

Lampiran 1 - Coding/Sketch Mikrokontroler pada Perancangan sistem kontrol turbin Berdasarkan Monitoring Level Air dan Debit Air

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

LiquidCrystal_I2C lcd(0x3F, 16, 4);
Servo motorServo;
/*****
* note:
* lcd-sda = A4
* lcd-scl = A5
*/

const int pin_buzzer = A0; // pin buzzer
const int trigger = A2; // pin trigger utk Sensor UltraSonic
const int echo = A3; // pin echo utk Sensor UltraSonic
const int LED_HIJAU= 8; // pin LED Hijau
const int LED_MERAH= 9; // pin LED Merah
const int pin_sensor_flow = 2; // pin data sensor flow
const int pin_servo = 4; // pin kontrol motor servo
const int tinggi_tiang = 37; //sesuaikan dengan aktual.

/***** Setting parameter bukaan Gate *****/
const int bukaan_default=45; // derajat bukaan awal kran
const int step_bukaan=10; // derajat per step
const int full_close=0; // derajat servo
const int full_open=90; // derajat servo
const int max_gate_openning=90; // derajat kran sesuai datasheet electricValve
const int max_debit=5; // Liter/Menit sesuai eksperimen
/***** Setting parameter bukaan Gate *****/
```

```

int count;

void setup() {

    Serial.begin(9600);
    lcd.begin();
    motorServo.attach(pin_servo);
    motorServo.write(full_close);
    pinMode(trigger, OUTPUT);
    pinMode(echo, INPUT);
    pinMode(pin_buzzer,OUTPUT);
    pinMode(LED_HIJAU,OUTPUT);
    pinMode(LED_MERAH,OUTPUT);
    lcd.backlight();
    initalisasi();
    lcd.clear();
    tampilkanDiLcd(0,0,"Monetoring Tando");
}
/**=====
 * === PROGRAM UTAMA =====
 * =====
 */
void loop() {
    int level = cekKedalaman();
    float debit = ambilDataDebit();

    if(level<10)action_dibawah10();
    if((level>=10)&&(level<=20))action_10sd20();
    if(level>20)action_diatas20();
    delay(100);
}

```

```

/**=====
 * --- END OF PROGRAM UTAMA -----
 * =====
 */

/** SUB PROGRAM MENGECEK LEVEL AIR **/
int cekKedalaman() {
long duration,jarak,kedalaman;
digitalWrite(trigger, LOW);
delayMicroseconds(10);
digitalWrite(trigger, HIGH);
delayMicroseconds(10);
digitalWrite(trigger, LOW);

while(digitalRead(echo));
duration = pulseIn(echo, HIGH); jarak = duration/29/2;
kedalaman = tinggi_tiang-jarak;
if(kedalaman<0)kedalaman=0;
delay(100);
return kedalaman; //
}

/** SUB PROGRAM MENGECEK DEBIT AIR **/
float ambilDataDebit() {
long perioda=millis();
count=0;
attachInterrupt(0,upCount,FALLING);
while(millis() - perioda < 500); detachInterrupt(0);
return (count*2/6); //nilai debit... freq/7.5, freq adalah jumlah pulse int dalam 1 detik

} // End Of ambilDataDebit()

```

```

void upCount(){
    count++;
}

/** SUB PROGRAM AKSI YANG DILAKUKAN JIKA LEVEL AIR DI BAWAH 2 m **/
void action_dibawah10(){
    int i=0;
    nyalakan_ledMerah();
    while(cekKedalaman()<11){
        tampilkan_kedalaman(cekKedalaman());
        tampilkan_debit(ambilDataDebit());
        tampilkan_gate(bukaan_default-step_bukaan*i);
        motorServo.write(bukaan_default-step_bukaan*i);
        if((bukaan_default-step_bukaan*i)<full_close){
            while(cekKedalaman()<10){
                motorServo.write(full_close);
                tampilkan_kedalaman(cekKedalaman());
                tampilkan_debit(ambilDataDebit());

                bunyikan_beeb(2);
                tampilkanDiLcd(5,3," ");
                tampilkanDiLcd(5,3,"!CutOff");
                delay(5000);
            }
        }
        i++;
        delay(5000);//aktualnya 5000
    }
    matikan_ledMerah();
}

```

