

## BAB III

### METODOLOGI PENELITIAN

#### A. Waktu dan Tempat Penelitian

##### 1. Waktu Penelitian

Penelitian ini dilaksanakan pada semester ganjil dan genap tahun akademik 2020/2021, tabel waktu penelitian sebagai berikut :

Tabel 3. 1 Jadwal Penelitian

Kegiatan	Tahun 2021						
	April	Mei	Juni	Juli	Agustus	September	Oktober
Studi Literatur							
Penulisan Proposal Skripsi							
Perancangan Alat							
Penentuan Alat dan Bahan							
Perancangan Perangkat Lunak							
Pengujian Alat							
Penulisan Skripsi							

##### 2. Tempat Penelitian

Dalam penelitian ini, penulis mengambil tempat di Lab Jaringan Fakultas Matematika dan Ilmu Komputer (FMIKOM), Universitas Nahdlatul Ulama Al Ghozali (UNUGHA).

#### B. Jenis Data

Jenis data yang digunakan dalam penelitian yaitu Data Primer. Data Primer adalah sumber data yang langsung memberikan data kepada pengumpul data. Peneliti menggunakan data primer karena peneliti mengumpulkan sendiri data-data

yang dibutuhkan yang bersumber langsung dari objek pertama yang akan diteliti. Dalam penelitian ini data primer berasal dari pembacaan sensor *Ultrasonic* dan sensor *Turbidity* yang diambil langsung dari alat monitoring pendeteksi dini bencana banjir berbasis *Internet of Things*.

### C. Jenis Penelitian

Penelitian dengan judul “Perancangan Sistem Monitoring Ketinggian Air sebagai Pendeteksi Dini Bencana Banjir Berbasis IoT (*Internet of Things*)” ini termasuk jenis penelitian *Research & Development*.

*Research & Development* adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tertentu (Sugiyono 2015).

### D. Alat dan Bahan

Pada perancangan dan pembuatan alat ini dibutuhkan beberapa alat dan bahan untuk membuat alat sistem monitoring ketinggian air sebagai pendeteksi dini bencana banjir berbasis *Internet of Things*. Daftar alat dan bahan yang digunakan dalam melakukan perancangan dan pembuatan alat ini sebagai berikut:

#### 1. Perangkat keras (*Hardware*)

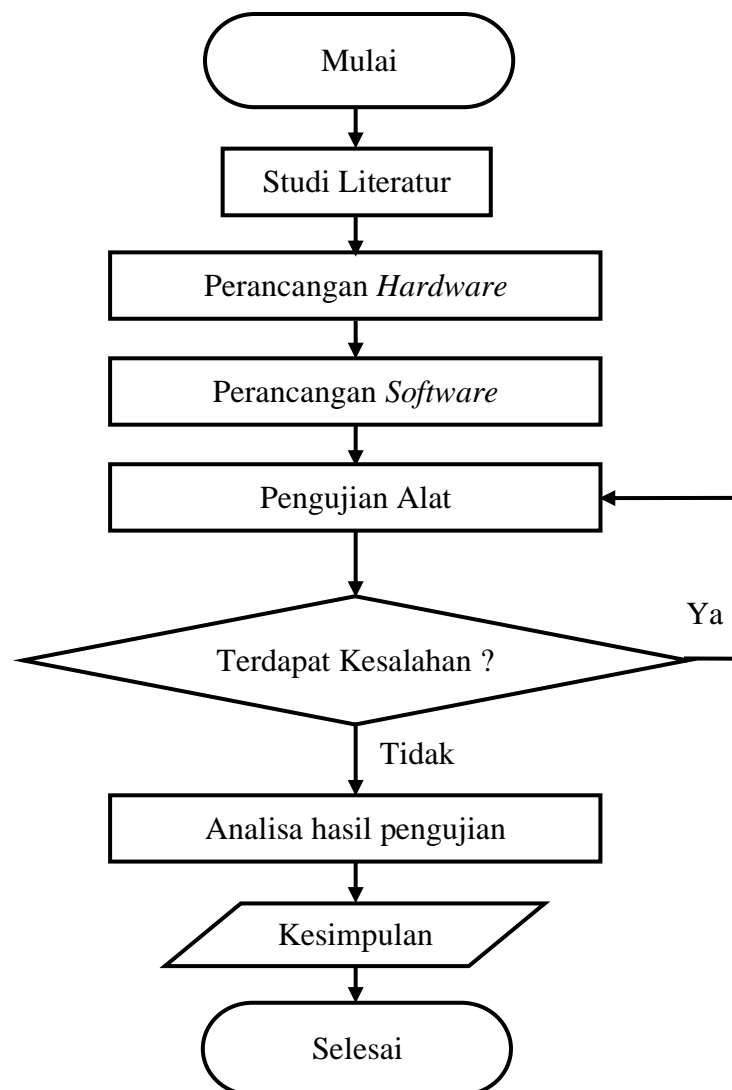
Perangkat keras yang digunakan dalam penelitian ini yaitu:

- a. Unit laptop dengan spesifikasi:
  - 1) Processor intel i3
  - 2) RAM 4
  - 3) HDD 1TB
- b. Node MCU ESP8266
- c. Sensor *Ultrasonic*
- d. Sensor *Turbidity*
- e. *Buzzer*
- f. *Liquid Cristal Display* (LCD)
- g. Dioda LED
- h. *Smartphone*

