

Koding Program

```
#include <Servo.h>
#include <Wire.h>
#include "RTClib.h"
//#include <SoftwareSerial.h>```
#include <OneWire.h>
#define pos_buka 160
#define pos_tutup 180
#define SensorPin A1      //pH meter Analog output to Arduino Analog Input 0
//#define SIM900 Serial

bool sudah_pagi = false;
bool sudah_siang = false;
bool sudah_malam = false;
//SoftwareSerial SIM900(2, 3);
#define SIM900 Serial3
String content = "";

Servo myservo; // create servo object to control a servo
// twelve servo objects can be created on most boards

int pos = 0; // variable to store the servo position

RTC_DS3231 rtc;

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

unsigned long int avgValue; //Store the average value of the sensor feedback
float b;
int buf[10], temp;

//SoftwareSerial SIM900( 14, 15); // RX, TX
String inputgsm = "";
boolean gsmComplete = false;
int jam1 = 0 ;
int menit1 = 0 ;
int jam2 = 0 ;
int menit2 = 0 ;
```

```
int jam3 = 0 ;
int menit3 = 0 ;

int t_jam1 = 0 ;
int t_menit1 = 0 ;
int t_jam2 = 0 ;
int t_menit2 = 0 ;
int t_jam3 = 0 ;
int t_menit3 = 0 ;

int jam_sekarang = 0 ;
int menit_sekarang = 0 ;

int pin_Suhu = 10;
float temperatur;
float phValue;
OneWire ds(pin_Suhu);

void setup() {
    // put your setup code here, to run once:
    myservo.attach(9); // attaches the servo on pin 9 to the servo
    myservo.write(pos_tutup);

    Serial.begin(9600);
#ifndef ESP8266
    while (!Serial); // for Leonardo/Micro/Zero
#endif

    //pinMode(A1, OUTPUT);
    Serial.println("Ready"); //Test the serial monitor
    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));
}

rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
SIM900.begin(9600);
// delay(5000);
//SIM900.println("AT+CMGF=1");
//while (!tunggu_balasan("OK")) {
//SIM900.println("AT+CMGF=1");
myservo.write(pos_tutup);

}

// send_sms("082261823243", "CYBORG CERIA");

void loop() {
    // put your main code here, to run repeatedly:
    ambil_waktu();
    Serial.print("jam 3 = ");Serial.println(jam3);
    Serial.print("menit 3 = ");Serial.println(menit3);

    Serial.print("jam 2 = ");Serial.println(jam2);
    Serial.print("menit 2 = ");Serial.println(menit2);

    Serial.print("jam 1 = ");Serial.println(jam1);
    Serial.print("menit 1 = ");Serial.println(menit1);

    Serial.print(sudah_malam);Serial.print(" ");Serial.print(sudah_siang);Serial.print(" ");
    Serial.println(sudah_pagi);

    if (jam_sekarang >= jam3 && menit_sekarang >= menit3 && !sudah_malam &&
    jam3 != 0 && menit3 != 0) {
        sudah_malam = true;
        sudah_siang = true;
        sudah_pagi = true;
    }
}

```

```

        buka_pakan(3000); //buka pakan 3 detik
    } else if (jam_sekarang >= jam2 && menit_sekarang >= menit2 && !sudah_siang
&& jam2 != 0 && menit2 != 0) {
    sudah_malam = false;
    sudah_siang = true;
    sudah_pagi = true;
    buka_pakan(3000); //buka pakan 3 detik
} else if (jam_sekarang >= jam1 && menit_sekarang >= menit1 && !sudah_pagi
&& jam1 != 0 && menit1 != 0) {
    sudah_malam = false;
    sudah_siang = false;
    sudah_pagi = true;
    buka_pakan(3000); //buka pakan 3 detik
} else if (jam_sekarang == 00 && menit_sekarang >= 1 && sudah_pagi &&
sudah_siang && sudah_malam) {
    sudah_malam = false;
    sudah_siang = false;
    sudah_pagi = false;
}

hitung_suhu();
hitung_ph();
SubmitHttpRequest();

}

void hitung_suhu() {
    temperatur = getTemp();
    Serial.println(temperatur);
}

void hitung_ph() {
    for (int i = 0; i < 10; i++) //Get 10 sample value from the sensor for smooth the
value
    {
        buf[i] = analogRead(SensorPin);
        delay(10);
    }
    for (int i = 0; i < 9; i++) //sort the analog from small to large