```

/** SUB PROGRAM AKSI YANG DILAKUKAN JIKA LEVEL AIR ANTARA 2-3 m **/
void action_10sd20(){
    nyalakan_ledHijau();
    while((cekKedalaman())>=10)&&(cekKedalaman())<=20){
tampilkan_kedalaman(cekKedalaman());
        float debit = ambilDataDebit();
        tampilkan_debit(debit);
        float gate = (debit/max_debit)*max_gate_openning;
        // float rasio=debit*100;
        // float gate =map(rasio,0,(6*100),0,90);

        motorServo.write(full_close+gate);

    if
(gate>max_gate_openning){tampilkan_gate(max_gate_openning);motorServo.write(max_gate_openning);}else {tampilkan_gate(gate);}
        delay(500);
    }
    matikan_ledHijau();
} // End Of action_10sd20

/** SUB PROGRAM AKSI YANG DILAKUKAN JIKA LEVEL AIR DI BAWAH 2 m **/
void action_diatas20(){
    nyalakan_ledMerah();
    int i=0;
    while(cekKedalaman())>19){
tampilkan_kedalaman(cekKedalaman());
        tampilkan_debit(ambilDataDebit());
        tampilkan_gate(bukaan_default+step_bukaan*i);

        motorServo.write(bukaan_default+step_bukaan*i);

```

```

if((bukaan_default+step_bukaan*i)>full_open){
  while(cekKedalaman(>20){
    motorServo.write(full_open);
    tampilkan_kedalaman(cekKedalaman());
    tampilkan_debit(ambilDataDebit());
    bunyikan_beeb(3);
    tampilkanDiLcd(5,3,"  ");
    tampilkanDiLcd(5,3,"!!MAX!");
    delay(2000);
  }
}
i++;
delay(5000); //aktualnya 5000
}
matikan_ledMerah();
// End Of action_diatas20()

/** SUB PROGRAM UNTUK MEMEPERINDAH TAMPILAN **/
void inialisasi(){
  Serial.println("Start");
  lcd.clear();
  tampilkanDiLcd(0,0,"Sistem Pemantau");
  tampilkanDiLcd(0,1,"Debit&Level Air");
  tampilkanDiLcd(-2,2,"PLTA GOLANG");
  tampilkanDiLcd(0,3,"MADIUN");
  lcd.backlight();delay(3000);
  for(int i=0;i<7;i++){lcd.noBacklight();delay(100); lcd.backlight();delay(100);}
  delay(2000);
  lcd.clear();
}

void bunyikan_beeb(int x){

```

```

for (int i=0;i<x;i++){
digitalWrite(pin_buzzer,HIGH);
delay(50);
digitalWrite(pin_buzzer,LOW);
delay(50);
}
}

```

```

void tampilkanDiLcd(int x, int y, String text){
  lcd.setCursor(x,y);
  lcd.print(text);
}

```

```

void tampilkan_kedalaman(int level){
  tampilkanDiLcd(0,1,"");
  tampilkanDiLcd(0,1,"LevelAir:"+String(level)+" Cm");
}

```

```

void tampilkan_debit(float debit){
  tampilkanDiLcd(-4,2,"");
  tampilkanDiLcd(-4,2,"DebitAir:"+String(debit,2)+"L/M");
}

```

```

void tampilkan_gate(int derajat){
  char buff[16];
  tampilkanDiLcd(-4,3,"");
  sprintf(buff,"OpenGate:%d",derajat);
  tampilkanDiLcd(-4,3,buff);
}

```

```

void nyalakan_ledHijau(){
  digitalWrite(LED_HIJAU,HIGH);
}

```

```

void matikan_ledHijau(){
  digitalWrite(LED_HIJAU,LOW);
}

```



```
}  
void nyalakan_ledMerah(){  
    digitalWrite(LED_MERAH,HIGH);  
}  
void matikan_ledMerah(){  
    digitalWrite(LED_MERAH,LOW);  
}
```

