

Lampiran

1. Coding Mikrokontroler

```
#include <DS3231.h>          // Memanggil RTC3231 Library
#include <Wire.h>            // i2C Connection Library
#include <LiquidCrystal_I2C.h> // Memanggil i2C LCD Library
#include <Button.h>         // Memanggil library Push Button
#include "GravityTDS.h"

#define DN_PIN 2 // Decrease Button Down
#define UP_PIN 3 // Increase Button Up
#define SET_PIN 4 // Setup Button Settings Jam
#define ALR_PIN 5 // Setup Button Alarm
#define PULLUP true
#define INVERT true
#define DEBOUNCE_MS 20
#define REPEAT_FIRST 500
#define REPEAT_INCR 100

#define TdsSensorPin A1
GravityTDS gravityTds;

#define trigPin 7
#define echoPin 6

#define ledAman 8
#define led1 9
#define led2 10
#define led3 11

float temperature = 25,tdsValue = 0;

unsigned long pulseTime = 0;
Button btnUP(UP_PIN, PULLUP, INVERT, DEBOUNCE_MS);
```

```
Button btnDN(DN_PIN, PULLUP, INVERT, DEBOUNCE_MS);
Button btnSET(SET_PIN, PULLUP, INVERT, DEBOUNCE_MS);
Button btnALR(ALR_PIN, PULLUP, INVERT, DEBOUNCE_MS);
enum {WAIT, INCR, DECR};
uint8_t STATE;
int count;
int lastCount = -1;
unsigned long rpt = REPEAT_FIRST;
```

```
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);
uint8_t dgrsChar[8] = {0x4, 0xa, 0x4, 0x0, 0x0, 0x0, 0x0, 0x0};
```

```
DS3231 rtc(SDA, SCL);
Time t;
```

```
uint32_t targetTime = 0;
uint8_t conv2d(const char* p) {
    uint8_t v = 0;
    if ('0' <= *p && *p <= '9')
        v = *p - '0';
    return 10 * v + *++p - '0';
}
```

```
uint8_t hh = 0, mm = 0, ss = 0, dd = 0, bb = 0, dy = 0;
int yy = 0;
String Day = " ";
uint8_t alarmHH = 9, alarmMM = 30;
int alarmLONG = 5;
uint8_t setMode = 0, setAlarm = 0, alarmMode = 0;
```

```
void setup() {
    Serial.begin(115200);
```

```
    gravityTds.setTemperature(temperature); // set the temperature and execute
        temperature compensation
    gravityTds.update(); //sample and calculate
    tdsValue = gravityTds.getTdsValue(); // then get the value
    Serial.print(tdsValue,0);
```

```
Serial.println("ppm");  
lcd.setCursor(0, 3);  
lcd.print("TDS Value:");  
lcd.setCursor(0, 4);  
lcd.print(tdsValue,0);  
lcd.print(" PPM");  
delay(1000);  
lcd.clear();
```

```
pinMode(trigPin, OUTPUT);  
pinMode(echoPin, INPUT);
```

```
pinMode(ledAman, OUTPUT);  
pinMode(led1, OUTPUT);  
pinMode(led2, OUTPUT);  
pinMode(led3, OUTPUT);
```

```
pinMode (13, OUTPUT);  
lcd.begin(20, 4);  
lcd.setBacklightPin(3, POSITIVE);  
lcd.setBacklight(HIGH);  
rtc.begin();  
lcd.createChar(0, dgrsChar);  
lcd.setCursor (0, 0);  
lcd.print(F(" Sistem "));  
lcd.setCursor (0, 1);  
lcd.print(F(" Hidroponik "));  
delay (2000);  
lcd.clear();  
digitalWrite(13, HIGH);  
delay (300);  
digitalWrite(13, LOW);  
delay (300);  
digitalWrite(13, HIGH);  
delay (300);  
digitalWrite(13, LOW);
```

```

    gravityTds.setPin(TdsSensorPin);
    gravityTds.setAref(5.0); //reference voltage on ADC, default 5.0V on Arduino
    UNO
    gravityTds.setAdcRange(1024); //1024 for 10bit ADC;4096 for 12bit ADC
    gravityTds.begin(); //initialization

    // rtc.setDOW (MINGGU);
    // rtc.setTime (23, 59, 55);    // Set Jam, Menit, Detik
    // rtc.setDate (14, 3, 2017);  // Set Tgl, Bulan, Tahun
}

void loop() {

    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    pulseTime = pulseIn(echoPin, HIGH);

    lcd.setCursor(0,3);
    lcd.print("Jarak = ");
    lcd.print(pulseTime / 58, DEC);
    lcd.print(" cm");
    lcd.setCursor(4,3);

    long duration, distance;
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distance = (duration/2) / 29.1;

    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);