2. Perangkat Lunak (*Software*)
  - a. Sistem operasi Windows 10 Pro
  - b. Arduino IDE

### E. Sistematika Penelitian

Perancangan sistem monitoring ketinggian air sebagai pendeteksi dini bencana banjir berbasis IoT dilakukan dengan tahapan dengan sistematika disajikan pada Gambar 3.1 dan penjelasan setiap tahapan sebagai berikut :



Gambar 3. 1 Diagram Alir Penelitian

## 1. Studi Literatur

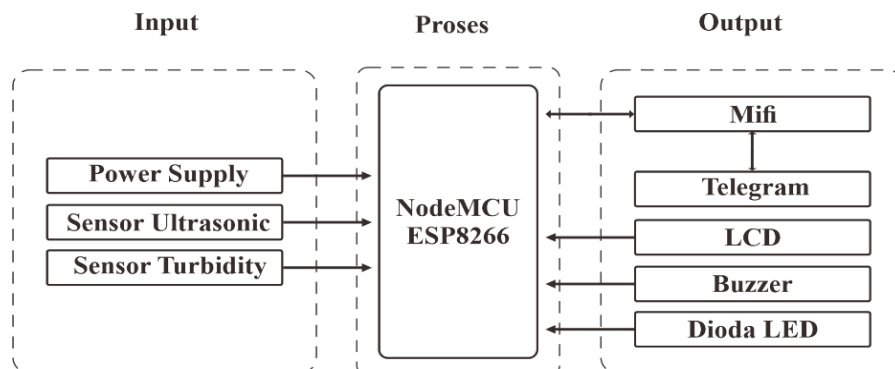
Studi literatur dilakukan setelah melakukan identifikasi masalah yang akan diteliti. Sebelum melakukan perancangan *hardware*, studi literatur untuk dilakukan karena pada tahapan ini dilakukan pencarian referensi yang berkaitan dengan judul penelitian penulis. Studi literatur diambil dari penelitian sebelumnya yang berkaitan dengan judul. Pada tahap ini perbandingan pada setiap penelitian sebelumnya untuk menentukan teknologi, penggunaan perangkat, dan kelemahan dari alat untuk penelitian ini.

Hasil studi literatur tersebut disajikan dalam Tabel 2. 1 Telaah Pustaka, sehingga dapat memudahkan pembaca dalam mengetahui penelitian yang berkaitan dengan penelitian yang dilakukan penulis.

## 2. Perancangan *Hardware*

Pada perancangan ini digunakan NodeMCU ESP8266 sebagai pengendali dari sistem. NodeMCU ESP8266 dipilih karena sudah terdapat wifi modul didalamnya, sehingga bisa bertugas sebagai *client* maupun *access point*. Untuk mengetahui ketinggian air pada suatu tempat digunakan sensor *Ultrasonic* dan untuk mengetahui tingkat kekeruhan air menggunakan sensor *Turbidity*. Sensor *Ultrasonic* dan sensor *Turbidity* akan mendeteksi tingkat ketinggian air dan tingkat kekeruhan air, maka semakin tinggi air dan tingkat kekeruhan yang dideteksi, maka potensi banjir akan terjadi.

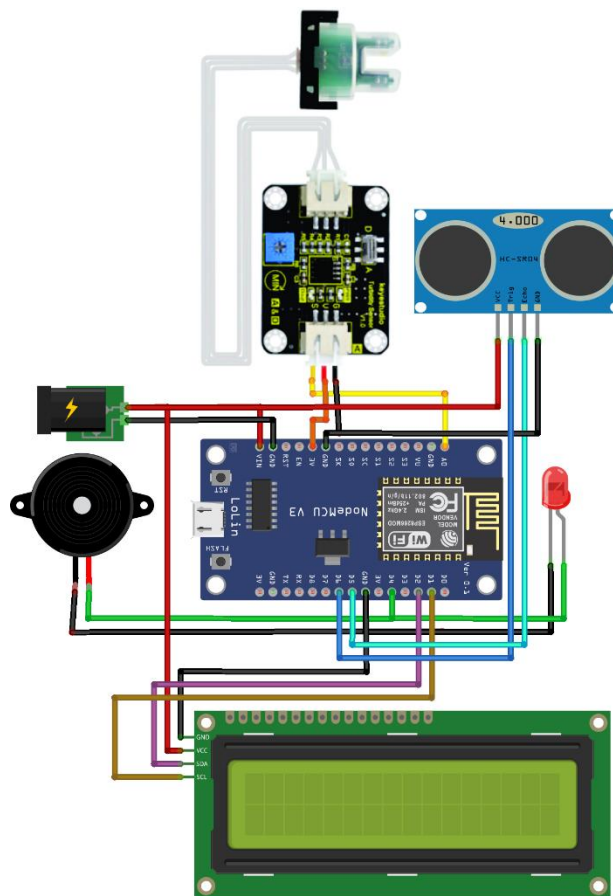
Informasi yang dibaca dari sensor *Ultrasonic* dan sensor *Turbidity* inilah yang nantinya akan diterjemahkan oleh NodeMCU ESP8266, lalu NodeMCU akan mengirimkan dan diterima aplikasi Telegram yang digunakan oleh petugas.



