

**SISTEM MONITORING CURAH HUJAN DAN BUKAAN GATE
BENDUNGAN PLTA BERBASIS IOT**



SKRIPSI

Diajukan sebagai salah satu syarat untuk menyelesaikan
Pendidikan diploma empat (D-4) Program Studi Teknik Pembangkit energi
Jurusan Teknik Mesin
Politeknik Negeri Ujung Pandang

Muhammad Sabarka 442 19 033
Samir Ramadhan Hz 442 19 041

**PROGRAM STUDI D-4 TEKNIK PEMBANGKIT ENERGI
JURUSAN TEKNIK MESIN
POLITEKNIK NEGERI UJUNG PANDANG
MAKASSAR
2023**

HALAMAN PENGESAHAN

Skripsi dengan judul **Sistem Monitoring Curah Hujan dan Buka-an Gate Bendungan PLTA Berbasis IoT** oleh Muhammad Sabarka NIM 442 190 33 dan Samir Ramadhan Hz NIM 442 190 41 dinyatakan layak untuk diujikan.

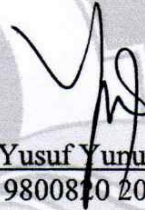
Makassar, 11 September 2023

Pembimbing I



Prof. Ir. Nur Hamzah, M.T., Ph.D.
NIP. 19631111 199003 1 002

Pembimbing II



Muh. Yusuf Yurus, S.ST., M.T.
NIP. 19800820 200501 1 001

Mengetahui
Koordinator Program Studi,



Ir. Chandra Buana, M.T.
NIP. 19650319 199103 1 003

HALAMAN PENERIMAAN

Pada hari ini, Selasa tanggal 19 September 2023, Tim Penguji Ujian Sidang Skripsi telah menerima dengan baik skripsi oleh mahasiswa: Muhammad Sabarka NIM 442 19 033 dan Samir Ramadhan Hz NIM 442 19 041 dengan judul Sistem Monitoring Curah Hujan dan Bukaan *Gate* Bendungan PLTA Berbasis IoT.

Makassar, 22 September 2023

Tim Penguji Ujian Sidang Skripsi:

1. Prof. Dr. Ir. Firman, M.T. Ketua (.....)
2. Dr. Jumadi Tangko, M.Pd Sekretaris (.....)
3. Prof. Ir. Makmur Saeni, M.T., Ph.D. Anggota (.....)
4. Ir. Andareas Pangkung, M.T. Anggota (.....)
5. Prof. Ir. Nur Hamzah, M.T., Ph.D. Pembimbing I (.....)
6. Muh. Yusuf Yunus, S.ST., M.T. Pembimbing II (.....)

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa karena berkat dan rahmat karunia-Nya, penulis skripsi ini yang berjudul “**Sistem Monitoring Curah Hujan dan Buakan Gate Bendungan PLTA Berbasis IoT**” dapat diselesaikan dengan baik.

Skripsi ini merupakan salah satu syarat yang baru dipenuhi guna menyelesaikan pendidikan Diploma IV pada Program Studi Teknik Pembangkit Energi, Jurusan Teknik Mesin, Politeknik Negeri Ujung Pandang.

Pada kesempatan ini, penulis menyampaikan penghargaan dan ucapan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT yang telah memberikan kesehatan, kekuatan serta inspirasi kepada kami untuk menyelesaikan tugas akhir ini.
2. Kedua orang tua tercinta, juga kepada saudara-saudara kami yang memberikan banyak bantuan berupa dorongan moral, bantuan materi, serta tak henti-hentinya memberikan doa yang tulus kepada kami dalam menyelesaikan laporan tugas akhir.
3. Bapak Ir. Ilyas Mansur, M.T. selaku Direktur Politeknik Negeri Ujung Pandang.
4. Bapak Dr. Ir. Syaharuddin Rasyid, M.T. selaku Ketua Jurusan Teknik Mesin Politeknik Negeri Ujung Pandang.
5. Bapak Ir. Chandra Buana, M.T. selaku Ketua Program Studi Teknik Pembangkit Energi Politeknik Negeri Ujung Pandang.
6. Bapak Prof. Ir. Nur Hamzah, M.T., Ph.D. selaku pembimbing 1 dan Bapak Muh. Yusuf Yunus, S.ST, M.T. selaku pembimbing 2 yang telah meluangkan waktunya untuk memberikan bimbingan dalam menyelesaikan skripsi ini.
7. Bapak Ir. Herman Nawir, M.T. selaku wali kelas.
8. Seluruh Dosen dan Staf Program Studi Teknik Pembangkit Energi yang telah memberikan ilmunya kepada penulis selama melaksanakan perkuliahan, dan telah membantu dalam menyediakan fasilitas sarana dan mengerjakan skripsi ini.
9. Seluruh rekan-rekan Mahasiswa Teknik Mesin Angkatan 2019 yang telah menjadi saudara-saudariku serta banyak memberikan motivasi, bantuan serta doanya, selama berada di Politeknik Negeri Ujung Pandang.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna, sehingga kami mengharapkan kritik dan saran yang bersifat membangun untuk perbaikan di masa mendatang. Akhir kata, semoga tulisan ini dapat bermanfaat sebagaimana mestinya.

Makassar, Agustus 2023

Penulis

DAFTAR ISI

HALAMAN PENGESAHAN.....	i
HALAMAN PENERIMAAN	ii
KATA PENGANTAR	iii
DAFTAR ISI.....	iv
DAFTAR TABEL.....	viii
DAFTAR GAMBAR	ix
DAFTAR SIMBOL, SATUAN, DAN SINGKATAN	xi
DAFTAR LAMPIRAN.....	xii
SURAT PERNYATAAN.....	xiii
SURAT PERNYATAAN.....	xiv
RINGKASAN	xv
SUMMARY	xvi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Ruang Lingkup Penelitian.....	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	4
BAB II TINJAUAN PUSTAKA.....	5

2.1 Fluida Dinamis	5
2.1.1 Persamaan Kontinuitas.....	5
2.1.2 Persamaan Energi.....	6
2.2 Bendungan	7
2.3 Curah Hujan	8
2.4 <i>Internet of Things (IoT)</i>	9
2.5 <i>Tipping Bucket</i>	11
2.6 Sensor HC-SR04.....	11
2.7 Sensor Water Flow	13
2.8 Modul <i>Wifi</i>	15
2.8.1 NodeMCU ESP8266.....	15
2.8.2 Wemos D1 R2	18
2.9 Arduino Nano.....	19
2.10 Motor <i>Stepper</i> Nema 17	22
2.11 Motor <i>Driver</i> L289N.....	24
2.12 Roda gigi <i>Bevel (Bevel Gear)</i>	25
2.13 Pompa.....	26
2.14 Software Arduino IDE	26
2.15 Platform <i>Thingspeak</i>	28
BAB III METODE PENELITIAN.....	30

3.1 Tempat dan Waktu	30
3.2 Alat dan Bahan.....	30
3.2.1 Alat.....	30
3.2.2 Bahan	31
3.2.3 Piranti Lunak.....	31
3.3 Parameter Yang Diukur.....	32
3.4 Prosedur Kerja.....	33
3.5 Tahap Perancangan	33
3.5.1 Perancangan Perangkat Keras (<i>Hardware</i>).....	34
3.5.2 Perancangan Perangkat Lunak (<i>Software</i>)	37
3.6 Tahap Pembuatan dan Perakitan	43
3.6.1 Rangka <i>Box Plant</i> dan <i>Panel Box</i>	43
3.6.2 Rangkaian Untuk Sistem Monitoring.....	44
3.7 Prosedur Pengujian	45
BAB IV HASIL DAN PEMBAHASAN	46
4.1 Hasil Pengujian Perangkat Keras (<i>Hardware</i>).....	46
4.1.1 Pengujian Sensor Ultrasonik HC-SR04	46
4.1.2 Pengujian Sensor Curah Hujan <i>Tipping Bucket</i>	48
4.1.3 Pengujian Sensor <i>Flowmeter</i>	50
4.2 Hasil Pengujian Pemrograman Perangkat Lunak (<i>Software</i>).....	52

4.2.1 Pengujian Pemrograman Arduino Nano	52
4.2.2 Pengujian Pemrograman Wemos D1 R2.....	53
4.2.3 Pengujian Pemrograman NodeMCU ESP8266.....	54
4.3 Hasil Pengujian Keseluruhan Sistem Monitoring.....	55
BAB V PENUTUP.....	59
5.1 Kesimpulan	59
5.2 Saran.....	60
DAFTAR PUSTAKA	61
LAMPIRAN.....	63



DAFTAR TABEL

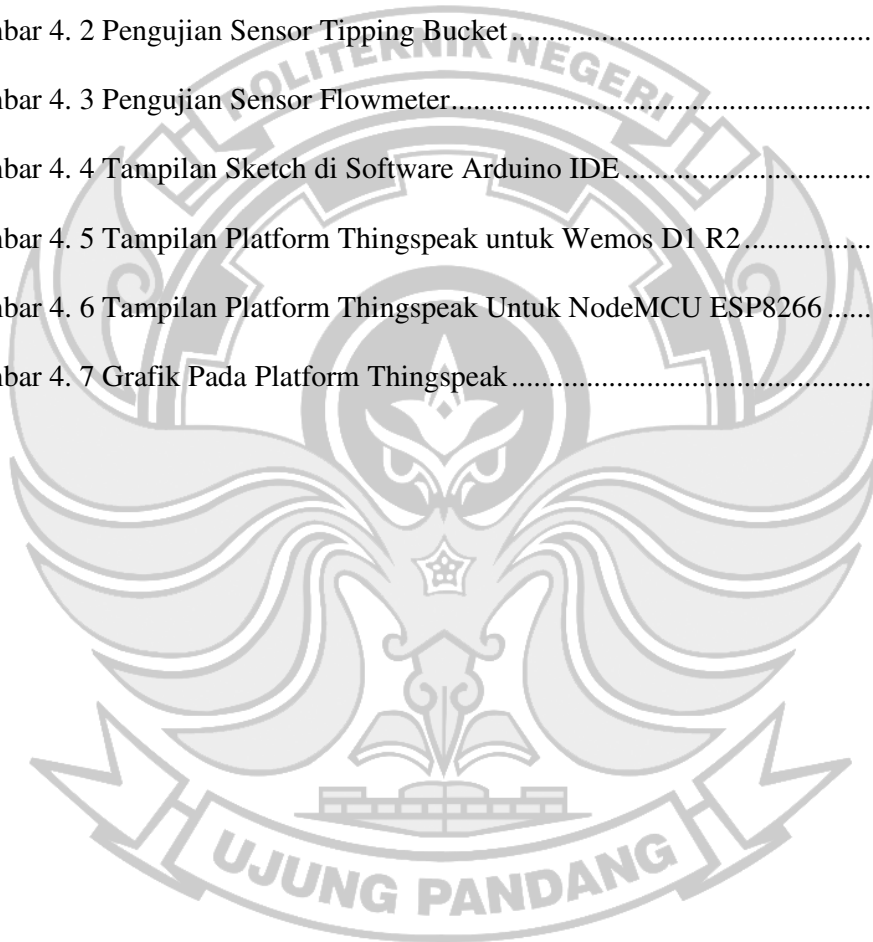
Tabel 2. 1 Spesifikasi Sensor Flowmeter.....	15
Tabel 3. 1 Daftar Alat.....	30
Tabel 3. 2 Daftar Bahan	31
Tabel 3. 3 Parameter Yang Diukur	32
Tabel 4. 1 Hasil Pengujian Sensor HC-SR04 Dengan Bukaan Gate 50%	47
Tabel 4. 2 Hasil Pengujian Sensor HC-SR04 Dengan Bukaan Gate 100%	47
Tabel 4. 3 Hasil Pengujian Sensor Curah Hujan Tipping Bucket.....	49
Tabel 4. 4 Hasil Pengujian Sensor Flowmeter YF-S201	50
Tabel 4. 5 Hasil Pengujian Sensor Flowmeter FS300A.....	50
Tabel 4. 6 Hasil Pengujian Sensor Flowmeter DN40	51
Tabel 4. 7 Hasil Pengujian Sistem Monitoring dengan Bukaan Gate 50%	55
Tabel 4. 8 Hasil Pengujian Sistem Monitoring dengan Bukaan Gate 100%	56



DAFTAR GAMBAR

Gambar 2. 1 Curah Hujan	9
Gambar 2. 2 Internet of Things (IoT).....	9
Gambar 2. 3 Tipping Bucket.....	11
Gambar 2. 4 Sensor HC-SR04	12
Gambar 2. 5 Konfigurasi pin-pin Flowmeter.....	13
Gambar 2. 6 Sensor Water Flowmeter; (a) DN40; (b) FS300A; (c) YF-S201	14
Gambar 2. 7 Modul Wifi NodeMCU ESP8266	16
Gambar 2. 8 Penempatan Pin NodeMCU	17
Gambar 2. 9 Modul Wifi Wemos D1 R2.....	19
Gambar 2. 10 Penempatan Pin Arduino Nano	20
Gambar 2. 11 Bentuk Fisik Arduino Nano	20
Gambar 2. 12 Konfigurasi Arduino Nano Atmel ATmega168.....	21
Gambar 2. 13 Motor Stepper Nema 17	23
Gambar 2. 14 Motor Driver L289N	24
Gambar 2. 15 Roda gigi kerucut (Bevel Gear)	25
Gambar 2. 16 Pompa.....	26
Gambar 2. 17 Software Arduino IDE	27
Gambar 2. 18 Platform Thingspeak	29
Gambar 3. 1 Prosedur Kerja.....	33
Gambar 3. 2 Desain rancangan struktur kerangka prototype bendungan	34
Gambar 3. 3 Skema Rangkaian Sistem Monitoring Berbasis IoT; (a) Pertama; ..b) Kedua.....	35

Gambar 3. 4 Skema Rangkaian Sistem Monitoring.....	36
Gambar 3. 5 Flowchart Arduino Nano.....	39
Gambar 3. 6 Flowchart Wemos D1 R2.....	40
Gambar 3. 7 Flowchart NodeMCU ESP8266.....	42
Gambar 4. 1 Pengujian Sensor HC-SR04.....	48
Gambar 4. 2 Pengujian Sensor Tipping Bucket.....	49
Gambar 4. 3 Pengujian Sensor Flowmeter.....	51
Gambar 4. 4 Tampilan Sketch di Software Arduino IDE.....	52
Gambar 4. 5 Tampilan Platform Thingspeak untuk Wemos D1 R2.....	53
Gambar 4. 6 Tampilan Platform Thingspeak Untuk NodeMCU ESP8266.....	54
Gambar 4. 7 Grafik Pada Platform Thingspeak.....	57



DAFTAR SIMBOL, SATUAN, DAN SINGKATAN

SIMBOL	SATUAN	KETERANGAN
C	mm	Curah Hujan
h	cm	Ketinggian Air
Q ₁	L/min	Debit Limpasan
Q ₂	L/min	Debit Aliran Hulu
Q ₃	L/min	Debit Hujan



DAFTAR LAMPIRAN

Lampiran 1 Sketch Arduino Nano	63
Lampiran 2 Sketch Wemos D1 R2	66
Lampiran 3 Sketch NodeMCU ESP8266.....	69
Lampiran 4 Gambar Teknik Rancangan Prototype Bendungan	74
Lampiran 5 Foto Kegiatan	75



SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama Lengkap : Muhammad Sabarka

NIM : 442 19 033

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam skripsi ini yang berjudul **Sistem Monitoring dan Bukaannya Gate Bendungan PLTA Berbasis IoT** merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing, dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan telah disebutkan dalam naskah dan dicantumkan dalam skripsi ini.

Jika pernyataan saya tersebut diatas tidak benar, saya siap menanggung resiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, 11 September 2023



Muhammad Sabarka
442 19 033

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama Lengkap : Samir Ramadhan Hz

NIM : 442 19 041

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam skripsi ini yang berjudul **Sistem Monitoring dan Buka-an Gate Bendungan PLTA Berbasis IoT** merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing, dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan telah disebutkan dalam naskah dan dicantumkan dalam skripsi ini.

