

**MAKALAH PROYEK SISTEM BENAM**



**UNIVERSITAS INDONESIA**

**MONITOR KEKERUHAN AIR**

**BERBASIS ARDUINO**

**Irfan Arif Maulana (1906379270)**

**TEKNIK ELEKTRO**

**UNIVERSITAS INDONESIA**

**DEPOK, 2021**

## ABSTRAK

Pada masa ini, tidak jarang perumahan di Indonesia yang mendapatkan sumber air dari sumur dengan menggunakan motor pompa air. Hal ini mengakibatkan diperlukannya penggunaan filter air dan perlu adanya sistem *backwash* berkala. Sistem monitoring tingkat kekeruhan air berbasis Arduino ini dapat mendeteksi dan mengukur tingkat kekeruhan air yang terkandung pada filter dan memberi informasi ke penghuni rumah terkait tingkat kekeruhan air tersebut. Seperti namanya, sistem ini menggunakan Arduino sebagai basis utama pengendaliannya. Disertakan pula sensor berupa sensor *turbidity* untuk mengukur tingkat kekeruhan air dan aktuator berupa LCD dan LED sebagai indikator kekeruhan.

Kata kunci: kekeruhan air, Arduino, sensor *turbidity*, aktuator LCD dan LED.

## DAFTAR ISI

HALAMAN JUDUL .....	i
ABSTRAK .....	ii
DAFTAR ISI .....	iii
DAFTAR GAMBAR.....	iv
DAFTAR TABEL .....	v
<b>BAB I. PENDAHULUAN .....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Perumusan Masalah .....	2
1.3. Tujuan Penelitian .....	2
1.4. Manfaat Penelitian .....	2
1.5. Batasan Penelitian .....	3
<b>BAB II. RANCANGAN SISTEM .....</b>	<b>4</b>
2.1. Dasar Teori.....	4
2.2. Arsitektur Sistem .....	5
2.3. Perangkat.....	6
2.4. Interkoneksi Perangkat pada Sistem .....	11
2.5. Algoritma Sistem .....	12
<b>BAB III. PENGUKURAN DAN HASIL .....</b>	<b>16</b>
3.1. Metode Pengukuran .....	16
3.2. Data Hasil Pengukuran .....	17
3.3. Grafik Hasil Pengukuran.....	19
3.4. Analisis Kerja Sistem.....	20
<b>BAB IV. PENUTUP.....</b>	<b>21</b>
4.1. Kesimpulan .....	21
4.2. Saran .....	21
DAFTAR REFERENSI.....	22

## DAFTAR GAMBAR

Gambar 2.1. Arsitektur Sistem Monitor Kekeruhan Berbasis Arduino .....	5
Gambar 2.2. Arduino UNO R3 <i>Board</i> .....	6
Gambar 2.3. Sensor <i>Turbidity</i> .....	6
Gambar 2.4. Grafik Hubungan Turbidity terhadap Tegangan.....	8
Gambar 2.5. <i>A/D Signal Converter</i> .....	8
Gambar 2.6. LCD 20x4 I2C .....	9
Gambar 2.7. LED .....	9
Gambar 2.8. <i>Breadboard</i> .....	10
Gambar 2.9. <i>Jumper Cables</i> .....	10
Gambar 2.10. Skema Arduino, Sensor, dan LCD .....	11
Gambar 2.11. Skema Arduino dan LED .....	12
Gambar 2.12. <i>Flowchart</i> Sistem.....	14
Gambar 2.13. Program Sistem Bagian <i>Setup</i> .....	14
Gambar 2.14. Program Sistem Bagian <i>Loop</i> .....	15
Gambar 3.1. Data Serial Monitor Air Jernih .....	17
Gambar 3.2. Data Serial Monitor Air Sedikit Keruh.....	17
Gambar 3.3. Data Serial Monitor Air Keruh Sekali .....	18
Gambar 3.4. Grafik Serial Plotter Air Jernih.....	19
Gambar 3.5. Grafik Serial Plotter Air Sedikit Keruh .....	19
Gambar 3.6. Grafik Serial Plotter Air Keruh Sekali .....	20

## DAFTAR TABEL

Tabel 2.1. Interkoneksi Perangkat .....	12
---	----

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Air merupakan salah satu kebutuhan pokok manusia. Sebab air menjadi sumber kehidupan, yaitu untuk minum serta untuk kebutuhan higienis atau kebersihan.

Pada masa kini, dalam memenuhi kebutuhan mencuci atau menjaga kebersihan, umumnya di Indonesia air yang diperoleh pada perumahan dapat bersumber dari dua cara. Yaitu pertama bersumberkan dari penyaluran air oleh PDAM atau Perusahaan Daerah Air Minum yang mana merupakan sebuah BUMD yang bergerak di bidang pelayanan air minum. Ada pun sumber kedua yang umum di Indonesia adalah menggunakan sumur sendiri.