```

```
pinMode(ledAman, OUTPUT);
pinMode(led1, OUTPUT);
pinMode(led2, OUTPUT);
pinMode(led3, OUTPUT);

if (distance >= 16)
{
  digitalWrite(ledAman, HIGH);
  digitalWrite(led1, LOW);
  digitalWrite(led2, LOW);
  digitalWrite(led3,LOW);
}
else {
  digitalWrite(ledAman,LOW);
}

if (distance <= 19)
{
  digitalWrite(led1, HIGH);
  digitalWrite(led2, LOW);
  digitalWrite(led3,LOW);
}

if (distance < 17) {
  digitalWrite(led2, HIGH);
  digitalWrite(led1, LOW);
  digitalWrite(led3,LOW);
}

if (distance < 9)
{
  digitalWrite(led3, HIGH);
  digitalWrite(led2, LOW);
  digitalWrite(led1, LOW);
}

delay(500);
```

```

t = rtc.getTime();
Day = rtc.getDOWStr(FORMAT_SHORT);

if (setMode == 0) {
    if (((rtc.getDOWStr(FORMAT_SHORT)) == "SEN")) {
        dy = 1;
    }
    if (((rtc.getDOWStr(FORMAT_SHORT)) == "SEL")) {
        dy = 2;
    }
    if (((rtc.getDOWStr(FORMAT_SHORT)) == "RAB")) {
        dy = 3;
    }
    if (((rtc.getDOWStr(FORMAT_SHORT)) == "KAM")) {
        dy = 4;
    }
    if (((rtc.getDOWStr(FORMAT_SHORT)) == "JUM")) {
        dy = 5;
    }
    if (((rtc.getDOWStr(FORMAT_SHORT)) == "SAB")) {
        dy = 6;
    }
    if (((rtc.getDOWStr(FORMAT_SHORT)) == "MIN")) {
        dy = 7;
    }
    hh = t.hour, DEC;
    mm = t.min, DEC;
    ss = t.sec, DEC;
    dd = t.date, DEC;
    bb = t.mon, DEC;
    yy = t.year, DEC;
}

if (setAlarm < 2) {
    lcd.setCursor(2, 0);
    if (setMode == 0)
        lcd.print((rtc.getDOWStr(FORMAT_SHORT))); // Hari
    else {

```

```

lcd.setCursor(2, 0);
lcd.print(Day); // Hari
lcd.setCursor(2, 0);
}
if (dy == 1) {           // hari
  lcd.print("SEN");
}
else if (dy == 2) {
  lcd.print("SEL");
}
else if (dy == 3) {
  lcd.print("RAB");
}
else if (dy == 4) {
  lcd.print("KAM");
}
else if (dy == 5) {
  lcd.print("JUM");
}
else if (dy == 6) {
  lcd.print("SAB");
}
else if (dy == 7) {
  lcd.print("MIN");
}

lcd.setCursor(5, 0);
lcd.print(", ");

if (dd < 10) {          // Tanggal
  lcd.print(F("0"));
}
lcd.print(dd); lcd.print(F(" "));

if (bb == 1) {         // Bulan
  lcd.print("JAN ");
}

```



```

}
lcd.print(hh); lcd.print(F(":"));
if (mm < 10) {
  lcd.print(F("0"));          // Menit
}
lcd.print(mm); lcd.print(F(":"));
if (ss < 10) {                // Detik
  lcd.print(F("0"));
}
lcd.print(ss); lcd.print(F(" ")); //
lcd.print(rtc.getTemp() - 2);    //
lcd.print((char)0);             // Temperatur
lcd.print(F("C"));              //
lcd.print(F(" "));              //

if (rtc.getTemp() > 40) {
  lcd.setCursor(0, 0);
  digitalWrite(13, HIGH);
  lcd.print(" Suhu Terlalu Panas ");
  delay (500);                  // Notifikasi Suhu Tinggi
  lcd.setCursor(0, 0);
  digitalWrite(13, LOW);
  lcd.print("          ");
  delay (500);
}
}

if ((hh == 0 && mm == 0 && ss == 0) || (hh == 1 && mm == 0 && ss == 0) ||
(hh == 2 && mm == 0 && ss == 0) ||
(hh == 3 && mm == 0 && ss == 0) || (hh == 4 && mm == 0 && ss == 0) ||
(hh == 5 && mm == 0 && ss == 0) ||
(hh == 6 && mm == 0 && ss == 0) || (hh == 7 && mm == 0 && ss == 0) ||
(hh == 8 && mm == 0 && ss == 0) ||
(hh == 9 && mm == 0 && ss == 0) || (hh == 10 && mm == 0 && ss == 0) ||
(hh == 11 && mm == 0 && ss == 0) ||
(hh == 12 && mm == 0 && ss == 0) || (hh == 13 && mm == 0 && ss == 0) ||
(hh == 14 && mm == 0 && ss == 0) ||
(hh == 15 && mm == 0 && ss == 0) || (hh == 16 && mm == 0 && ss == 0) ||