Gambar 3. 2 Diagram blok alat

Gambar 3. 2 merupakan diagram blok alat yang akan dibuat. Sensor yang digunakan yaitu sensor *Ultrasonic* dan sensor *Turbidity* sebagai pendeteksi tingkat ketinggian air beserta kekeruhannya. Sensor *Turbidity* mengambil data tingkat kekeruhan air dengan membaca sifat optik air akibat dispersi cahaya dan dapat dinyatakan sebagai perbandingan cahaya yang dipantulkan terhadap cahaya yang tiba, pengambilan data ketinggian permukaan air memperhatikan jarak sensor dengan dasar pengaliran sebagai patokan nilai. Nilai jarak sensor *Ultrasonic* dengan permukaan air akan diketahui ketika sensor membaca gelombang *Ultrasonic*. Jika nilai dari kedua jarak tersebut telah diketahui selanjutnya mencari nilai jarak ketinggian air, nilai jarak ketinggian air didapat dari jarak letak sensor *Ultrasonic* sampai dasar sungai. Hasil dari pembacaan kedua sensor kemudian di kirim ke NodeMCU ESP8266, kemudian NodeMCU ESP8266 mengirimkan notifikasi ke Telegram. Fungsi dari alat dan bahan yang digunakan yaitu :

- a. NodeMCU ESP8266



Gambar 3. 3 Rangkain NodeMCU ESP8266

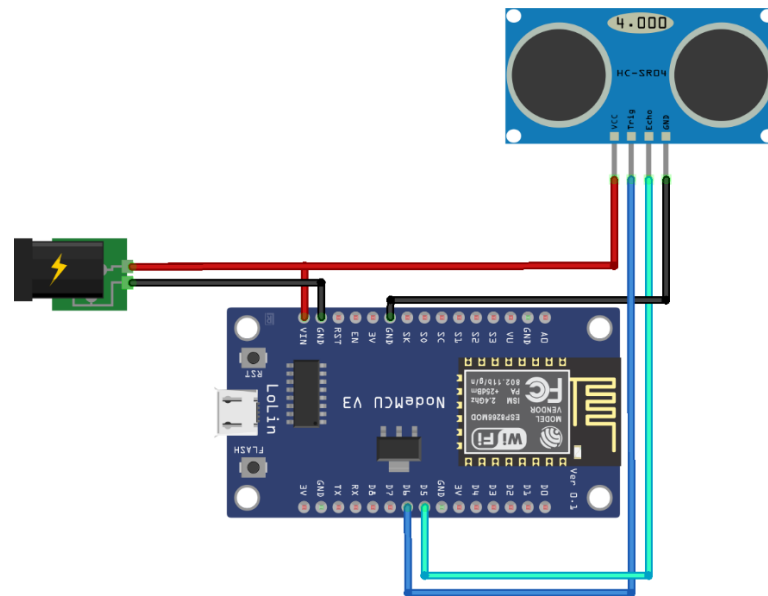
NodeMCU ESP8266 digunakan sebagai penerima informasi dari sensor dan sebagai pengirim ke aplikasi Telegram. NodeMCU ESP8266 sebagai pengendali dari sistem, dipilih karena sudah terdapat wifi modul didalamnya, sehingga bisa bertugas sebagai *client* maupun *access point*. NodeMCU ESP8266 sebagai pusat pengolahan dari data setiap komponen alat membutuhkan *power supply* untuk menjalankan fungsi dari alat. NodeMCU ESP8266 dihubungkan ke tombol *power supply* dengan menghubungkan *port* positif ke VIN dan *port* negatif ke G yang terdapat pada NodeMCU ESP8266 sehingga arus power dapat dihidupkan atau dimatikan dengan tombol yang terdapat pada alat.

NodeMCU ESP8266 menghubungkan komponen alat pengambilan data tingkat ketinggian air dan tingkat kekeruhan air yaitu sensor *Ultrasonic* dengan menghubungkan port sensor *Ultrasonic* Vcc, Trig, Echo, Gnd ke pin D7, D6, D5, G dan sensor *Turbidity* ke pin A0, G, VU yang terdapat pada NodeMCU ESP8266. Hasil dari pembacaan kedua sensor akan dikirim dan ditampilkan pada layar LCD dalam bentuk tulisan dan angka. Oleh karena itu harus menghubungkan port SCL, SDA, VCC, GND pada LCD ke pin D1, D2, D3, D4 NodeMCU ESP8266.

*Buzzer* dan LED berfungsi sebagai alarm dengan cara menghubungkan *port* positif *Buzzer* dan LED ke pin RX dan port negatif dihubungkan ke pin TX. Pin RX dengan pin D7 ke NodeMCU ESP8266 agar *Buzzer* dan LED terhubung dengan sensor *Ultrasonic*. *Buzzer* akan merespon dengan bunyi dan LED akan merespon dengan menyala sebagai tanda *warning* ketika hasil dari pembacaan sensor *Ultrasonic* berada pada ketinggian air diatas 40 cm. *Buzzer* dan LED akan terus berbunyi dan menyala ketika tingkat ketinggian belum berada pada level siaga banjir dan status aman banjir.

