    display.stopscroll();
}

```

Program Test RTC DS3231

```

#include <Wire.h>
#include "RTClib.h"

RTC_DS3231 rtc;

char daysOfTheWeek[7][12] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday"};

void setup () {

#ifndef ESP8266
    while (!Serial); // for Leonardo/Micro/Zero
#endif
}

```

```

Serial.begin(9600);

delay(3000); // wait for console opening

if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
    while (1);
}

if (rtc.lostPower()) {
    Serial.println("RTC lost power, lets set the time!");
    // following line sets the RTC to the date & time this sketch was compiled
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    // This line sets the RTC with an explicit date & time, for example to set
    // January 21, 2014 at 3am you would call:
    // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
}
}

void loop () {
    DateTime now = rtc.now();

    Serial.print(now.year(), DEC);
    Serial.print('/');
    Serial.print(now.month(), DEC);
    Serial.print('/');
    Serial.print(now.day(), DEC);
    Serial.print(" (");
    Serial.print(daysOfTheWeek[now.dayOfTheWeek()]);
    Serial.print(") ");
    Serial.print(now.hour(), DEC);
    Serial.print(':');
    Serial.print(now.minute(), DEC);
    Serial.print(':');
    Serial.print(now.second(), DEC);
    Serial.println();

    Serial.print(" since midnight 1/1/1970 = ");
    Serial.print(now.unixtime());
    Serial.print("s = ");
    Serial.print(now.unixtime() / 86400L);
    Serial.println("d");

    // calculate a date which is 7 days and 30 seconds into the future
    DateTime future (now + TimeSpan(7,12,30,6));

    Serial.print(" now + 7d + 30s: ");
    Serial.print(future.year(), DEC);
    Serial.print('/');
    Serial.print(future.month(), DEC);
}

```

```

Serial.print('/');
Serial.print(future.day(), DEC);
Serial.print(' ');
Serial.print(future.hour(), DEC);
Serial.print(':');
Serial.print(future.minute(), DEC);
Serial.print(':');
Serial.print(future.second(), DEC);
Serial.println();

Serial.println();
delay(3000);
}

```

Program gabungan OLED dan RTC DS3231 :

Program ini jika dijalankan akan menampilkan waktu pada LCD OLED.

```

#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#define OLED_RESET 0
Adafruit_SSD1306 display(OLED_RESET);
#define NUMFLAKES 10
#define XPOS 0
#define YPOS 1
#define DELTAY 2

#define LOGO16_GLCD_HEIGHT 16
#define LOGO16_GLCD_WIDTH 16

#include <Wire.h>
#include "RTClib.h"
RTC_DS3231 rtc;
#define detik now.second()
#define menit now.minute()
#define jam now.hour()
#define tanggal now.day()
#define hari daysOfTheWeek[now.dayOfTheWeek()]
#define bulan now.month()
#define tahun now.year()
char daysOfTheWeek[7][12] = {"Minggu", "Senin", "Selasa", "Rabu", "Kamis", "Jum'at",
"Saturday"};
char buff[10];
#if (SSD1306_LCDHEIGHT != 64)
//#error("Height incorrect, please fix Adafruit_SSD1306.h!");
#endif
void setup() {
 // put your setup code here, to run once:
Serial.begin(115200);

```

```

display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
    while (1);
}
if (rtc.lostPower()) {
    Serial.println("RTC lost power, lets set the time!");
    // following line sets the RTC to the date & time this sketch was compiled
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    // This line sets the RTC with an explicit date & time, for example to set
    // January 21, 2014 at 3am you would call:
    // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
}

void loop() {
    // put your main code here, to run repeatedly:
    DateTime now = rtc.now();
    if (jam <10 && menit <10){
        sprintf(buff, "%s 0%d:0%d", hari,jam,menit);
    }
    else if (jam <10 && menit >= 10){
        sprintf(buff, "%s 0%d:%d", hari,jam,menit);
    }
    else if (jam >=10 && menit <10){
        sprintf(buff, "%s %d:0%d", hari,jam,menit);
    }
    else if (jam >=10 && menit >=10){
        sprintf(buff, "%s %d:%d", hari,jam,menit);
    }
    display.clearDisplay();
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.setCursor(0,0);
    display.println(buff);
    display.display();
    delay(2000);
}

```

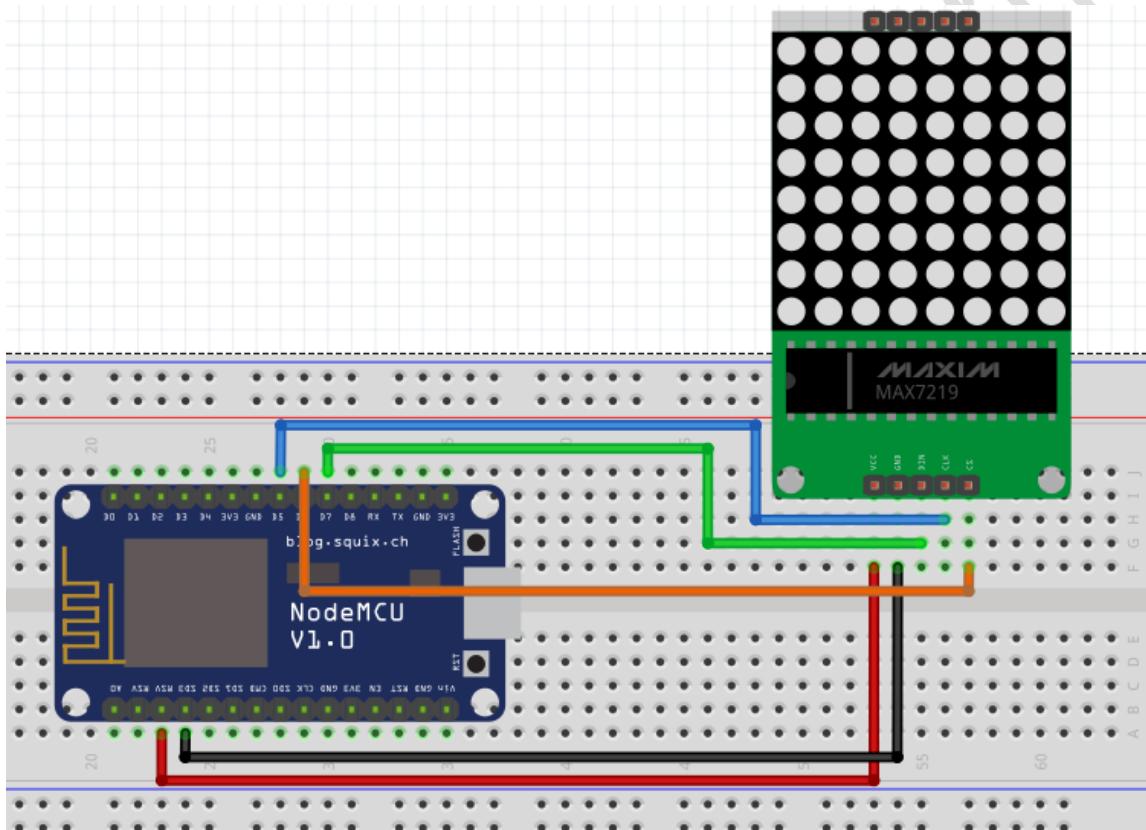
BAB 5

SPI

Program Mengakses Dotmatrix 8x8 Max7219

Berikut adalah skema wiring ESP8266 NodeMCU / Wemos dengan Dot Matrix 8x8 IC Max7219

- D5 ESP8266 >>> CLK Dot Mtrix
- D6 ESP8266 >>> CS Dot Mtrix
- D7 ESP8266 >>> DIN Dot Mtrix
- VIN / VV ESP8266 >>> CLK Dot Mtrix
- GND ESP8266 >>> GND Dot Mtrix



Gambar 11. Rangkaian NodeMCU dan Dot Matrix Max7219

Program ESP8266 NodeMCU / Wemos D1 dengan Dot Matrix Max7219

Download library : https://github.com/nickgammon/MAX7219_Dot_Matrix

: <https://github.com/nickgammon/bitBangedSPI>

```
#include <SPI.h>
#include <bitBangedSPI.h>
#include <MAX7219_Dot_Matrix.h>
const byte chips = 4;

// 12 chips (display modules), hardware SPI with load on D10
MAX7219_Dot_Matrix display(chips, D6, D7, D5); // Chips / LOAD
```

```

char message []= "www.anakkendali.com";

void setup ()
{
  display.begin ();
} // end of setup

unsigned long lastMoved = 0;
unsigned long MOVE_INTERVAL = 1; // mS
int messageOffset;

void updateDisplay ()
{
  display.sendSmooth (message, messageOffset);

  // next time show one pixel onwards
  if (messageOffset++ >= (int) (strlen (message) * 8))
    messageOffset = - chips * 8;
} // end of updateDisplay

void loop ()
{

  // update display if time is up
  if (millis () - lastMoved >= MOVE_INTERVAL)
  {
    updateDisplay ();
    lastMoved = millis ();
  }

  // do other stuff here

} // end of loop

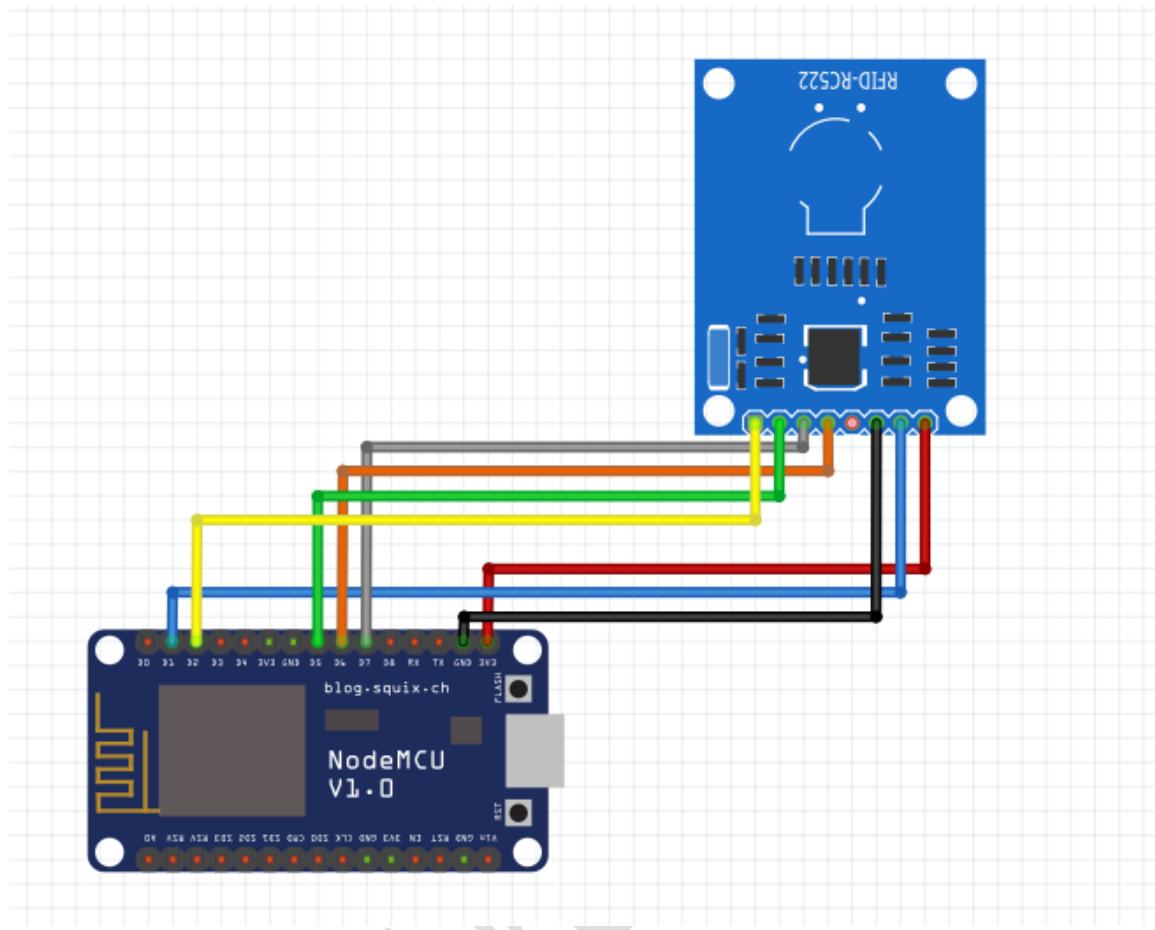
```

Program Mengakses RFID RC522

Berikut adalah skema wiring NodeMCU / Wemos dengan modul RFID RC522

D1 NodemCu----- RST RFID
D2 NodemCu----- SDA/SS RFID
D5 NodemCu----- SCK RFID
D6 NodemCu----- MISO RFID
D7 NodemCu----- MOSI RFID
3V NodemCu----- 3.3V RFID
GND NodemCu---- GND RFID

Untuk lebih jelasnya silahkan lihat gambar skema wiring berikut ini :



Gambar 12. Rangkaian NodeMCU dan RFID RC522

Program NodeMCU / Wemos D1 dengan RFID RC522

```
#include <SPI.h>
#include <MFRC522.h>

#define SS_PIN 4
#define RST_PIN 5

MFRC522 rfid(SS_PIN, RST_PIN);
MFRC522::MIFARE_Key key;

void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);
    SPI.begin();
    rfid.PCD_Init();
    Serial.println("I am waiting for card...");
}

void loop() {
    // put your main code here, to run repeatedly:
    if (!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial())
}
```

```
return;

// Serial.print(F("PICC type: "));
MFRC522::PICC_Type piccType = rfid.PICC_GetType(rfid.uid.sak);
// Serial.println(rfid.PICC_GetTypeName(piccType));

// Check is the PICC of Classic MIFARE type
if (piccType != MFRC522::PICC_TYPE_MIFARE_MINI &&
    piccType != MFRC522::PICC_TYPE_MIFARE_1K &&
    piccType != MFRC522::PICC_TYPE_MIFARE_4K) {
    Serial.println(F("Your tag is not of type MIFARE Classic."));
    return;
}