```

```

(hh == 17 && mm == 0 && ss == 0) ||
(hh == 18 && mm == 0 && ss == 0) || (hh == 19 && mm == 0 && ss == 0) ||
(hh == 20 && mm == 0 && ss == 0) ||
(hh == 21 && mm == 0 && ss == 0) || (hh == 22 && mm == 0 && ss == 0) ||
(hh == 23 && mm == 0 && ss == 0)) {
digitalWrite(13, HIGH);
delay (300);                // Buzzer Berbunyi Otomastis Setiap Jam
digitalWrite(13, LOW);
delay (300);
digitalWrite(13, HIGH);
delay (300);
digitalWrite(13, LOW);
delay (100);
}
}

setupClock();
Alarm (alarmHH, alarmMM, alarmLONG);

if (setAlarm < 2 && setMode != 0) {
  delay (100);
}
if (setAlarm < 2 && setMode == 1) {
  lcd.setCursor(2, 0);        // Set Hari Cursor
  lcd.print(F("  "));
  delay(100);
}
if (setAlarm < 2 && setMode == 2) {
  lcd.setCursor(7, 0);       // Set Tanggal Cursor
  lcd.print(F("  "));
  delay(100);
}
lcd.setCursor(10, 0);
if (setAlarm < 2 && setMode == 3) { // Set Bulan Cursor
  lcd.print(F("  "));
  delay(100);
}
}

```

```

if (setAlarm < 2 && setMode == 4) {
  lcd.setCursor(14, 0);          // Set Tahun Cursor
  lcd.print(F("  "));
  delay(100);
}
if (setAlarm < 2 && setMode == 5) {
  lcd.setCursor(2, 1);          // Set Jam Cursor
  lcd.print(F("  "));
  delay(100);
}
if (setAlarm < 2 && setMode == 6) {
  lcd.setCursor(5, 1);          // Set Menit Cursor
  lcd.print(F("  "));
  delay(100);
}
if (setAlarm < 2 && setMode == 7) {
  lcd.setCursor(8, 1);          // Set detik Cursor
  lcd.print(F("  "));
  delay(100);
}

if (setMode == 0 && setAlarm != 0 && setAlarm != 1) {
  delay (100);
}

// if (setMode == 0 && setAlarm == 1) {
//   lcd.clear();
//   lcd.setCursor(0, 0);          // Notifikasi Alarm OFF
//   lcd.print(F(" Alarm OFF  "));
//   delay(1500);
//   setMode = 0;
// }

if (setMode == 0 && setAlarm == 2) {
  lcd.setCursor(11, 0);          // Set Alarm Jam Cursor
  lcd.print(F("  "));
  delay(100);
}

```

```

if (setMode == 0 && setAlarm == 3) {
    lcd.setCursor(14, 0);           // Set Alarm Menit Cursor
    lcd.print(F(" "));
    delay(100);
}
if (setMode == 0 && setAlarm == 4 && alarmLONG > 9) {
    lcd.setCursor(11, 1);         // Set Lama Alarm Cursor
    lcd.print(F(" "));
    delay(100);
}
if (setMode == 0 && setAlarm == 4 && alarmLONG < 10) {
    lcd.setCursor(11, 1);         // Set Lama Alarm Cursor
    lcd.print(F(" "));
    delay(100);
}

Serial.print (setMode);
Serial.print (" ");
Serial.println (setAlarm);
}

void setupClock (void) {
    btnUP.read();
    btnDN.read();
    btnSET.read();
    btnALR.read();

    if (setMode == 8) {
        lcd.setCursor (0, 0);
        lcd.print (F(" PENGATURAN JAM "));
        lcd.setCursor (0, 1);
        lcd.print (F("  DI SIMPAN  "));
        digitalWrite(13, HIGH);
        delay (300);
        digitalWrite(13, LOW);
        delay (2000);
        rtc.setDOW (dy);
        rtc.setTime (hh, mm, ss);   // Simpan Pengaturan Jam ke RTC
    }
}

```

```

rtc.setDate (dd, bb, yy);
lcd.clear();
setMode = 0;
}

if (setAlarm == 5) {
  lcd.setCursor (0, 0);
  lcd.print (F("  ALARM DISIMPAN  "));
  lcd.setCursor (0, 1);
  if (alarmHH < 10) {
    lcd.print(F("0"));
  }
  lcd.print (alarmHH);
  lcd.print (F(":"));
  if (alarmMM < 10) {          // Alarm disimpan
    lcd.print(F("0"));
  }
  lcd.print (alarmMM);
  lcd.print (F(" LAMA ALARM "));
  lcd.print(alarmLONG); lcd.print(F("M"));
  digitalWrite(13, HIGH);
  delay (300);
  digitalWrite(13, LOW);
  delay (3000);
  lcd.clear();
  setAlarm = 0;
  alarmMode = 1;
}

if (setAlarm > 0) {
  alarmMode = 0;
}

switch (STATE) {

case WAIT:
  if (btnSET.wasPressed())
  {