Informasi yang dibaca dari sensor *Ultrasonic* dan sensor *Turbidity* inilah yang nantinya akan dibaca dan diterjemahkan oleh NodeMCU ESP8266 lalu dikirim ke Telegram.

b. Sensor *Ultrasonic*

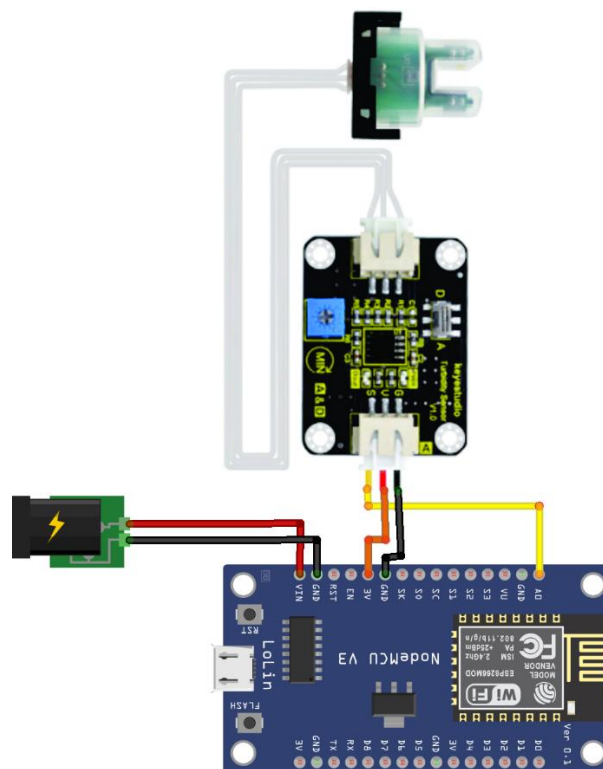


Gambar 3. 4 Rangkaian sensor *Ultrasonic*

Sensor *Ultrasonic HC-SR04* terdiri dari dua buah *transducer* yang masing-masing berfungsi sebagai *transmitter* yang berperan untuk pemancar frekuensi gelombang ultrasonic dan sebuah *receiver* yang berfungsi sebagai penerima pantulan (*returnecho*) frekuensi gelombang ultrasonic dari objek yang diukur jaraknya terhadap sensor. Sensor *Ultrasonic* dengan port Vcc, Trig, Echo, Gnd akan dihubungkan dengan pin NodeMCU ESP8266 yaitu D7, D6, D5 dan G.

Sensor *Ultrasonic* digunakan untuk mengawasi ketinggian air, lalu hasil tangkapan sensor diproses oleh NodeMCU ESP8266. Disaat ketinggian air berada 0-10 cm maka akan muncul notifikasi status aman di LCD yang terpasang. Jika ketinggian air dibawah batas atas yaitu saat ketinggian air mencapai 20-40 cm, maka akan muncul notifikasi bahaya banjir di LCD serta *Buzzer* yang terpasang akan menyala.

Disaat yang hampir bersamaan, data ketinggian air akan diolah terlebih dahulu di NodeMCU ESP8266, setelah itu hasil dari olahan data tersebut dikirim menuju pengguna yang terinstall aplikasi Telegram.

c. Sensor *Turbidity*Gambar 3. 5 Rangkaian sensor *Turbidity*

Sensor *Turbidity* digunakan untuk mengetahui kualitas air. Sensor *Turbidity* terdapat bagian pemancar dan penerima inframerah. Dalam penggunaannya sensor *Turbidity* menggunakan prinsip jumlah intensitas cahaya infra merah yang mengenai permukaan bagian penerima. Hasil dari tangkapan sensor *Turbidity* akan dikirimkan dan di proses oleh NodeMCU ESP8266.

Sensor *Turbidity* dapat mendeteksi tingkat kekeruhan air yang menjadi tanda akan terjadinya banjir, peningkatan kekeruhan air sungai merupakan salah satu indikator peringatan dini adanya potensi banjir. Kekeruhan air di hulu sungai meningkat seiring terjadinya pengikisan tanah akibat tingginya intensitas hujan. Intensitas hujan yang tinggi akan meningkatkan debit air di hulu sungai. Debit air yang tinggi akan mengalir ke hilir sungai sebagai banjir. Air sungai ketika banjir tidak memiliki suatu nilai standar kekeruhan air yang baku sehingga kekeruhan air banjir berbeda-beda untuk setiap sungai (Ammari 2019).

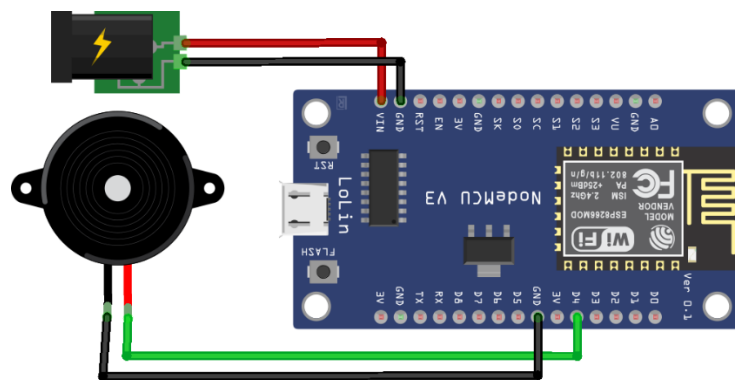
Supaya hasil pembacaan dari sensor *Turbidity* dapat ditampilkan pada layar LCD dan dapat memberikan informasi hasil pembacaan sensor *Turbidity* pada



*Client* maka port sensor *Turbidity* harus dihubungkan pada pin A0, VU dan G pada NodeMCU ESP8266.

Setelah dihubungkan NodeMCU ESP8266 akan membaca dan menterjemahkan hasil dari pembacaan sensor *Turbidity*, kemudian akan ditampilkan pada layar LCD dalam bentuk angka dan huruf, lalu NodeMCU ESP8266 secara bersamaan akan mengirimkan notifikasi kepada *client* yang telah meng-*install* dan terdaftar pada aplikasi Telegram yang telah terintegrasi dengan alat.