//id kartu dan yang akan dikirim ke database
String strID = "";
for (byte i = 0; i < 4; i++) {
    strID +=
        (rfid.uid.uidByte[i] < 0x10 ? "0" : "") +
        String(rfid.uid.uidByte[i], HEX) +
        (i != 3 ? ":" : "");
}

strID.toUpperCase();
Serial.print("Tap card key: ");
Serial.println(strID);
delay(1000);
}
```

www.anakkendali.com

BAB 6 SERIAL

Program Mengakses Bluetooth HC-05

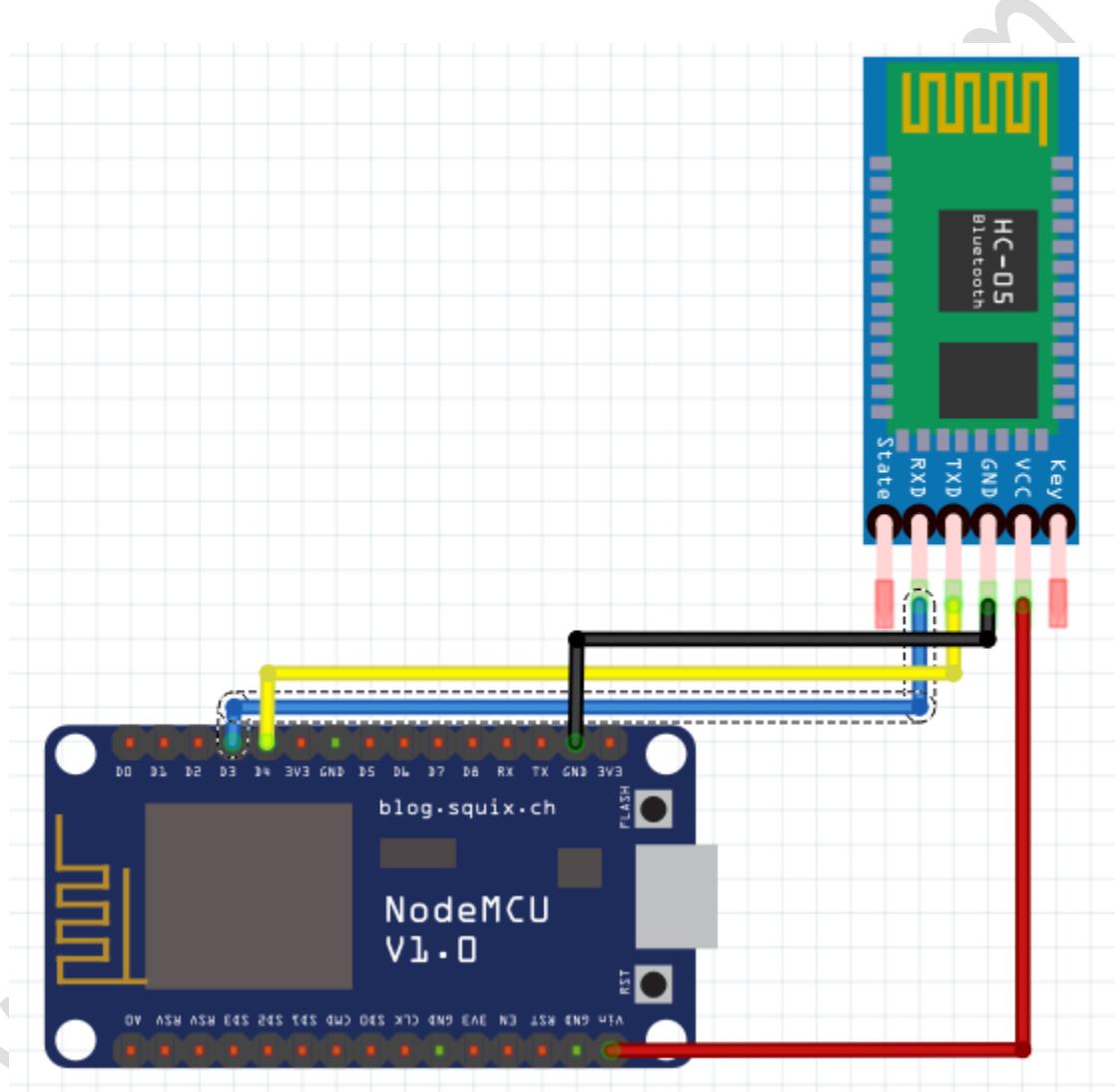
Berikut adalah skema wring NodeMCU / Wemos D1 dengan Bluetooth HC-05

TX Bluetooth ----- D3 NodeMCU

RX Bluetooth ----- D4 NodeMCU

VIN / VV Bluetooth ---- 5V NodeMCU

GND Bluetooth ----- GND NodeMCU



Gambar 13. Rangkaian NodeMCU dan Bluetooth HC-05

Program NodeMCU / Wemos D1 Bluetooth HC-05

```
#include <SoftwareSerial.h>
SoftwareSerial serial(D4, D3); // Rx,Tx
```

```

String data;

void setup(){
    // put your setup code here, to run once:
    serial.begin(9600);
    Serial.begin(9600);
}

void loop() {
    // put your main code here, to run repeatedly:
    while (serial.available()>0){
        delay(10);
        char c = serial.read();
        data+=c;
    }
    if (data.length()>0){
        Serial.println(data);
        data="";
    }
}

```

Program Komunikasi NodeMCU / Wemos D1 dengan Arduino Uno

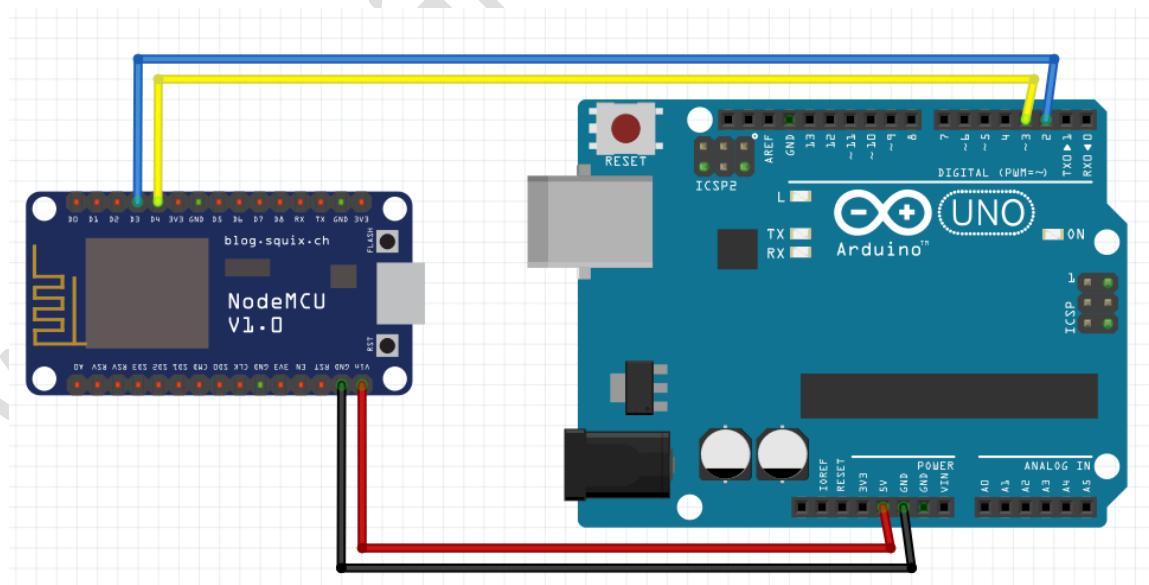
Berikut adalah skema wiring NodeMCU / Wemos dengan Arduino Uno

GND NodeMCU ----- GND Arduino Uno

VIN / VV NodeMCU --- 5V Arduino Uno

D3 NodeMCU ----- 2 Arduino Uno

D4 NodeMCU ----- 3 Arduino



Gambar 14. Rangkaian NodeMCU dan Arduino Uno

Program untuk NodeMCU / Wemos D1 sebagai Master

```
#include <SoftwareSerial.h>
SoftwareSerial serial(D4, D3); // Rx,Tx
```

```

String data;

void setup() {
    // put your setup code here, to run once:
    serial.begin(9600);
    Serial.begin(9600);
}

void loop() {
    // put your main code here, to run repeatedly:
    while (serial.available()>0){
        delay(10);
        char c = serial.read();
        data+=c;
    }
    if (data.length()>0){
        Serial.println(data);
        data="";
    }
}

```

Program untuk Arduino Uno sebagai Slave

```

#include <SoftwareSerial.h>
SoftwareSerial serial(2, 3); // Rx,Tx

String data;

void setup() {
    // put your setup code here, to run once:
    serial.begin(9600);
    Serial.begin(9600);
}

void loop() {
    // put your main code here, to run repeatedly:
    serial.print("Hello ESP8266");
    delay(2000);
}

```

Program diatas jika dijalankan maka NodemCU akan menerima data dari Arduino Uno (“Hello ESP8266”) dan ditampilkan pada Serial Monitor ESP8266.

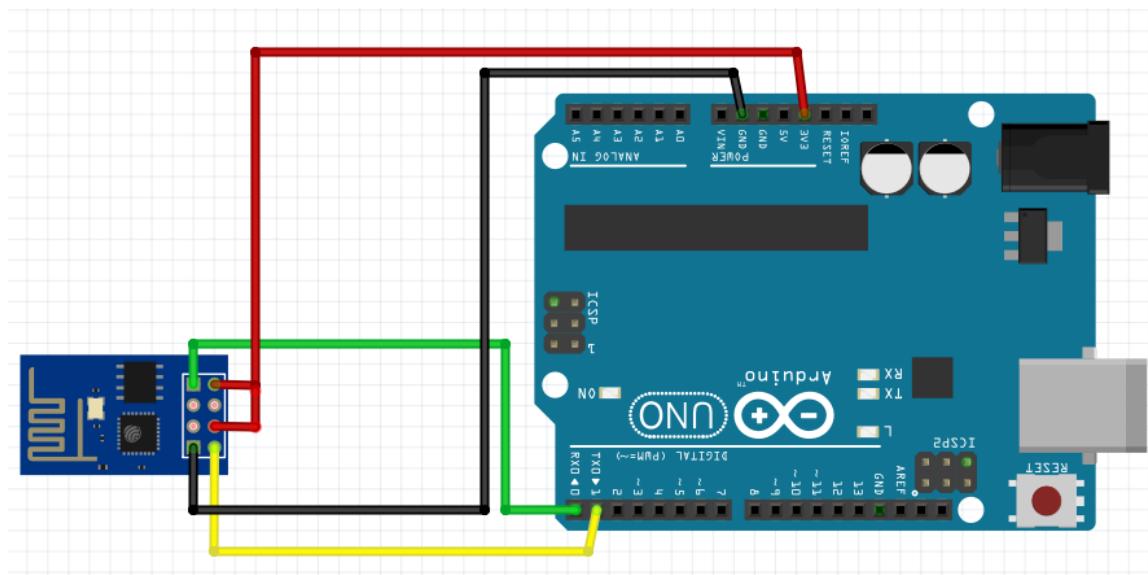
BAB 7

Wifi

Konfigurasi ESP8266 ESP-01 dengan Arduino Uno

Pada ESP8266 ESP-01 bisa di akses dengan Arduino Uno dengan perintah AT-Command, dan biasanya ESP8266 menggunakan Baudrate 115200.
berikut adalah skema wiring ESP8266 ESP-01 dengan Arduino Uno.

3.3V Arduino ----- VCC ESP8266
GND Arduino ----- GND ESP8266
TX Arduino ----- TX ESP8266
RX Arduino ----- RX ESP8266
3.3V Arduino ----- EN ESP8266



Gambar 15. Rangkaian ESP8266 ESP-01 dan Arduino Uno

Silahkan upload program kosong seperti berikut

```
void setup() {  
    // put your setup code here, to run once:  
  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
}
```

Sebelum upload jangan lupa untuk melepas kabel TX dan RX yang terhubung ke ESP8266 ESP-01. Karena jika terpasang maka komunikasi antar Arduino Uno dan Komputer terganggu sehingga tidak bisa meng-Upload program.

Setelah selesai upload kirim perintah AT melalui Serial Monitor, jika keluar respon Ok maka komunikasi antar Arduino Uno dan ESP8266 ESP-01 berhasil, selanjutnya bisa mengirim perintah AT-Command lainnya.

Perintah AT-Command	Respon	Fungsi
AT	Ok	Test AT-Command
AT+RST	Ok ets Jan 8 2013, r st cause:4, boot mode: (3,7) wd़t reset load 0x40100000, len 24444, room 1 6 tail 12 chksm 0xe0 ho 0 tail 12 room 4 load 0x3ffe8000, len 3168, room 12 tail 4 chksm 0x93 load 0x3ffe8c60, len 4956, room 4 tail 8 chksm 0xbd csum 0xbd ready	Reset Modul ESP8266
AT+GMR	Ok 00160901	Versi Firmware
AT+CWMODE=?	+CWMODE:(1-3)	Mengecek List Valid
AT+CWMODE?	+CWMODE:1	Mengecek ESP dalam Mode apa 1. STA (Client) 2. AP (Acces Point) 3. STA + AP
AT+CWMODE= nomer 1-3	Ok	Menentukan Mode ESP (Silahkan gunakan mode 1)
AT+CWLAP	Wifi yang tersedia Ok	Mengecek Wifi yang Tersedia
AT+CWJAP="ssid","pswd"	Wifi Connected Wifi Got Ip	Menghubungkan ESP ke Wifi yang tersedia
AT+UART=9600,8,1,0,0	Ok	Merubah BaudRate ke 9600

Tabel 2. AT-Command dasar

Program Mengirim Data ke Thingspeak ESP8266 ESP-01

Setelah mengikuti langkah konfigurasi ESP8266 ESP-01 dengan Arduino kita mencoba mengirim data Sensor ke Website thingspeak.com

Silahkan kunjungi web thingspeak.com dan buat chanel untuk menerima data yang dikirim dari Arduino dan ESP8266 kemudian catat Api Key nya dan masukan ke dalam program Arduino.

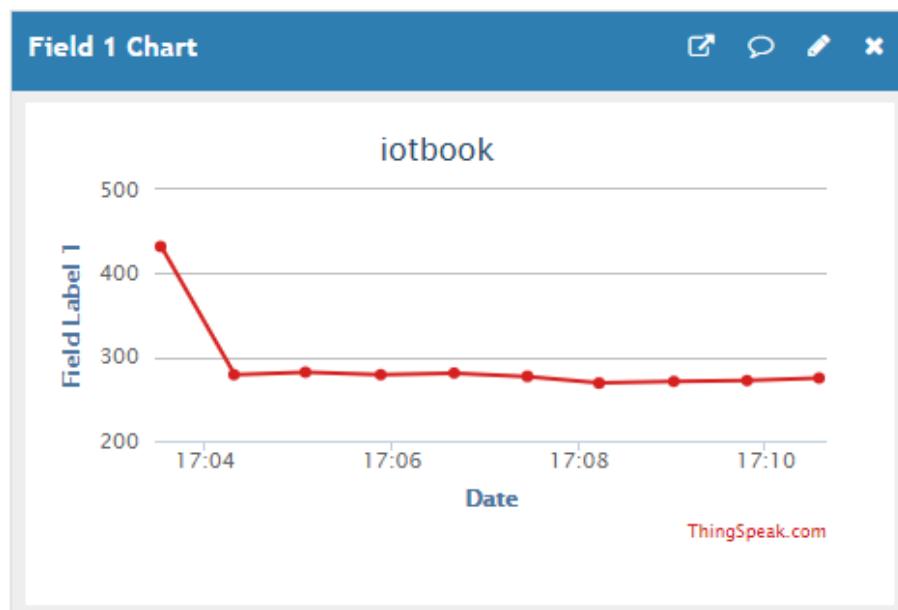
Channel Stats

Created: 18 minutes ago

Updated: 18 minutes ago

Last entry: 6 minutes ago

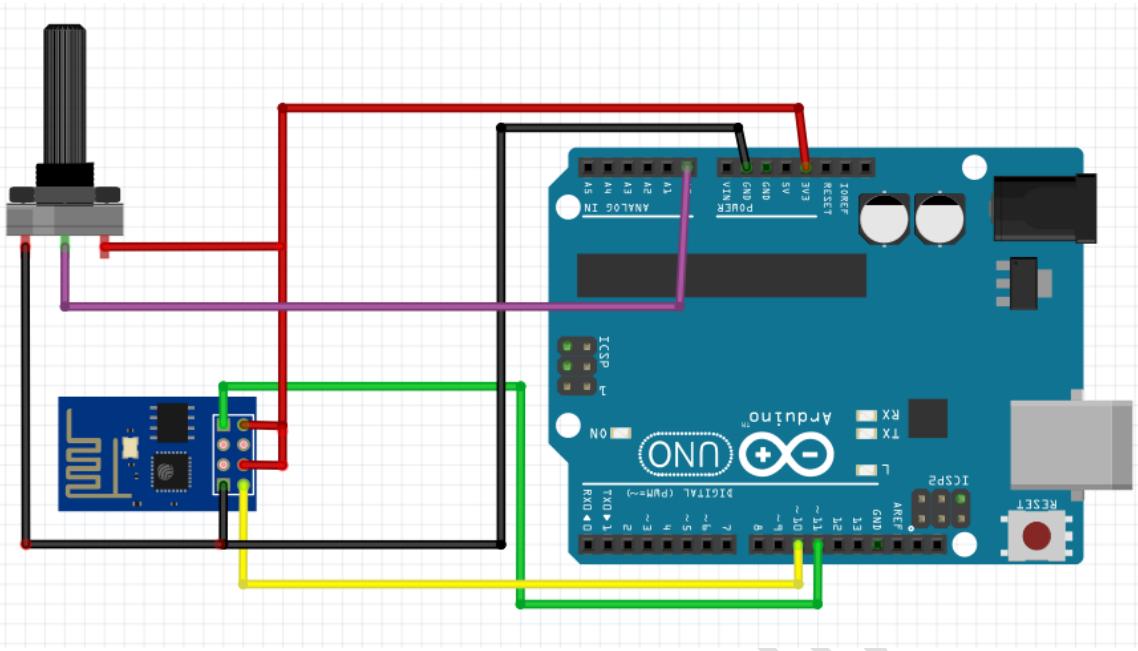
Entries: 1



Gambar 16. Chanel thingspeak.com

Sebelum itu buat terlebih dahulu skema wiring Arduino Uno dan ESP8266 ESP-01

- 3.3V Arduino ----- VCC ESP8266
- GND Arduino ----- GND ESP8266
- 10 Arduino ----- TX ESP8266
- 11 Arduino ----- RX ESP8266
- 3.3V Arduino ----- EN ESP8266



Gambar 17. Rangkaian ESP8266 ESP-01, Arduino Uno Potensiometer

Berikut adalah Program Arduino Uno dan ESP8266 kirim data ke thingspeak.com