Sistem perolehan air melalui sumur pribadi ini menggunakan sebuah pompa atau motor yang menggerakkan air tanah ke tangki penyimpanan. Air yang didapat ini kemudian disaring bersih agar terhindar dari kotoran-kotoran yang terdapat pada tanah seperti lumut, lumpur, debu, dan sebagainya. Proses penyaringan air ini dilakukan melalui sebuah saringan atau filter air. Seiring berjalannya waktu, filter air ini akan mengalami penyumbatan saluran air akibat lumut dan kotoran tanah lainnya yang terkoleksi secara akumulatif pada filter air. Akibatnya, debit air yang diteruskan dari tangki penyimpanan air ke rumah menjadi sedikit. Untuk menyelesaikan masalah ini, diperlukan suatu proses yang dinamakan sebagai *backwash* atau mengalirkan air dari tangki penyimpanan ke filter namun dari arah yang berlawanan. Dengan begitu, air akan mendorong kotoran ada filter dan membuka kembali saluran air yang asalnya tersumbat. Kotoran yang didapat pada proses *backwash* ini dibuang melalui pipa khusus agar tidak tercampur dengan pipa yang mengarah pada sistem pipa rumah.

Di sisi lain, mengingat kembali perkembangan teknologi, proses *backwash* tersebut akan sangat baik jika diintegrasikan dengan suatu sistem kendali. Baik itu menggunakan sebuah *controller* atau komputer sederhana yang menjadikannya

sebuah perangkat berbasis IoT. Ataupun dengan sistem *automation* yang memiliki manfaat berupa memudahkan kehidupan manusia.

Salah satu teknologi yang tepat akan hal tersebut yang sedang menjadi *trend* masa kini adalah menggunakan Arduino, sebuah perangkat keras yang disertakan dengan perangkat lunak programming yang dapat memberdayakan sinyal analog maupun digital dari suatu sensor untuk kemudian diberikan kepada aktuator. Dengan menggunakan Arduino, proses otomasi pada kehidupan sehari-hari menjadi lebih mudah dan terjangkau mengingat harganya yang relatif murah.

## **1.2. Perumusan Masalah**

1. Apakah sistem berbasis Arduino dapat mendeteksi penyumbatan saluran air pada filter air?
2. Apakah sistem berbasis Arduino dapat memberikan informasi terkait penyumbatan saluran air pada filter air ke penghuni rumah?

## **1.3. Tujuan Penelitian**

1. Mengukur tingkat penyumbatan air yang ada pada filter air; dan
2. Memberikan pemberitahuan kepada penghuni rumah terkait penyumbatan air pada filter air.

## **1.4. Manfaat Penelitian**

1. Mengedukasi secara singkat tentang Arduino *board* dan Arduino IDE;
2. Mengedukasi secara singkat tentang sensor dan aktuator yang berkaitan dengan sistem yang dirancang;

3. Memberikan pengetahuan tentang cara perangkaian Arduino serta sensor dan aktuator; dan
4. Memberikan solusi untuk masalah yang telah dirumuskan.

#### **1.5. Batasan Penelitian**

1. Sensor yang digunakan merupakan sensor tingkat kekeruhan atau kejernihan air mengakibatkan sistem ini tidak dapat digunakan untuk semua jenis filter air.



## **BAB II**

### **RANCANGAN SISTEM**

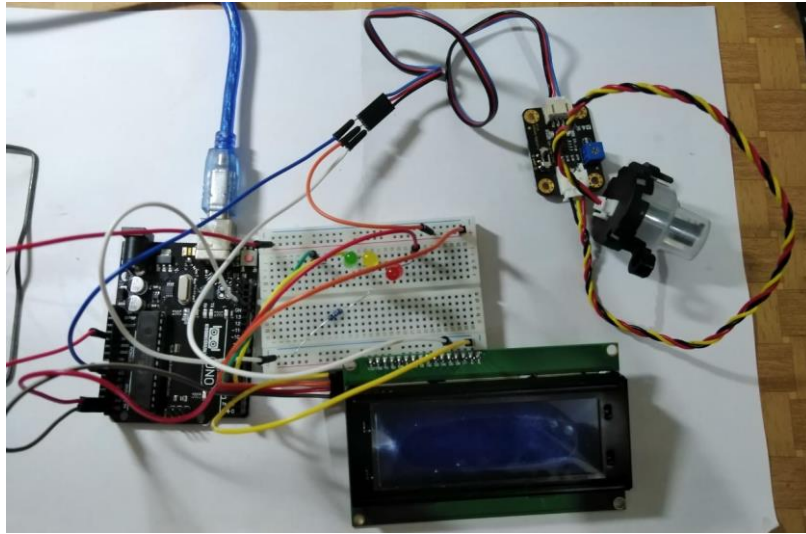
#### **2.1. Dasar Teori**

Sistem berbasis Arduino yang dirancang pada makalah ini mengukur tingkat kekeruhan air. Dalam mengukur tingkat kekeruhan air sendiri terdapat suatu parameter yang harus dipertimbangkan. Satuan standar global yang digunakan untuk mengukur kekeruhan air adalah NTU atau *Nephelometric Turbidity Units*. Di mana semakin besar nilai NTU, semakin keruh pula sampel air yang sedang diukur.

Pada makalah ini, demi keserhanaan pembahasan dasar teori, pembahasan nilai NTU hanya sebatas hubungan nilai NTU itu sendiri terhadap kekeruhan sampel air yang diukur. Selebihnya, terkait hubungan nilai NTU terhadap penggunaan air untuk kehidupan sehari-hari, metode pengukuran NTU, serta faktor-faktor penyebab nilai NTU dapat dilihat pada dokumen resmi dari WHO (World Health Organization) yang dilampirkan pada bagian referensi makalah ini. Namun, makalah ini akan tetap membahas sedikit tentang cara kerja sensor *turbidity* yang digunakan.