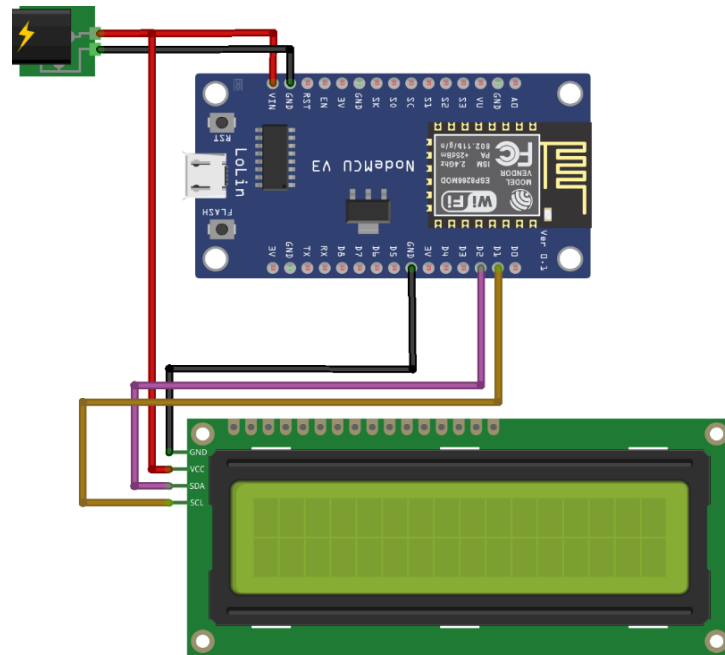
d. *Buzzer*



Gambar 3. 6 Rangkaian *Buzzer*

*Buzzer* adalah komponen yang dapat mengubah sinyal listrik menjadi getaran suara. Cara kerja *Buzzer* dalam sistem monitoring ketinggian air ini adalah ketika sensor mendeteksi ketinggian air sudah dalam status bahaya banjir, maka *Buzzer* akan berbunyi keras sebagai tanda bahaya. Supaya *Buzzer* dapat aktif pada saat tingkat ketinggian air berada diatas 40 cm maka port *Buzzer* positif harus di hubungkan dengan pin RX dan port negatif dihubungkan dengan pin TX pada NodeMCU ESP8266.

e. *Liquid Cristal Display (LCD)*

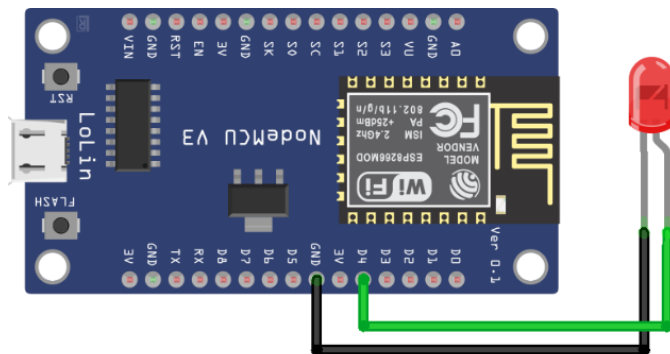


Gambar 3. 7 Rangkaian LCD

LCD berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik. Cara kerja LCD dalam sistem monitoring ketinggian air ini adalah dengan membaca hasil dari pembacaan sensor *Ultrasonic* dan sensor *Turbidity* kemudian di tampilkan di layer LCD.

Supaya LCD dapat menampilkan data dari pembacaan sensor *Ultrasonic* dan sensor *Turbidity* maka harus menghubungkan port SCL, SDA, VCC dan GND ke pin D1, D2, D3 dan D4 pada NodeMCU ESP8266.

f. *Dioda LED*

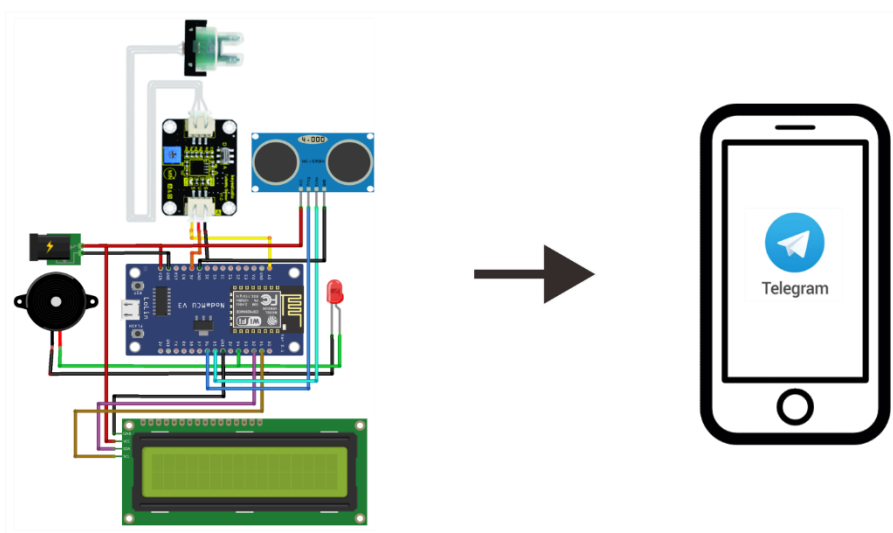


Gambar 3. 8 Rangkaian dioda LED

Dioda LED sebagai *output* pada alat, digunakan satu buah LED berwarna merah LED merah akan menyala ketika kode ketinggian air bahaya banjir sudah mencapai batas maksimum yaitu diatas 40 cm.