```
#include <SoftwareSerial.h> // memasukan library sofwareserial
SoftwareSerial wifi(10,11); // RX, TX

#define WiFiSSID "KOST_RAMA"
#define WiFiPassword "bayardulu500"
#define DestinationIP "184.106.153.149" // ip web thingspeak.com
#define TS_Key "8SHCIPVXZBEQRRY8" // api key dari thingspeak.com buat chanel untuk mendapatkanya

int sensorPin = A0;
int nilai_sensor;
boolean connected=false;

void setup()
{
  wifi.begin(9600);
  wifi.setTimeout(5000);
  Serial.begin(9600);
  Serial.println("ESP8266 Client Demo");
  delay(1000);
  // periksa apakah modul ESP8266 aktif
  wifi.println("AT+RST");
  delay(1000);
  if(wifi.find("OK"))
  {
    Serial.println("Modul siap");
  }
  else
```

```

{
Serial.println("Tidak ada respon dari modul");
while(1);
}
delay(1000);
//setelah modul siap, kita coba koneksi sebanyak 5 kali

for(int i=0;i<5;i++)
{
connect_to_WiFi();
if (connected)
{
break;
}
}
if (!connected){
while(1);
}
delay(5000);

// set the single connection mode
wifi.println("AT+CIPMUX=0");
delay(1000);
}

void loop()
{
String cmd = "AT+CIPSTART=|\"TCP|\",|\"";
cmd += DestinationIP ;
cmd += "|,80";
wifi.println(cmd);
Serial.println(cmd);
if(wifi.find("Error"))
{
Serial.println("Koneksi error.");
return;
}

nilai_sensor = analogRead(sensorPin); // Anda bisa menggantinya dengan nilai analog.
// dalam contoh ini, kita menggunakan nilai random 1...10
cmd = "GET /update?key=";
cmd += TS_Key;
cmd += "&field1=";
cmd += nilai_sensor;
cmd += "|r|\n";    // jangan lupa, setiap perintah selalu diakhiri dengan CR+LF
wifi.print("AT+CIPSEND=");
wifi.println(cmd.length());
if(wifi.find(">"))
{
Serial.print(">");
}
}

```

```

else
{
    wifi.println("AT+CIPCLOSE");
    Serial.println("koneksi timeout");
    delay(1000);
    return;
}
wifi.print(cmd);
delay(2000);

while (wifi.available())
{
    char c = wifi.read();
    Serial.write(c);
    if(c=='r') Serial.print('\n');
}
Serial.println("-----end");
delay(10000);
}

void connect_to_WiFi()
{
    wifi.println("AT+CWMODE=1");
    String cmd="AT+CWJAP=\"";
    cmd+=WiFiSSID;
    cmd+="\",\"";
    cmd+=WiFiPassword;
    cmd+="\"";
    wifi.println(cmd);
    Serial.println(cmd);
    if(wifi.find("OK"))
    {
        Serial.println("Sukses, terkoneksi ke WiFi.");
        connected= true;
    }
    else
    {
        Serial.println("Tidak dapat terkoneksi ke WiFi. ");
        connected= false;
    }
}

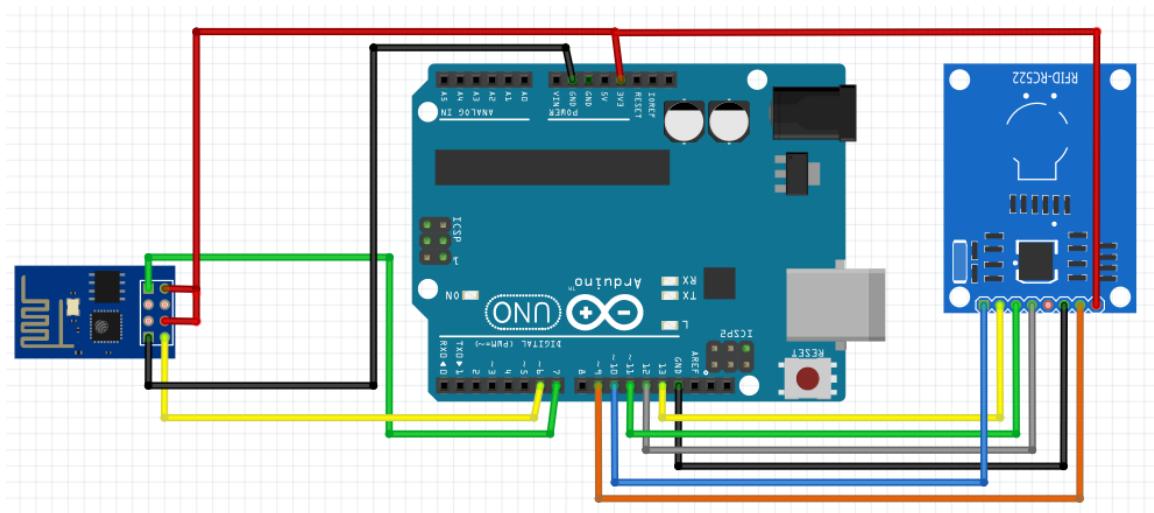
```

Program Mengirim Data ke Database MySQL ESP8266 ESP-01

Yang perlu disiapkan untuk mengirim data Sensor ke Database MySQL adalah sebagai berikut :

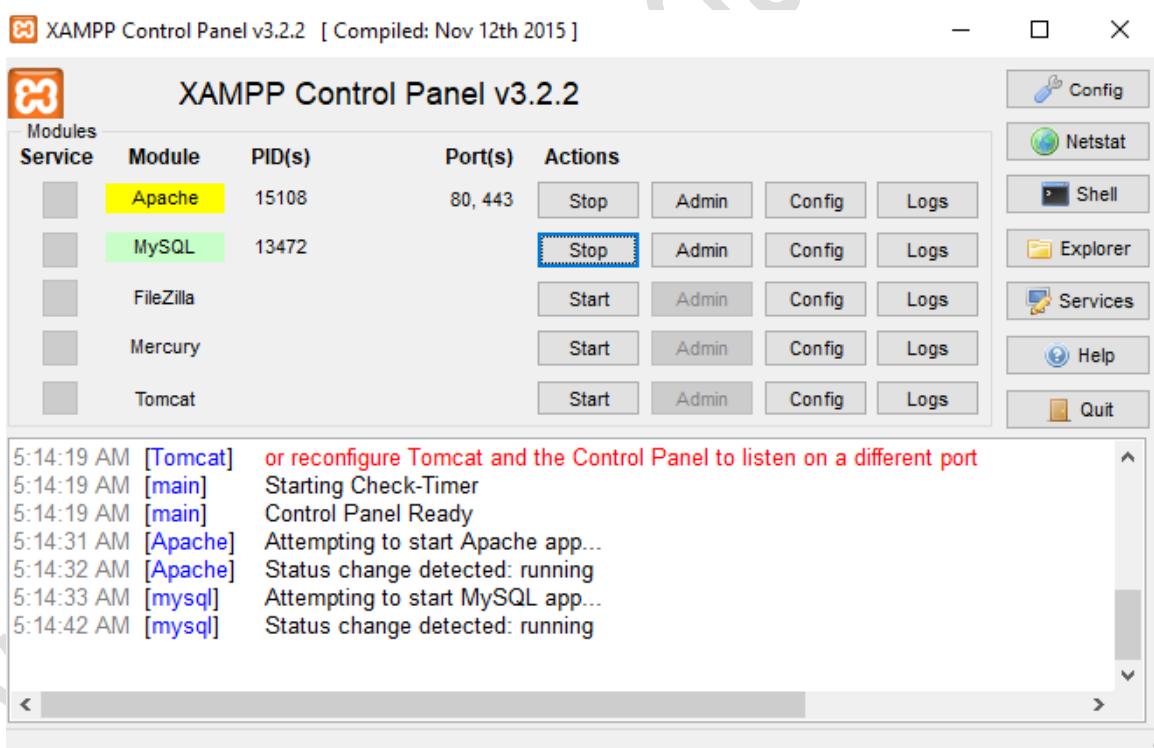
- Editor Text (Notepad, Sublime dll)
- Xampp
- Program PHP [3]
- Database

Untuk skema wiring Arduino Uno dan ESP8266 ESP-01



Gambar 16. Chanel thingspeak.com

Aktifkan Apache dan MySQL pada aplikasi XAMPP



Buat database dan tabelnya pada halaman phpMyAdmin

Tabel: rfid

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	no	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
2	date	datetime			Tidak	CURRENT_TIMESTAMP			Ubah Hapus Lainnya
3	idcard	varchar(20)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
4	val	int(11)			Tidak	Tidak ada			Ubah Hapus Lainnya

Tabel: user

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(11)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Lainnya
2	username	varchar(50)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
3	password	varchar(200)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Lainnya

Buat folder baru bernama esp8266 pada direktori C:\xampp\htdocs. kemudian buat Program PHP dengan nama koneksi.php dan simpan didalam folder esp8266

```
<?php

$servername = "localhost";
$username = "root";
$password = "";
$database = "ebook";

$conn = mysqli_connect($servername, $username, $password, $database);

function query($query){
    global $conn;
    $hasil = mysqli_query($conn, $query);
    $rows = [];
    while( $row = mysqli_fetch_assoc($hasil)){
        $rows[] = $row;
    }
    return $rows;
}
```

```

}

function hapus($no){
    global $conn;
    mysqli_query($conn, "DELETE FROM rfid WHERE no = $no");

    return mysqli_affected_rows($conn);
}

function tambah($data){
    global $conn;
    $idcard = htmlspecialchars($data["idcard"]);
    $value = htmlspecialchars($data["val"]);
    $now = new DateTime();
    $datenow = $now->format("Y-m-d H:i:s");
    $query = "INSERT INTO rfid VALUES(',$datenow','$idcard', '$value')";

    mysqli_query($conn, $query);

    return mysqli_affected_rows($conn);
}

function ubah($data){
    global $conn;

    $no = $data["no"];
    $idcard = htmlspecialchars($data["idcard"]);
    $value = htmlspecialchars($data["val"]);

    $query = "UPDATE rfid SET idcard = '$idcard', val = '$value' WHERE no = $no ";

    mysqli_query($conn, $query);

    return mysqli_affected_rows($conn);
}

function registrasi($data){
    global $conn;

    $username = strtolower(stripslashes( $data["username"]));
    $password = mysqli_real_escape_string($conn, $data["password"]);
    $password2 = mysqli_real_escape_string($conn, $data["password2"]);

    $result = mysqli_query ($conn, "SELECT username FROM user WHERE username =
'$username' ");

    if (mysqli_fetch_assoc ($result) ){
        echo "
<script>
        alert('Username sudah Terdaftar');
    </script>

```

```

        </script>
        ";
        return false;
    }

    if( $password !== $password2){
        echo "
        <script>
        alert('Konfirmasi Password Tidak Sesuai')
        </script>
        ";

        return false;
    }

    $password = password_hash($password, PASSWORD_DEFAULT);
    mysqli_query($conn, "INSERT INTO user VALUES('$username', '$password')");

    return mysqli_affected_rows($conn);
}

function readrfid($query){
    global $conn;
    $hasil = mysqli_query($conn, $query);
    $rows = [];
    while( $row = mysqli_fetch_assoc($hasil)){
        $rows[] = $row;
    }
    return $rows;
}

?>

```

Buat program PHP dengan nama index.php kemudian simpan didalam folder esp8266

```

<?php

session_start();

if (!isset($_SESSION["login"])){
    header("Location: login.php");
    exit;
}

require 'koneksi.php';

$kartu = query("SELECT * FROM rfid");

```

```

?>

<!DOCTYPE html>
<html>
<head>
    <title>Halaman Admin</title>
</head>
<body>

    <a href="logout.php">Logout</a>

    <h1>Daftar ID Card</h1>
    <a href="tambah.php">Tambah ID Kartu Manual</a>

    <table border="1" cellpadding="5" cellspacing="0">

        <tr>
            <th>No</th>
            <th>Date</th>
            <th>Id Card</th>
            <th>Saldo</th>
            <th>Aksi</th>
        </tr>

<?php $i = 1; ?>
<?php foreach ( $kartu as $data ) :{>
} ?>
        <tr>
            <td><?= $i; ?></td>
            <td><?= $data["date"]; ?></td>
            <td><?= $data["idcard"]; ?></td>
            <td><?= $data["val"]; ?></td>
            <td>
                <a href="ubah.php?no=<?= $data["no"]; ?>">Edit</a> |
                <a href="hapus.php?no=<?= $data["no"]; ?>">Hapus</a>
            </td>
        </tr>
<?php $i++; ?>
<?php endforeach; ?>