Jika pernyataan saya tersebut diatas tidak benar, saya siap menanggung resiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, 11 September 2023



Samir Ramadhan Hz
442 19 041

SISTEM MONITORING CURAH HUJAN DAN BUKAAN GATE BENDUNGAN PLTA BERBASIS IOT

RINGKASAN

Bendungan merupakan salah satu bagian yang penting bagi Pembangkit Listrik Tenaga Air (PLTA) dalam menyediakan pasokan air yang dibutuhkan oleh turbin. Pada sisi bendungan terdapat pengukur ketinggian air untuk memperoleh informasi mengenai ketinggian air pada bendungan. Kemudahan, pengaturan *gate* bendungan sering kali dilakukan secara manual. Tujuan dari Tugas Akhir ini yaitu memudahkan pemantauan ketinggian air, curah hujan serta debit air dengan pemanfaatan *Internet of Things* (IoT) pada platform *Thingspeak* yang akan menampilkan pembacaan sensor.

Metode pembuatan Sistem Monitoring Curah Hujan dan Buka-an Gate Bendungan PLTA Berbasis IoT melalui beberapa tahap: (1) perancangan miniatur bendungan pembangkit listrik tenaga air; (2) perancangan sistem monitoring; (3) implementasi dan pengujian sistem monitoring.

Hasil pengujian Sistem Monitoring Curah Hujan dan Buka-an Gate Bendungan PLTA Berbasis IoT yaitu ketinggian air, debit limpasan, debit aliran hulu, debit curah hujan dan curah hujan. Hasil pembacaan sensor dapat dimonitor melalui platform *Thingspeak* berbasis IoT. Bentuk data yang dimonitor berupa grafik. Sistem ini akan mengoperasikan *gate* secara otomatis jika ketinggian air yang diinginkan telah tercapai. Sistem ini terbagi menjadi dua kondisi yaitu buka-an *gate* setengah (50%) dengan lebar *gate* terbuka sebesar 1,865 cm dan buka-an *gate* penuh (100%) dengan lebar *gate* terbuka sebesar 3,73 cm. Buka-an *gate* setengah akan terbuka jika pembacaan sensor HC-SR04 berada di antara 7,5 cm hingga 7,8 cm sedangkan untuk buka-an *gate* penuh akan terbuka jika pembacaan sensor HC-SR04 berada di antara 9 cm hingga 9,5 cm. Namun kondisi air yang tidak tenang dapat mengakibatkan pembacaan sensor HC-SR04 mengalami eror yang cukup signifikan. Selain itu waktu pengiriman data dari modul *wifi* membutuhkan waktu selama 15 detik akibatnya pembacaan sensor pada platform *thingspeak* tidak memberikan data yang lebih *real time*.

Kata kunci: bendungan, sistem monitoring, *gate*, IoT, *Thingspeak*

RAINFALL AND GATE OPENING SYSTEM OF HYDROPOWER DAM WITH IOT

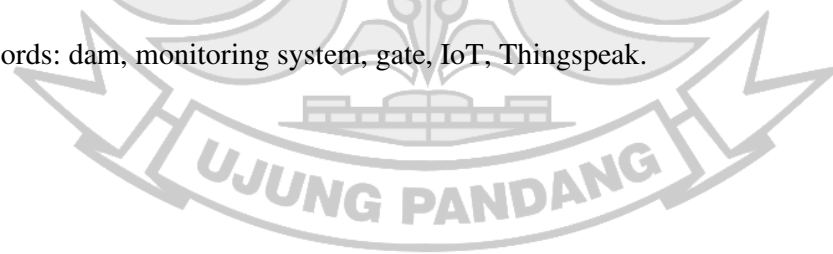
SUMMARY

The dam is an important component for a Hydropower Plant in providing the water supply needed for the turbines. On the dam's side, there is a water level gauge to obtain information about the water level at the dam. Furthermore, the dam gate control is often done manually. The aim of this Final Project is to facilitate monitoring of water levels, rainfall, and water discharge by utilizing the Internet of Things (IoT) on the Thingspeak platform, which will display sensor readings.

The method of creating a Rainfall and Gate Opening System of Hydropower Dam with IoT goes through several stages: (1) designing a miniature hydropower dam; (2) designing the monitoring system; (3) implementing and testing the monitoring system.

The results of the Rainfall and Gate Opening System of Hydropower Dam with IoT testing include water level, runoff flow rate, upstream flow rate, rainfall flow rate, and rainfall. Sensor readings can be monitored through Thingspeak platform. The monitored data is presented in the form of graphs. This system will automatically operate the gate when the desired water level is reached. The system is divided into two conditions: a half-open gate (50%) with a gate width of 1.865 cm and a fully open gate (100%) with a gate width of 3.73 cm. The half-open gate will open if the HC-SR04 sensor reading is between 7.5 cm and 7.8 cm, while the fully open gate will open if the HC-SR04 sensor reading is between 9 cm and 9.5 cm. However, turbulent water conditions can result in significant errors in the HC-SR04 sensor readings. Additionally, data transmission from the Wi-Fi module takes 15 seconds, resulting in less real-time data on the Thingspeak platform.

Keywords: dam, monitoring system, gate, IoT, Thingspeak.



BAB I PENDAHULUAN

1.1 Latar Belakang

Air sungai dapat dimanfaatkan untuk berbagai macam keperluan di luar alur sungai dengan menyadap air sungai dan mengalirkannya ke tempat yang dituju. Penyadapan dapat dilakukan secara langsung apabila elevasi permukaan air sungai terletak lebih tinggi dari pada daerah yang memerlukannya sehingga dapat dialirkan secara gravitasi. Namun, apabila daerah tujuan lebih tinggi elevasinya, diperlukan suatu rekayasa untuk menaikkan elevasi muka air sungai.

Bendungan adalah sebuah bangunan yang melintang sungai yang berfungsi untuk meninggikan elevasi muka air sungai agar dapat dialirkan ke tempat yang diperlukan, misalnya ke area persawahan secara gravitasi. Struktur bendung biasanya terbuat dari pasangan batu kali, bronjong atau beton. Bendung dapat dibedakan antara bendung tetap, artinya tidak ada pintu untuk mengatur ketinggian dan debit air serta bendung gerak yang dilengkapi dengan pintu pengatur debit air dan ketinggian air. Bendung difungsikan untuk berbagai kepentingan, antara lain irigasi, air baku untuk air minum, pemutar turbin, penggelontoran kota, dan sebagainya (Isnugroho, 2015).

Bendungan pada Pembangkit Listrik Tenaga Air (PLTA) memiliki fungsi sebagai penyuplai air dari tenaga utama yang dibutuhkan oleh PLTA, oleh karena itu level ketinggian air pada bendungan sangat berdampak terhadap pengoperasian unit generator. Saat ini, beberapa PLTA masih menggunakan metode secara konvensional untuk memonitoring level ketinggian air dengan cara menggunakan

alat pengukuran manual yang ditempatkan di tepi bendungan. Faktanya, monitoring level ketinggian secara konvensional tidak efektif dan efisien karena tidak terpantau secara *real time* (Mayang, 2020).

Tindakan yang dibutuhkan untuk menghadapi permasalahan tersebut adalah dengan memantau ketinggian air pada bendungan. Kemudian pengendalian ketinggian air pada bendungan dilakukan dengan sistem otomatis pada pintu air bendungan sehingga ketika ketinggian air berada pada kondisi level yang telah ditentukan maka *gate* atau pintu air dapat terbuka atau tertutup secara otomatis, disisi lain curah hujan memiliki dampak pada ketinggian air pada bendungan.

Berdasarkan permasalahan tersebut, kami membuat prototipe Sistem Monitoring Curah Hujan dan Buka-an Gate Bendungan PLTA Berbasis IoT dengan menggunakan platform *Thingspeak* dan dapat diakses dari *smartphone* dan PC atau laptop. Dengan demikian perancangan alat dapat mempermudah dalam mengetahui, memonitoring dan menyimpang data yang dibutuhkan.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan di atas, maka dapat diambil rumusan masalah sebagai berikut:

1. Bagaimana sistem monitoring parameter-parameter berbasis IoT menggunakan platform *Thingspeak*?
2. Bagaimana kinerja bukaan *gate* dengan memantau ketinggian air menggunakan sensor HC-SR04 dengan platform *Thingspeak*?

1.3 Ruang Lingkup Penelitian

Adapun ruang lingkup pada penelitian ini sebagai berikut:

1. Difokuskan untuk bagaimana memonitoring curah hujan dan bukaan *gate* yang dapat dimonitor dari jauh sehingga memberi kemudahan dalam melakukan pengoperasiannya.
2. Jumlah curah hujan yang jatuh diasumsikan sama dengan yang diterima oleh waduk.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah dijelaskan di atas, maka tujuan yang dicapai adalah:

1. Dapat memonitoring parameter-parameter yang dibutuhkan dengan menggunakan platform *Thingspeak*.
2. Mengatur secara otomatis bukaan *gate* dengan memanfaatkan hasil pembacaan sensor HC-SR04 berbasis IoT dengan platform *Thingspeak*.

1.5 Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian ini, yaitu:

1. Dapat menambah pengetahuan serta menambah wawasan dalam pengembangan energi terbarukan.
2. Dapat dimanfaatkan sebagai kegiatan praktikum dan penelitian selanjutnya.



BAB II TINJAUAN PUSTAKA

2.1 Fluida Dinamis

Fluida dikatakan dinamis jika fluida itu bergerak secara terus – menerus terhadap sekitarnya. Fluida yang bergerak sering disebut dinamika fluida atau fluida dinamik. Beberapa sifat fluida yang ada.

1. Ketertampatan, yaitu kemampuan fluida untuk mengalami perubahan volume ketika ditekan (dimampatkan).
2. Kekentalan atau viskositas, kekentalan terkait dengan gesekan antar bagian zat alir. Kekentalan terkait dengan gesekan antar bagian fluida. Kekentalan terlihat dampaknya kalau terdapat perbedaan kecepatan antar bagiannya. Fluida yang kental akan lebih sulit mengalir. Itulah sebabnya, kekentalan setara dengan gaya gesekan antar bagian zat alir.
3. Gerakan fluida dikatakan stasioner bila pola gerakannya tidak berubah terhadap waktu. Kecepatan fluida itu disetiap titik dalam ruang tetap sama.

Terdapat dua hukum yang berlaku di dalam fluida dinamis, yaitu:

2.1.1 Persamaan Kontinuitas

Persamaan kontinuitas dihasilkan dari prinsip kekekalan massa. Untuk aliran mantap, massa fluida yang melalui semua bagian dalam arus fluida per stauan waktu adalah sama. Ini bisa dievaluasi sebagai:

$$\rho_1 A_1 V_1 = \rho_2 A_2 V_2 = \text{konstan} \dots \dots \dots (2. 1)$$

Untuk fluida tak kompresibel maka $\rho_1 = \rho_2$, sehingga persamaan tersebut menjadi:

$$Q = A_1V_1 = A_2V_2 = \text{konstan} \dots\dots\dots (2. 2)$$

Dimana:

Q = Debit air (m³/s)

A = Luas Penampang (m²)

v = Kecepatan aliran (m/s)

(Sumber:Giles, 1984)

2.1.2 Persamaan Energi

Persamaan energi dihasilkan dari penerapan prinsip kekekalan energi pada aliran fluida. Energi yang dimiliki oleh suatu fluida yang mengalir terdiri dari energi dalam dan energi-energi akibat tekanan, kecepatan dan kedudukan. Dalam arah aliran, prinsip energi diringkas dengan suatu persamaan umum sebagai berikut:

$$\left(\frac{P_1}{\rho g} + \frac{v_1^2}{2g} + z_1\right) = \left(\frac{P_2}{\rho g} + \frac{v_2^2}{2g} + z_2\right) \dots\dots\dots (2. 3)$$

Dimana:

P = tekanan air (N/m²)

ρ = densitas (kg/m³)

V = kecepatan air (m/s)

g = percepatan gravitasi (m/s²)

z = ketinggian air (m)

Persamaan ini, untuk aliran mantap fluida tak kompresibel yang perubahan energi dalamnya bisa diabaikan. Persamaan ini dikenal sebagai *teorema Bernoulli*.

(Giles,1984)

2.2 Bendungan

Bendungan adalah bangunan yang berupa urugan tanah, urugan batu, beton, dan atau pasangan batu yang dibangun selain untuk menahan dan menampung air. Bendungan berfungsi sebagai penangkap air dan menyimpannya di musim hujan waktu air sungai mengalir dalam jumlah besar dan yang melebihi kebutuhan baik untuk keperluan. Bendungan digunakan untuk keperluan irigasi, air minum industri, tempat rekreasi, pembangkit listrik tenaga air dan sebagainya.



Gambar 2. 1 Bendungan
(Sumber: www.kajianpustaka.com)

Dalam menjalankan fungsinya sebagai PLTA, waduk dikelola untuk mendapatkan kapasitas listrik yang dibutuhkan. Pembangkit listrik tenaga air (PLTA) adalah suatu sistem pembangkit listrik yang biasanya terintegrasi dalam bendungan dengan memanfaatkan energi mekanis aliran air untuk memutar turbin, diubah menjadi energi listrik melalui generator. Adapun rumus teoritis untuk bendungan segiempat sebagai berikut:

$$Q = \frac{2}{3} c \cdot b \sqrt{2g} \left[\left(H + \frac{v^2}{2g} \right)^{\frac{3}{2}} - \left(\frac{v^2}{2g} \right)^{\frac{3}{2}} \right] \dots\dots\dots (2.4)$$

Dimana:

Q = debit aliran (m³/s)

c = koefisien

b = lebar bendungan (m)

H = head pada bendungan (m)

V = Kecepatan aliran (m/s)

(Sumber: Giles,1984)

2.3 Curah Hujan

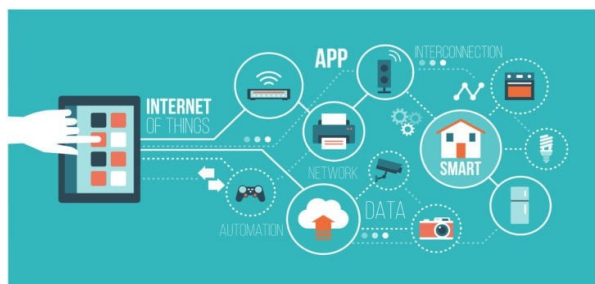
Suatu bentuk presipitasi yang berwujud cairan disebut hujan. Air hujan terbentuk jika titik air yang terpisah tersebut jatuh dari awan ke bumi. Ketinggian air hujan yang terkumpul pada tempat yang datar, tidak meresap, tidak menguap serta tidak mengalir disebut curah hujan. Banyak curah hujan persatuan jangka waktu tertentu disebut intensitas hujan. Besarnya intensitas hujan dipengaruhi oleh lama waktu curah hujan dan frekuensi terjadinya. Titik terjadinya intensitas curah hujan yang tinggi berlangsung dengan durasi pendek pada umumnya terletak pada daerah yang tidak luas, Begitu juga sebaliknya, pada area yang luas jarang sekali terjadi hujan dengan intensitas yang tinggi, namun memiliki durasi waktu yang cukup panjang. Bahkan sulit ditemukan kombinasi antara intensitas hujan yang tinggi dengan waktu yang panjang. Adapun macam-macam hujan menurut besarnya curah hujan, antara lain yaitu hujan ringan, hujan sedang dan hujan lebat. (sumber: Arisandi, 2019).