LED berfungsi sebagai tanda bahaya yang terdapat pada alat dengan menghubungkan port positif ke pin RX dan port negatif ke pin TX pada NodeMCU ESP8266.

g. *Smartphone*



Gambar 3. 9 Proses pengiriman notifikasi

*Smartphone* digunakan sebagai penerima pemberitahuan dari alat. *Smartphone* yang di pakai adalah Realme 7 dengan spesifikasi RAM 8 dengan prosesor Helio G95. *Smartphone* diinstal aplikasi Telegram yang sudah terintegrasi dengan alat. Aplikasi Telegram dipilih karena mempunyai beberapa keunggulan yaitu mengirim pesan lebih cepat karena berbasis cloud, lebih ringan dijalankan dan dapat diakses dari berbagai perangkat (Fitriansyah and Aryadillah 2020). *Smartphone* akan menerima pemberitahuan pada saat ada pergantian tingkat ketinggian air.

Untuk menghubungkan alat dengan *smartphone* memerlukan prosedur, diantaranya :

1. Konfigurasi Telegram, konfigurasi Telegram diperlukan untuk mendapatkan Telegram Bot ( Token ) dan ID Telegram. Token yang didapatkan dari

Telegram Bot nantinya akan ditanamkan di program NodeMCU ESP8266 dan ID Telegram menjadi identitas pada saat NodeMCU ESP8266 mengirimkan notifikasi ke Telegram.

2. *Library* Telegram yang dibutuhkan adalah CTBot dan ArduinoJson.
3. Pembuatan program Arduino, pada langkah terakhir ini harus membuat program menerima dan mengirim pesan Telegram dengan menggunakan *software* Arduino IDE.

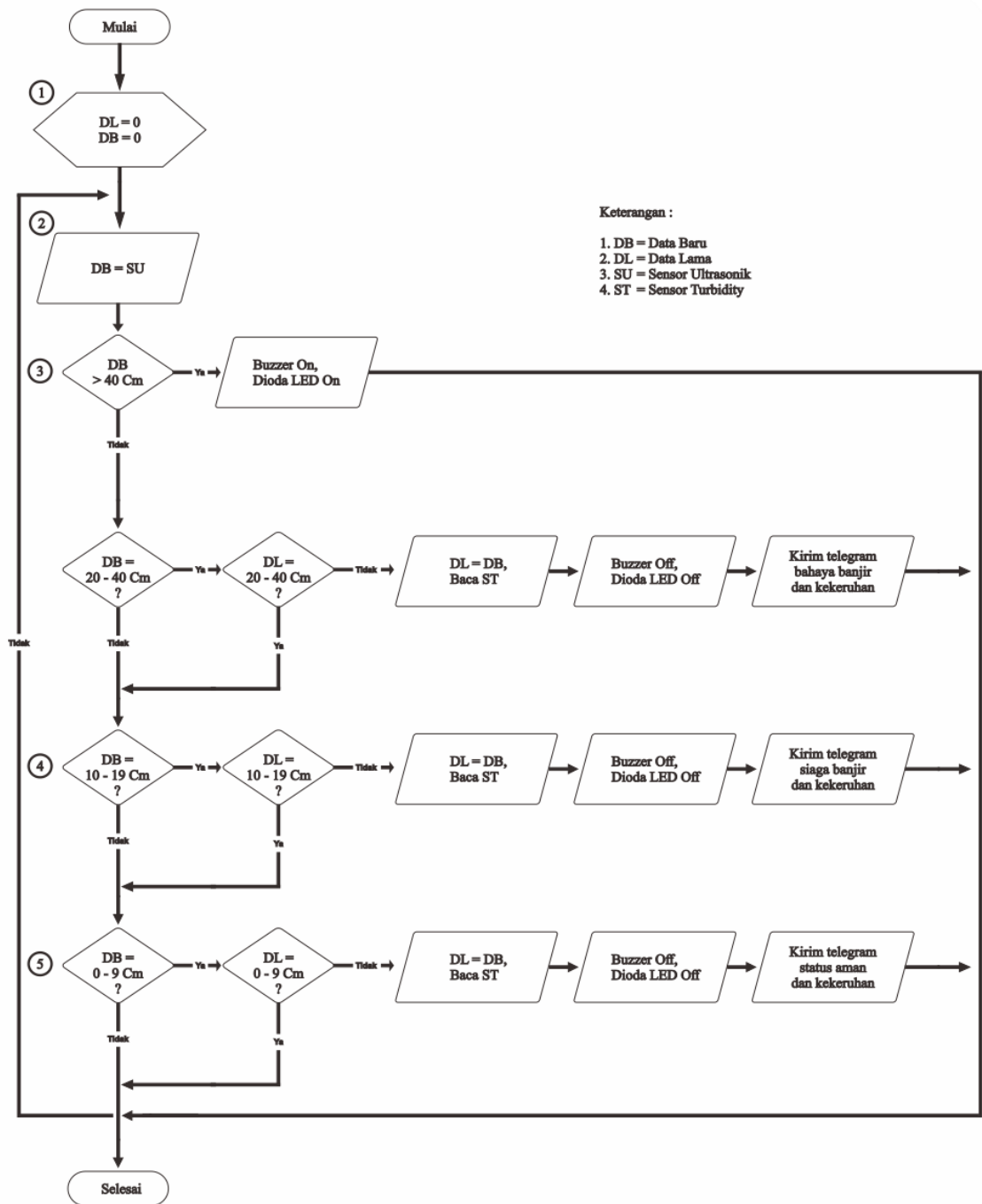
### 3. Perancangan *Software*

Setelah perancangan *Hardware* maka selanjutnya dilanjutkan dengan perancangan *Software* atau kode agar sensor dapat bekerja. Kode ditulis dengan menggunakan Bahasa C+. Penulisan kode dilakukan menggunakan aplikasi Arduino IDE. Kode dibuat untuk dapat memfungsikan sensor *Ultrasonic*, sensor *Turbidity*, *Buzzer* dan keseluruhan alat agar dapat terhubung ke pengguna. Alat harus dapat mendeteksi tingkat ketinggian air beserta kekeruhannya.