    </table>

</body>
</html>

```

Buat program PHP dengan nama tambah.php kemudian simpan didalam folder esp8266

```
<?php
```

```

session_start();

if (!isset($_SESSION["login"])){
    header("Location: login.php");
    exit;
}

require 'koneksi.php';

if (isset($_POST["submit"])){
{
    if (tambah ($_POST) > 0)
    {
        echo "
<script>
            alert('Kartu Berhasil di Tambahkan');
            document.location.href = 'index.php';
</script>
        ";
    } else
    {
        echo "
<script>
            alert('Kartu Gagal DI Tambahkan');
            document.location.href = 'tambah.php';
</script>
        ";
    }
}
}

?>

<!DOCTYPE html>
<html>
<head>
    <title>Tambah Kartu ID</title>
    <style>
        label{
            display: block;
        }
    </style>
</head>
<body>

    <h1>Tambah ID Kartu</h1>

    <form action="" method="post">
        <ul>
            <li>

```

```

        <label for ="idcard">ID Card</label>
        <input type ="text" name="idcard" id="idcard" required="">
    </li>

    <li>
        <label for ="val">Value</label>
        <input type="Text" name="val" id="val" required="">
    </li>

    <li>
        <button type="submit" name="submit">Tambah
Kartu</button>
    </li>

</ul>
</form>

</body>
</html>

```

Buat program PHP dengan nama ubah.php kemudian simpan didalam folder esp8266

```

<?php

session_start();

if (!isset($_SESSION["login"])){
    header("Location: login.php");
    exit;
}

require 'koneksi.php';

$no = $_GET["no"];

$skrt = query("SELECT * FROM rfid WHERE no = $no")[0];

if (isset($_POST["submit"]))
{
    if (ubah($_POST) > 0)
    {
        echo "
<script>
    alert('Kartu Berhasil di Ubah');
    document.location.href = 'index.php';
</script>
";
    }
}

```

```

        }
    }
    echo "
<script>
    alert('Kartu Gagal DI Ubah');
    document.location.href = 'ubah.php';
</script>
";
}
}

?>

<!DOCTYPE html>
<html>
<head>
    <title>Ubah Kartu ID</title>
    <style>
        label{
            display: block;
        }
    </style>
</head>
<body>

    <h1>Ubah ID Kartu</h1>

    <form action="" method="post">
        <input type="hidden" name="no" value="<?= $krt["no"]; ?>">
        <ul>
            <li>
                <label for ="idcard">ID Card</label>
                <input type ="text" name="idcard" id="idcard" required="" value="<?= $krt["idcard"]; ?>">
            </li>

            <li>
                <label for ="val">Value</label>
                <input type="Text" name="val" id="val" required="" value="<?= $krt["val"]; ?>">
            </li>

            <li>
                <button type="submit" name="submit">Ubah Kartu</button> <input type="button" value="Cancel" onclick="window.location.href='index.php'" />
            </li>
        </ul>
    </form>

```

```
</body>
</html>
```

Buat program PHP dengan nama hapus.php kemudian simpan didalam folder esp8266

```
<?php

session_start();

if (!isset($_SESSION["login"])){
    header("Location: login.php");
    exit;
}

require 'koneksi.php';

$no = $_GET ["no"];

if (hapus($no)> 0){
    echo "
<script>
    alert('Kartu Berhasil di Hapus');
    document.location.href = 'index.php';
</script>
";
}

else {
    echo "
<script>
    alert('Kartu Gagal di Hapus');
    document.location.href = 'index.php';
</script>
";
}

?>
```

Buat program PHP dengan nama login.php kemudian simpan didalam folder esp8266

```
<?php

session_start();

if (isset($_SESSION["login"])){
    header("Location: index.php");
    exit;
}
```

```

require 'koneksi.php';

if (isset($_POST["login"])){
    $username = $_POST["username"];
    $password = $_POST["password"];

    $hasil = mysqli_query($conn, "SELECT * FROM user WHERE username = '$username'");
};

    if (mysqli_num_rows($hasil) === 1){

        $row = mysqli_fetch_assoc($hasil);
        if (password_verify($password, $row["password"])){

            $_SESSION["login"] = true;

            header("Location: index.php");
            exit;
        }
    }

    $error = true;
}

?>

<!DOCTYPE html>
<html>
<head>
    <title>Halaman Login</title>

    <style>
        label{
            display: block;
        }
    </style>
</head>
<body>

    <h1>Halaman Login</h1>

    <?php if (isset($error)): ?>
        <p style="color : red; font-style: italic">Username / Password Salah</p>
    <?php endif; ?>

    <form action="" method="post">

        <ul>
            <li>

```

```

        <label for="username">Username</label>
        <input type="text" name="username" id="username" required="">
    </li>

    <li>
        <label for="password">Password</label>
        <input type="password" name="password" id="password"
required="">
    </li>

        <button type="submit" name="login">Login</button>
        <input type="button" value="Sign Up"
onclick="window.location.href='registrasi.php'" />
    </ul>

</form>

</body>
</html>
```

Buat program PHP dengan nama registrasi.php kemudian simpan didalam folder esp8266

```

<?php

require 'koneksi.php';

if (isset($_POST["register"])) {

    if (registrasi($_POST) > 0) {
        echo "
        <script>
            alert('User Baru Berhasil di Tambahkan');
        </script>
        ";
    } else
    {
        echo mysqli_error($conn);
    }
}

?>

<!DOCTYPE html>
<html>
<head>
    <title>Halaman Registrasi</title>
    <style>
```

```

        label{
            display: block;
        }
    </style>

</head>
<body>

<h1> Halaman Registrasi</h1>

<form action="" method="post">

    <ul>
        <li>
            <label for="username">Username :</label>
            <input type="text" name="username" id="username" required="">
        </li>

        <li>
            <label for="password">Password :</label>
            <input type="password" name="password" id="password"
required="">
        </li>

        <li>
            <label for="password2">Konfirmasi Password</label>
            <input type="password" name="password2" id="password2"
required="">
        </li>

        <li>
            <button type="submit" name="register">Register</button>
            <input type="button" value="Login"
onclick="window.location.href='login.php'" />
        </li>
    </ul>

</form>

</body>
</html>

```

Buat program PHP dengan nama logout.php kemudian simpan didalam folder esp8266

```

<?php

session_start();
$_SESSION = [];

```

```

session_unset();
session_destroy();

header("Location: login.php");
exit;

?>

```

Buat program PHP dengan nama rfidadd.php kemudian simpan didalam folder esp8266

```

<?php

require 'koneksi.php';

$now = new DateTime();

$rfid = $_GET['idcard'];
$value = $_GET['val'];

$datenow = $now->format("Y-m-d H:i:s");

$sql = "INSERT INTO rfid VALUES ('','$datenow', '$rfid', '$value')";

$result = mysqli_query($conn, $sql);
if (!$result) {
    die('Invalid query: ');
}
echo "<h1>THE DATA HAS BEEN SENT!!</h1>";
mysqli_close($conn);

?>

```

Buat program PHP dengan nama rfidread.php kemudian simpan didalam folder esp8266

```

<?php

require 'koneksi.php';

$idcard = $_GET["idcard"];

$hasil = mysqli_query($conn, "SELECT * FROM rfid WHERE idcard = '$idcard' ");

$row = mysqli_fetch_array($hasil);

echo "ID CARD :", $row["idcard"],      ", "Saldo :", $row["val"];

?>

```

Program Arduino Uno, RFID dan ESP8266 ESP-01 Kirim data RFID ke Database

```
/*
 * visit www.anakkendali.com
 * 2018
 *
 */
#include <SPI.h>
#include <MFRC522.h>

#define SS_PIN 10
#define RST_PIN 9

MFRC522 rfid(SS_PIN, RST_PIN);
MFRC522::MIFARE_Key key;

#include <SoftwareSerial.h> // memasukan library sofwareserial
SoftwareSerial wifi(6,7 ); // RX, TX

#define WiFiSSID "KOST_RAMA"
#define WiFiPassword "bayardulu500"
#define DestinationIP "192.168.1.9" // ip komputer yang terinstall xampp

boolean connected=false;

void setup()
{
  wifi.begin(9600);
  wifi.setTimeout(5000);
  Serial.begin(9600);
  Serial.println("ESP8266 Client Demo");
  delay(1000);
  // periksa apakah modul ESP8266 aktif
  wifi.println("AT+RST");
  delay(1000);
  if(wifi.find("ready"))
  {
    Serial.println("Modul siap");
  }
  else
  {
    Serial.println("Tidak ada respon dari modul");
    while(1);
  }
  delay(1000);
  //setelah modul siap, kita coba koneksi sebanyak 5 kali

  for(int i=0;i<5;i++)
  {
    connect_to_WiFi();
    if (connected)
    {
```

```

break;
}
}
if (!connected){
while(1);
}
delay(5000);

// set the single connection mode
wifi.println("AT+CIPMUX=0");
delay(1000);

SPI.begin();
rfid.PCD_Init();
Serial.println("I am waiting for card...");
}

void loop()
{
if (!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial())
    return;

// Serial.print(F("PICC type: "));
MFRC522::PICC_Type piccType = rfid.PICC_GetType(rfid.uid.sak);
// Serial.println(rfid.PICC_GetTypeName(piccType));

// Check is the PICC of Classic MIFARE type
if (piccType != MFRC522::PICC_TYPE_MIFARE_MINI &&
    piccType != MFRC522::PICC_TYPE_MIFARE_1K &&
    piccType != MFRC522::PICC_TYPE_MIFARE_4K) {
    Serial.println(F("Your tag is not of type MIFARE Classic."));
    return;
}

//id kartu dan yang akan dikirim ke database
String strID = "";
for (byte i = 0; i < 4; i++) {
    strID +=
        (rfid.uid.uidByte[i] < 0x10 ? "0" : "") +
        String(rfid.uid.uidByte[i], HEX) +
        (i != 3 ? ":" : "");
}

strID.toUpperCase();
Serial.print("Tap card key: ");
Serial.println(strID);
String Val = "5000"; // misalkan nilai val constant
kirim_data(strID, Val);
delay(1000);
}

```

```

void kirim_data(String a, String b){

    String cmd = "AT+CIPSTART=\"TCP\",\"";
    cmd += DestinationIP ;
    cmd += "\",80";
    wifi.println(cmd);
    Serial.println(cmd);
    if(wifi.find("Error"))
    {
        Serial.println("Koneksi error.");
        return;
    }
    cmd = "GET /esp8266/rfidadd.php?idcard=";
    cmd += a;
    cmd += "&val=";
    cmd += b;
    cmd += "|r|\n";    // jangan lupa, setiap perintah selalu diakhiri dengan CR+LF
    wifi.print("AT+CIPSEND=");
    wifi.println(cmd.length());
    if(wifi.find(">"))
    {
        Serial.print(">");
    }
    else
    {
        wifi.println("AT+CIPCLOSE");
        Serial.println("koneksi timeout");
        delay(1000);
        return;
    }
    wifi.print(cmd);
    delay(2000);

    while (wifi.available())
    {
        char c = wifi.read();
        Serial.write(c);
        if(c=='r') Serial.print('\n');
    }
    Serial.println("-----end");
    delay(10000);
}

void connect_to_WiFi()
{
    wifi.println("AT+CWMODE=1");
    String cmd="AT+CWJAP=\"";
    cmd+=WiFiSSID;
    cmd+="\",\"";
    cmd+=WiFiPassword;
    cmd+="\"\";

}

```

```

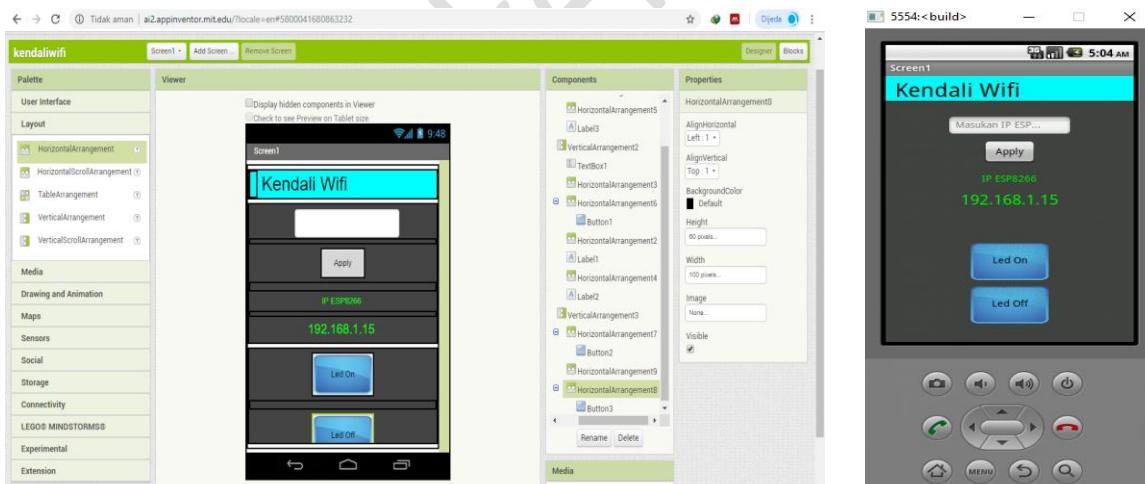
wifi.println(cmd);
Serial.println(cmd);
if(wifi.find("OK"))
{
Serial.println("Sukses, terkoneksi ke WiFi.");
connected= true;
}
else
{
Serial.println("Tidak dapat terkoneksi ke WiFi. ");
connected= false;
}
}

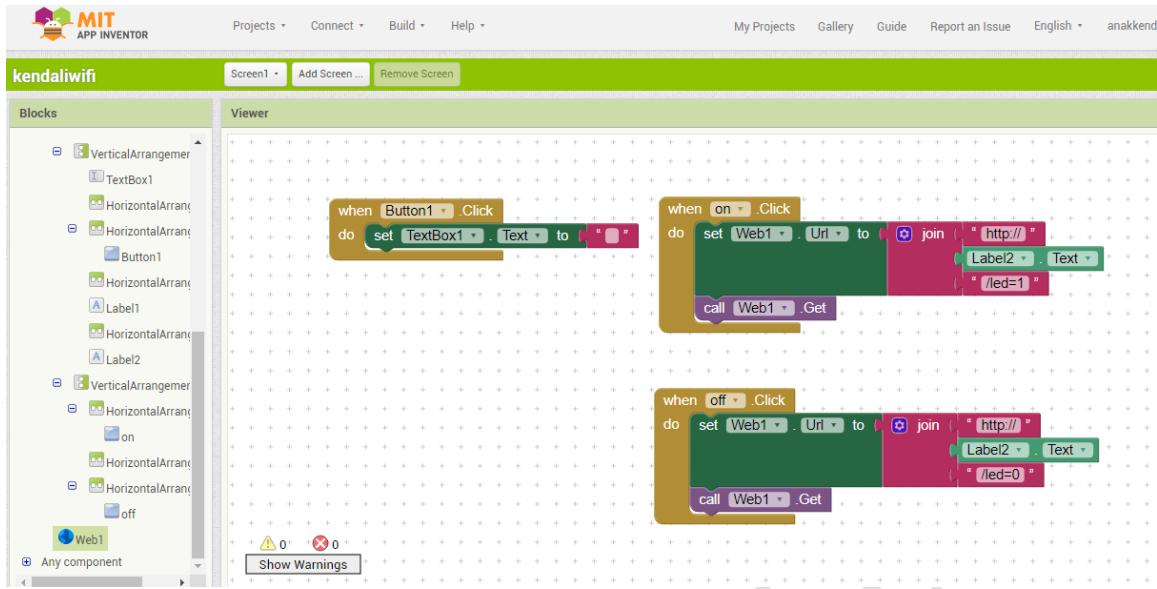
```

Kontrol Relay/LED Arduino Uno dan ESP8266 ESP-01 Alikasi Android

Kontrol relay dengan Arduino dan ESP8266 di lakukan melalui Aplikasi Android. Pertama buatlah aplikasi android menggunakan App Inventor yang bisa di kunjungi alamat webnya <http://ai2.appinventor.mit.edu>.

Berikut adalah contoh aplikasi Android yang di buat dengan app inventor, project nya juga bisa di download pada alamat <http://safelinku.net/2jyAz93N4>





Dari contoh aplikasi di atas silahkan untuk di kembangkan sendiri, karena buku ini tidak membahas lebih lanjut mengenai pembuatan aplikasi android.

Berikut program Arduino Uno + ESP8266 ESP-01 Kontrol Relay Aplikasi Android