```

```

    setMode = setMode + 1;
}
if (btnALR.wasPressed())
{
    setAlarm = setAlarm + 1;
}
if (btnUP.wasPressed())
    STATE = INCR;
else if (btnDN.wasPressed())
    STATE = DECR;
else if (btnUP.wasReleased())
    rpt = REPEAT_FIRST;
else if (btnDN.wasReleased())
    rpt = REPEAT_FIRST;
else if (btnUP.pressedFor(rpt)) {
    rpt += REPEAT_INCR;
    STATE = INCR;
}
else if (btnDN.pressedFor(rpt)) {
    rpt += REPEAT_INCR;
    STATE = DECR;
}
break;

```

case INCR:

```

if (setAlarm < 2 && setMode == 1 && dy < 7)dy = dy + 1;
if (setAlarm < 2 && setMode == 2 && dd < 31)dd = dd + 1;
if (setAlarm < 2 && setMode == 3 && bb < 12)bb = bb + 1;
if (setAlarm < 2 && setMode == 4 && yy < 2050)yy = yy + 1;
if (setAlarm < 2 && setMode == 5 && hh < 23)hh = hh + 1;
if (setAlarm < 2 && setMode == 6 && mm < 59)mm = mm + 1;
if (setAlarm < 2 && setMode == 7 && ss < 59)ss = ss + 1;
if (setMode == 0 && setAlarm == 2 && alarmHH < 23)alarmHH = alarmHH
+ 1;
if (setMode == 0 && setAlarm == 3 && alarmMM < 59)alarmMM =
alarmMM + 1;
if (setMode == 0 && setAlarm == 4 && alarmLONG < 59)alarmLONG =
alarmLONG + 1;

```

```
STATE = WAIT;
break;
```

```
case DECR:
```

```
if (setAlarm < 2 && setMode == 1 && dy > 0)dy = dy - 1;
if (setAlarm < 2 && setMode == 2 && dd > 0)dd = dd - 1;
if (setAlarm < 2 && setMode == 3 && bb > 0)bb = bb - 1;
if (setAlarm < 2 && setMode == 4 && yy > 2000)yy = yy - 1;
if (setAlarm < 2 && setMode == 5 && hh > 0)hh = hh - 1;
if (setAlarm < 2 && setMode == 6 && mm > 0)mm = mm - 1;
if (setAlarm < 2 && setMode == 7 && ss > 0)ss = ss - 1;
if (setMode == 0 && setAlarm == 2 && alarmHH > 0)alarmHH = alarmHH -
1;
if (setMode == 0 && setAlarm == 3 && alarmMM > 0)alarmMM = alarmMM
- 1;
if (setMode == 0 && setAlarm == 4 && alarmLONG > 0)alarmLONG =
alarmLONG - 1;
STATE = WAIT;
break;
```

```
}
}
```

```
void Alarm (uint8_t alarmHH, uint8_t alarmMM, int alarmLONG) {
```

```
if (alarmMode == 1 && hh == alarmHH && (mm - alarmMM >= 0 ) && (mm -
alarmMM <= alarmLONG )) {
digitalWrite(13, HIGH); // Menyalakan Alarm
delay (500);
digitalWrite (13, LOW);
delay (500);
if (ss % 2 == 0) {
lcd.noBacklight();
}
else {
lcd.backlight ();
}

btnALR.read();
```

```

if (btnALR.wasPressed())
{
    alarmMode = 0;
    lcd.backlight();
}
}

else {
    digitalWrite (13, LOW);
    lcd.backlight();
}

if (setMode == 0 && setAlarm != 0 && setAlarm != 1) {
    lcd.setCursor (0, 0);
    lcd.print(F("Set Alarm "));           // Menampilkan Set Jam Alarm
    if (alarmHH < 10) {
        lcd.print(F("0"));
    }
    lcd.print (alarmHH); lcd.print(F(":"));
    if (alarmMM < 10) {                   // Menampilkan Set Menit Alarm
        lcd.print(F("0"));
    }
    lcd.print (alarmMM); lcd.print(F(" WIB"));
    lcd.setCursor (0, 1);
    lcd.print(F("Lama Alarm "));         // Menampilkan Set Lama Alarm
    if (alarmLONG < 10) {
        lcd.print(F("0"));
    }
    lcd.print(alarmLONG); lcd.print(F(" Menit"));
}
}
}

```