```

```

{
    for (int j = i + 1; j < 10; j++)
    {
        if (buf[i] > buf[j])
        {
            temp = buf[i];
            buf[i] = buf[j];
            buf[j] = temp;
        }
    }
}

{ avgValue = 0;
for (int i = 2; i < 8; i++)           //take the average value of 6 center sample
    avgValue += buf[i];
phValue = (float)avgValue * 5.0 / 1024 / 6; //convert the analog into millivolt
phValue = 3.5 * phValue;
Serial.println(phValue);
}
}

void buka_pakan(int waktu) {
    myservo.write(pos_buka);
    delay(waktu);
    myservo.write(pos_tutup);
}

void ambil_waktu() {
    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();
jam_sekarang = now.hour();
menit_sekarang = now.minute();

}

bool tunggu_balasan(String balasan) {
    float waktu = millis();
    bool sesuai = false;
    while (!gsmComplete || millis() - waktu < 5000) {
        gsmEvent();
    }
    if (gsmComplete) {
        Serial.println(inputgsm);
        Serial.println(balasan);
        if (inputgsm.indexOf(balasan) != -1) {
            sesuai = true;
        }
        inputgsm = "";
        gsmComplete = false;
    }
    return sesuai;
}

void gsmEvent() {
    while (SIM900.available()) {
        char inChar = (char)SIM900.read();
        inputgsm += inChar;
        if (inChar == '\n') {
            gsmComplete = true;
        }
    }
}

//fungsi untuk kirim sms
void send_sms(String nomor, String isi_pesan) {

```

```

//delay(100);
SIM900.print("AT+CMGS=\\"");
SIM900.print(nomor);
SIM900.println("\\"");
tunggu_balasan(">");
SIM900.println(isi_pesan);
delay(100);
SIM900.println((char)26);
tunggu_balasan("OK");
}

float getTemp() {

    byte data[12];
    byte addr[8];

    if ( !ds.search(addr)) {
        //no more sensors on chain, reset search
        ds.reset_search();
        return -1000;
    }

    if ( OneWire::crc8( addr, 7) != addr[7]) {
        Serial.println("CRC is not valid!");
        return -1000;
    }

    if ( addr[0] != 0x10 && addr[0] != 0x28) {
        Serial.print("Device is not recognized");
        return -1000;
    }

    ds.reset();
    ds.select(addr);
    ds.write(0x44, 1);

    byte present = ds.reset();
    ds.select(addr);
    ds.write(0xBE);
}

```

```

for (int i = 0; i < 9; i++) {
    data[i] = ds.read();
}

ds.reset_search();

byte MSB = data[1];
byte LSB = data[0];

float TRead = ((MSB << 8) | LSB);
float Temperature = TRead / 16;

return Temperature;

}

void setting_sim900() {
    SIM900.println("AT+CSQ"); // Signal quality check

    delay(100);

    ShowSerialData(); // this code is to show the data from gprs shield, in order to easily
    see the process of how the gprs shield submit a http request, and the following is for
    this purpose too.
    //delay(5000);
    SIM900.println("AT+SAPBR=1,1");//setting the SAPBR
    delay(500);
    ShowSerialData();
}

void SubmitHttpRequest()
{
    setting_sim900();

    ShowSerialData();

    SIM900.println("AT+HTTPINIT"); //init the HTTP request
}

```

```
delay(1000);
ShowSerialData();
SIM900.println("AT+HTTPPARA=\"cid\",1");// setting the httppara, the second
parameter is the website you want to access
delay(1000);
ShowSerialData();

SIM900.print("AT+HTTPPARA=\"URL\",\"sistem-monitoring-smart-fish-
farm.000webhostapp.com/update_sensor.php?suhu=\"");// setting the httppara, the
second parameter is the website you want to access
SIM900.print(temperatur);// setting the httppara, the second parameter is the website
you want to access
SIM900.print("&ph=");// setting the httppara, the second parameter is the website
you want to access
SIM900.print(phValue);// setting the httppara, the second parameter is the website you
want to access
SIM900.println("");// setting the httppara, the second parameter is the website you
want to access
delay(1000);

ShowSerialData();

SIM900.println("AT+HTTPACTION=0");//submit the request
//delay(5000);//the delay is very important, the delay time is base on the return from
the website, if the return datas are very large, the time required longer.
//while(!SIM900.available());
//while(!SIM900.find("+HTTPACTION"));
String masuk = "";
// String RedState = content.substring();
//SIM900.find("AT+HTTPREAD");
long waktu_hitung = millis();
while (waktu_hitung + 10000 > millis ()) {
if (SIM900.available())
{
    //Serial.write(SIM900.read());
    //char a = (char)SIM900.read();
    masuk = masuk + char (SIM900.read());
    if (masuk.indexOf("OK") != -1) {
```

```

        break;
    }
}
}

Serial.println(masuk);
masuk = "";
//ShowSerialData();
waktu_hitung = millis ();

int temp_waktu = 20000;
while (waktu_hitung + temp_waktu > millis ()) {
    if (SIM900.available())
    {

        //Serial.write(SIM900.read());
        //char a = (char)SIM900.read();
        masuk = masuk + char (SIM900.read());
        if (masuk.indexOf("+HTTPACTION") != -1) {
//        break;
        waktu_hitung = millis();
        temp_waktu = 1000;
    }
}
}

//ShowSerialData();
Serial.print("action : ");
Serial.println(masuk);
content = "";
SIM900.println("AT+HTTPREAD");// read the data from the website you access
masuk = "";
//ShowSerialData();
waktu_hitung = millis ();
while (waktu_hitung + 10000 > millis ()) {
    if (SIM900.available())
    {

        //Serial.write(SIM900.read());
        //char a = (char)SIM900.read();
        masuk = masuk + char (SIM900.read());
//        if (masuk.indexOf("+HTTPREAD") != -1) {