Gambar 2. 1 Curah Hujan
(Sumber: Arisandi, 2019)

2.4 *Internet of Things (IoT)*

IoT adalah konsep yang menghubungkan semua perangkat ke internet dan memungkinkan perangkat *IoT* berkomunikasi satu sama lain melalui internet. *IoT* adalah jaringan raksasa dari perangkat yang terhubung ke semua yang mengumpulkan dan membagikan data tentang bagaimana suatu perangkat tersebut digunakan dan lingkungan dimana perangkat tersebut dioperasikan. Karena memungkinkan perangkat dikontrol dari jarak jauh dengan internet, maka hal tersebut menciptakan peluang untuk langsung menghubungkan dan mengintegrasikan dunia fisik ke sistem berbasis komputer menggunakan sensor dan internet. (Sumber: [otomasi, 2018](#))



Gambar 2. 2 *Internet of Things (IoT)*
(Sumber: [otomasi.sv.ugm.ac.id](#))

Internet of Things (IoT) bekerja dengan memanfaatkan intruksi pemrograman yang setiap perintahnya bisa menghasilkan interaksi sesama perangkat terhubung secara otomatis tanpa adanya intervensi pengguna. Adapun faktor vital yang menjadi kelancaran perangkat *IoT* adalah jaringan internet yang menjadi *connector* antara sistem dan perangkat sehingga dapat dioperasikan dari jarak jauh. Sementara, manusia hanya menjadi monitor untuk setiap perilaku perangkat saat mereka bekerja. Adapun platform yang diakses diantaranya adalah *Ubidots, Firebase, Greeknesia, Thingspeak* dan *Blynk*. Ada beberapa unsur pembentuk *IoT* yaitu sebagai berikut:

1. Sensor

Sebuah perangkat yang diberikan sensor dapat mengidentifikasi informasi tertentu seperti suhu, tekanan, jarak, gerakan dan lain sebagainya.

2. Kecerdasan Buatan (*Artificial Intelligence/AI*)

Perangkat *IoT* yang dibekali *Artificial Intelligence* mampu melakukan analisis data yang lebih kompleks seperti pengumpulan data sehingga penggunaan *Internet of Things* berjalan dengan baik.

3. Konektivitas

Konektivitas berupa komponen pendukung pada sistem *IoT* untuk menghubungkan informasi yang terjadi pada sistem *IoT*. Konektivitas yang diperlukan harus memiliki stabilitas yang baik, tetapi tidak dalam ukuran yang besar.

4. Perangkat Mikro

Dengan hadirnya perangkat mikro atau perangkat berukuran kecil dapat meningkatkan ketepatan, skalabilitas dan fleksibilitas dalam performa *IoT*.

2.5 *Tipping Bucket*

Penakar curah hujan tipe tipping bucket ini bekerja dengan memanfaatkan sensor reed switch untuk memberikan masukan pada mikrokontroler yaitu berupa perubahan tahanan ketika bejana berjungkit. Sensor curah hujan tipe ini bekerja saat air jatuh melalui corong pada sensor dan mengisi ember jungkit yang terdapat pada sensor. Pada ember jungkit terdapat sebuah magnet yang akan menghubungkan reed switch yang berada di dalam sensor sehingga ketika magnet melewati saklar tersebut, saklar akan aktif dan mengirimkan sinyal.



Gambar 2. 3 *Tipping Bucket*
(Sumber: Prayogo, 2019)

2.6 Sensor HC-SR04

Sensor ultrasonik HC-SR04 adalah salah satu sensor yang paling populer digunakan pada proyek-proyek elektronik yang berhubungan dengan pengukuran jarak. Sensor ini memiliki kemampuan untuk mengukur jarak dengan cepat dan akurat, dan sering digunakan pada robotika, kendaraan otomatis, dan alat-alat yang

memerlukan pengukuran jarak yang akurat. Pada artikel ini, akan dibahas tentang sensor ultrasonik HC-SR04 dan cara menghubungkannya dengan Arduino serta kode program yang diperlukan untuk mengoperasikan sensor ini. Sensor ultrasonik HC-SR04 adalah sensor yang menggunakan gelombang suara ultrasonik untuk mengukur jarak antara sensor dan objek yang diukur. Sensor ini memiliki dua modul utama yaitu modul pengirim dan modul penerima. Modul pengirim akan mengirimkan sinyal ultrasonik ke objek yang diukur, dan kemudian modul penerima akan menerima sinyal pantulan dari objek tersebut. Sensor ini kemudian menghitung waktu yang dibutuhkan untuk sinyal ultrasonik kembali ke modul penerima dan mengkonversi waktu tersebut menjadi jarak. (sumber: Purwanto, 2019)

Jarak yang bisa ditangani berkisar antara 2 cm hingga 400 cm, dengan tingkat presisi sebesar 0,3 cm. sudut deteksi yang bisa ditangani tidak lebih dari 15°. Arus yang dibutuhkan tidak lebih dari 2mA dan tegangan yang dibutuhkan sebesar +5V. Jumlah pin adalah 4. Untuk mengetahui konfigurasi masing-masing pin pada sensor HC-SR04 dapat dilihat pada Gambar 2.5 berikut.



Gambar 2. 4 Sensor HC-SR04
(sumber: Purwanto, 2019)

1. Pin 1 : VCC, dihubungkan tegangan +5 V.

2. Pin 2 : Trig, berfungsi untuk mengirimkan gelombang suara.
3. Pin 3 : Echo, berfungsi untuk menerima pantulan gelombang suara.
4. Pin 4 : GND, dihubungkan ke ground.

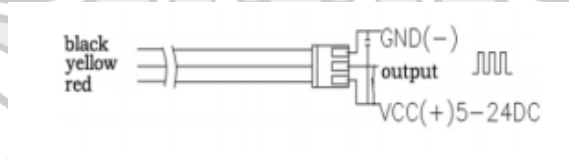
Jarak antara sensor dan objek yang memantulkan gelombang suara dihitung dengan rumus berikut:

$$Jarak = \frac{Kecepatan\ Suara \times Waktu}{2} \dots\dots\dots (2. 5)$$

Dalam hal ini, waktu yang ditempuh dari saat sinyal ultrasonic dipancarkan hingga kembali dan kecepatan suara adalah 343 m/s. (Sumber: Purwanto, 2019)

2.7 Sensor Water Flow

Secara umum sensor di defenisikan sebagai jenis tranduser yang digunakan untuk mengubah besaran mekanis, magnetis, panas, sinar, dan kimia menjadi tegangan dan arus listrik. Untuk sensor *flow water*, Air mengalir melalui suatu luasan penampang sehingga digunakan sensor debit air dalam sistem ini.



Gambar 2. 5 Konfigurasi pin-pin Flowmeter
(sumber: <https://jurnal.polines.ac.id>)

Debit merupakan volume fluida yang mengalir per satuan waktu. Sensor *flow water* merupakan sensor yang digunakan untuk mengukur debit air yang mengalir pada pipa. Sensor *flow water* terdiri dari bagian katup plastik (*valve body*), rotor air dan sebuah sensor *half effect*. Ketika air mengalir melalui rotor maka rotor akan berputar dan kecepatan dari rotor akan sesuai dengan aliran air yang masuk

melewati rotor. Pulsa dari rotor akan diterima oleh sensor hall effect untuk selanjutnya diproses di mikrokontroler.



Gambar 2. 6 Sensor *Water Flowmeter*; (a) DN40; (b) FS300A; (c) YF-S201

Sensor *Hall effect* merupakan salah satu *tranduser* yang sering digunakan untuk mendeteksi medan magnet. *Hall Effect* dapat digunakan untuk mendeteksi gerakan atau putaran apabila gerakan atau putaran tersebut dipengaruhi oleh medan magnet. Adanya aliran fluida yang mengalir pada sensor mengakibatkan kincir pada sensor akan berputar, putaran pada kincir akan menimbulkan medan magnet pada kumparan yang terdapat pada water flow sensor. Medan magnet tersebut yang dikonversi oleh hall effect menjadi pulsa. Untuk mengukur debit air yang mengalir di dalam pipa dapat menggunakan persamaan berikut

$$Q = \frac{\text{Pulse Frequency}}{CF} \dots\dots\dots(2. 6)$$

Dimana Q adalah debit air mengalir yang di nyatakan dengan liter/menit, Pulse Frequency adalah jumlah putaran per periode yang dinyatakan dengan Hz, sedangkan CF adalah *calibration factor*. (Sumber: Prasetyo, 2020)

Adapun spesifikasi lengkap dari sensor-sensor di atas adalah sebagai berikut:

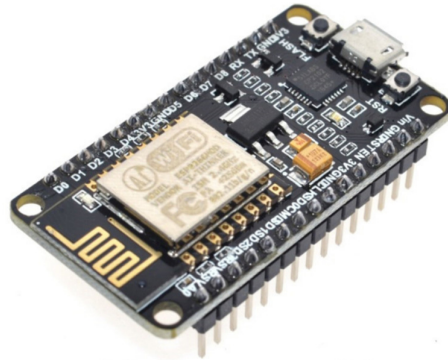
Tabel 2. 1 Spesifikasi Sensor *Flowmeter*

Spesifikasi	DN40	FS300A	YF-S201
Tegangan Kerja	5 - 24 V	3 - 18 V	5 - 18 V
Tekanan Air	1,75 MPa	1,20 MPa	1,75 MPa
<i>Working Flow Rate</i>	1 - 120 L/min	1 - 60 L/min	1 - 30 L/min
Rentang Suhu Kerja	-20° s/d 85° C	0° s/d 80° C	-25° s/d 80° C

2.8 Modul *Wifi*

2.8.1 NodeMCU ESP8266

NodeMCU ESP8266 merupakan modul *wifi* yang berfungsi sebagai komponen tambahan mikrokontroler, *wifi* membutuhkan koneksi TCP/IP supaya bisa terhubung dengan koneksi internet. Modul ini membutuhkan daya sekitar 3,3 V dengan memiliki tiga mode *wifi* yaitu *Station*, *Access Point* dan *Both* (Keduanya). Pada modul ini terdapat prosesor, memori serta GPIO (*General Purpose Input Ouput*) dimana jumlah pin bergantung dengan jenis ESP8266 yang digunakan. Sehingga modul ini bisa berdiri sendiri tanpa menggunakan mikrokontroler apapun karena sudah memiliki perlengkapan layaknya mikrokontroler.

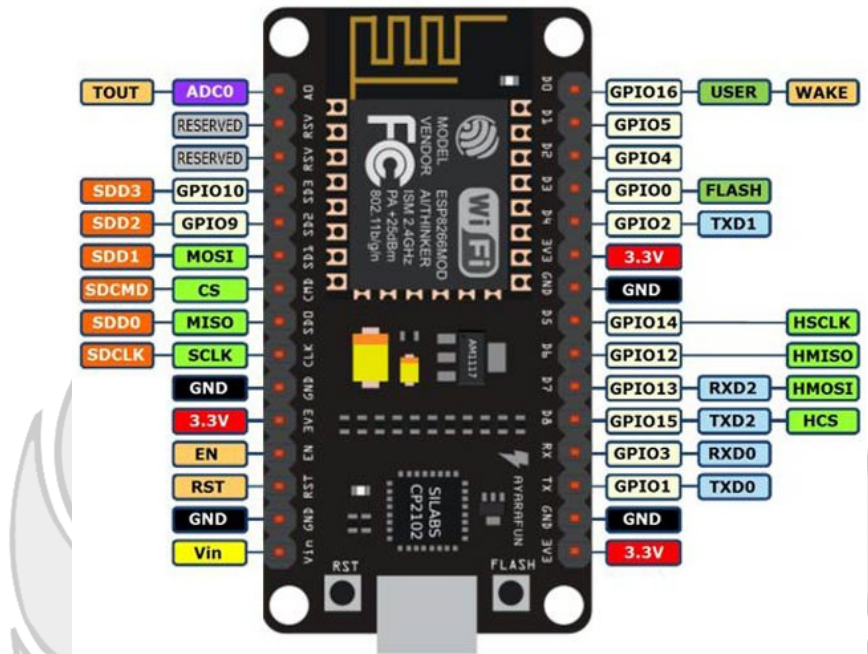


Gambar 2. 7 Modul *Wifi* NodeMCU ESP8266
(Sumber: indobot.co.id/blog)

Adapun spesifikasi modul NodeMCU ESP8266 adalah sebagai berikut:

1. Mikrokontroler: Tensilica 32-bit RISC CPY Xtensa LX106
2. Tegangan operasi: 3,3 V
3. Tegangan masukan: 7-12 V
4. Pin Digital I/O (DIO): 16
5. Pin Analog Input (ADC): 1
6. UARTs: 2
7. SPIs: 1
8. I2Cs: 1
9. Flash memori: 4 mb
10. SRAM: 64 kb
11. *Clock Speed*: 80 MHz
12. PCB Antenna

NodeMCU ESP8266 memiliki beberapa pin, dimana penempatan pin untuk NodeMCU adalah sebagai berikut:



Gambar 2. 8 Penempatan Pin NodeMCU
(Sumber: indobot.co.id/blog)

Keterangan:

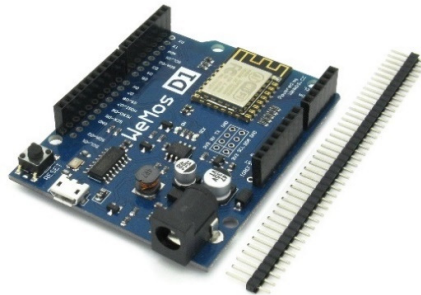
1. Micro-USB, fungsinya sebagai power yang dapat terhubung dengan USB *port*. Selain itu, biasanya juga digunakan untuk melakukan pengiriman *sketch* atau memantau data serial dengan serial dengan serial monitor di aplikasi Arduino IDE.
2. 3,3 V, digunakan sebagai tegangan untuk device lainnya. Ada 3 tempat untuk 3,3 V. Biasanya juga dituliskan hanya 3V.
3. GND, *Ground* sebagai tegangan 0 atau nilai negatif untuk mengalirkan arus.

4. Vin, sebagai *External Power* yang akan mempengaruhi *Output* dari seluruh pin. Cara menggunakannya yaitu dengan menghubungkannya dengan tegangan 7 hingga 12 volt.
5. EN & RST, merupakan pin yang digunakan untuk reset program di mikrokontroler.
6. A0, Analog pin digunakan untuk membaca input secara analog.
7. GPIO 1 – GPIO 16, merupakan pin yang dapat digunakan sebagai *input* dan *output*. Pin ini dapat melakukan pembacaan dan pengiriman data secara analog juga.
8. SD1, CMD, SD0 & CLK, pin untuk komunikasi SPI (*Serial Peripheral Interface*) dimana akan menggunakan *clock* untuk sinkronisasi deteksi bit pada *receiver*.
9. TXD0, RXD0, TXD2 & RXD2, sebagai *interface* UART, pasangannya adalah TXD0 dengan RXD0 dan TXD2 dengan RXD2.
10. TXD1, digunakan untuk upload *firmware*/program.
11. SDC, SCL (I2C Pins), digunakan untuk device yang membutuhkan I2C.

2.8.2 Wemos D1 R2

Mikrokontroler Wemos D1 R2 merupakan salah satu board yang dapat berfungsi dengan arduino khususnya untuk project yang mengusung konsep IOT. Wemos dapat *running stand-alone* berbeda dengan modul wifi lain yang masih membutuhkan mikrokontroler sebagai pengontrol atau otak dari rangkaian tersebut, wemos dapat *running stand-alone* karena didalamnya sudah terdapat

CPU yang dapat memprogram melalui *serial port* atau via OTA (*Over The Air*) serta transfer program secara *wireless*. (sumber: Kusuma, T., & Mulis, M. T, 2018).



Gambar 2. 9 Modul *Wifi* Wemos D1 R2
(sumber: <https://99tech.com>.)