Pada sistem ini, pengukuran nilai NTU dilakukan dengan menggunakan sensor *turbidity* yang terhubung dengan Arduino. Selebihnya cara kerja sensor *turbidity* ini akan dibahas pada bagian arsitektur sistem.

## 2.2. Arsitektur Sistem



**Gambar 2.1.** Arsitektur Sistem Monitor Kekeruhan Air Berbasis Arduino

Alat pada sistem monitor kekeruhan air berbasis Arduino ini memiliki tiga bagian utama, yaitu Arduino itu sendiri, sensor, serta aktuator. Otak dari sistem ini terletak pada Arduino yang mana berfungsi untuk mengoperasikan dan menghubungkan interkoneksi antara sensor dan aktuator. Sistem ini memperoleh input atau masukan berupa sinyal analog yang diukur oleh sensor *turbidity*. Sinyal analog ini kemudian diproses melalui Arduino untuk dikalkulasi serta dievaluasi seberapa tinggi tingkat kekeruhan air yang telah diukur. Kemudian berdasarkan evaluasi tadi, Arduino akan mengirimkan sinyal ke aktuator untuk memberikan data atau informasi terkait tingkat kekeruhan air tersebut. Aktuator yang digunakan ada dua, yaitu LCD dan LED.

Perkabelan pada sistem ini juga dihubungkan dengan *breadboard* untuk mempermudah manajemen kabel yang digunakan. Terutama untuk mempermudah pembagian sumber tegangan VCC +5V maupun GND untuk perangkat-perangkat yang digunakan. Untuk mempermudah pula, LED yang digunakan dipasang pada *breadboard* tersebut.

### 2.3. Perangkat

1. Arduino UNO R3 ATmega328 / ATmega16U2



**Gambar 2.2.** Arduino UNO R3 Board

Digunakan sebagai pusat pengendalian atau kontrol dari sistem yang mana akan memberikan perintah untuk sistem interkoneksi sensor dan aktuator.

2. Arduino Turbidity Sensor SKU SEN0189



**Gambar 2.3.** Sensor Turbidity

Digunakan untuk mengukur tingkat kekeruhan air melalui satuan NTU dan memberikan sinyal listrik sebagai keluarannya.

Sensor ini dimanufaktur oleh DFRobot beserta dengan *converter* sinyal analog/digitalnya. Secara singkatnya, sensor ini bekerja dengan cara mengirimkan atau men-*transmit* sinyal dari salah satu lengannya untuk kemudian diterima oleh lengan satunya. Sinyal yang dikirimkan ini dapat berinteraksi dengan partikel yang terlarut dalam air sehingga mengakibatkan disturbansi terhadap sinyal. Berdasarkan disturbansi ini, maka tingkat kekeruhan air dapat diukur. Semakin

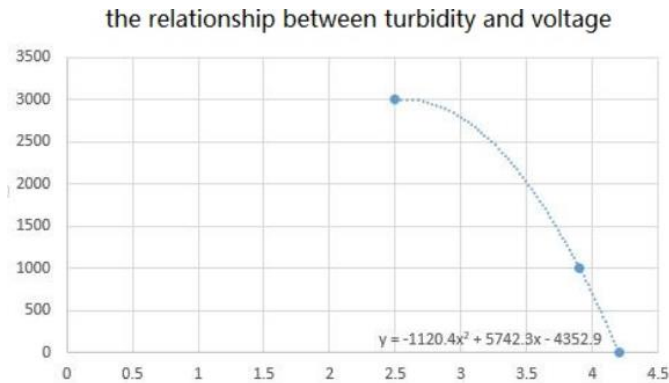
besar disturbansi sinyal ini, semakin keruh atau semakin tinggi nilai NTU, akibatnya sinyal tegangan yang dikeluarkan oleh sensor ini semakin kecil.

Perlu diketahui bahwa kerja sensor ini dipengaruhi oleh suhu sampel air yang digunakan. Untuk mengeliminasi perubahan data akibat parameter yang tidak diperlukan tersebut, maka dibutuhkan variabel kontrol berupa suhu sampel yang konstan, yaitu pada suhu ruangan.

Terkait spesifikasi, sensor *turbidity* ini memiliki spesifikasi sebagai berikut:

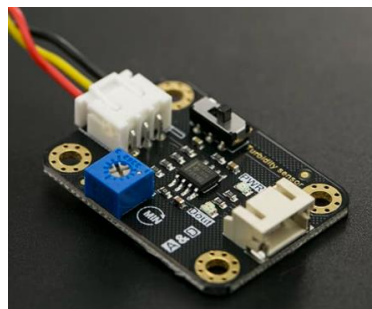
- Tegangan kerja : 5V DC
- Arus kerja : 40mA (MAX)
- Waktu respon : <500ms
- Resistansi insulasi : 100M (MIN)
- Metode keluaran : 0 - 4,5V Analog dan HIGH/LOW Digital
- Temperatur kerja : 5 hingga 90 derajat Celsius
- Temperatur penyimpanan : -10 hingga 90 derajat Celcius
- Berat : 30g