```
/*
*
* www.anakkendali.com
*
* 2018 Indonesia
*/

#include <SoftwareSerial.h> // memasukan library sofwareserial
SoftwareSerial Serial1(6,7); // RX, TX

int index1, index2;
String firstdata, seconddata;

#define SSID_WIFI "KOST_RAMA"
#define PASSWORD_WIFI "bayardulu500"

#define DEBUG true

void konek_ke_wifi(){
    String data="AT+CWJAP=""";
    data+=SSID_WIFI;
    data+=",";;
    data+=PASSWORD_WIFI;
    data+="";
    kirimPerintah(data,10000, DEBUG);
}
```

```

void setup() {
    // put your setup code here, to run once:
    Serial1.begin(9600);
    Serial.begin(9600);
    kirimPerintah("AT+RST|r|n", 2000, DEBUG);
    kirimPerintah("AT+CWMODE=1|r|n", 1000, DEBUG); // Set ke AP / Client
    konek_ke_wifi();
    kirimPerintah("AT+CIFSR|r|n", 500, DEBUG);
    kirimPerintah("AT+CIPMUX=1|r|n", 500, DEBUG);
    kirimPerintah("AT+CIPSERVER=1,80|r|n", 500, DEBUG);
    Serial.println("Arduino Kontrol Wifi SIAP !!!");

    pinMode(13, OUTPUT);
}

String kirimPerintah(String perintah, const int waktu, boolean debug){

    String respon ="";
    Serial1.print(perintah);

    long time = millis();
    while ((time+waktu)>millis())
    {
        while (Serial1.available())
        {
            char c = Serial1.read();
            respon+=c;
        }
    }

    if (debug){
        Serial.print(respon);
    }
    return respon;
}

String data;

void loop() {
    // put your main code here, to run repeatedly:
    while (Serial1.available()>0){
        delay(10);
        char c = Serial1.read();
        data+=c;
    }
    if (data.length()>0){
        // Serial.println(data);

        index1 = data.indexOf('/');
        index2 = data.indexOf(' ', index1+1);

```

```

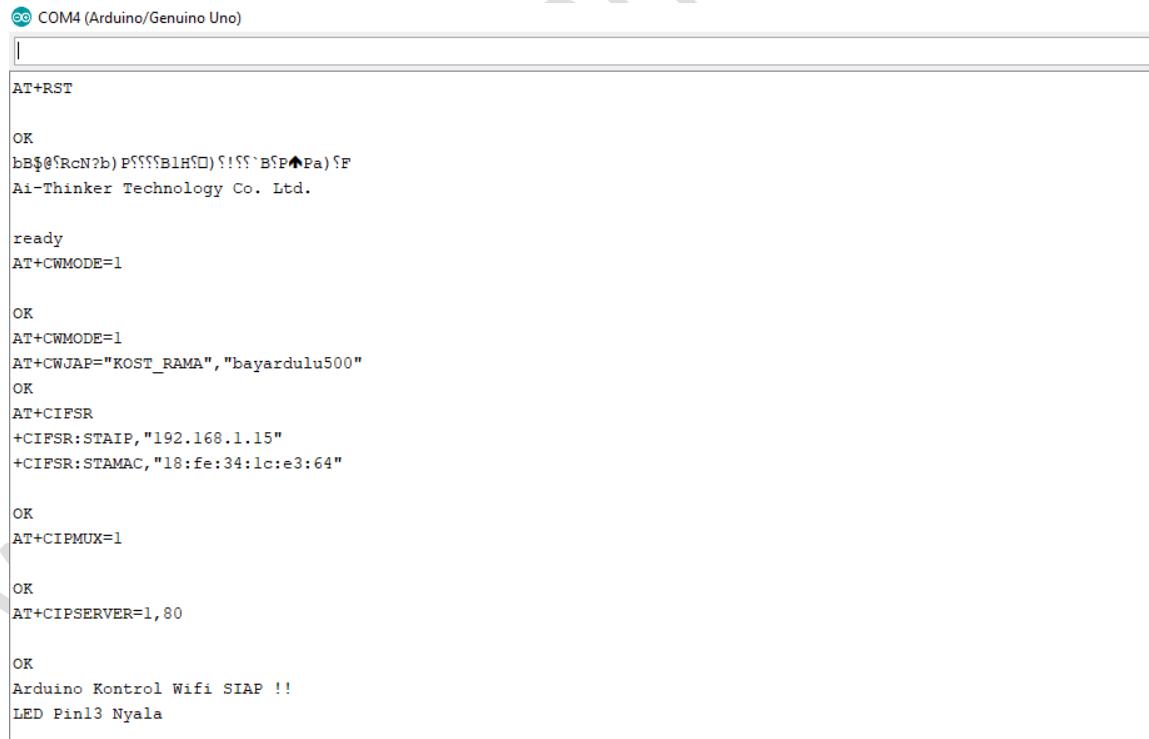
// firstdata = data.substring(index1);
seconddata = data.substring(index1+1, index2);

// Serial.print("first data: ");
// Serial.println(firstdata);
// Serial.print("second data: ");
// Serial.println(seconddata);
if (seconddata == "led=1"){
    Serial.println("LED Pin13 Nyala");
    digitalWrite(13, 1);
}
if (seconddata == "led=0"){
    Serial.println("LED Pin13 Mati");
    digitalWrite(13, 0);
}

data="";
}
}

```

Program di atas jika di lihat pada Serial Monitor Arduino hasilnya seperti gambar berikut :



```

COM4 (Arduino/Genuino Uno)

AT+RST
OK
bB$@RcN?b) PFFFFB1HfO) f!f`BfP▲Pa) fF
Ai-Thinker Technology Co. Ltd.

ready
AT+CWMODE=1

OK
AT+CWMODE=1
AT+CWJAP="KOST_RAMA", "bayardulu500"
OK
AT+CIFSR
+CIFSR:STAIP,"192.168.1.15"
+CIFSR:STAMAC,"18:fe:34:1c:e3:64"

OK
AT+CIPMUX=1

OK
AT+CIPSERVER=1, 80

OK
Arduino Kontrol Wifi SIAP !!
LED Pin13 Nyala

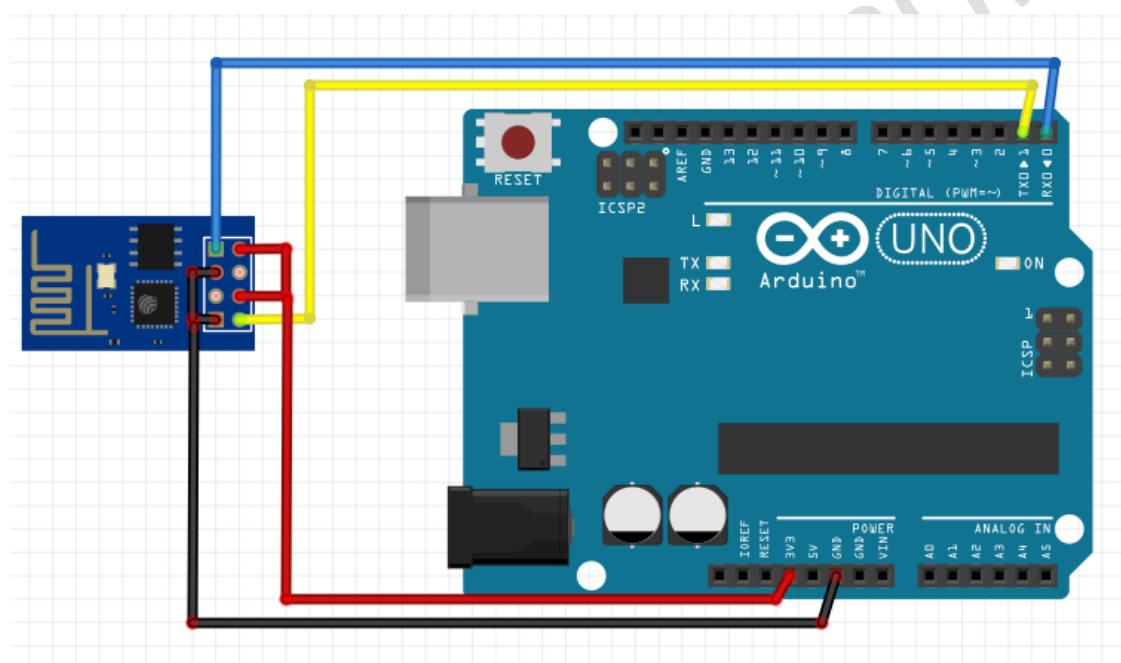
```

Kontrol Relay/LED Arduino Uno dan ESP8266 ESP-01 dengan Telegram

Pada tutorial ini akan membuat masing-masing program untuk Arduino Uno dan juga ESP8266 ESP-01, jadi yang dihubungkan dengan Telegram adalah ESP-01 kemudian dari ESP-01 mengirim data ke Arduino Uno untuk melakukan suatu perintah tertentu (mengontrol LED pada pin 13).

Buatlah skema rangkaian seperti berikut ini:

3.3V Arduino ----- VCC ESP8266
GND Arduino ----- GND ESP8266
TX Arduino ----- TX ESP8266
RX Arduino ----- RX ESP8266
3.3V Arduino ----- EN ESP8266
GND Arduino ----- GPIO0 ESP8266



Rangkaian untuk Upload Program ke ESP-01

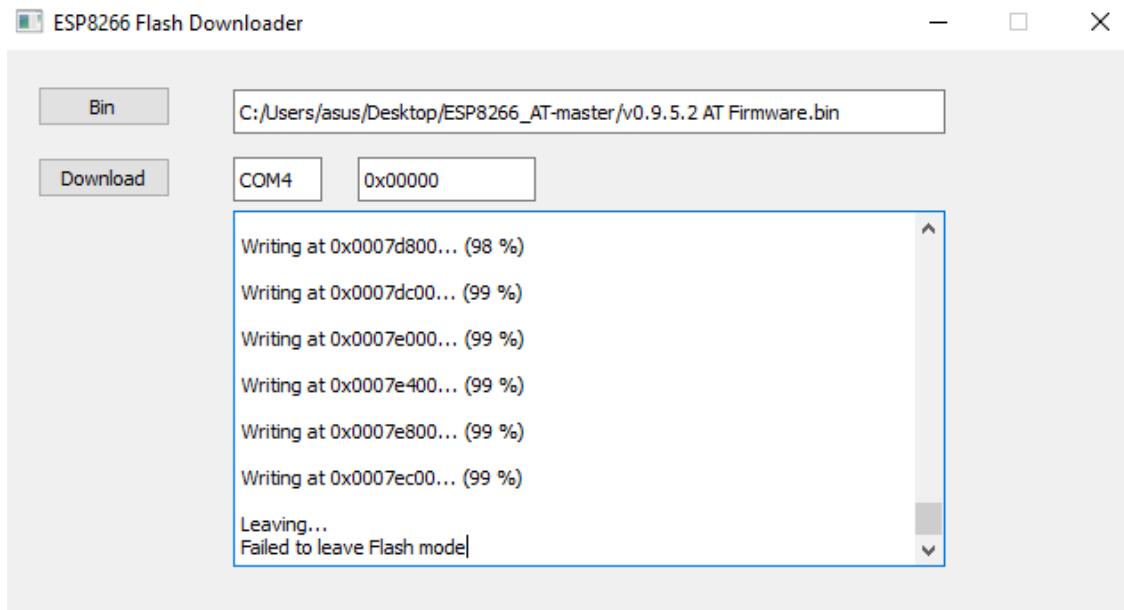
Download File pada alamat <http://idsly.org/BUGB>

File tersebut berfungsi untuk melakukan Flash Firmware ESP8266.

Perlu diingat saat kita menggunakan Arduino Uno sebagai programmer ESP8266 maka terlebih dahulu Arduino Uno harus di isi program Kosong. Lihat tutorial sebelumnya.

Silahkan lakukan Re-Flash pada ESP8266 ESP-01. Cara nya cukup mudah, sesuaikan COM PORT yang terdeteksi kemudian masukan File Firmware nya. dan klik Download.

Jika tampil seperti gambar berikut berarti ESP8266 berhasil di program, dan kita bisa mengisi proram pada ESP-01 menggunakan Arduino Uno.



ESP8266 ESP-01 berhasil di isi Firmware

Jika gagal atau muncul failed, cobalah untuk mencabut USB Arduino kemudian menghubungkanya kembali.

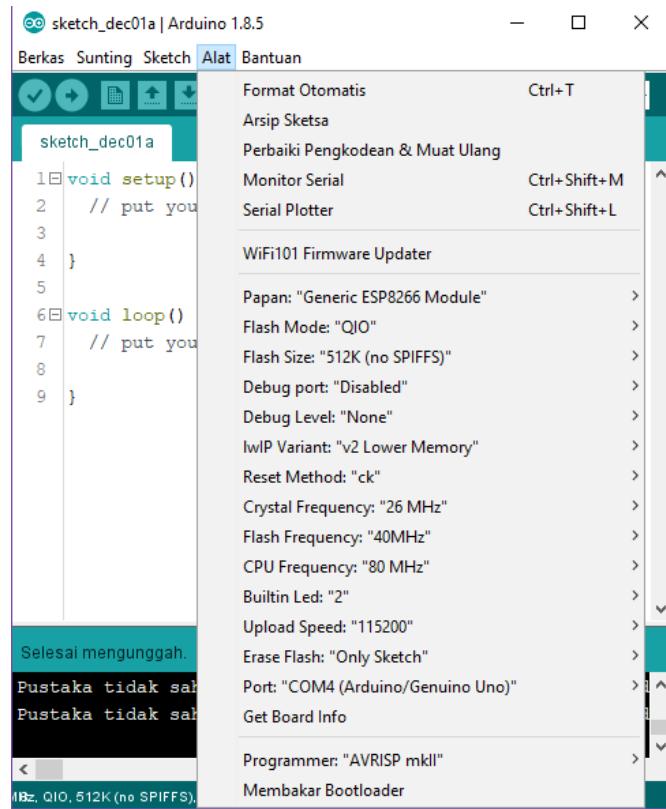
Mengisi Firmware tujuanya untuk mereset Konfigurasi ESP-01, dan jika belum bisa juga untuk reFlash ESP8266 silahkan coba masuk ke mode AT-Command seperti tutorial Sebelumnya dan berikan perintah
“AT+RESTORE”

Berikut Library Telegram ESP8266 yang perlu di pasang pada Arduino IDE

ArduinoJson : <http://idsly.bid/TZ8I>

Universal Telegram Both : <http://idsly.bid/LcNxWs>

Pasanglah kedua library yang sudah di download kemudian rubahlah pilihan board pada tool arduino menjadi **Generic ESP8266** lihat gambar berikut :



Kemudian silahkan upload program untuk menghubungkan esp8266 dengan TelegramBot :