Tingkat ketinggian air dalam pembacaan sensor *Ultrasonic* dalam ukuran Cm (Centimeter), dikarenakan alat monitoring ketinggian air sebagai pendeteksi banjir ini masih berskala prototipe. Sehingga pengujian ketinggian air berbeda dengan ketinggian air pada bendungan air Desa Kamulyan.

Sensor *Ultrasonic* dan sensor *Turbidity* mendeteksi ketinggian air dan tingkat kekeruhan air, lalu hasil pembacaan sensor diproses oleh NodeMCU. Jika tingkat ketinggian air diantara 0 – 9 cm beserta kekeruhannya (Aman Banjir), 10 – 19 Cm beserta kekeruhannya (Siaga Banjir), 20 – 40 Cm beserta kekeruhannya (Bahaya Banjir). Dan jika ketinggian air sudah berada diatas batas maksimum Bahaya Banjir maka lampu indikator akan menyala dan *Buzzer* akan berbunyi sebagai alarm tanda bahaya.

Pada saat yang bersamaan, data level ketinggian air akan dikirim oleh NodeMCU ke sebuah *smartphone* yang telah ter-install aplikasi Telegram melalui jaringan *wifi*. Setiap data ketinggian air akan selalu diupdate dan mengirimkan informasi ke NodeMCU ESP8266, kemudian data update akan dikirim ke aplikasi Telegram jika terdapat pergantian ketinggian air dari kriteria status aman, kriteria status siaga banjir dan kriteria status bahaya banjir. Berikut pada Gambar 3.10 ditampilkan *flowchart* program sistem :



Gambar 3. 10 *Flowchart* Program

Gambar 3.10 merupakan *flowchart* program sistem, diawali dari mulai lalu menginisiasi NodeMCU ESP8266. Dalam kinerja alat ada 5 tahapan diantaranya :

- a. Tahap 1

Tahap 1 merupakan proses pengenalan program. Tahap 1 bertujuan agar setiap bagian dan perintah dalam pemrograman tersebut dapat dioperasikan serta sebagai tanda bahwa pemrograman tersebut siap untuk di jalankan. Fungsi lain dari tahap inisiasi adalah untuk mengatur definisi dari setiap *port* yang digunakan pada NodeMCU ESP8266. Dalam tahap ini variabel Data Lama (DL) = variabel Data Baru (DB) dengan nilai Data Baru (DB) = 0 dan Data Lama (DL) = 0 untuk mengawali sebuah program sistem awal.

Adapun sebuah kode dalam tahap 1 yang berfungsi untuk menjalankan alat sehingga dapat bekerja sesuai dengan tugas dan tujuan alat tersebut. Berikut program untuk menjalankan alat pada tahap 1 :

```
const char* ssid = "Koberto";           // SSID
const char* password = "Koberto257";   // Password
#define BOTtoken "2025284839:AAH7C6rV-evI3Eoz516GU6oqSyhLPXXf-J8" //
your Bot Token (Get from Botfather)
//#define CHAT_ID "-401546885"         //
Use @myidbot to find out the chat ID of an individual or a group

int botRequestDelay = 1000;             // Checks for new messages
every 1 second.
unsigned long lastTimeBotRan;

int adc, adc_in, DL = 0;
float ntu = 0, DB = 0;
float tinggi;
int rcnt_lcd = 0;

int aman = 9;
int siaga_bawah = 10;
int siaga_atas = 19;
int bahaya = 20;
float ts = 0;
int x = 0;
String kondisi = "Aman";
int t1 = 0;
```

b. Tahap 2

Tahap 2 merupakan tahap menginputkan data. Dalam tahap ini, data Baru (DB) sama dengan memeriksa hasil pembacaan Sensor *Ultrasonic* (SU). Hasil pembacaan Sensor *Ultrasonic* (SU) akan disimpan sebagai variabel Data Baru (DB).

Adapun sebuah kode dalam tahap 2 yang berfungsi untuk menjalankan alat sehingga dapat bekerja sesuai dengan tugas dan tujuan alat tersebut. Berikut program untuk menjalankan alat pada tahap 2 :

```
if (DB >= 510) {
  DB = map(DB, 510, 520, 40, 0);
}
else if (DB >= 470) {
  DB = map(DB, 470, 510, 100, 40);
}
else if (DB > 0) {
  DB = map(DB, 0, 470, 540, 100);
}
Serial.print("  ");
Serial.print(DB);
Serial.print("- ");
DB = DB / 10.00;
if (DB <= 0.4) DB = 0;
ts = DB;
```