Dikutip dari wiki DFRobot mengenai sensor *turbidity*, terdapat hubungan antara level *turbidity* dengan tegangan keluaran yang dihasilkan oleh sensor ini. Hubungan tersebut dapat dilihat pada grafik berikut:



**Gambar 2.4.** Grafik Hubungan *Turbidity* terhadap Tegangan

### 3. Analog/Digital Signal Converter



**Gambar 2.5.** A/D Signal Converter

Digunakan untuk dihubungkan ke sensor *turbidity*. Sinyal yang diberikan oleh sensor *turbidity* kemudian dapat diteruskan oleh *converter* ini untuk diteruskan sinyalnya ke Arduino. *Converter* ini dilengkapi dengan dua mode pengiriman sinyal keluaran, yaitu sinyal analog dan digital. Pada makalah ini, mode yang digunakan adalah mode analog. Selain itu, *converter* ini juga dilengkapi dengan *potentiometer* yang digunakan untuk mengkalibrasi sinyal yang didapat dan diteruskan ke Arduino. Dalam kata lain, mengatur tingkat sensitivitas sensor terhadap pengukurannya. Perlu diketahui, bahwa *potentiometer* ini hanya dapat digunakan pada mode digital.

Bersamaan dengan sensor *turbidity*, *converter* sinyal ini juga dimanufaktur oleh DFRobot.

#### 4. LCD 20x4 I2C with Serial Interface Module



**Gambar 2.6.** LCD 20x4 I2C

Digunakan untuk menampilkan keluaran nilai NTU dan tegangan yang terukur oleh sensor *turbidity*.

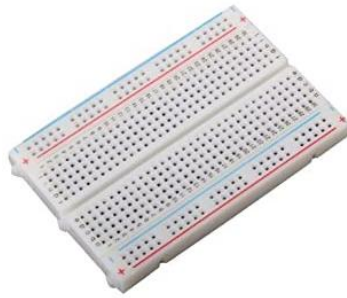
#### 5. LED



**Gambar 2.7.** LED

Digunakan untuk menampilkan tingkat kekeruhan air yang terukur melalui perbedaan warna. Di mana hijau berarti air yang relatif jernih, kuning berarti air keruh sedang, dan merah berarti air keruh sekali.

## 6. Breadboard



**Gambar 2.8.** *Breadboard*

Digunakan untuk memudahkan pembagian sumber tegangan VCC +5V dan GND ke perangkat-perangkat yang digunakan serta untuk memudahkan pula peletakan LED yang digunakan.

## 7. Jumper Cables



**Gambar 2.9.** *Jumper Cables*

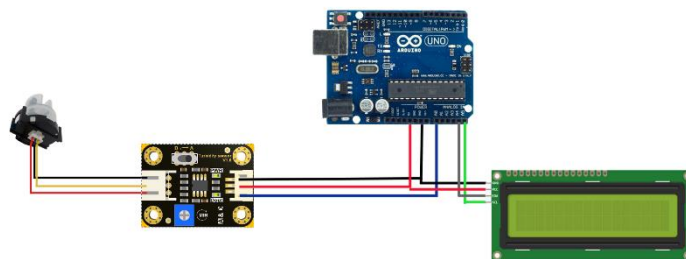
Digunakan untuk menghubungkan perangkat-perangkat yang digunakan ke Arduino maupun ke *breadboard*.

## 2.4. Interkoneksi Perangkat pada Sistem

Secara singkatnya, sistem ini menghubungkan Arduino dengan sensor dan aktuatornya, LED dan LCD. Untuk mempermudah pembagian sumber tegangan VCC dan GND, maka digunakan sebuah *breadboard* yang menghubungkan anoda dan katoda dari LED, LCD, serta sensor yang digunakan. Pada *breadboard* tersebut juga ketiga LED diletakkan. Jenis LCD yang digunakan menjadi sebuah perhatian di sini. Di mana LCD sudah terkoneksi dengan I2C. Hal ini diterapkan agar mengurangi perkabelan yang mesti dilakukan. LCD biasa akan membutuhkan proses *solder* dengan menghubungkan 16 kabel atau pin. Sedangkan dengan adanya integrasi dengan I2C, kabel yang mesti dihubungkan ke Arduino dikurangi menjadi hanya 4 buah.

Pin keluaran analog dari sensor dihubungkan ke pin input analog A0 pada Arduino. Ada

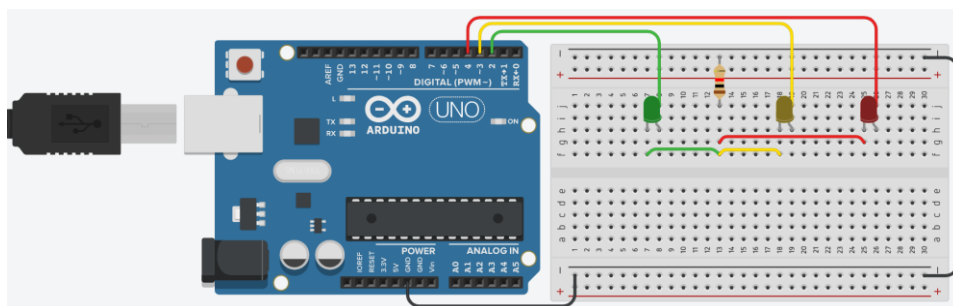
pun pada SCL dan SDA I2C LCD dihubungkan ke input analog A5 dan A4 secara berurutan. Meskipun VCC dan GND dari sensor dan LCD VCC dirangkai pada *breadboard*, namun dapat digambarkan secara sederhana melalui skema rangkaian berikut:



**Gambar 2.10.** Skema Arduino, Sensor, dan LCD

Ada pula rangkaian Arduino dengan ketiga LED yang digunakan dirangkai menggunakan *breadboard*. LED berwarna hijau, kuning, dan merah secara berurutan dihubungkan masing-masing anodanya ke pin digital 2, 3, dan 4. Rangkaian ini dapat direpresentasikan dengan skema rangkaian berikut:





**Gambar 2.11.** Skema Arduino dan LED

Berikut merupakan hubungan pin dan kabel semua komponen yang ada direpresentasikan dalam tabel:

**Tabel 2.1.** Interkoneksi Komponen

Arduino UNO R3	Sensor <i>Turbidity</i>	LCD I2C	LED Hijau	LED Kuning	LED Merah
VCC +5V	VCC	VCC			
GND	GND	GND	<i>Cathode</i>	<i>Cathode</i>	<i>Cathode</i>
A0	<i>Signal</i>				
A4		SDA			
A5		SCL			
D2			<i>Anode</i>		
D3				<i>Anode</i>	
D4					<i>Anode</i>

## 2.5. Algoritma Sistem

Ide sederhana dari algoritma yang diterapkan pada sistem ini adalah pertama, sistem akan meminta input melalui sensor *turbidity*. Sinyal input masukan ini berupa sinyal analog. Sinyal ini kemudian diproses dengan proses matematika pada Arduino sekaligus dievaluasi nilai yang didapat dari hasil perhitungan tadi. Evaluasi nilai ini berupa penentuan berapa besar nilai NTU yang terukur berdasarkan persamaan yang menghubungkan nilai NTU dengan besar tegangan yang terukur yang sudah ditampilkan pada Gambar 2.3. Evaluasi ini juga menentukan salah satu dari ketiga

LED yang dinyalakan. Terakhir, nilai NTU yang didapat beserta tegangan yang terukur akan ditampilkan melalui LCD. Selain itu, untuk kepentingan *debug*, dilakukan pula perintah *mem-print* nilai-nilai yang terukur pada Serial Monitor pada Arduino.

Oleh karena fluktuasi nilai yang terukur oleh sensor sangat mungkin terjadi, maka proses pengambilan data dapat dilakukan dengan mengumpulkan data sebanyak jumlah yang sudah ditentukan kemudian merata-ratakan nilai tersebut. Banyaknya sampel yang diambil ini ditentukan oleh *programmer* melalui Arduino IDE.

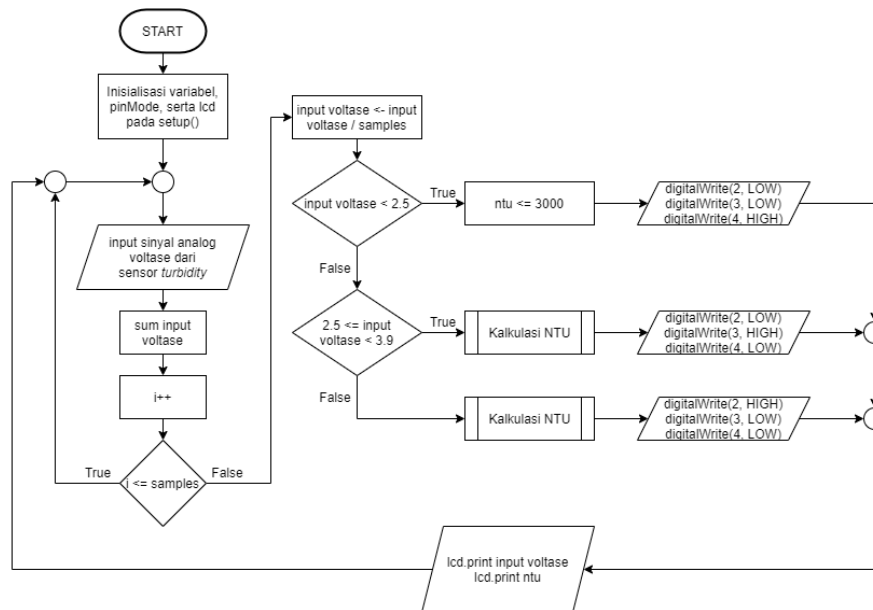
Melihat kembali hubungan NTU dengan besar tegangan yang terukur yang diperlihatkan pada Gambar 2.3., didapat bahwa persamaan NTU terhadap tegangan adalah sebagai berikut:

$$NTU = -1120.4V_{in}^2 + 5742.3V_{in} - 4352.9$$

Di mana  $V_{in}$  merupakan tegangan masukan analog yang terukur oleh sensor *turbidity*.

Perlu diketahui bahwa Arduino memproses sinyal analog dengan data dengan ukuran 10bit. Sedangkan nilai yang diperlukan untuk pengolahan data merupakan besar tegangan, maka dilakukan sebuah konversi nilai dari skala 0 hingga 1023 ( $2^{10}$ ) menjadi 0V hingga 5V. Selain itu, untuk membatasi jumlah angka yang ditampilkan pada LCD, maka dilakukan fungsi yang membulatkan angka ke maksimal sebanyak 2 angka di belakang koma. Hal ini dilakukan sebab terdapat keterbatasan karakter yang bisa ditampilkan pada LCD serta untuk memudahkan pengamatan.