Adapun spesifikasi Wemos D1 R2 adalah:

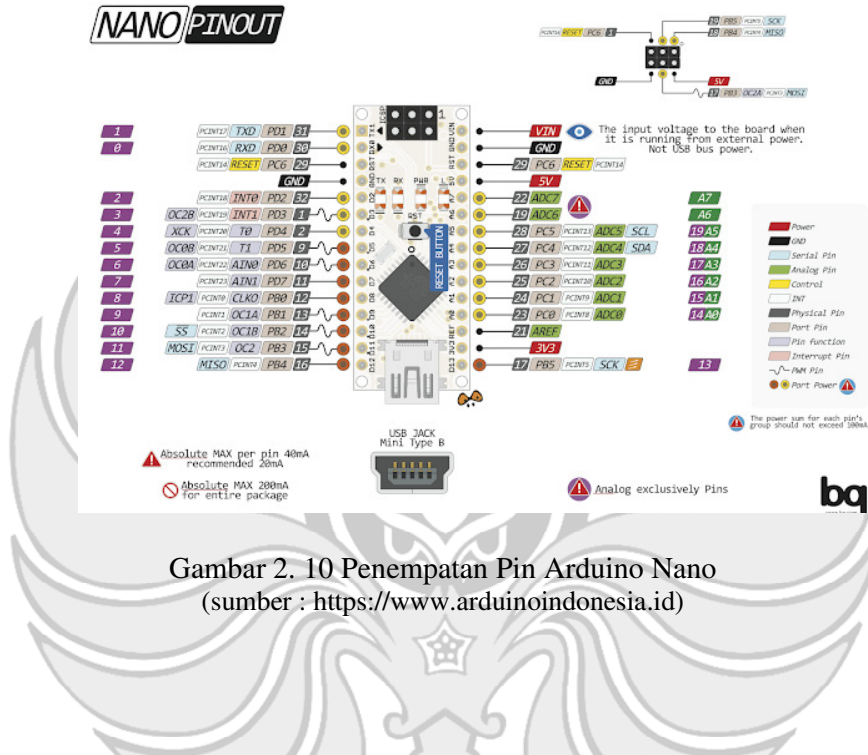
1. CPU RISC 32 bit yang berjalan pada 80MHz
2. 64Kb RAM Instruksi dan 96Kb RAM Data
3. 4MB flash memory
4. Wi-Fi
5. Terdapat 16 pin digital input dan output (GPIO)
6. Konektifitas peripheral I2C,SPI dan I2S
7. 1 Analog Digital Converter (ADC) dengan inputan maksimal 3,3vDC
8. Kompatibel dengan shield-shield Arduino UNO yang beredar di pasaran
9. Port Power Supply dengan tegangan 9v - 24vDC, seperti Arduino UNO

2.9 Arduino Nano

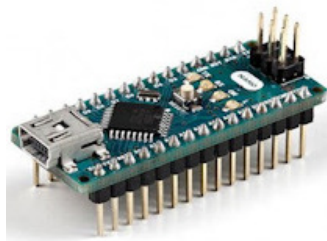
Arduino Nano adalah board microcontroller yang berukuran kecil, lengkap, dan salah satu board yang menggunakan IC ATmega328P (Arduino Nano V3). Ini memiliki fungsi yang kurang lebih sama dengan Arduino UNO, tetapi

dalam *packaging* yang berbeda. Arduino Nano ini bekerja dengan kabel USB Mini-B dan bukan yang standar. (sumber: Arduino Indonesia, 2019).

Berikut ini adalah Pin Out Arduino Nano:



Gambar 2. 10 Penempatan Pin Arduino Nano
(sumber : <https://www.arduinoindonesia.id>)



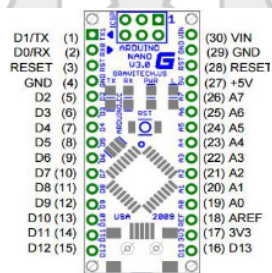
Gambar 2. 11 Bentuk Fisik Arduino Nano
(sumber: <https://www.arduinoindonesia.id>)

Arduino Nano memiliki Spesifikasi sebagai berikut:

1. Mikrokontroler (Arduino Nano) menggunakan IC Atmel ATmega 168.
2. Tegangan pada pin output saat operasi 5 VDC.

3. Input tegangan yang dianjurkan 7 sampai 12 VDC.
4. Batas input tegangan minimal 6 VDC sampai maksimal 20 VDC.
5. Memiliki pin output digital I/O 14 pin, dimana yang 6 pin memberi output PWM.
6. Memiliki pin input analog 8 pin.
7. Per pin output digital memiliki batas arus DC 40 mA.
8. Flash Memory 16 yang 2 KB digunakan oleh bootloader.
9. Memori SRAM 1 KB.
10. EEPROM 512 byte.
11. Proses kecepatan operasi yang dimiliki 16 MHz.
12. Spesifikasi fisik yaitu berdiameter besar fisik 0,73 inchi x 1,70 inchi, dengan Panjang 45 mm, serta lebar 18 mm dan bobot 5 g.

Port masukan dan keluaran Arduino Nano Pada Gambar 2.13. Konfigurasi Arduino Nano Atmel ATmega168 memiliki pin-pin yang ditunjukkan dibawah ini:



Gambar 2. 12 Konvigurasi Arduino Nano Atmel ATMega168
(sumber: <https://jeeeu.umsida.ac.id>)

Penjelasan konvigurasi pin 1–30 pada Arduino Nano Atmel ATMega168 mempunyai beberapa fungsi khusus yaitu :

1. Pin1-2,5-16 (Pin D0-D13) dapat difungsikan I/O (Digital Input/Output Port 0-13).
2. Pin3,28 (Pin Reset) yang difungsikan Input Reset(aktif rendah).
3. Pin4, 29 (Pin GND) difungsikan Input *ground*.
4. Pin17, (Pin Output3.3 VDC) yang berfungsi output +3.3 VDC dari FTDI.
5. Pin18 (Pin AREF) yang difungsikan Input referensi masukan ADC.
6. Pin 19-26 (Pin A7-A0) yang berfungsi pin input analog 0-7.
7. Pin 27 (Pin VCC) berfungsi Input/Output (Pin Input-an +5VDC dari catudaya/*power supply*, atau Pin Output-an +5VDC dari Regulator *On-board*).

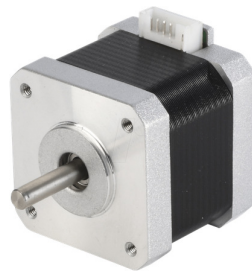
2.10 Motor Stepper Nema 17

Motor stepper adalah suatu perangkat elektromekanis yang bekerja dengan mengubah pulsa elektronis menjadi gerakan mekanis diskrit. Motor stepper bergerak berdasarkan urutan pulsa yang diberikan kepada motor sehingga untuk menggerakkan motor stepper diperlukan pengendali motor stepper yang membangkitkan pulsa-pulsa periodik. Penggunaan motor stepper memiliki beberapa keunggulan dibandingkan dengan penggunaan motor DC biasa.

Keuntungan dari stepper motor adalah kemampuannya untuk menghasilkan gerakan rotasi yang presisi dan dapat dikontrol dengan sangat baik, sehingga stepper motor banyak digunakan pada aplikasi yang memerlukan gerakan yang tepat dan terkontrol seperti pada mesin CNC, printer 3D, dan robotika. Namun, kekurangan dari stepper motor adalah memiliki noise (suara) yang lebih besar dan biaya yang lebih mahal dibandingkan dengan motor listrik konvensional. Selain itu,

NEMA 17 juga memiliki beberapa kekurangan, seperti harganya yang lebih mahal dibandingkan dengan motor DC biasa dan memiliki noise (suara) yang cukup besar selama operasinya. Namun, dengan keunggulan dan kelebihan yang dapat menghasilkan gerakan presisi dan terkontrol, NEMA 17 tetap menjadi pilihan yang sering digunakan pada berbagai aplikasi yang memerlukan kontrol gerakan yang akurat.

Motor stepper adalah motor listrik yang bergerak dalam peningkatan posisi rotor (step) yang ditentukan secara tepat melalui derajat perputaran. Jumlah step yang dilakukan oleh motor stepper diatur melalui komputer atau mikrokontroler. Pulsa yang digunakan berasal dari rangkaian osilator yang dikontrol oleh tegangan analog, pengontrol digital, atau mikropesesor. Motor stepper Nema 17 memiliki 6 kabel utama yang berjalan pada tegangan 12V. Motor ini dapat beroperasi pada tegangan yang lebih rendah, namun torsi akan turun. Nema 17 memiliki step angle $1,8^\circ$ sehingga untuk satu putaran dibutuhkan 200 step. (sumber: Prabowo, R. R. 2023)



Gambar 2. 13 Motor Stepper Nema 17
(sumber: www.omc-stepperonline.com)

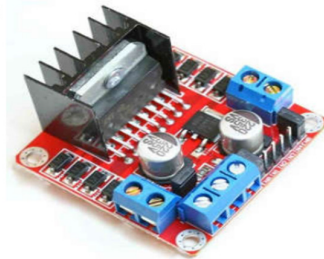
Adapun spesifikasi dari motor *stepper* nema 17:

- Ukuran lempeng depan (faceplate): 1,7 inci atau sekitar 42,3 mm
- Jumlah tahap (step): 200 tahap per putaran

- Torsi (torque): berkisar antara 0,3 hingga 0,6 Nm
- Arus nominal: berkisar antara 0,5 hingga 1,5 A
- Tegangan nominal: berkisar antara 2 hingga 4 VDC
- Tahanan: berkisar antara 1 hingga 5 ohm
- Kecepatan rotasi: berkisar antara 500 hingga 3000 RPM (tergantung pada arus dan tegangan yang diberikan)

2.11 Motor Driver L289N

Driver motor L298N merupakan driver motor yang paling populer digunakan untuk mengontrol kecepatan dan arah pergerakan motor. Kelebihan dari driver motor L298N ini adalah cukup presisi dalam mengontrol motor. Selain itu kelebihan driver motor L298N adalah mudah untuk dikontrol. Untuk mengontrol driver motor L298N ini dibutuhkan 6 buah Pin mikrokontroler dua buah untuk Pin Enable (satu buah untuk motor A dan satu buah yang lain untuk motor B karena driver motor L298N ini dapat mengontrol dua buah motor dc) 4 buah untuk mengatur kecepatan motor tersebut. Pada prinsipnya rangkaian driver motor L298N ini dapat mengatur tegangan dan arus sehingga kecepatan dan arah motor dapat diatur.



Gambar 2. 14 Motor Driver L289N
(Sumber: Widiarto, 2018)

Modul driver L298N yang dipakai pada robot vacuum seperti Gambar 2.10, menggunakan Chip ST L298N yang dapat secara langsung mengontrol dua motor dc 3-30V, arus yang dapat melewati motor driver L298N 2A/Bridge dan menyediakan antarmuka keluaran 5 V, kita dapat mengontrol dengan mudah kecepatan dan arah pergerakan motor dc, kita juga dapat mengontrol 2-Phasa motor dc. (Sumber: Widiarto, 2018)

2.12 Roda gigi *Bevel* (*Bevel Gear*)

Secara umum, bevel gear memiliki bentuk yang mengerucut sehingga bisa juga disebut roda gigi kerucut. Salah satu contoh bevel gear yang paling sederhana adalah *straight bevel gear* dan sering digunakan untuk mentransmisikan daya antarporos yang berpotongan. Sudut antara poros yang berpotongan umumnya adalah 90° namun untuk sudut tertentu lainnya juga bisa.



Gambar 2. 15 Roda gigi kerucut (*Bevel Gear*)
(Sumber: Aji, 2017)

Roda gigi kerucut lurus adalah jenis roda gigi yang mudah dan sederhana pembuatannya dan memberikan hasil yang baik dalam pemakaiannya bila dipasangkan secara tepat dan teliti. Sama halnya dengan rodagigi lurus, roda gigi ini menjadi bising pada kecepatan garis puncak yang tinggi.

2.13 Pompa

Pompa adalah mesin atau peralatan mekanis yang digunakan untuk menaikkan cairan dari dataran rendah ke dataran tinggi atau untuk menaikkan tekanan cairan dari cairan bertekanan rendah ke cairan yang bertekanan tinggi dan juga sebagai penguat laju aliran pada suatu sistem jaringan perpindahan. Hal Ini dicapai dengan membuat suatu tekanan yang rendah pada sisi masuk atau suction dan tekanan yang tinggi pada sisi keluar atau discharge dari pompa.



Gambar 2. 16 Pompa

2.14 Software Arduino IDE

Arduino IDE (*Integrated Development Environment*) merupakan *software* yang khusus digunakan untuk merancang program melalui Arduino, dengan bahasa lain Arduino IDE menjadi sebuah media untuk melakukan program *board* Arduino. Arduino IDE berfungsi untuk editor *text* guna mengedit, membuat, dan memvalidasi sebuah kode program.



Gambar 2. 17 Software Arduino IDE
(Sumber: www.inserbit.com)

Arduino IDE juga dapat melakukan sebuah aksi upload ke board Arduino. Kode program yang digunakan dalam proses pemrograman Arduino dapat disebut sebagai Arduino “*Sketch*” atau juga dapat dikatakan sebagai *source code Arduino*. Ekstensi yang digunakan sebagai *file source code* yaitu “.ino”

Arduino IDE dibuat menggunakan bahasa pemrograman Java. Software ini juga sudah *support* dengan *library* C/C++ yang biasanya disebut dengan istilah *Writing* sebagai proses operasional input dan output agar lebih sederhana. Menu-menu yang ada pada software Arduino IDE:

1) *Verify*

Verify yang tersedia pada Arduino IDE digunakan sebagai pemeriksaan kode program yang telah kami buat. Apakah sudah benar atau belum, jika belum nanti akan muncul sebuah message box yang memberitahukannya.

2) *Upload*

Menu upload idgunakan sebagai untuk untuk mengupload sketch yang telah selesai ke board Arduino. Jika menggunakan menu upload maka akan otomatis Arduino IDE akan melakukan sebuah verify lebih dulu sebelum melakukan proses upload.

3) *New*

Menu new berfungsi sebagai membuat halaman sketch baru.

4) *Open*

Menu open fungsinya digunakan untuk membuat proyek sketch yang sudah tersimpan di dalam folder sebelumnya.

5) *Save*

Menu save berguna untuk melakukan penyimpanan sketch yang sudah dibuat.

6) *Serial Monitor*

Menu serial monitor digunakan untuk membuka serial monitor untuk menampilkan sebuah jendela yang dikirimkan atau dipertukarkan antara sketch dan Arduino pada port serialnya. Biasanya serial monitor ini digunakan bertujuan untuk melakukan sebuah debugging tanpa menggunakan layar monitor baik dalam menampilkan sebuah proses, pembacaan, bahkan error.

7) *Keterangan Aplikasi*

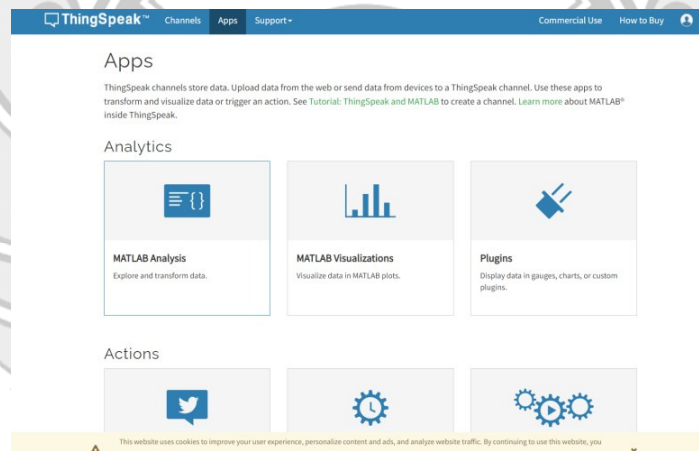
Bagian ini berguna sebagai penampilan sebuah proses yang sedang berjalan di Arduino IDE dalam bentuk keterangan.