```
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>

// Initialize Wifi connection to the router
char ssid[] = "KOST_RAMA"; // your network SSID (name)
char password[] = "bayardulu500"; // your network key

// Initialize Telegram BOT
#define BOTtoken "518247654:AAFnJ3bwQoixeX4nDDLb-JdmA1vPsD85gRo" // your Bot Token (Get from Botfather)

WiFiClientSecure client;
UniversalTelegramBot bot(BOTtoken, client);

int Bot_mtbs = 1000; //mean time between scan messages
long Bot_lasttime; //last time messages' scan has been done
bool Start = false;

const int ledPin = 13;
int ledStatus = 0;

void handleNewMessages(int numNewMessages) {
```

```

for (int i=0; i<numNewMessages; i++) {
    String chat_id = String(bot.messages[i].chat_id);
    String text = bot.messages[i].text;

    String from_name = bot.messages[i].from_name;
    if (from_name == "") from_name = "Guest";
    Serial.print(text);
}
}

void setup() {
Serial.begin(9600);

// Set WiFi to station mode and disconnect from an AP if it was Previously
// connected
WiFi.mode(WIFI_STA);
WiFi.disconnect();
delay(100);

// attempt to connect to Wifi network:
Serial.print("Connecting Wifi: ");
Serial.println(ssid);
WiFi.begin(ssid, password);

while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
}

Serial.println("");
Serial.println("WiFi connected");
Serial.print("IP address: ");
Serial.println(WiFi.localIP());

}

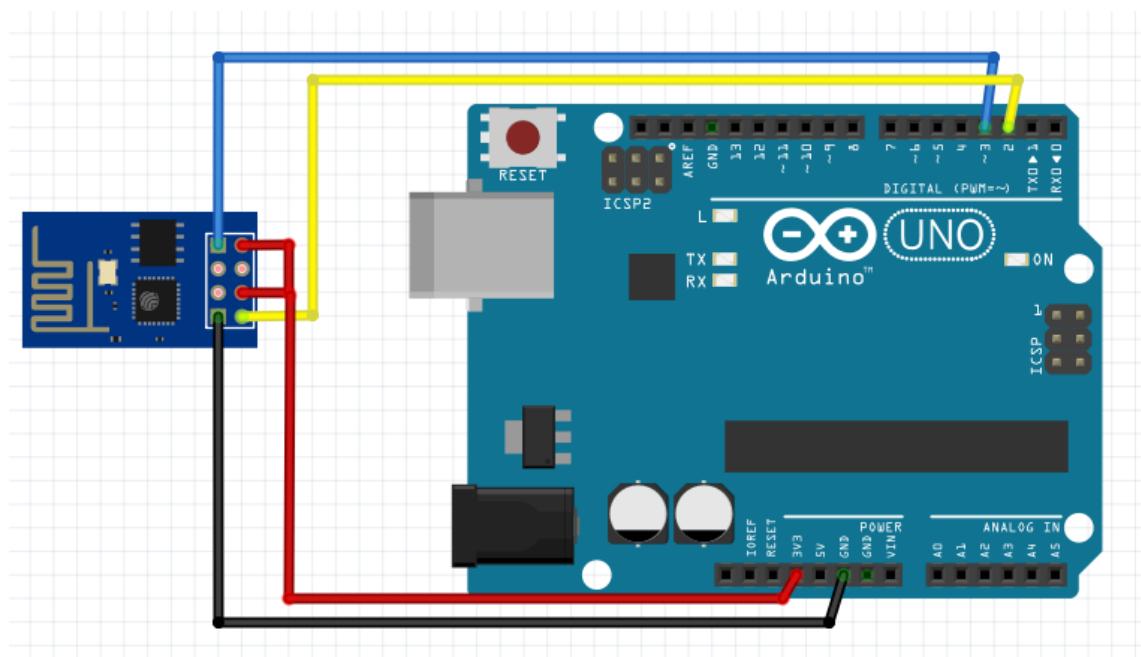
void loop() {
if (millis() > Bot_lasttime + Bot_mtbs) {
    int numNewMessages = bot.getUpdates(bot.last_message_received + 1);

    while(numNewMessages) {
        handleNewMessages(numNewMessages);
        numNewMessages = bot.getUpdates(bot.last_message_received + 1);
    }

    Bot_lasttime = millis();
}
}

```

Selanjutnya kita akan mengisi program pada Arduino Uno itu sendiri, dan rubahlah Sekma wiring nya menjadi seperti pada gambar *Arduino Uno membaca data ESP-01*



Gambar *Arduino Uno membaca data ESP-01*

Kemudian upload program berikut ke arduino Uno untuk membaca data yang masuk dari Telegram ke ESP-01 kemudian di Arduino Uno kita bisa membuat perintah seperti menyalaikan LED atau menyalaikan Relay dari data tersebut

```
#include <SoftwareSerial.h>
SoftwareSerial serial(2,3);

String data;
char c;

void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);
    serial.begin(9600);

    pinMode(13, OUTPUT);
}

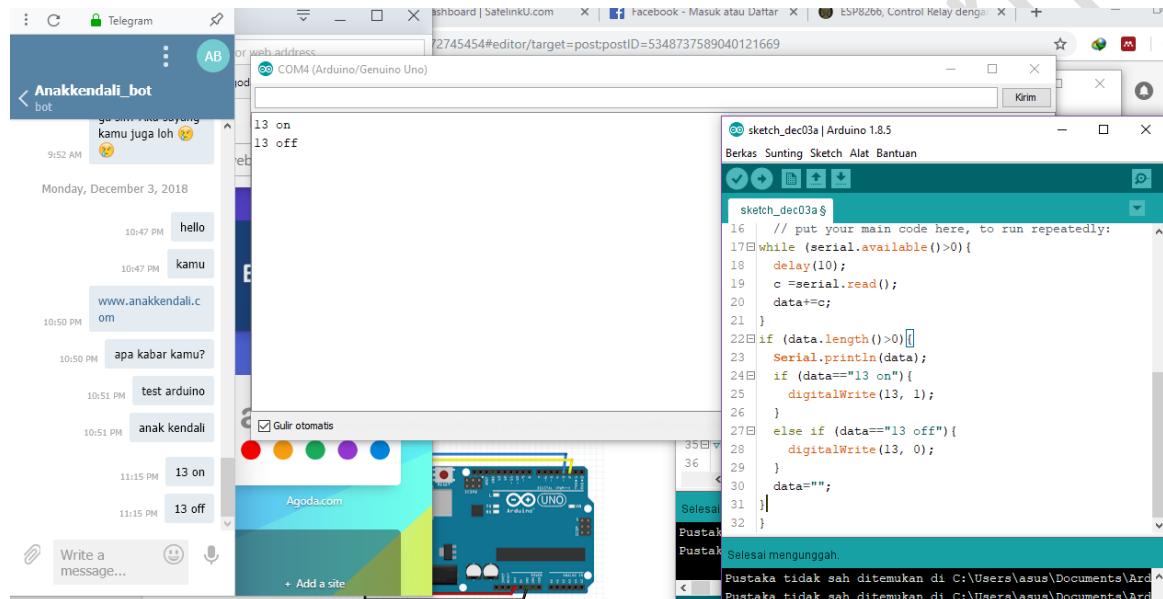
void loop() {
    // put your main code here, to run repeatedly:
    while (serial.available()>0){
        delay(10);
        c =serial.read();
        data+=c;
    }
    if (data.length()>0){
        Serial.println(data);
    }
}
```

```

if (data=="13 on"){
    digitalWrite(13, 1);
}
else if (data=="13 off"){
    digitalWrite(13, 0);
}
data="";
}
}

```

Berikut adalah gambar hasil dari kedua program diatas.

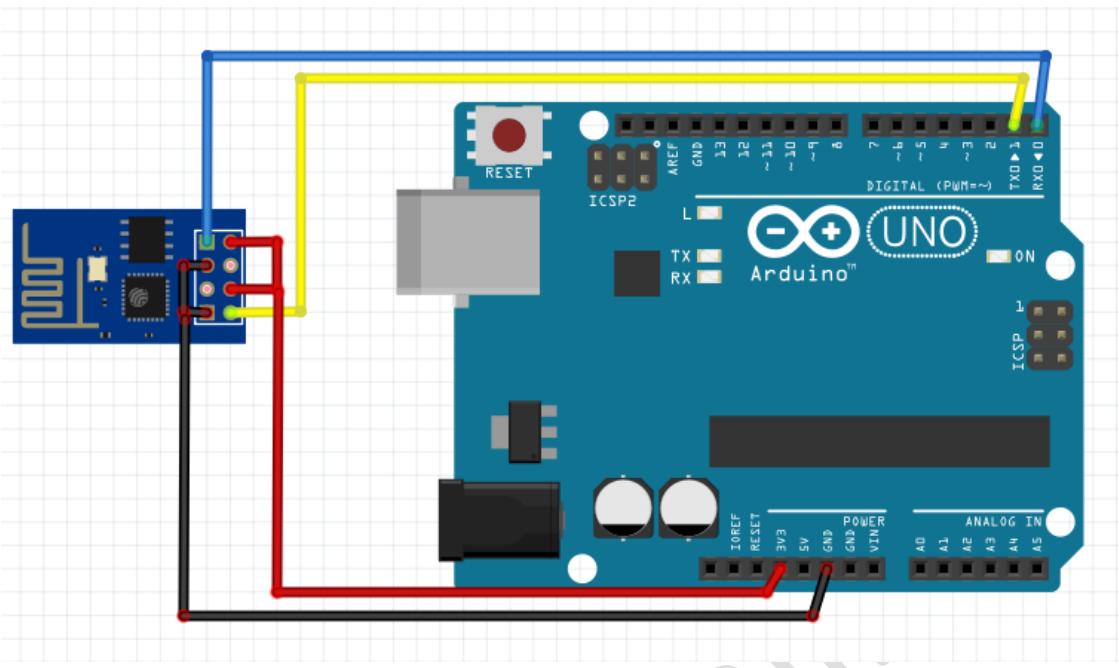


Menghubungkan ESP8266 ESP-01 ke Cayenne

Cayenne[4] merupakan website yang menyediakan layanan IOT sama hal nya seperti Thingspeak, tidak hanya untuk ESP8266 tapi juga kita bisa menggunakan Raspberry Pi, Lora, dan masih banyak yang lainnya.

untuk dapat menggunakan layanan Cayenne pertama yang harus dilakukan adalah membuat Akun terlebih dahulu. Silahkan mendaftarkan akun dengan mengunjungi website resminya di <https://mydevices.com/cayenne/signup/>

karena menggunakan ESP8266 mode program dan Arduino Uno sebagai Programmer nya maka rangkaianya seperti pada gambar dibawah ini. Dan jangan lupa Arduino Uno tidak boleh menyimpan program (Kosong). Lihat tutorial sebelumnya.



Setelah itu silahkan Download librarynya [Disini!](#)

install library Cayene ESP8266 ke Arduino IDE kemudian buka **File - Example – Cayenne MQTT ESP – ESP8266**

akan tampak program seperti pada gambar di bawah ini

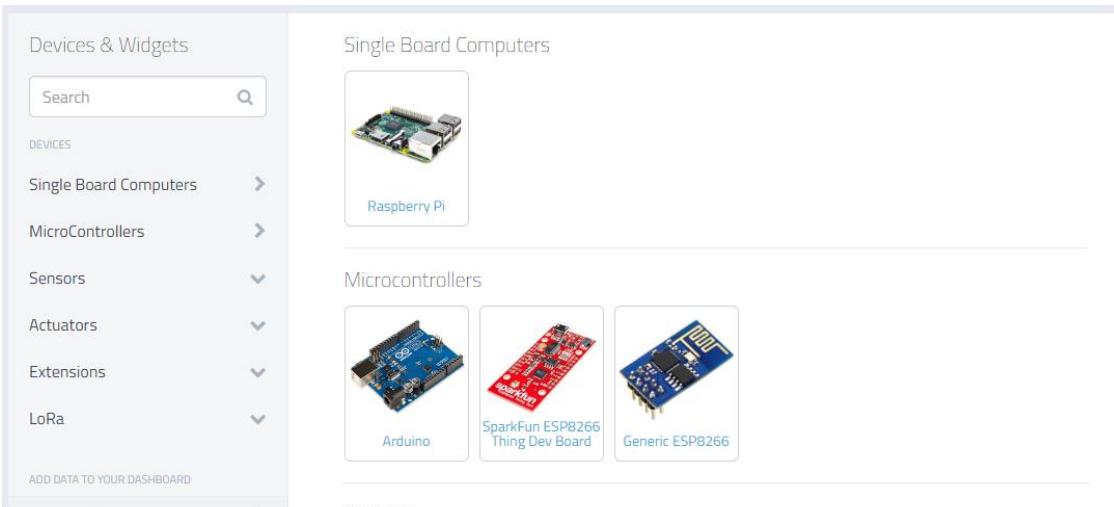
```

ESP8266 | Arduino 1.8.5
Berkas Sunting Sketch Alat Bantuan
ESP8266 §
6 #include <CayenneMQTTESP8266.h>
7
8 // WiFi network info.
9 char ssid[] = "ssid";
10 char wifiPassword[] = "wifiPassword";
11
12 // Cayenne authentication info. This should be obtained from :
13 char username[] = "MQTT_USERNAME";
14 char password[] = "MQTT_PASSWORD";
15 char clientID[] = "CLIENT_ID";
16
17 unsigned long lastMillis = 0;
18
19 void setup() {
20   Serial.begin(9600);
21   Cayenne.begin(username, password, clientID, ssid, wifiPassw
22 }
23
24 void loop() {
25   Cayenne.loop();
26 }

Selesai mengunggah.
Pustaka tidak sah ditemukan di C:\Users\asus\Documents\Arduino\lib
Pustaka tidak sah ditemukan di C:\Users\asus\Documents\Arduino\lib

```

Pada halaman web Cayenne silahkan buat project kemudian pilih Board mikrokonroler Generic ESP8266 seperti terlihat pada gambar berikut



Silahkan isi **Username**, **password** dan **clientID** seperti yang disediakan oleh Cayenne, berikut adalah contoh

selanjutnya kita akan mencoba mengirim data random dari ESP8266-01 ke cayenne. Berikut program lengkapnya.