Berdasarkan deskripsi algoritma yang sudah dikemukakan tadi, algoritma ini dapat direpresentasikan dengan *flowchart* sebagai berikut:



**Gambar 2.12.** *Flowchart* Sistem

Oleh karena terdapat perangkat I2C, maka pada bagian program mesti dimasukkan *library* yang bersesuaian. Dalam hal ini, *library* yang digunakan adalah Wire.h dan LiquidCrystal\_I2C.h.

Dengan mengimplementasikan algoritma tersebut ke *program* pada Arduino IDE, maka didapat *program* seperti berikut:

Bagian header dan fungsi *setup*:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 20, 4); // 0x27: I2C address, 20: columns, 4: rows

const int analogInPin = A0;
float inputVoltage;
int samples = 600;
float ntu; // Nephelometric Turbidity Units

void setup() {
  Serial.begin(9600);
  pinMode(analogInPin, INPUT);

  pinMode(2, OUTPUT); // GREEN
  pinMode(3, OUTPUT); // YELLOW
  pinMode(4, OUTPUT); // RED

  lcd.init(); // Initializes the LCD
  lcd.backlight(); // Activates backlight
  lcd.home();
}
```

**Gambar 2.13.** Program Sistem Bagian *Setup*

Bagian fungsi *loop* dan fungsi modular:

```
void loop() {
  inputVoltage = 0;

  // Averages 600 samples
  for(int i = 0; i < samples; i++) {
    inputVoltage += (float)analogRead(analogInPin) * (5.0 / 1023.0);
  }

  inputVoltage = inputVoltage/samples;
  inputVoltage = decRound(inputVoltage, 2);

  // Calculates NTU value
  if(inputVoltage < 2.5){
    ntu = 3000;
    digitalWrite(2, LOW);
    digitalWrite(3, LOW);
    digitalWrite(4, HIGH);
  } else if(inputVoltage >= 2.5 && inputVoltage < 3.9) {
    ntu = -1120.4 * square(inputVoltage) + 5742.3 * inputVoltage - 4352.9;
    digitalWrite(2, LOW);
    digitalWrite(3, HIGH);
    digitalWrite(4, LOW);
  } else {
    ntu = -1120.4 * square(inputVoltage) + 5742.3 * inputVoltage - 4352.9;
    digitalWrite(2, HIGH);
    digitalWrite(3, LOW);
    digitalWrite(4, LOW);
  }

  digitalWrite(4, LOW);
}

// For debugging purposes
Serial.print("Voltage: ");
Serial.println(inputVoltage);
Serial.print("NTU: ");
Serial.println(ntu);
Serial.print("\n");

// Prints Voltage to LCD
lcd.clear();
lcd.setCursor(0,0);
lcd.print(inputVoltage);
lcd.print(" V");

// Prints NTU to LCD
lcd.setCursor(0,1);
lcd.print(ntu);
lcd.print(" NTU");
delay(10);

float decRound(float input, int decPlace) {
  float multiplier = powf(10.0f, decPlace);
  input = roundf(input * multiplier) / multiplier;

  return input;
}
```

**Gambar 2.14.** Program Sistem Bagian *Loop*

## BAB III

### PENGUKURAN DAN HASIL

#### 3. 1. Metode Pengukuran

Terdapat dua buah parameter yang menjadi perhatian utama pada sistem ini, parameter tersebut merupakan nilai tegangan yang terukur pada sensor *turbidity* serta nilai NTU dari hasil perhitungan. Kedua nilai dari parameter ini dapat diukur dengan dua cara, yaitu dengan melihat langsung nilai yang ditampilkan oleh LCD atau dengan melihat Serial Monitor pada Arduino IDE. Pada makalah ini, ditetapkan bahwa cara yang lebih baik untuk melihat dan mengoleksi adalah dengan memperhatikan nilai yang ditampilkan pada Serial Monitor maupun Serial Plotter pada Arduino IDE. Sebab Serial Monitor maupun Serial Plotter menunjukkan nilai-nilai sejak Arduino berjalan. Sedangkan data yang ditunjukkan LCD hanya pada waktu tertentu saja.

Ketika pengukuran, diberikan tiga buah sampel air dengan tingkat kekeruhan yang berbeda sebagai variabel bebas percobaan. Yaitu, air jernih, air sedikit keruh, dan air keruh sekali. Sensor *turbidity* akan dipindahkan letak pengukurannya berdasarkan ketiga sampel ini agar variabel terikat yang dihasilkan dapat bervariasi.

### 3. 2. Data Hasil Pengukuran

Data hasil pengukuran yang didapat dibagi menjadi tiga buah data, yaitu berdasarkan ketiga sampel air dengan tingkat kekeruhan yang berbeda-beda.

- Air jernih