2.15 Platform *Thingspeak*

Platform Thingspeak merupakan sebuah layanan yang berisi aplikasi dan API yang sifatnya open source untuk menyimpan dan mengambil data dari berbagai perangkat yang menggunakan HTTP (Hypertext Transfer Protocol) melalui internet atau melalui LAN (Local Area Network). Dengan menggunakan ThingSpeak, seseorang dapat membuat aplikasi logging sensor, aplikasi pelacakan lokasi, dan jaringan sosial dari segala sesuatu yang terhubung ke internet dengan pembaruan

status. ThingSpeak berfungsi sebagai pengumpul data yang mengumpulkan data dari perangkat node dan juga memungkinkan data yang akan diambil ke dalam lingkungan perangkat lunak untuk analisis historis data. (sumber: www.thingspeak.com)

Thingspeak sebagai platform cloud memungkinkan kita untuk melakukan monitoring dan controlling pada parameter-parameter yang akan kita ukur seperti layaknya sistem IoT. Thingspeak dapat menerima data dan dapat menampilkan data untuk dikirimkan ke device atau interface lainnya.



Gambar 2. 18 Platform Thingspeak
(sumber: www.thingspeak.com)

BAB III METODE PENELITIAN

3.1 Tempat dan Waktu

Lokasi pembuatan dan pengujian alat dilaksanakan di Laboratorium Energi, Jurusan Teknik Mesin, Politeknik Negeri ujung Pandang. Waktu pembuatan dan pengerjaan Tugas Akhir ini dilaksanakan selama 6 bulan mulai bulan Maret 2023 sampai dengan Juli 2023.

3.2 Alat dan Bahan

Alat dan bahan yang diperlukan dalam pembuatan sistem monitoring curah hujan dan bukaan *gate* bendungan pembangkit listrik tenaga air berbasis IoT adalah sebagai berikut:

3.2.1 Alat

Tabel 3. 1 Daftar Alat

No.	Nama	Spesifikasi	Jumlah
1.	Seperangkat <i>Tools</i>	-	1
2.	Laptop	-	1
3.	Android	-	1

3.2.2 Bahan

Tabel 3. 2 Daftar Bahan

No.	Nama	Spesifikasi	Jumlah
1	Akrilik	-	-
2	Besi siku	-	-
3	Arduino	Nano V3 ATMEGA328P	1
4	Modul <i>wifi</i>	NodeMCU ESP8266, Wemos D1 R2	2
5	Motor <i>Stepper</i>	Stepper Motor Nema 17	1
6	<i>Driver</i> Motor <i>Stepper</i>	L289N	1
7	Kabel	-	-
8	Sensor ultrasonik	HC-SR04	1
9	Sensor curah hujan	<i>Tipping Bucket</i>	1
10	Sensor <i>Water Flow</i>	YF-S201, FS300A, DN40	3
11	<i>Gear bevel</i>	-	2
12	Besi drat	-	1
13	Pompa	-	1

3.2.3 Piranti Lunak

- 1) Software Arduino IDE
- 2) Software Website *Thingspeak*

3.3 Parameter Yang Diukur

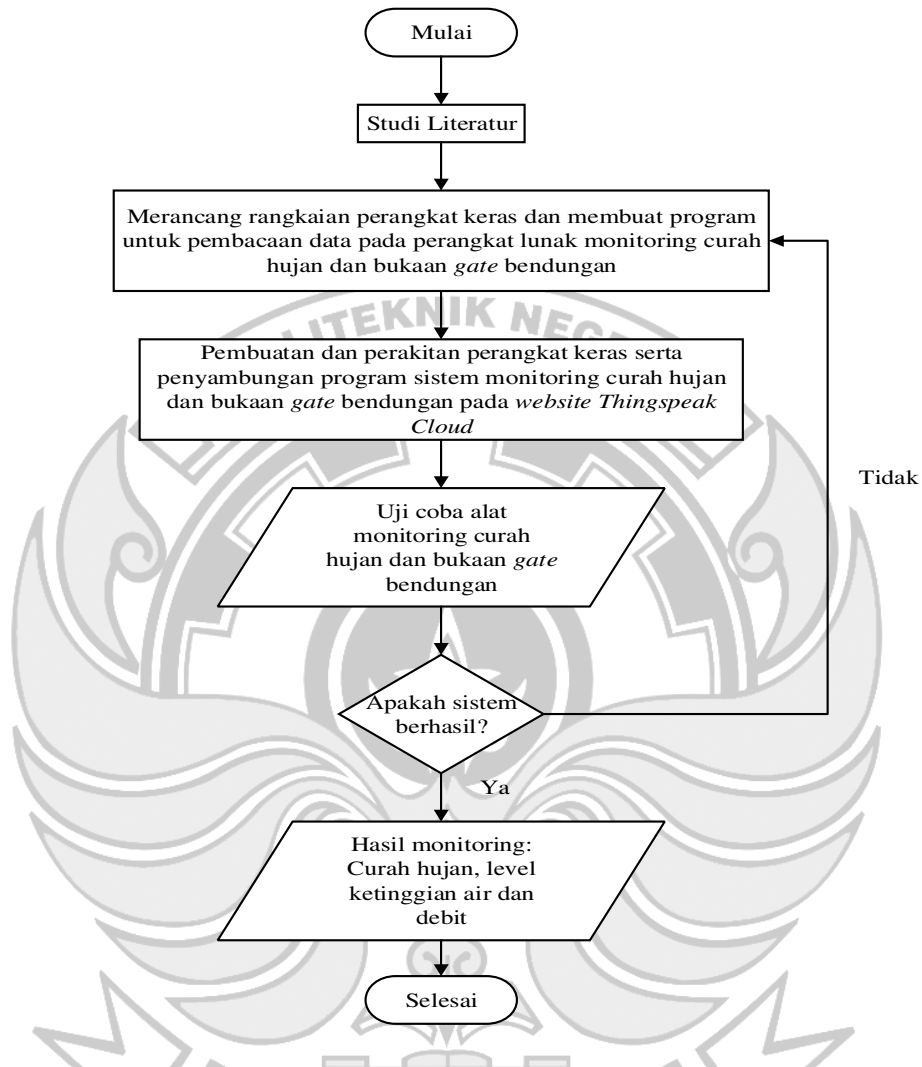
Pengambilan data dilakukan setelah perancangan, pembuatan alat, perakitan dan pengujian. Parameter yang akan diukur sebagai berikut

Tabel 3. 3 Parameter Yang Diukur

No.	Parameter	Simbol	Satuan	Keterangan
1.	Curah Hujan	C	mm	<i>Tipping Bucket</i>
2.	Ketinggian Air	h	cm	Sensor HC-SR04
3.	Debit Limpasan	Q_1	L/min	Sensor DN40
4.	Debit Aliran Hulu	Q_2	L/min	Sensor SF300A
5.	Debit Aliran Hujan	Q_3	L/min	Sensor YF-S201



3.4 Prosedur Kerja



Gambar 3. 1 Prosedur Kerja

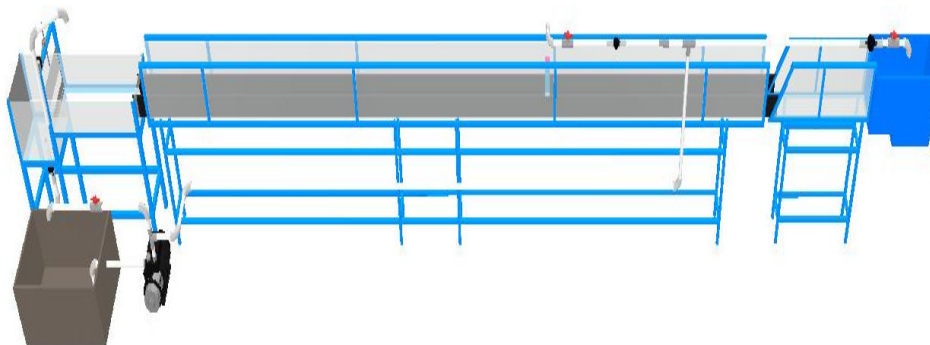
3.5 Tahap Perancangan

Tahap perancangan merupakan proses mendesain rangkaian dengan kata lain menganalisis dan membuat pola rancangan rangkaian yang merupakan langkah awal sebelum digunakan untuk membantu kerja operator sistem. Perancangan ini

terbagi atas dua bagian yaitu perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*).

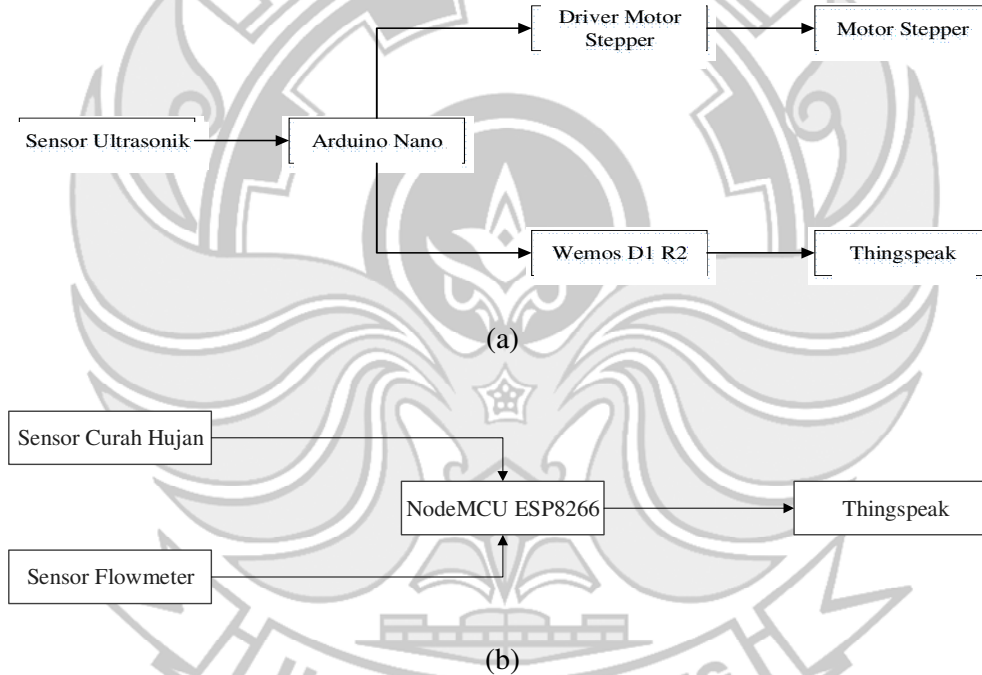
3.5.1 Perancangan Perangkat Keras (*Hardware*)

Rancangan sistem perangkat keras (*hardware*) pada sistem monitoring curah hujan dan bukaan *gate* bendungan pembangkit listrik tenaga air berbasis IoT (*Internet of Things*) digunakan beberapa komponen yaitu sensor ketinggian air berupa sensor *ultrasonik*, sensor *water flow*, sensor curah hujan, motor *stepper*, *driver motor*, Arduino Nano, NodeMCU ESP8266, Wemos D1 R2, *box plant* bendungan, *box panel*, *bevel gear*, pompa, *smartphone* dan *computer/laptop*. Rangkaian yang dibuat untuk memonitor curah hujan dan bukaan *gate* bendungan dari jarak jauh menggunakan *Internet of Things* dengan platform *Thingspeak* yang terhubung modul *wifi*. Data pembacaan sensor akan diteruskan ke mikrokontroler untuk pengumpulan data dan dikirim ke *Thingspeak cloud* yang merupakan tempat penyimpanan data agar dapat dimonitoring dari jarak jauh menggunakan *device smartphone* atau laptop.



Gambar 3. 2 Desain rancangan struktur kerangka prototype bendungan

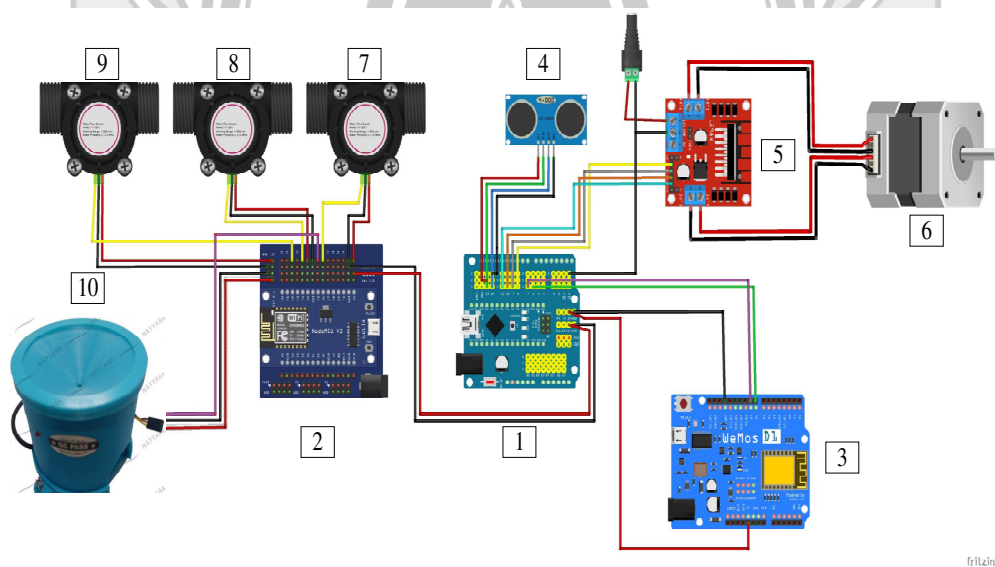
Untuk perencanaan kerangka box plant bendungan dapat dilihat pada Gambar 3.2, merupakan ilustrasi bagian kerangka dari bahan besi siku yang berfungsi sebagai tempat meletakkan box plant bendungan dan box panel terbuat dari bahan akrilik yang berisi komponen-komponen elektronika seperti modul *wifi*/mikrokontroler, kabel-kabel dan *driver motor stepper*. Skema dari rangkaian Sistem Monitoring Curah Hujan dan Buka-an Gate Bendungan PLTA Berbasis IoT sebagai berikut:



Gambar 3. 3 Skema Rangkaian Sistem Monitoring Berbasis IoT; (a) Pertama; b) Kedua

Berdasarkan Gambar 3.4 (a), Arduino Nano merupakan pusat dari sistem, ini dikarenakan Arduino Nano berperan sebagai mikrokontroler. Hasil pembacaan dari sensor ultrasonik akan terkirim ke Arduino Nano untuk diproses dan keluarannya akan diteruskan menuju ke Wemos D1 R2 dan *Driver Motor Stepper*.

Wemos D1 R2 merupakan modul *wifi* yang berperan untuk menghubungkan ke internet akan mengirim hasil pembacaan sensor menuju platform Thingspeak. Kemudian *driver* motor *stepper* yang berfungsi untuk membuka atau menutup *gate* beroperasi secara otomatis jika hasil pembacaan sensor ultrasonik telah mencapai pembacaan yang telah ditentukan. Sedangkan Gambar 3.4 (b), NodeMCU ESP8266 merupakan pusat dari sistem, karena NodeMCU ESP8266 bukan hanya berperan sebagai mikrokontroler tetapi sebagai modul *wifi*. Hasil pembacaan dari sensor-sensor akan terkirim ke NodeMCU ESP8266 kemudian NodeMCU ESP8266 akan mengirim hasil pembacaan tersebut ke Thingspeak. Untuk hasil realisasi dari skema rangkaian sistem monitoring adalah sebagai berikut:



Gambar 3. 4 Skema Rangkaian Sistem Monitoring

Keterangan:

1. Arduino Nano V3 ATMEGA328P
2. NodeMCU ESP8266

3. Wemos D1 R2
4. Sensor Ultrasonik HC-SR04
5. *Driver* L289N
6. Motor *Stepper* Nema 17
7. Sensor *Flowmeter* FS300A
8. Sensor *Flowmeter* YF-S201
9. Sensor *Flowmeter* DN40
10. *Tipping Bucket*

3.5.2 Perancangan Perangkat Lunak (*Software*)

Terdapat dua tahapan dalam perancangan perangkat lunak yaitu proses pembuatan program di *software* Arduino IDE dan pembuatan sistem pemantauan melalui platform *Thingspeak*.