```
//#define CAYENNE_DEBUG
#define CAYENNE_PRINT Serial
#include <CayenneMQTTESP8266.h>

// WiFi network info.
char ssid[] = "anakkendali.com";
char wifiPassword[] = "chibakutensei";

// Cayenne authentication info. This should be obtained from the Cayenne Dashboard.
char username[] = "ccaaaaf0-0d81-11e9-898f-c12a468aadce";
char password[] = "f252edfa0e10fab2b5d8e5d1d21824f826ccf740";
char clientID[] = "ofedefbo-odc9-11e9-809d-of8fe4c30267";
```

```

unsigned long lastMillis = 0;

void setup() {
    Serial.begin(9600);
    Cayenne.begin(username, password, clientID, ssid, wifiPassword);
}
unsigned long waktulama=0;
void loop() {
    Cayenne.loop();

    if (millis() - waktulama >= 100){
        waktulama = millis();

        float data = random (0.00, 200.00);

        Cayenne.celsiusWrite(0, data);
    }
}

```

berikut adalah hasil data yang diterima oleh cayene

Fversi Web Dekstop

selain versi Web Dekstop Cayenne juga menyediakan versi aplikasi Android yang dapat di download pada Playstore.

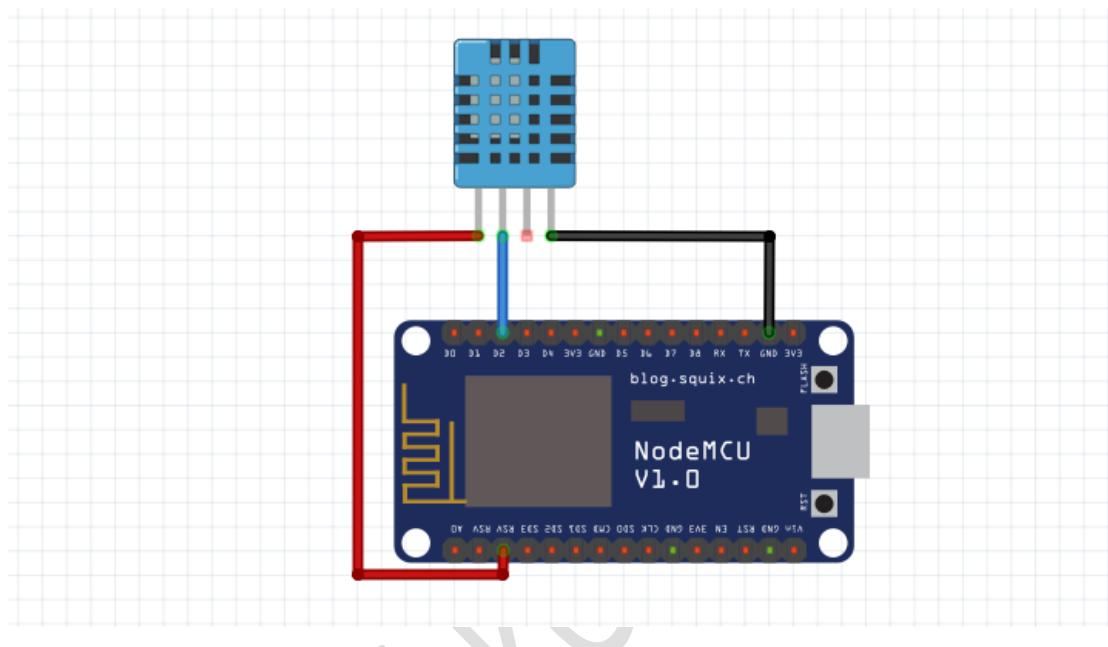
tutorial dengan menggunakan ESP8266 – 01 dicukupkan sampai sini, dan mengenai Cayenne akan di lengkapi pada tutorial dengan NodeMCU/Wemos D1 mini.

Program NodeMCU / Wemos D1 Mini Kirim Data DHT11 ke Thingspeak

Tutorial ini adalah monitoriag data Suhu dan Kelembaban menggunakan sensor DHT11 dan NodeMCU / Wemos D1 Mini ESP8266.

Terlebih dahulu buatlah Chanel yang berisi dua Field, masing masing untuk data Suhu dan data Kelembaban.

untuk rangkaian NodeMCU dan sensor DHT11 seperti pada gambar di bawah ini.



Tidak perlu cemas jika sensor DHT11 hanya memiliki 3 kaki karena sensor sudah di buatkan PCB dengan tambahan resistor.

selanjutnya silahkan upload program berikut ke modul NodeMCU ESP8266.

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include "DHT.h"

#define DHTPIN D2
#define DHTTYPE DHT11

WiFiClient client;

String thingSpeakAddress = "http://api.thingspeak.com/update?";
String writeAPIKey;
String tsfield1Name;
String request_string;

HTTPClient http;
DHT dht(DHTPIN, DHTTYPE);
```

```

void setup()
{
    Serial.begin(9600);
    WiFi.disconnect();
    WiFi.begin("anakkendali.com", "chibakutensei");
    while (!(WiFi.status() == WL_CONNECTED)) {
        delay(300);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("Connected");
    dht.begin();
}

void loop()
{
    delay(2000);
    float h = dht.readHumidity();
    float t = dht.readTemperature();
    if (isnan(h) || isnan(t)) {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }
    kirim_thingspeak(t, h);
}

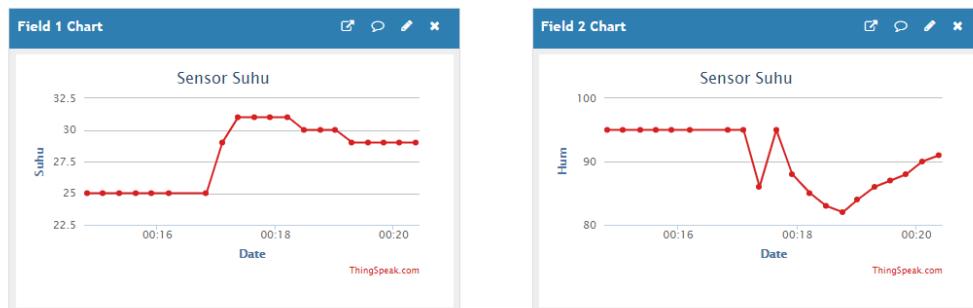
void kirim_thingspeak(float suhu, float hum) {
    if (client.connect("api.thingspeak.com", 80)) {
        request_string = thingSpeakAddress;
        request_string += "key=";
        request_string += "ABYNHJEFBK3Q97NE";
        request_string += "&";
        request_string += "field1";
        request_string += "=";
        request_string += suhu;
        request_string += "&";
        request_string += "field2";
        request_string += "=";
        request_string += hum;
        http.begin(request_string);
        http.GET();
        http.end();
        delay(5000);
        Serial.print("data yang dikirim, Suhu : ");
        Serial.print(suhu);
        Serial.print("\t");
        Serial.print("Kelembaban : ");
        Serial.println(hum);
    }
}

```



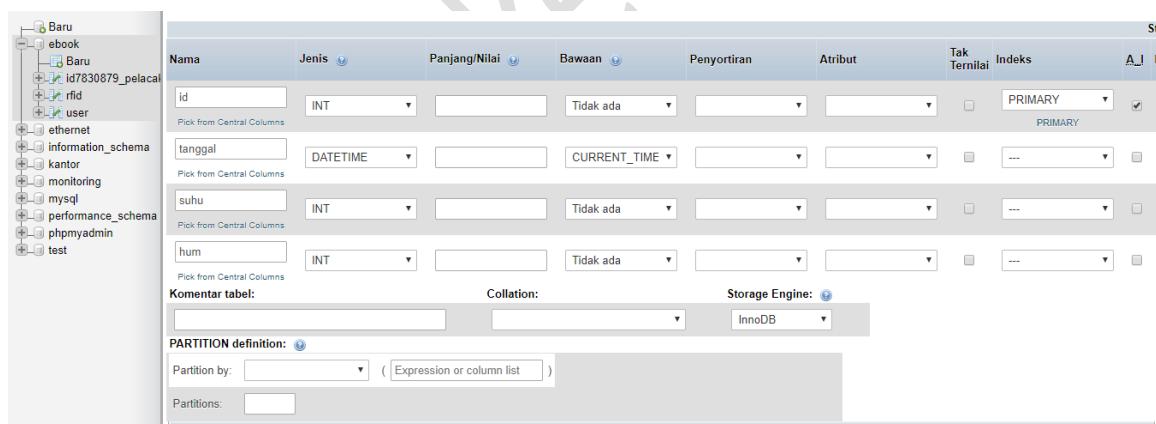
Untuk hasil pada field1 dan field2 terlihat seperti gambar berikut

Updated: 5 minutes ago
Last entry: 5 minutes ago
Entries: 3

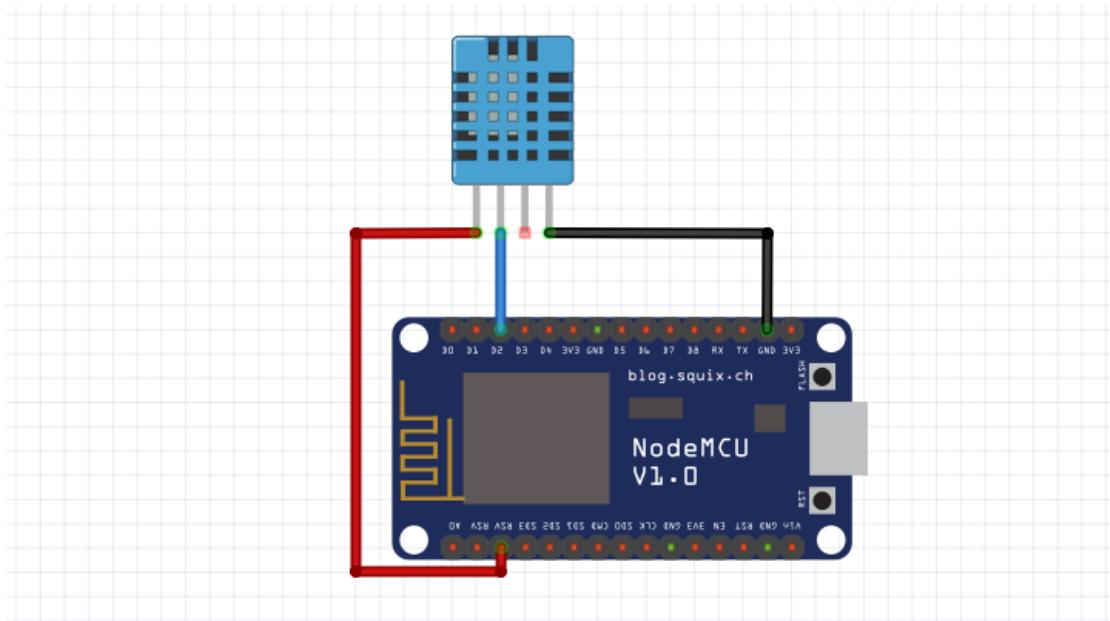


Program NodeMCU / Wemos D1 Mini Kirim Data DHT11 ke Database MySQL

Sebelum membuat program untuk NodeMCU ESP8266 terlebih dahulu buatlah database MySQL menggunakan Server XAMPP. Seperti terlihat pada gambar dibawah ini.



Untuk rangkaian DHT11 dan Nodemcu kirim data ke Database MySQL terlihat seperti gambar dibawah ini.



kemudian buatlah dua program PHP sebagai jembatan antara Database dengan NodeMCU.
yang pertama silahkan beri nama "koneksi.php" dan di simpan dalam folder
htdocs/monitordht11 (monitordht11 adalah folder buatan kita sendiri)

```
<?php

$servername = "localhost";
$database = "ebook";
$username = "root";
$password = "";

$konek = mysqli_connect ($servername, $username, $password, $database);

if ($konek!=false){
echo "berhasil";
} else {
echo "gagal";}

?>
```

Yang kedua silahkan beri nama "index.php" dan disimpan pada folder yang sama seperti file "koneksi.php"

```
<?php

include ("koneksi.php");

$now = new DateTime();
$suhu = $_GET['suhu'];
$hum = $_GET['hum'];
```

```

$datenow = $now->format("Y-m-d H:i:s");

mysqli_query($koneksi, "INSERT INTO dht11(tanggal,suhu,hum)
VALUES('$datenow','$suhu','$hum')");

?>

```

Kemudian upload program berikut ke modul NodeMCU.

```

#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>

WiFiClient client;
#include "DHT.h"

#define DHTPIN D2
#define DHTTYPE DHT11

String request_string;
const char* host = "192.168.1.6";

HTTPClient http;
DHT dht(DHTPIN, DHTTYPE);

void setup()
{
    Serial.begin(9600);
    WiFi.disconnect();
    WiFi.begin("anakkendali.com", "chibakutensei");
    while ((!(WiFi.status() == WL_CONNECTED))) {
        delay(300);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("Connected");
    dht.begin();
}

void loop()
{
    delay(2000);
    float h = dht.readHumidity();
    float t = dht.readTemperature();
    if (isnan(h) || isnan(t)) {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }

    if (!client.connect(host,80)) {
        Serial.println("Gagal Konek");

```

```

    return;
}

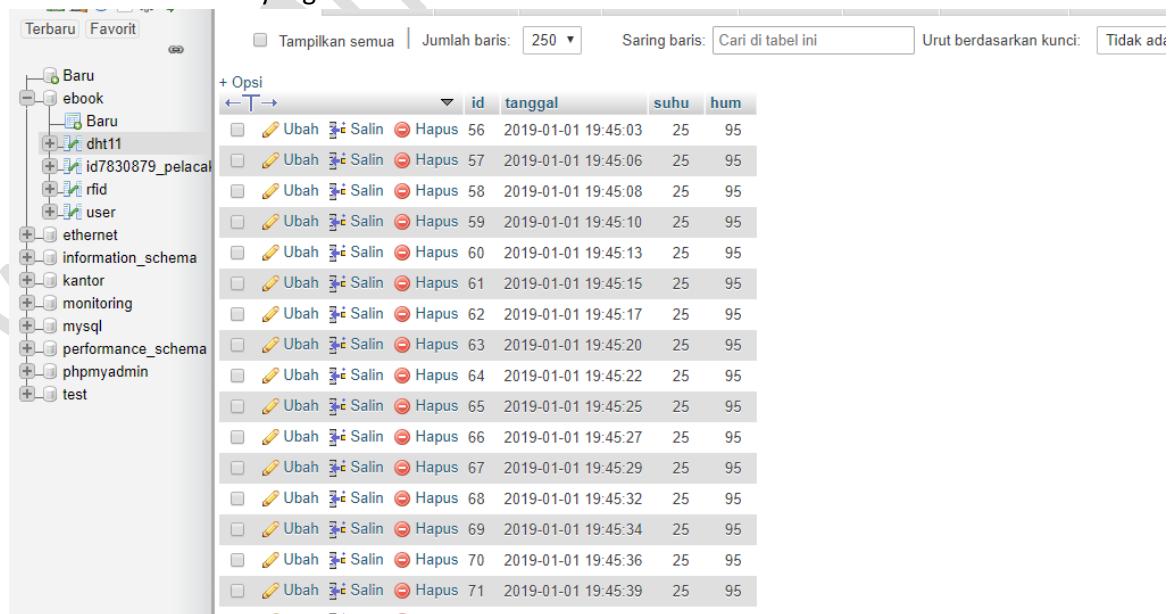
request_string = "/monitordht11/index.php?suhu=";
request_string += t;
request_string += "&hum=";
request_string += h;

// Serial.print("requesting URL: ");
// Serial.println(request_string);
client.print(String("GET ") + request_string + "HTTP/1.1\r\n" + "Host: " + host + "\r\n" +
"Connection: close\r\n\r\n");
Serial.print("data yang dikirim, Suhu : ");
Serial.print(t);
Serial.print("\t");
Serial.print("Kelembaban : ");
Serial.println(h);

unsigned long timeout = millis();
while (client.available() == 0){
  if (millis() - timeout > 5000) {
    Serial.println(">>> Client Timeout !");
    client.stop();
    return;
  }
}
}
}

```

hasil yang diterima oleh database seperti pada gambar di bawah ini.
 tanggal otomatis berdasarkan waktu saat data dikirim. Sedangkan ID nilainya akan bertambah ketika ada data baru yang masuk.