```
Voltage: 4.17  
NTU: 109.97  
  
Voltage: 4.17  
NTU: 109.97  
  
Voltage: 4.17  
NTU: 109.97  
  
Voltage: 4.17  
NTU: 109.97  
  
Voltage: 4.17  
NTU: 109.97  
  
Voltage: 4.17  
NTU: 109.97  
  
Voltage: 4.17  
NTU: 109.97  
  
Voltage: 4.17  
NTU: 109.97
```

**Gambar 3.1.** Data Serial Monitor Air Jernih

- Air sedikit keruh sedikit

```
Voltage: 3.70  
NTU: 1555.33  
  
Voltage: 3.71  
NTU: 1529.73  
  
Voltage: 3.70  
NTU: 1555.33  
  
Voltage: 3.71  
NTU: 1529.73  
  
Voltage: 3.70  
NTU: 1555.33  
  
Voltage: 3.70  
NTU: 1555.33  
  
Voltage: 3.70  
NTU: 1555.33  
  
Voltage: 3.70  
NTU: 1555.33
```

**Gambar 3.2.** Data Serial Monitor Air Sedikit Keruh

- Air keruh sekali

```

Voltage: 2.02
NTU: 3000.00

Voltage: 2.01
NTU: 3000.00

Voltage: 2.01
NTU: 3000.00

Voltage: 2.01
NTU: 3000.00

Voltage: 2.00
NTU: 3000.00

Voltage: 2.00
NTU: 3000.00

Voltage: 1.99
NTU: 3000.00

Voltage: 1.99
NTU: 3000.00

```

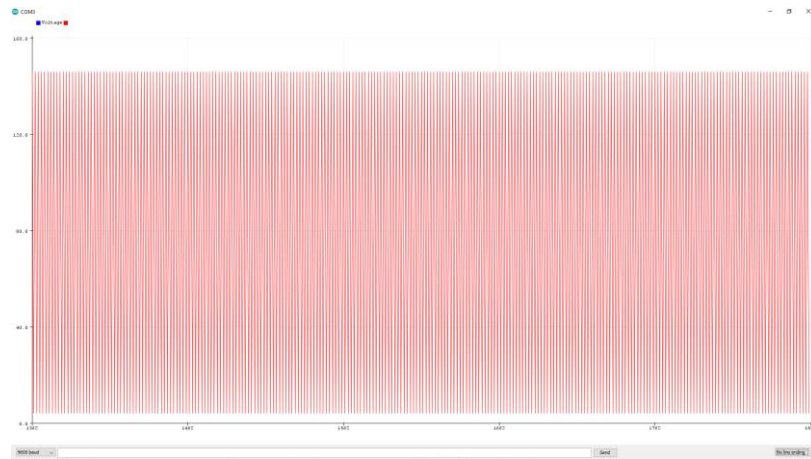
**Gambar 3.3.** Data Serial Monitor Air Keruh Sekali

Dari data-data yang didapat dari ketiga sampel air tersebut, terlihat bahwa secara berurutan, dari sampel air yang paling jernih ke yang paling keruh, terdapat variasi variabel terikat yang dihasilkan. Di mana semakin keruh sampel air yang diberikan, semakin kecil tegangan keluaran oleh sensor *turbidity*. Sedangkan nilai NTU atau tingkat kekeruhan air semakin tinggi yang menandakan air semakin keruh.

### 3.3. Grafik Hasil Pengukuran

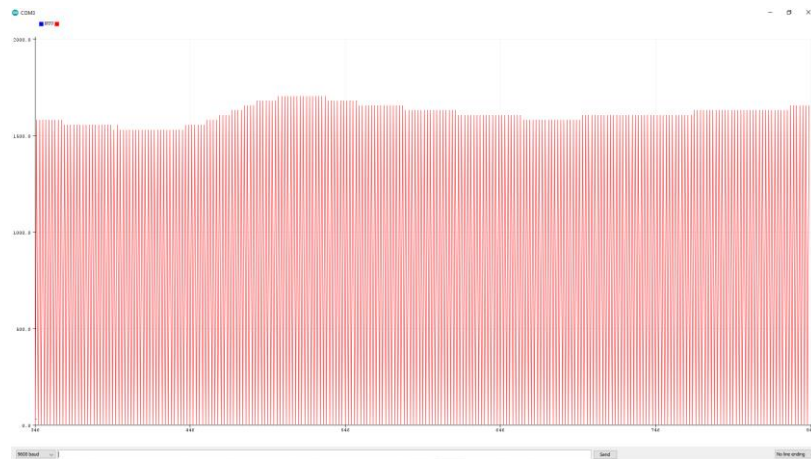
Sama seperti data hasil pengukuran, grafik hasil pengukuran dibagi menjadi tiga buah sesuai dengan tiga sampel yang digunakan, yaitu berdasarkan tingkat kekeruhan air yang berbeda-beda.

- Air jernih



**Gambar 3.4.** Grafik Serial Plotter Air Jernih

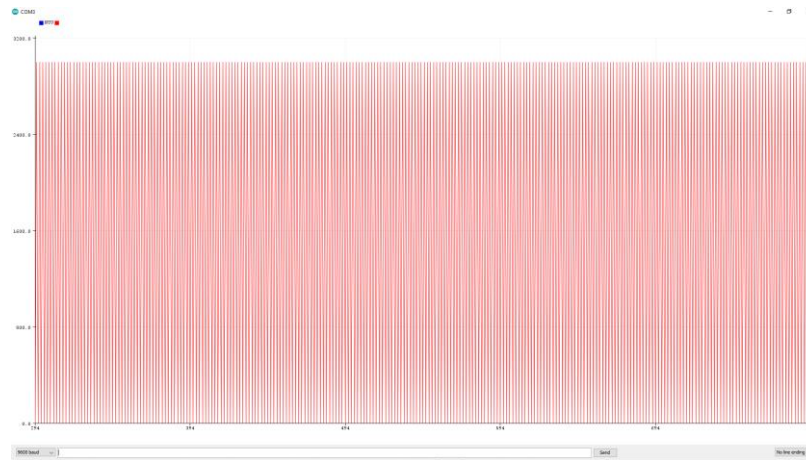
- Air keruh sedikit



**Gambar 3.5.** Grafik Serial Plotter Air Sedikit Keruh



- Air keruh sekali



**Gambar 3.6.** Grafik Serial Plotter Air Keruh Sekali

Ketiga grafik dari ketiga sampel air tersebut menunjukkan dua buah data, yaitu data tegangan keluaran oleh sensor *turbidity* bersamaan dengan nilai NTU dari sampel air. Hanya saja, keduanya memiliki rentang nilai yang sangat berbeda. Di mana nilai NTU berada pada skala 0 – 3000 sedangkan nilai tegangan pada skala 0 – 5. Oleh karena itu, seolah-olah hanya nilai NTU-nya saja yang terlihat yang mana direpresentasikan oleh garis merah. Ada pun garis biru yang merepresentasikan nilai tegangan berada pada tidak terlihat.

Sama seperti pada bagian data hasil pengamatan, hubungan tingkat kekeruhan pada sampel air terhadap nilai tegangan maupun NTU juga terlihat di sini.

### 3. 4. Analisis Kerja Sistem

Melalui analisis data dan grafik yang telah dipaparkan pada bagian sebelumnya, dapat dikatakan alat ini telah memiliki kerja yang cukup bersesuaian dengan keinginan. Sistem ini berhasil memperlihatkan hubungan variabel bebas dengan variabel terikat yang sesuai. Di mana kekeruhan air memiliki hubungan terbalik terhadap nilai tegangan yang dikeluarkan. Kekeruhan air juga memiliki hubungan senilai dengan nilai NTU.