1) Perancangan Program di *Software* Arduino IDE

Pada perancangan program Arduino IDE terdapat beberapa program yang akan dibuat yaitu pembuatan program sistem monitoring curah hujan yang mempengaruhi naiknya permukaan air pada bendungan dan monitoring ketinggian air pada bendungan untuk pengontrolan bukaan *gate* bendungan secara otomatis ditampilkan pada *device smartphone* yang terhubung dengan mosul *wifi* melalui platform *Thingspeak*. Prosedur perancangan perangkat lunak untuk program monitoring bukaan *gate* sebagai berikut:

- a) Menginstal *Software Arduino IDE* pada Komputer.
- b) Membuat program baru.

c) Program dibuat pada halaman *sketch*, Struktur Arduino IDE terdiri dari beberapa bagian yaitu:

1. *Header*

Header berisikan *library* dari setiap komponen yang digunakan.

2. *Void setup ()*

Pada bagian *Void setup* berfungsi untuk menyambungkan variabel pembacaan sensor dan komunikasi dengan perangkat keras mikrokontroler dan hanya dijalankan sekali selama program dijalankan.

3. *Void loop ()*

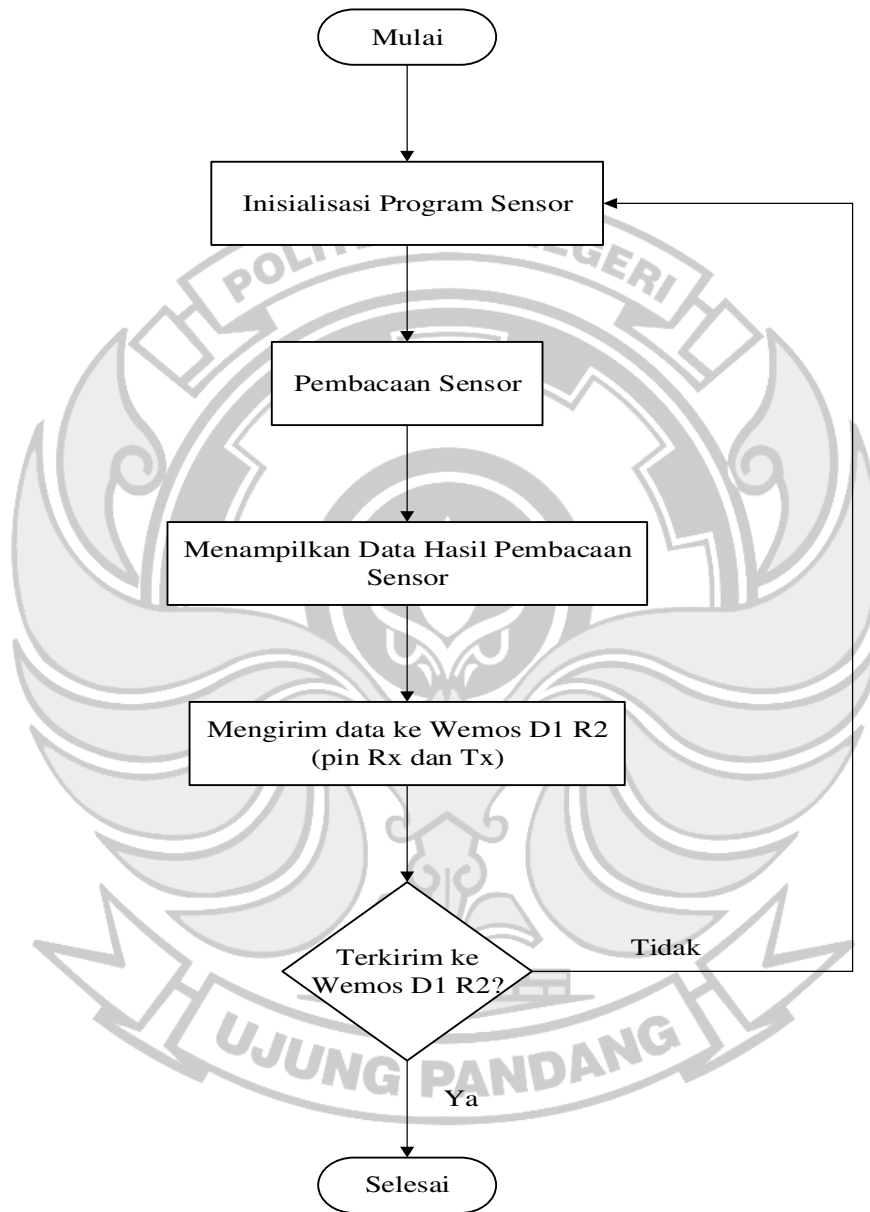
Pada bagian *void loop ()* merupakan sebuah fungsi berjalan secara berulang ulang dan terus menerus dan bisa disebut juga fungsi ini merupakan program utama yang berisi tentang perintah untuk pembacaan sensor.

d) Setelah itu melakukan *verify/compile* untuk menemukan apakah terdapat error pada pemograman.

e) Program di-upload untuk melakukan pengujian.

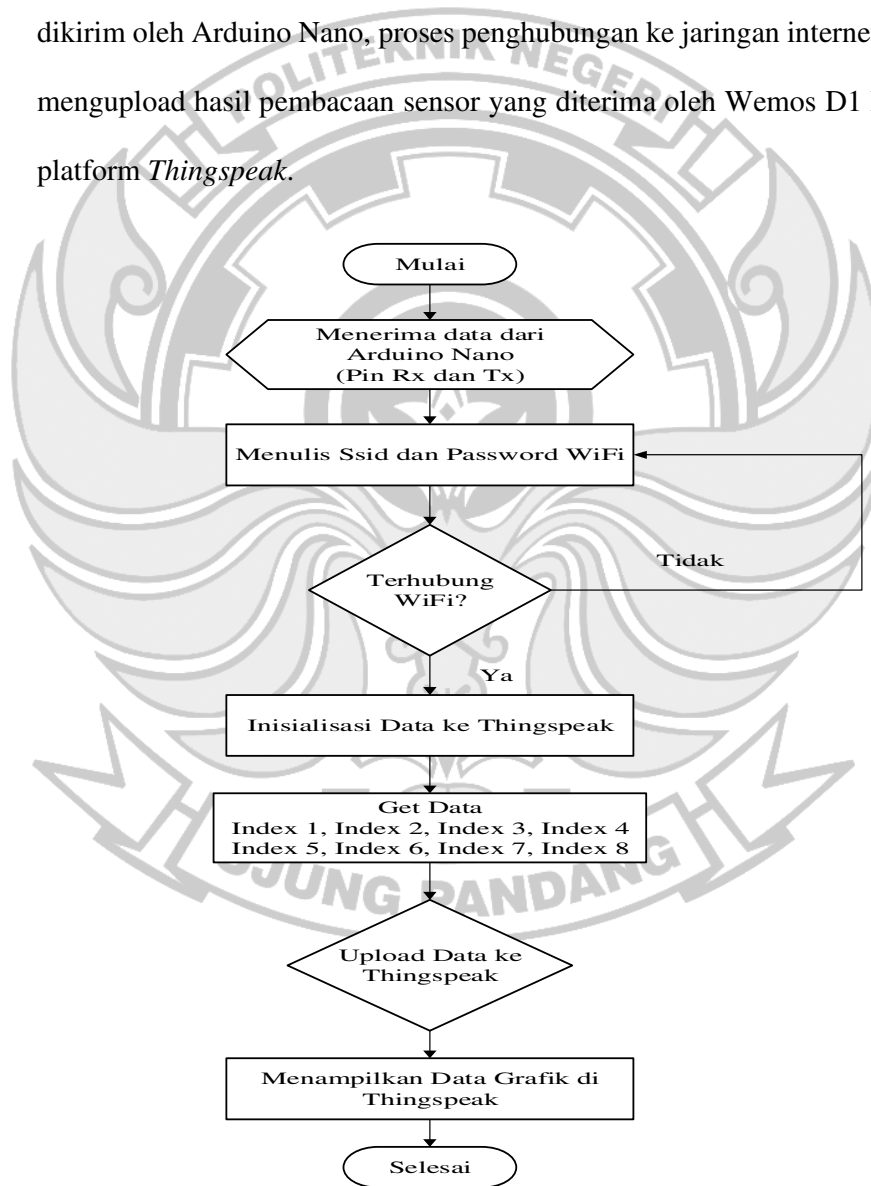
Pada sistem monitoring curah hujan dan bukaan *gate* bendungan berbasis IoT terdapat 3 buah *sketch pemograman*. *Sketch* pertama berisi program yang diupload ke dalam Arduino Nano, dalam *sketch* ini program yang diupload berupa *library* motor *stepper*, sensor ultrasonik dan *Json*. Bagain *void setup* berisikan perintah untuk menjalankan dan menghubungkan perangkat yang terhubung ke Arduino Nano. Bagian *void loop* berisikan program untuk pembacaan sensor ultrasonik dan

pengoperasian motor *stepper* kemudian pembacaan sensor ultrasonik akan dikirimkan kepada modul *wifi* Wemos D1 R2.



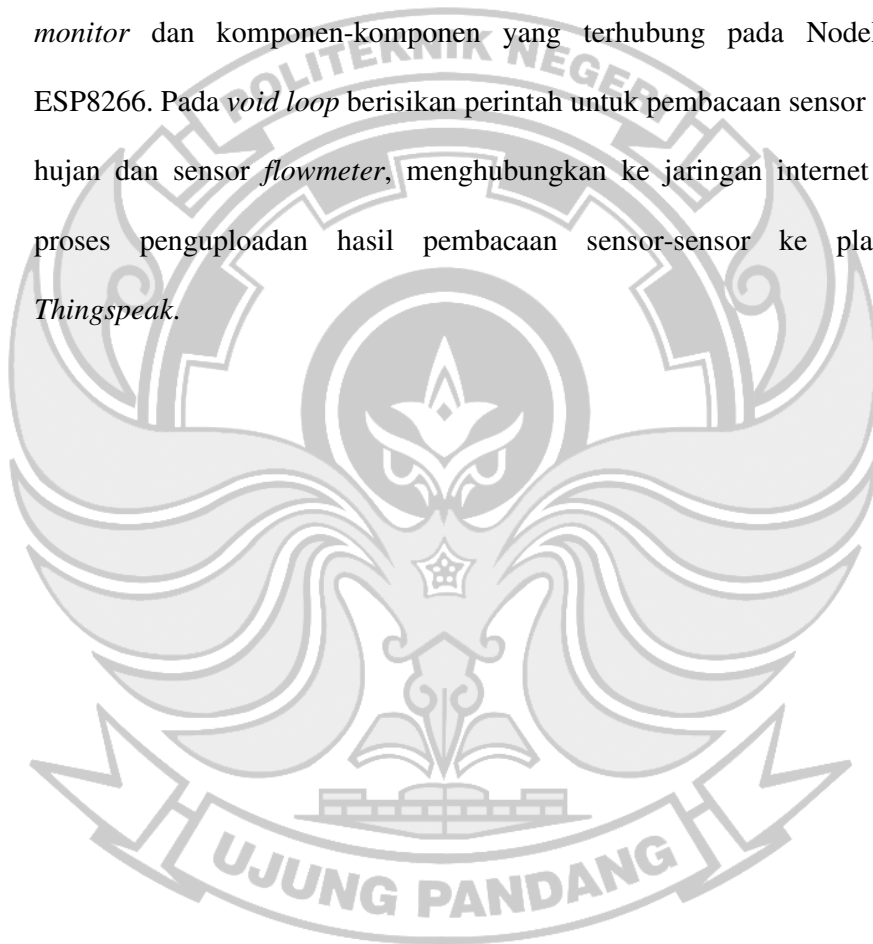
Gambar 3. 5 *Flowchart* Arduino Nano

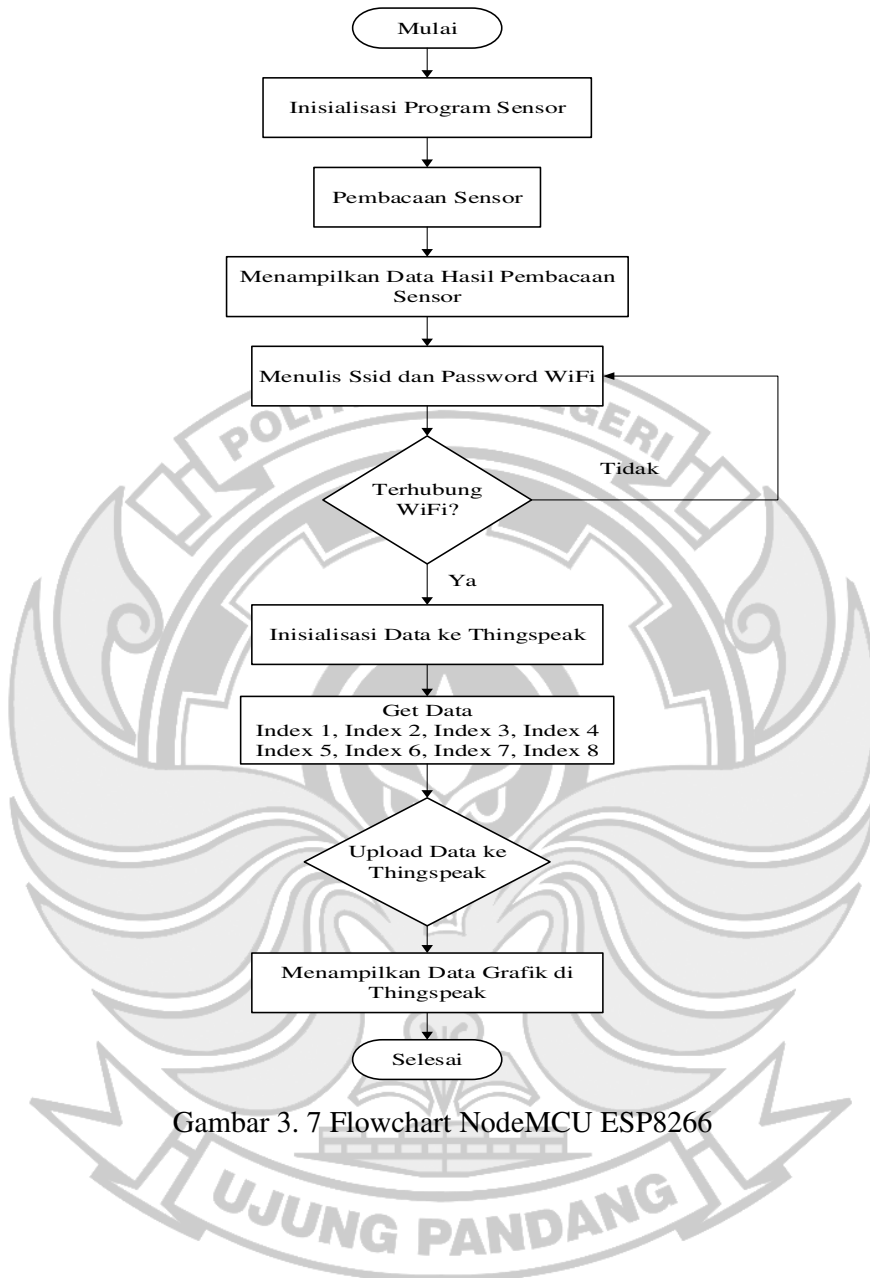
Sketch kedua merupakan program yang diupload ke Wemos D1 R2, dimana hasil pembacaan sensor yang diterima berasal dari Arduino Nano. Pada *sketch* kedua ini berisi *library Thingspeak*, *wifi* dan *Json*. Pada *void setup* berisikan perintah inialisasi *serial monitor* dan *Thingspeak*. Kemudian *void loop* berisikan perintah untuk penerimaan hasil data yang dikirim oleh Arduino Nano, proses penghubungan ke jaringan internet serta mengupload hasil pembacaan sensor yang diterima oleh Wemos D1 R2 ke platform *Thingspeak*.