The screenshot shows the MySQL Workbench application. On the left, there is a tree view of the database schema, including tables like Baru, ebook, dht11, id7830879_pelacak, rfid, user, ethernet, information_schema, kantor, monitoring, mysql, performance_schema, phpmyadmin, and test. On the right, there is a table named 'Baru' with columns: id, tanggal, suhu, and hum. The table contains 71 rows of data, each with a timestamp between 2019-01-01 19:45:03 and 2019-01-01 19:45:39, and values for suhu and hum both set to 25 and 95 respectively. Each row has edit, copy, and delete icons.

	Opsi		id	tanggal	suhu	hum
<input type="checkbox"/>		Ubah	56	2019-01-01 19:45:03	25	95
<input type="checkbox"/>		Ubah	57	2019-01-01 19:45:06	25	95
<input type="checkbox"/>		Ubah	58	2019-01-01 19:45:08	25	95
<input type="checkbox"/>		Ubah	59	2019-01-01 19:45:10	25	95
<input type="checkbox"/>		Ubah	60	2019-01-01 19:45:13	25	95
<input type="checkbox"/>		Ubah	61	2019-01-01 19:45:15	25	95
<input type="checkbox"/>		Ubah	62	2019-01-01 19:45:17	25	95
<input type="checkbox"/>		Ubah	63	2019-01-01 19:45:20	25	95
<input type="checkbox"/>		Ubah	64	2019-01-01 19:45:22	25	95
<input type="checkbox"/>		Ubah	65	2019-01-01 19:45:25	25	95
<input type="checkbox"/>		Ubah	66	2019-01-01 19:45:27	25	95
<input type="checkbox"/>		Ubah	67	2019-01-01 19:45:29	25	95
<input type="checkbox"/>		Ubah	68	2019-01-01 19:45:32	25	95
<input type="checkbox"/>		Ubah	69	2019-01-01 19:45:34	25	95
<input type="checkbox"/>		Ubah	70	2019-01-01 19:45:36	25	95
<input type="checkbox"/>		Ubah	71	2019-01-01 19:45:39	25	95

Program NodeMCU / Wemos D1 Kontrol LED Aplikasi Android

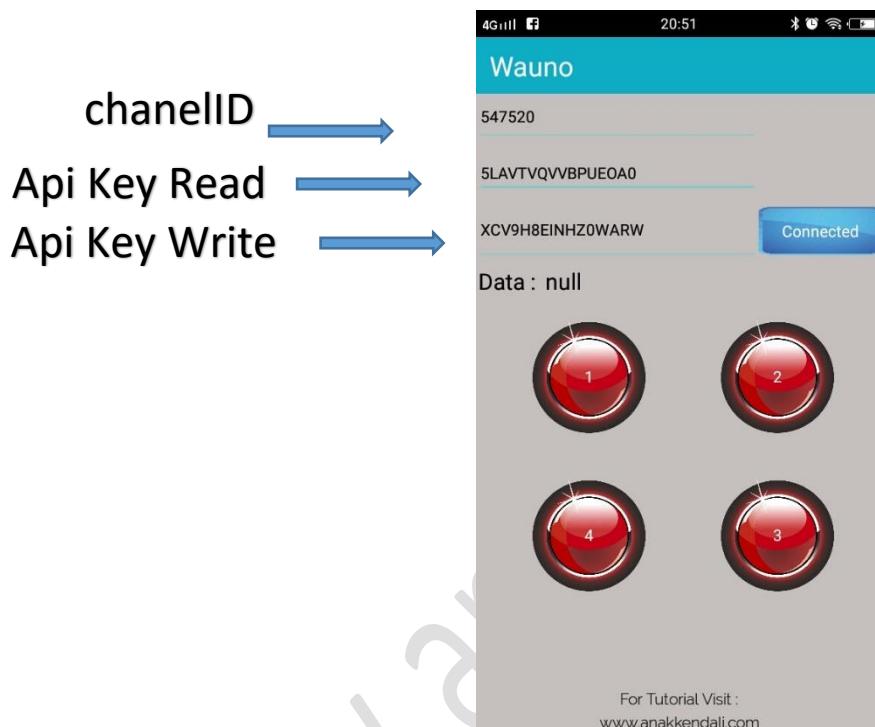
Pada tutorial kali ini juga akan menggunakan server Thingspeak sebagai jembatan pengirim dan penerima data. Karena prinsip kerjanya, dari aplikasi android mengirim data ke thingspeak kemudian esp8266 membaca data yang di terima oleh thingspeak untuk melakukan perintah menyalakan dan mematikan LED, LED dapat di ganti dengan Relay, Motor atau aktuator lainnya.

pertama buat terlebih dahulu aplikasi Android dengan menggunakan App Inventor. Disini akan disediakan file mentah yang bisa di upload ke App Inventor kemudian bisa di build menjadi aplikasi android.

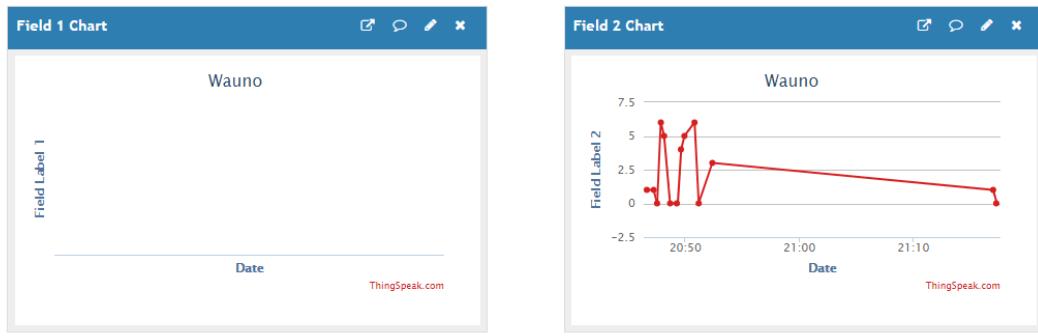
Download Aplikasi Android Kontrol Led Internet (APK) [Disini!](#)

Download Aplikasi Android Kontrol Led Internet (aia) [Disini!](#)

tampilan aplikasi kontrol led dengan internet pada smrphone android



selanjutnya buatlah dua buah field untuk data monitoring dan untuk kontrol LED. Pada aplikasi tersebut terdapat label “Data” dan empat buah tombol. Label data berisi **Null** karena tidak ada nilai pada field1. Aplikasi ini menyediakan fitur tersebut untuk mencoba jika ingin menggunakan aplikasi android sebagai monitor data, seperti data suhu atau data sensor lainnya.
untuk menggunakan fitur monitor aplikasi ini, bisa mengikuti tutorial kirim data ESP8266 NodeMCU ke thingspeak pada bab sebelumnya.



berikut adalah program ESP8266 NodeMCU untuk kontrol LED melalui aplikasi Android berbasis Internet

```
#include <ESP8266WiFi.h>

#include <ThingSpeak.h>

#include <ESP8266HTTPClient.h>

WiFiClient client;

String thingSpeakAddress= "http://api.thingspeak.com/update?";
String writeAPIKey;
String tsfield1Name;
String request_string;

HTTPClient http;

void setup()
{
  Serial.begin(9600);
  ThingSpeak.begin(client);

  WiFi.disconnect();
  delay(3000);
  WiFi.begin("anakkendali.com","chibakutensei");
  while ((!(WiFi.status() == WL_CONNECTED))){
    Serial.print(".");
    delay(300);
  }
  Serial.println("connected!");
  pinMode(D4, OUTPUT);
  pinMode(D5, OUTPUT);
  pinMode(D6, OUTPUT);
  pinMode(D7, OUTPUT);
}
```

```
unsigned long waktulama=0;

void loop()
{
    int val = (ThingSpeak.readIntField(547520,2,"5LAVTVQVVBPUCEO Ao"));

    if (val == 1){
        digitalWrite(D4, 1);
    }
    else if (val == 0){
        digitalWrite(D4, 0);
    }
    else if (val == 3){
        digitalWrite(D5, 1);
    }
    else if (val == 4){
        digitalWrite(D5, 0);
    }
    else if (val == 5){
        digitalWrite(D6, 1);
    }
    else if (val == 6){
        digitalWrite(D6, 0);
    }
    else if (val == 7){
        digitalWrite(D7, 1);
    }
    else if (val == 8){
        digitalWrite(D7, 0);
    }
    Serial.println(val);
}
```

kelemahan menggunakan server thingspeak adalah data yang dikirim maupun yang diterima sangat lama, sehingga saat melakukan perintah harus menunggu data sebelumnya selesai di proses.

Program NodeMCU / Wemos D1 Kontrol LED dengan Cayenne

Sebelumnya sudah dijelaskan tentang cara mengirim data dari ESP-01 ke server Cayenne, dan kali ini kita akan membuat program untuk menyalakan LED yang di kontrol oleh Cayenne. yang pertama harus kita lakukan adalah menyiapkan Project pada akun Cayenne dengan menggunakan Generic ESP8266 sebagai microkontroler nya dan menambahkan Aktuator Relay. Selebihnya ikuti gambar berikut.

The screenshot shows the Cayenne dashboard interface. On the left, there's a sidebar with a dropdown for 'Add new...' and a list of devices: 'Generic ESP8266'. Below this is a search bar labeled 'Search Devices'. In the center, under 'Custom Widgets', there are five items listed:

- Gauge Display Widget Example
- 2 State Display Widget Example
- Button Controller Widget Example
- Slider Controller Widget Example
- Motion Example

Selanjutnya silahkan klik Add Widget. Dan akan muncul channel0 sebagai Button untuk menyalaan LED pada pin D4 NodeMCU

The screenshot shows the Cayenne dashboard with the 'Overview' tab selected. On the left, it shows the same sidebar and device list as the previous screenshot. The main area displays a single 'LED' widget. At the bottom, it says 'Last data packet sent: January 2, 2019 11:30:07 PM'.

selanjutnya silahakn buka software Arduino IDE dan upload program berikut pada board NodeMCU ESP8266

```

//#define CAYENNE_DEBUG
#define CAYENNE_PRINT Serial
#include <CayenneMQTTESP8266.h>

// WiFi network info.
char ssid[] = "anakkendali.com";
char wifiPassword[] = "chibakutensei";

// Cayenne authentication info. This should be obtained from the Cayenne Dashboard.
char username[] = "ccaaaaf0-0d81-11e9-898f-c12a468aadce";
char password[] = "f252edfa0e10fab2b5d8e5d1d21824f826ccf740";
char clientID[] = "ofedefbo-odc9-11e9-809d-of8fe4c30267";

unsigned long lastMillis = 0;

void setup() {
    Serial.begin(9600);
    Cayenne.begin(username, password, clientID, ssid, wifiPassword);
    pinMode(D4, OUTPUT);
}
unsigned long waktulama = 0;
void loop() {
    Cayenne.loop();
}

CAYENNE_IN_DEFAULT()
{
    CAYENNE_LOG("Channel %u, value %d", request.channel, getValue.asInt());
    digitalWrite(D4, getValue.asInt());
}

```

Tutorial ini menutup materi pada EBOOK ESP8266 dan mungkin akan ada ebook bagian ke 2 nya yang akan melanjutkan berbagai tutorial mengenai IOT, tidak hanya Board ESP8266, namun juga akan di bahas mengenai IOT dengan GSM maupun ETHERNET.

Tentang Penulis

Chaerul Anam. Lahir 01 Desember 1996 di Indramayu, Jawa Barat. Pernah sekolah di Pesantren Al – Ishlah Tajug Sudimampir Indramayu. Dan sekarang sedang menjalani studi Teknik Elektro di Universitas Islam Indonesia. Mengambil Konsentrasi Kendali menjadi pilihan untuk mempelajari bidang Mikrokontroler Seperti Atmega, Arduino, STM32, NodeMCU, PLC, Labview. Selain belajar pemograman hardware penulis juga sangat suka dengan dunia Blogging.

Penulis dapat di hubungi melalui :

WA : 083862832235

Email : chaerulanam1412@gmail.com

www.anakkendali.com

Referensi

- [1] "Getting Started with MicroPython on ESP32 and ESP8266 | Random Nerd Tutorials." [Online]. Available: <https://randomnerdtutorials.com/getting-started-micropython-esp32-esp8266/>. [Accessed: 22-Oct-2018].
- [2] "D1 mini [WEMOS Electronics]." [Online]. Available: https://wiki.wemos.cc/products:d1:d1_mini. [Accessed: 23-Oct-2018].
- [3] "Belajar PHP untuk PEMULA : Intro - YouTube." [Online]. Available: <https://www.youtube.com/watch?v=l1W2OwV5rgY&list=PLFIM0718LjIUqXfmEIBE3-uzERZPh3vp6>. [Accessed: 25-Oct-2018].
- [4] "Cayenne Docs." [Online]. Available: <https://mydevices.com/cayenne/docs/getting-started/#getting-started-esp8266-get-started-in-minutes>. [Accessed: 01-Jan-2019].
- [5] "ESP8266-ESP-01-ESP8266-WIFI-module-Serial-Wi-Fi-Wireless-Transceiver-Module-for-Arduino.jpg_640x640.jpg (640×640)." [Online]. Available: https://ae01.alicdn.com/kf/HTB1mwr2NFXXXXc4XpXXq6xFXXXn/ESP8266-ESP-01-ESP8266-WIFI-module-Serial-Wi-Fi-Wireless-Transceiver-Module-for-Arduino.jpg_640x640.jpg. [Accessed: 21-Oct-2018].
- [6] "1501699031787010991.jpg (400×400)." [Online]. Available: <https://gloimg.gbtcdn.com/gb/pdm-product-pic/Electronic/2017/06/13/goods-img/1501699031787010991.jpg>. [Accessed: 21-Oct-2018].
- [7] "F7FIBJYIMTE8XJB.LARGE.jpg (535×404)." [Online]. Available: <https://cdn.instructables.com/F7F/IBJY/IMTE8XJB/F7FIBJYIMTE8XJB.LARGE.jpg>. [Accessed: 21-Oct-2018].
- [8] "df713b06-d98b-4e7e-bff4-274b5d3fe3bf.JPG (600×600)." [Online]. Available: <https://img.banggood.com/images/oaupload/banggood/images/29/9F/df713b06-d98b-4e7e-bff4-274b5d3fe3bf.JPG>. [Accessed: 21-Oct-2018].