## **BAB IV**

### **PENUTUP**

#### **4. 1. Kesimpulan**

Sistem monitor kekeruhan air berbasis Arduino dibuat untuk memonitor tingkat kekeruhan air dengan memanfaatkan sensor *turbidity* dan menampilkan informasi terkait sinyal tegangan yang dikeluarkan oleh sensor pada sebuah LCD bersamaan dengan nilai NTU atau tingkat kekeruhan air. Digunakan pula tiga buah LED dengan warna-warna yang berbeda sebagai indikator level kekeruhan air yang diukur. Sistem ini berhasil memperlihatkan hubungan variabel bebas dan variabel terikat yang bersesuaian dengan teori. Di mana semakin keruh air, semakin rendah sinyal tegangan yang dikeluarkan oleh sensor, namun semakin tinggi pula nilai NTU.

#### **4. 2. Saran**

Sistem monitor kekeruhan air ini menggunakan sensor *turbidity* yang mana penerapannya masih kurang tepat sehingga tidak semua jenis filter dapat dipasang sistem ini. Penggunaan sensor yang paling umum pada industri untuk permasalahan ini adalah sensor tekanan air. Penulis masih perlu mempelajari lebih lanjut mengenai penerapan sensor yang tepat untuk permasalahan keteknikan pada bidang ini.

## DAFTAR REFERENSI

- World Health Organization. (2017). *Water Quality and Health – Review of Turbidity: Information for regulators and water suppliers*. [online] Available at: [https://www.who.int/water\\_sanitation\\_health/publications/turbidity-information-200217.pdf](https://www.who.int/water_sanitation_health/publications/turbidity-information-200217.pdf) [Accessed 1 July 2021].
- World Health Organization. *Turbidity Measurement*. [online] Available at: [https://www.who.int/water\\_sanitation\\_health/hygiene/emergencies/fs2\\_33.pdf](https://www.who.int/water_sanitation_health/hygiene/emergencies/fs2_33.pdf) [Accessed 1 July 2021].
- Lenntech. (2021). *Turbidity*. [online] Available at: <https://www.lenntech.com/turbidity.htm> [Accessed 1 July 2021].
- DFRobot. *Turbidity Sensor SKU SEN0189*. [online] Available at: [https://wiki.dfrobot.com/Turbidity\\_sensor\\_SKU\\_\\_SEN0189](https://wiki.dfrobot.com/Turbidity_sensor_SKU__SEN0189) [Accessed 30 June 2021].
- Fahad, Engr. (2020). *Turbidity Sensor with Arduino for Quality Monitoring, Turbidity Meter*. [online] Available at: <https://www.electronicclinic.com/turbidity-sensor-with-arduino-for-water-quality-monitoring-turbidity-meter/> [Accessed 30 June 2021].
- Rajesh. (2020). *Measuring Turbidity of Water to Determine Water Quality using Arduino and Turbidity Sensor*. [online] Available at: <https://circuitdigest.com/microcontroller-projects/measuring-turbidity-of-water-to-determine-water-quality-using-arduino-turbidity-sensor> [Accessed 30 June 2021].