Gambar 3. 6 Flowchart Wemos D1 R2

Sketch ketiga merupakan program yang diupload ke NodeMCU ESP8266, dimana NodeMCU ESP8266 memiliki dua peranan yaitu sebagai mikrokontroler dan modul *wifi*. Pada *sketch* ini berisikan *library Thingspeak, wifi*, deklarasi sensor curah hujan dan sensor *flowmeter*. Pada *void setup* berisikan perintah untuk menghubungkan *Thingspeak serial monitor* dan komponen-komponen yang terhubung pada NodeMCU ESP8266. Pada *void loop* berisikan perintah untuk pembacaan sensor curah hujan dan sensor *flowmeter*, menghubungkan ke jaringan internet serta proses penguploadan hasil pembacaan sensor-sensor ke platform *Thingspeak*.





Gambar 3. 7 Flowchart NodeMCU ESP8266

2) Pembuatan sistem pemantauan platform *Thingspeak*.

Berikut langkah-langkah untuk perancangan sistem monitoring dengan platform *Thingspeak*:

- a) Membuka *browser* pada PC dengan memasukkan alamat *web Thingspeak*.
- b) Membuat akun *Thingspeak* di *MathWorks*.
- c) *Login* pada *Thingspeak* dengan akun yang telah dibuat.
- d) Membuka *new channel* pada *Thingspeak*.
- e) Membuat nama dan membuat *channel* sesuai dengan kebutuhan.
- f) Beralih ke *software* Arduino IDE yang digunakan untuk pemrograman.
- g) Setelah semua program dibuat, program akan diupload.
- h) *Channel* pada *Thingspeak* akan menampilkan hasil data dalam bentuk grafik dan data.

3.6 Tahap Pembuatan dan Perakitan

3.6.1 Rangka *Box Plant* dan *Panel Box*

Prosedur pembuatan dan perakitan rangka box plant dan *panel box* adalah sebagai berikut:

- 1) Menyiapkan alat dan bahan yang diperlukan
- 2) Memotong bahan akrilik sesuai ukuran yang telah direncanakan, untuk membuat box plant dan *panel box*.
- 3) Menggabungkan potongan akrilik yang telah dibuat menggunakan lem bakar dan lem khusus akrilik.

- 4) Membuat rangka untuk box plant, dengan memotong besi siku sesuai dengan ukuran dimensi box plant yang dibuat.
- 5) Menyusun potongan besi siku sebagai tempat meletakkan box plant lalu menyambung besi siku dengan cara dilas dan menggunakan baut mur.
- 6) Setelah menentukan dan menyesuaikan lurus dari tempat meletakkan box plant, selanjutnya melakukan pemotongan dan pemasangan besi siku yang akan dijadikan sebagai kaki rangka.
- 7) Setelah melakukan pemasangan dan penyambungan rangka, selanjutnya rangka melalui proses pengecatan untuk menghindari terjadinya korosi pada besi siku.
- 8) Setelah proses untuk rangka selesai, selanjutnya box plant dan *panel box* dipasang pada rangka.

3.6.2 Rangkaian Untuk Sistem Monitoring.

Prosedur perakitan sistem monitoring curah hujan dan bukaan *gate* bendungan sebagai berikut:

- 1) Menyiapkan alat dan bahan yang akan digunakan.
- 2) Merangkai sistem monitoring yang terdiri dari sensor-sensor, Arduino Nano, Wemos D1 R2, NodeMCU ESP8266, dan motor *stepper* pada *box panel* monitoring.
- 3) Kemudian *box panel* dipasang pada bagian bawah *box plant* bendungan

3.7 Prosedur Pengujian

Setelah melakukan pembuatan dan perakitan selesai maka akan dilakukan dengan pengujian alat. Pengujian alat dilakukan untuk mengetahui unjuk dari sistem monitoring curah hujan dan bukaan *gate* bendungan ini apakah sudah dapat bekerja dengan baik atau tidak sesuai dengan fungsinya. Adapun langkah-langkah yang akan dilakukan adalah sebagai berikut:

- 1) Memasang rancangan bangun sistem monitoring curah hujan dan bukaan *gate* bendungan pembangkit listrik tenaga air berbasis IoT pada lokasi pengujian.
- 2) Menyalakan pompa air.
- 3) Mengatur bukaan katup *bypass*.
- 4) Motor *stepper* akan membuka *gate* secara otomatis sesuai dengan ketinggian air yang telah ditentukan.
- 5) Pengujian terdapat dua kondisi bukaan *gate* yaitu 50% dan 100%.
- 6) Pada pengujian bukaan *gate* 100%, buka katup *nozzle* pada sensor curah hujan
- 7) Mencatat hasil pembacaan sensor-sensor pada *thingspeak*.
- 8) Membuat kesimpulan dari hasil pengujian.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas mengenai hasil dan pembahasan Sistem Monitoring Curah Hujan dan Buka-an Gate Bendungan PLTA Berbasis IoT yang terdiri dari perangkat keras dan perangkat lunak.

Pengujian alat sistem monitoring ini terdiri dari beberapa tahapan pengujian, dimulai dari pengujian sensor yang digunakan, pengujian pemrograman, hingga pengujian sistem secara keseluruhan. Dari hasil pengujian yang diperoleh maka dapat dilakukan analisis kinerja dari bagian-bagian sistem yang saling berintegrasi sehingga terbentuk sistem monitoring curah hujan dan buka-an *gate*.

4.1 Hasil Pengujian Perangkat Keras (*Hardware*)

4.1.1 Pengujian Sensor Ultrasonik HC-SR04

Pengujian sensor ultrasonik HC-SR04 dilakukan dengan tujuan untuk menentukan ketelitian sensor saat membaca ketinggian air pada bendungan. Sensor ultrasonik ini diletakkan disekitar pintu air. Pengujian sensor ini dilakukan dengan cara membandingkan hasil pembacaan sensor dengan hasil pembacaan menggunakan penggaris. Pengujian dibagi menjadi dua kondisi yaitu saat pintu air terbuka 50% dan pintu air terbuka 100% dan setiap kondisi dibutuhkan 10 kali pengujian proses terjadi buka tutup *gate*.

Tabel 4. 1 Hasil Pengujian Sensor HC-SR04 Dengan Bukaannya *Gate* 50%

No	h_{sensor} (cm)	h_{aktual} (cm)	Selisih	Eror (%)
1.	7,6	8	0,4	4,93
2.	7,6	7,9	0,3	3,72
3.	7,6	8	0,4	4,93
4.	7,6	7,9	0,3	3,72
5.	7,6	8,1	0,5	6,10
6.	7,6	7,8	0,2	2,49
7.	7,6	7,8	0,2	2,49
8.	7,6	7,9	0,3	3,51
9.	7,6	7,8	0,2	2,27
10.	7,6	7,9	0,3	3,51
Rata-rata persentase eror				3,76

Tabel 4. 2 Hasil Pengujian Sensor HC-SR04 Dengan Bukaannya *Gate* 100%

No	h_{sensor} (cm)	h_{aktual} (cm)	Selisih	Eror (%)
1.	9,2	8,4	0,8	9,52
2.	9,2	8,5	0,7	7,68
3.	9,2	8,6	0,6	6,43
4.	9,2	8,6	0,6	6,43
5.	9,2	8,5	0,7	7,68
6.	9,2	8,5	0,7	7,68
7.	9,2	8,4	0,8	8,96
8.	9,2	8,6	0,6	6,43
9.	9,2	8,6	0,6	6,43
10.	9,2	8,5	0,7	7,68
Rata-rata persentase eror				7,49

Berdasarkan Tabel 4.1 dan Tabel 4.2 hasil pengujian sensor ultrasonik HC-SR04, dapat diketahui bahwa rata-rata persentase eror pada saat bukaan *gate* 50% dengan nilai sebesar 3,76% lebih kecil daripada bukaan *gate* 100% dengan nilai 7,49%. Hal ini terjadi karena yang menjadi objek dari sensor HC-SR04 merupakan air, permukaan dari air terus berubah-ubah seiring bertambahnya debit air pada hulu aliran. Pembacaan dari sensor HC-SR04 tidak begitu baik jika permukaan dari objeknya itu tidak datar.



Gambar 4. 1 Pengujian Sensor HC-SR04

4.1.2 Pengujian Sensor Curah Hujan *Tipping Bucket*.

Pengujian sensor curah hujan *tipping bucket* dilakukan dengan tujuan untuk menentukan ketelitian sensor saat membaca jumlah curah hujan yang terjadi pada bendungan. Sensor ini diletakkan ditengah alat. Pengujian sensor ini dilakukan dengan cara menuangkan air sebanyak 100 mL pada *tipping bucket* dengan diameter permukaan corong sebesar 80 mm kemudian menghitung berapa jumlah *tipping* yang terjadi. Pengujian ini membanding beberapa hasil pembacaan sensor.

Tabel 4. 3 Hasil Pengujian Sensor Curah Hujan *Tipping Bucket*

Pembacaan Sensor 1				Pembacaan Sensor 2			
V (mL)	n	V _b (mL/tip)	C (mm)	V (mL)	n	V _b (mL/tip)	C (mm)
100	24	4,17	0,83	100	21	4,76	0,95

Berdasarkan Tabel 4.3 hasil pengujian sensor *tipping bucket* membandingkan beberapa pembacaan sensor. Keduanya mengukur volume air yang sama yaitu sebanyak 100 mL, tetapi perbedaan kedua sensor ini terletak pada jumlah *tipping* (n) yang terjadi. Hal ini disebabkan oleh beberapa faktor antara lain kesalahan akibat air yang menempel pada sensor, jungkat jungkit sensor curah hujan tidak bekerja dengan baik dikarenakan kesalahan pembuatan mekaniknya atau tidak tepatnya pengaturan yang dilakukannya serta *bucket* menampung air secara berlebih sehingga jungkat jungkit tidak dapat berkerja.



Gambar 4. 2 Pengujian Sensor Tipping Bucket

4.1.3 Pengujian Sensor *Flowmeter*

Pengujian sensor *flowmeter* dilakukan dengan tujuan untuk menentukan ketelitian sensor saat membaca jumlah debit aliran. Pengujian sensor ini dilakukan dengan cara mengalirkan air melalui sensor *flowmeter* dimana air kelurannya akan mengalir ke wadah, kemudian hitung volume air serta waktu yang dibutuhkan hingga mencapai volume air tersebut. Setelah menemukan volume air dan waktu, kemudian menentukan debit yang diperoleh dan membandingkannya dengan hasil pembacaan sensor.

Tabel 4. 4 Hasil Pengujian Sensor *Flowmeter* YF-S201

Q_{sensor} (L/min)	Q_{aktual} (L/min)	Eror (%)
4,00	4,00	0,07
4,10	4,22	3,14
4,06	4,22	4,01
4,06	4,00	1,50
4,07	4,00	1,68
Rata-rata eror		2,08

Tabel 4. 5 Hasil Pengujian Sensor *Flowmeter* FS300A

Q_{sensor} (L/min)	Q_{aktual} (L/min)	Eror (%)
26,33	28,23	7,18
29,59	28,48	3,78
29,58	27,48	7,13
28,84	28,23	2,13
27,61	29,98	8,55
Rata-rata eror		5,75

Tabel 4. 6 Hasil Pengujian Sensor *Flowmeter* DN40

Q_{sensor} (L/min)	Q_{aktual} (L/min)	Eror (%)
15,07	15,89	5,45
16,07	16,88	5,07
16,41	17,87	8,91
16,30	14,90	8,59
16,38	14,90	9,04
Rata-rata eror		7,41

Berdasarkan Tabel 4.4, Tabel 4.5 dan Tabel 4.6 merupakan hasil pengujian dari ketiga sensor yaitu *flowmeter* YF-S201, *flowmeter* FS300A dan *flowmeter* DN40. Dimana masing-masing sensor *flowmeter* memiliki rata-rata eror 2,08%, 5,75% dan 7,41%. Ada beberapa faktor yang dapat menyebabkan turunnya kinerja dari sensor *flowmeter* antara lain terdapat kotoran pada rotor sensor, terjadinya perubahan sifat fluida seperti viskositas, suhu dan sebagainya serta tekanan air yang tidak stabil



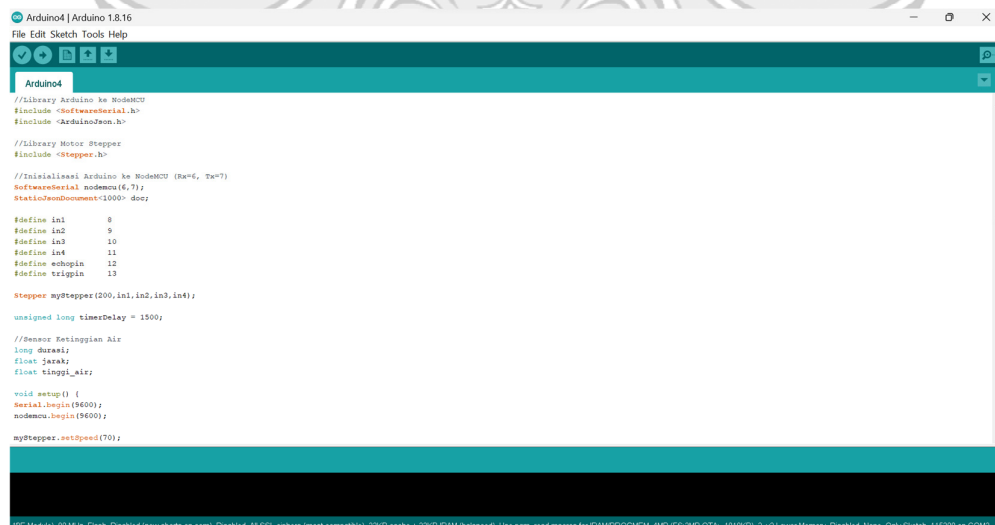
Gambar 4. 3 Pengujian Sensor *Flowmeter*

4.2 Hasil Pengujian Pemrograman Perangkat Lunak (*Software*)

Pengujian *software* ini terdiri dari tiga pengujian yaitu pengujian pemrograman pada Arduino Nano, Wemos D1 R2 dan NodeMCU ESP8266.