### c. Tahap 3

Hal yang dilakukan pada tahap 3 adalah memeriksa isi variabel DB. Apabila DB dari pembacaan sensor *Ultrasonic* berada diatas ketinggian 40 cm, maka *Buzzer* akan berbunyi dan LED akan menyala. Tetapi jika sensor *Ultrasonic* membaca tingkat ketinggian air tidak berada diatas 40 cm maka baca DB dengan kriteria 20 – 40. Jika DB berada di dalam kriteria 20 – 40 cm maka lakukan pemeriksaan DL, jika DL berada di dalam kriteria 20 - 40 cm, maka NodeMCU ESP8266 akan memeriksa DB dengan kriteria 10 – 19 cm. Jika DL tidak berada pada kriteria 20 – 40 cm, maka NodeMCU ESP8266 melakukan pengkopian DL = DB, kemudian melakukan pembacaan sensor *Turbidity* untuk mengetahui tingkat kekeruhan air. Hasil pembacaan kedua sensor lalu dikirim ke aplikasi Telegram dengan menampilkan status “Bahaya Banjir” dan tingkat kekeruhan air. Berikut program untuk menjalankan alat pada tahap 3 :

```

if (DB >= bahaya) {
  if (DL >= bahaya) {}
  else {
    DL = DB; x = 1;
    kondisi = "Bahaya";
    lcd.setCursor(8, 1); lcd.print("~Wait...");
  }
}
// ===== Buzzer
if (ts <= 40) digitalWrite(buzzerPin, LOW);
if (ts > 40) digitalWrite(buzzerPin, HIGH);
if (x == 1) {
  x = 0;
  lcd.setCursor(8, 1);
  lcd.print("~Send~~~");
  String statuss = "Status " + kondisi + " Banjir \n";
  statuss += "Ketinggian air " + String(ts, 1) + " Cm \n";
  statuss += "Kekeruhan air " + String(ntu, 0) + " NTU \n";
  bot.sendMessage(CHAT_ID, statuss, "");
  Serial.print("Send ");
  Serial.println(kondisi);
  if (x == 0) {
    lcd.setCursor(8, 1); lcd.print("~Standby");
  }
  else {
    lcd.setCursor(8, 1); lcd.print("~Wait...");
  }
}
}

```



#### d. Tahap 4

Tahap 4 yaitu proses pembacaan jika variabel DB berada di dalam kriteria 10 – 19 cm, maka lakukan pemeriksaan DL. Jika DL berada di dalam kriteria 10 -19 cm maka akan memeriksa DB dengan kriteria 0 – 10 cm. Dan jika DL tidak berada di dalam kriteria 10 – 19 cm, maka NodeMCU ESP8266 melakukan pengkopian DL = DB, kemudian melakukan pembacaan sensor *Turbidity* untuk mengetahui tingkat kekeruhan air. Hasil pembacaan kedua sensor lalu dikirim ke aplikasi Telegram dengan menampilkan status “Siaga Banjir” dan tingkat kekeruhan air.

Adapun sebuah kode dalam tahap 4 yang berfungsi untuk menjalankan alat sehingga dapat bekerja sesuai dengan tugas dan tujuan alat tersebut. Berikut program untuk menjalankan alat pada tahap 4 :

```

        if ((DL >= siaga_bawah) && (DL <= siaga_atas)) {}
    else {
        DL = DB; x = 1;
        kondisi = "Siaga";
        lcd.setCursor(8, 1); lcd.print("~Wait...");
    }
}
if (x == 1) {
    x = 0;
    lcd.setCursor(8, 1);
    lcd.print("~Send~~~");
    String statuss = "Status " + kondisi + " Banjir \n";
    statuss += "Ketinggian air " + String(ts, 1) + " Cm \n";
    statuss += "Kekeruhan air " + String(ntu, 0) + " NTU \n";
    bot.sendMessage(CHAT_ID, statuss, "");
    Serial.print("Send ");
    Serial.println(kondisi);
    if (x == 0) {
        lcd.setCursor(8, 1); lcd.print("~Standby");
    }
    else {
        lcd.setCursor(8, 1); lcd.print("~Wait...");
    }
}
}

```

### e. Tahap 5

Tahap 5 yaitu proses pembacaan jika variabel DB berada di dalam kriteria 0 – 10 cm, maka lakukan pemeriksaan DL. Jika DL berada di dalam kriteria 0 -10 cm maka akan kembali ke pembacaan variabel DB. Dan jika DL tidak berada di dalam kriteria 0 – 10 cm, maka NodeMCU ESP8266 melakukan pengkopian DL = DB, kemudian melakukan pembacaan sensor *Turbidity* untuk mengetahui tingkat kekeruhan air. Hasil dari pembacaan kedua sensor lalu dikirim ke aplikasi Telegram dengan menampilkan status “Aman Banjir” dan tingkat kekeruhan air.

Adapun sebuah kode dalam tahap 5 yang berfungsi untuk menjalankan alat sehingga dapat bekerja sesuai dengan tugas dan tujuan alat tersebut. Berikut program untuk menjalankan alat pada tahap 5 :

```

if (DB <= aman) {
    if (DL <= aman) {}
    else {
        DL = DB; x = 1;
        kondisi = "Aman";
        lcd.setCursor(8, 1); lcd.print("~Wait...");
    }
}
if (x == 1) {
    x = 0;
    lcd.setCursor(8, 1);
    lcd.print("~Send~~~");
    String statuss = "Status " + kondisi + " Banjir \n";
    statuss += "Ketinggian air " + String(ts, 1) + " Cm \n";
    statuss += "Kekeruhan air " + String(ntu, 0) + " NTU \n";
    bot.sendMessage(CHAT_ID, statuss, "");
    Serial.print("Send ");
    Serial.println(kondisi);
    if (x == 0) {
        lcd.setCursor(8, 1); lcd.print("~Standby");
    }
    else {
        lcd.setCursor(8, 1); lcd.print("~Wait...");
    }
}
}

```