```

```

        //      break;
        //    }
      }
}

Serial.print("masuk = ");
Serial.println(masuk);

content = getValue(masuk, '|', 1);
Serial.println(content);
//Serial.println(getValue(content, '|', 2));

String waktu1 = getValue(content, ',', 0);
String waktu2 = getValue(content, ',', 1);
String waktu3 = getValue(content, ',', 2);
t_jam1 = getValue(waktu1, ':', 0).toInt();
t_menit1 = getValue(waktu1, ':', 1).toInt();
t_jam2 = getValue(waktu2, ':', 0).toInt();
t_menit2 = getValue(waktu2, ':', 1).toInt();
t_jam3 = getValue(waktu3, ':', 0).toInt();
t_menit3 = getValue(waktu3, ':', 1).toInt();
Serial.print(t_jam1); Serial.print(" ");
Serial.print(t_menit1); Serial.print("\t");
Serial.print(t_jam2); Serial.print(" ");
Serial.print(t_menit2); Serial.print("\t");
Serial.print(t_jam3); Serial.print(" ");
Serial.print(t_menit3); Serial.println(" ");

if (t_jam1 != 0 && t_menit1 !=0 ){
  jam1 = t_jam1;
  menit1 = t_menit1;
}
if (t_jam2 != 0 && t_menit2 !=0 ){
  jam2 = t_jam2;
  menit2 = t_menit2;
}
if (t_jam3 != 0 && t_menit3 !=0 ){
  jam3 = t_jam3;
  menit3 = t_menit3;
}

```

```

//cek_data();

//delay(1000);
ShowSerialData();
//cek_data();
//ShowSerialData();

SIM900.println("");
delay(100);
SIM900.println("AT+HTTPTERM"); //init the HTTP request

delay(1000);
ShowSerialData();
}

void cek_data()
{
// String RedState = content.substring();
//SIM900.find("AT+HTTPREAD");
//long waktu_bantuan = millis();
while (SIM900.available() )
{
//Serial.write(char (SIM900.read()));
//if (){
content += (char)SIM900.read();
// if (content.indexOf("|") > 5){
// break;
//}
//}

//Serial.write(SIM900.read());
//char a = (char)SIM900.read();

}
Serial.print("content:");
Serial.println(content);
Serial.print("INDEX:");
Serial.println(content.indexOf("|"));

```

```

//content = getValue(content, '|', 0);
// Serial.println(getValue(content, '|', 0));
Serial.println(getValue(content, '|', 1));
// Serial.println(getValue(content, '|', 2));

// String waktu1 = getValue(content, ',', 0);
// String waktu2 = getValue(content, ',', 1);
// String waktu3 = getValue(content, ',', 2);
// jam1 = getValue(waktu1, ':', 0).toInt();
// menit1 = getValue(waktu1, ':', 1).toInt();
// jam2 = getValue(waktu2, ':', 0).toInt();
// menit2 = getValue(waktu2, ':', 1).toInt();
// jam3 = getValue(waktu3, ':', 0).toInt();
// menit3 = getValue(waktu3, ':', 1).toInt();
Serial.print(jam1); Serial.print(" ");
Serial.print(menit1); Serial.print("\t");
Serial.print(jam2); Serial.print(" ");
Serial.print(menit2); Serial.print("\t");
Serial.print(jam3); Serial.print(" ");
Serial.print(menit3); Serial.println(" ");
content = "";
}

```

```

void ShowSerialData()
{
    while (SIM900.available())
        Serial.write(char (SIM900.read()));
}

String getValue(String data, char separator, int index)
{
    Serial.print("data : ");
    Serial.print(data);

    int found = 0;
    int strIndex[] = {0, -1};
    int maxIndex = data.length() - 1;

    for (int i = 0; i <= maxIndex && found <= index; i++) {

```

```
if (data.charAt(i) === separator || i == maxIndex) {  
    found++;  
    strIndex[0] = strIndex[1] + 1;  
    strIndex[1] = (i == maxIndex) ? i + 1 : i;  
}  
}  
Serial.print("found");  
Serial.println(found);  
return found > index ? data.substring(strIndex[0], strIndex[1]) : "";
```