4.2.1 Pengujian Pemrograman Arduino Nano

Pada pemrograman Arduino Nano memiliki *sketch* yang berperan untuk menjalankan perintah untuk sensor ultrasonik dan mengoperasikan motor *stepper* secara otomatis dengan bantuan motor *driver*. Jika hasil pembacaan sensor ultrasonik telah mencapai ketinggian yang telah ditentukan, maka sensor ultrasonik akan mengirim sinyal ke Arduino Nano kemudian sinyal tersebut akan diproses oleh Arduino Nano dan setelah diproses Arduino Nano akan memerintahkan motor *driver* untuk mengoperasikan motor *stepper*. Kemudian Arduino Nano akan mengirim hasil pembacaan dari sensor ultrasonik tersebut ke Wemos D1 R2.



```
Arduino4 | Arduino 1.8.16
File Edit Sketch Tools Help

Arduino4
//library Arduino ke NodeMCU
#include <SoftwareSerial.h>
#include <Arduino.h>

//library Motor Stepper
#include <Stepper.h>

//inisialisasi Arduino ke NodeMCU (Ba=4, Tx=7)
SoftwareSerial nodemcu(4,7);
StaticJsonDocument<1000> doc;

#define in1 8
#define in2 9
#define in3 10
#define in4 11
#define echopin 12
#define trigpin 13

Stepper myStepper(200,in1,in2,in3,in4);

unsigned long timesDelay = 1500;

//Sensor Ketinggian Air
long durasi;
float jarak;
float tinggi_air;

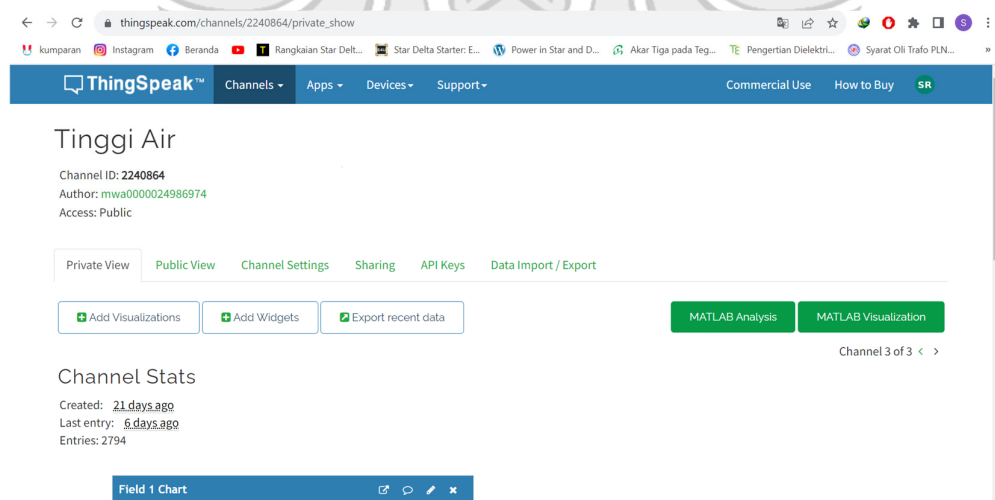
void setup() {
  Serial.begin(9600);
  nodemcu.begin(9600);

  myStepper.setSpeed(70);
}
```

Gambar 4. 4 Tampilan *Sketch* di *Software* Arduino IDE

4.2.2 Pengujian Pemrograman Wemos D1 R2

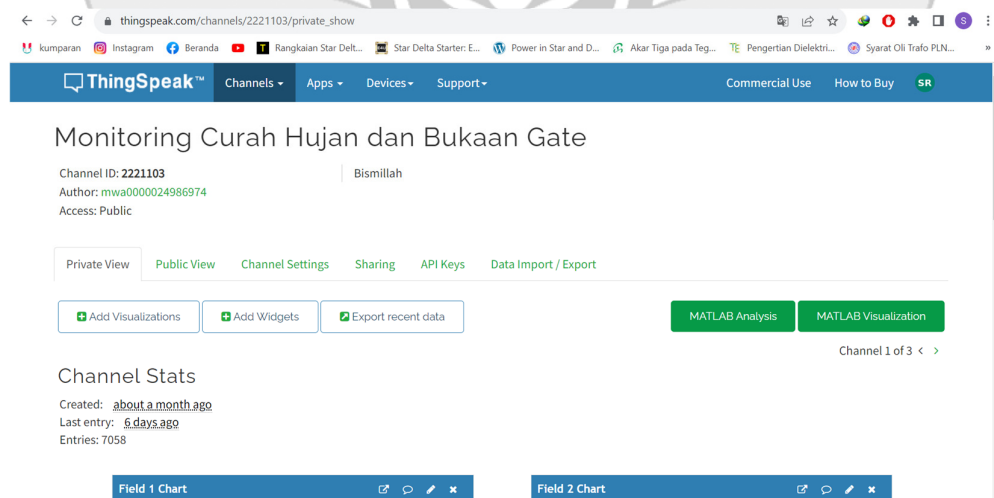
Sketch dari Wemos D1 R2 merupakan pemrograman yang akan menerima hasil pembacaan sensor ultrasonik dari Arduino Nano, dan selanjut data tersebut akan diolah dan ditampilkan pada platform *Thingspeak*. *Sketch* dari Wemos D1 R2 ini terdapat *user*, *address* nama *wifi* dan *password* yang akan digunakan untuk menghubungkan *wifi/hotspot*. Untuk menghubungkan program ini ke platform *Thingspeak* harus memasukkan “*myChannelNumber*” membantu mengalamatkan *channel* yang digunakan pada platform *Thingspeak* dan “*myWriteAPIKey*” untuk menampilkan data ke *channel*. Baik “*myChannelNumber*” maupun “*myWriteAPIKey*” dapat memperoleh ID-nya di platform *Thingspeak*. Nilai pembacaan dari platform *Thingspeak* akan menampilkan hasil pengolahan data dalam bentuk grafik dan data-data tersebut dapat diunduh dalam bentuk Microsoft Excel. Pengiriman data yang akan tampil ke *Thingspeak* membutuhkan minimal 15 detik.



Gambar 4. 5 Tampilan Platform *Thingspeak* untuk Wemos D1 R2

4.2.3 Pengujian Pemrograman NodeMCU ESP8266

Sama halnya dengan pengujian pemrograman Arduino Nano dan Wemos D1 R2, pengujian pemrograman NodeMCU ESP8266 memiliki keunikan yaitu dapat berperan sebagai mikrokontroler dan modul *wifi* sekaligus. Dimana *sketch* pemrograman NodeMCU ESP8266 terdapat program untuk menjalankan perintah pada sensor curah hujan *tipping bucket*, sensor *flowmeter* YF-S201, sensor *flowmeter* FS300A, sensor *flowmeter* DN40. Selain itu juga terdapat *user* dan *password* yang membantu menghubungkan ke jaringan *wifi/hotspot*, serta memiliki “*myChannelNumber*” maupun “*myWriteAPIKey*” yang memiliki ID yang berbeda dari Wemos D1 R2. Salah satu perbedaan dari *sketch* NodeMCU ESP8266 yaitu tidak terdapat *Json* yang berperan sebagai komunikasi antar dua mikrokontroler.



Gambar 4. 6 Tampilan Platform *Thingspeak* Untuk NodeMCU ESP8266

4.3 Hasil Pengujian Keseluruhan Sistem Monitoring.

Hasil pengujian keseluruhan alat ini meliputi pengujian *hardware* dan pengujian *software* yang telah dirancang. Pengujian yang dilakukan meliputi pembacaan sensor ultrasonik HC-SR04 dimana hasil pembacaannya berupa ketinggian air, pembacaan sensor *Tipping Bucket* dimana hasil pembacaannya berupa curah hujan, sensor *flowmeter* DN40 yang akan mengukur debit air yang mengalir setelah *gate*, sensor *flowmeter* FA300A yang akan mengukur debit aliran pada hulu aliran, sensor *flowmeter* YF-S201 yang akan mengukur debit aliran hujan serta *driver* L289N yang akan beroperasi secara otomatis jika hasil pembacaan ketinggian air telah mencapai batas yang telah ditentukan.

Pengujian ini dilakukan dengan dua kondisi bukaan *gate* yaitu bukaan *gate* 50% dan bukaan *gate* 100%. Setiap kondisi akan dilakukan sebanyak 10 kali pengujian.

Tabel 4. 7 Hasil Pengujian Sistem Monitoring dengan Bukaan *Gate* 50%

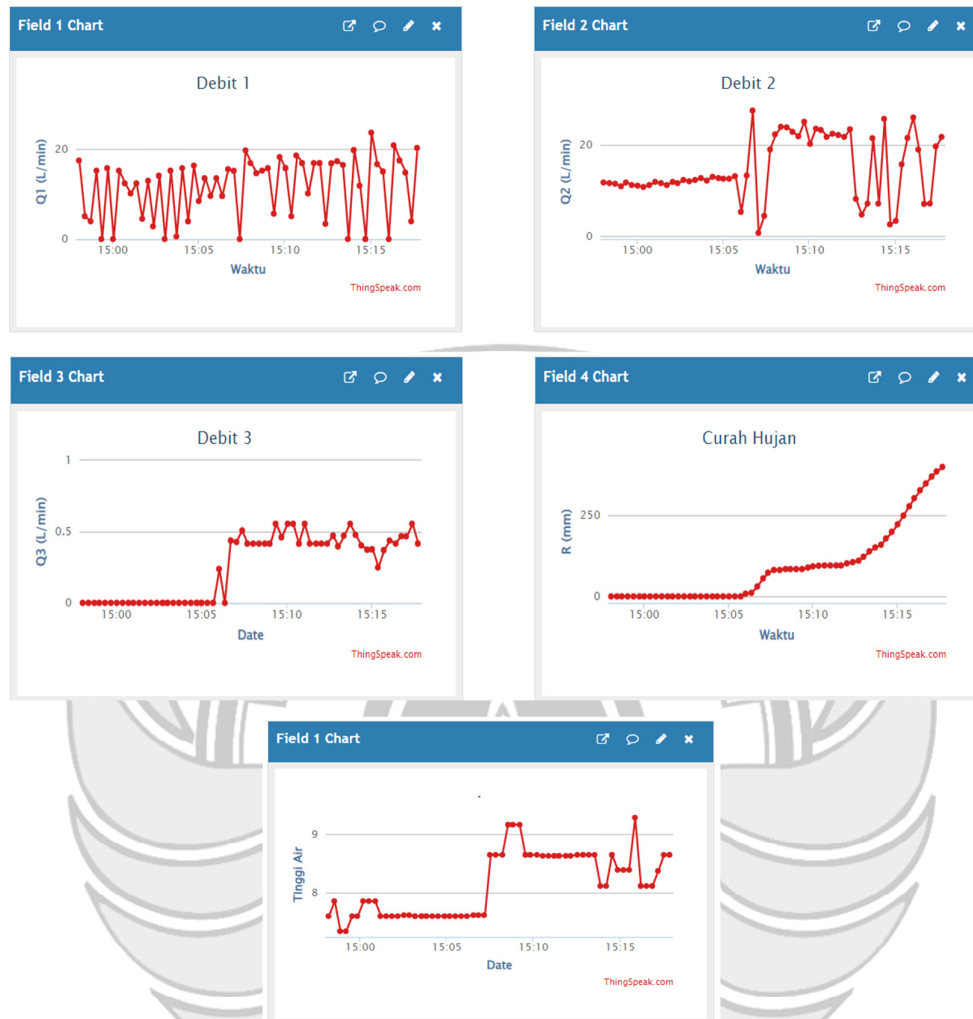
No	h (cm)	Q ₁ (L/min)	Q ₂ (L/min)	Q ₃ (L/min)	C (mm)
1	7,4	15,24	10,96	0	0
2	7,6	15,80	11,24	0	0
3	7,9	15,24	10,82	0	0
4	7,6	12,42	11,66	0	0
5	7,6	12,98	11,93	0	0
6	7,6	14,11	12,35	0	0
7	7,6	15,24	12,35	0	0
8	7,6	15,80	12,21	0	0
9	7,6	16,37	12,77	0	0
10	7,6	13,55	12,63	0	0
Rata-rata	7,61	14,68	11,89	0	0

Tabel 4. 8 Hasil Pengujian Sistem Monitoring dengan Buka *Gate* 100%

No	h (cm)	Q ₁ (L/min)	Q ₂ (L/min)	Q ₃ (L/min)	C (mm)
1	8,6	16,93	22,34	0,42	80,96
2	9,2	15,80	22,89	0,42	83,72
3	8,6	18,27	25,11	0,46	88,32
4	8,6	18,63	23,31	0,42	94,76
5	8,6	16,93	22,20	0,42	94,76
6	8,6	17,35	4,76	0,40	101,20
7	8,6	19,83	7,17	0,48	104,88
8	8,6	23,72	3,39	0,38	109,48
9	8,1	20,88	19,01	0,42	121,44
10	8,6	20,32	21,78	0,42	138,00
Rata-rata	8,61	18,87	17,20	0,42	101,75

Hasil pengukuran diatas merupakan hasil uji coba yang dilakukan pada kondisi yang berbeda yaitu saat bukaan *gate* sebesar 50% atau selebar 1,865 cm dan saat bukaan *gate* sebesar 100% atau selebar 3,73 cm. Pada Tabel 4.7 diketahui bahwa *gate* akan terbuka setengah jika ketinggian air telah mencapai antara 7,4 cm hingga 7,6 cm. Debit limpasan terjadi akibat terbukanya *gate* dan besarnya nilai debit limpasan bergantung terhadap tinggi level air yang terjadi pada bendungan. Kemudian ketinggian air juga dipengaruhi oleh seberapa besar debit pada hulu aliran dan pada kondisi ini belum terjadinya simulasi hujan.

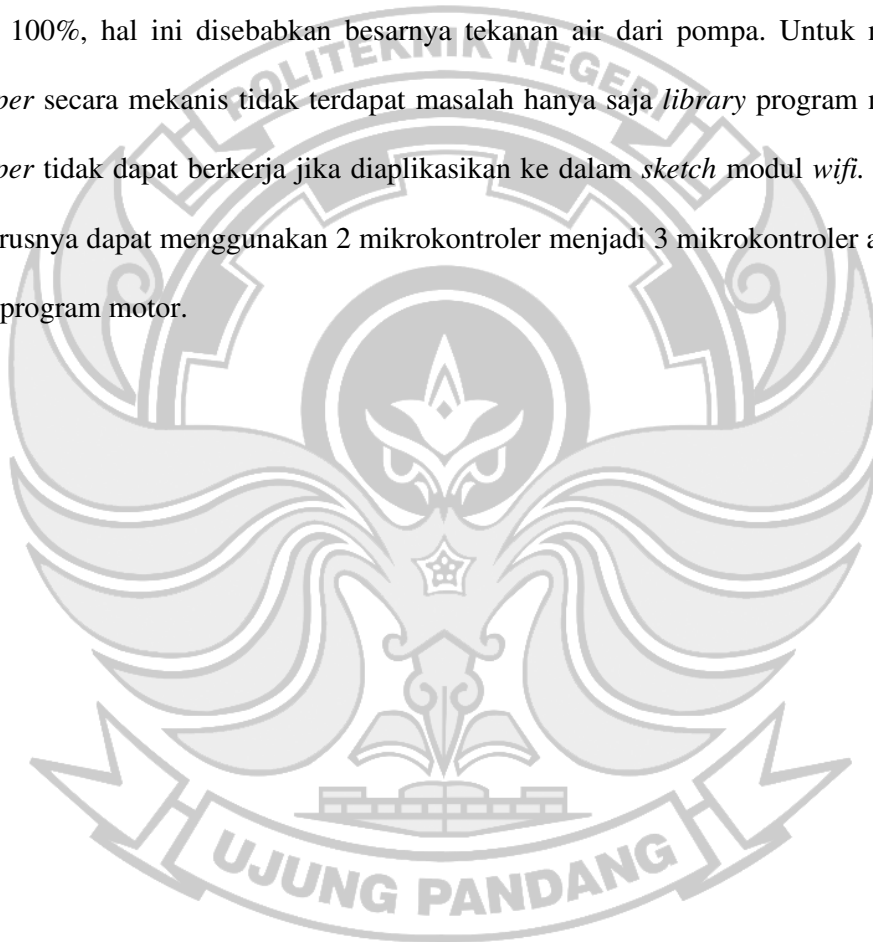
Sedangkan pada Tabel 4.8, *gate* akan terbuka penuh jika ketinggian air telah mencapai antara 8,6 cm hingga 9,2 cm. Dimana saat kondisi ini telah terjadi simulasi hujan sehingga aliran hulu meningkat dan akan berdampak pada limpasan akibat naiknya level air.



Gambar 4. 7 Grafik Pada Platform Thingspeak

Dari semua pengujian yang telah dilakukan, terdapat beberapa masalah yang muncul antara lain pengiriman data ke *Thingspeak* memiliki *delay* selama 15 detik sehingga sulit untuk memperoleh data yang lebih *real time* sesuai dengan yang diinginkan, contohnya sistem akan mulai beroperasi jika pembacaan sensor HC-SR04 antara 7,5 cm hingga 7,8 pada saat bukaan 50% dan jika pembacaan sensor HC-SR04 9 cm hingga 9,5 cm maka sistem akan melakukan kondisi bukaan 100%, sehingga saat hasil pembacaan sensor HC-SR04 pada *Thingspeak* sulit

menampilkan data ketinggian air yang sesuai dengan hasil pembacaan sensor HC-SR04 yang sebenarnya. Kemudian sensor ultrasonik HC-SR04 memiliki eror yang cukup besar diakibatkan air yang merupakan objek dari sensor tidak dalam kondisi tenang dan akan jauh lebih tidak tenang jika aliran hulu lebih besar. Dan juga *flowmeter* FS300A dan YF-S201 akan memiliki eror yang besar pada saat bukaan *gate* 100%, hal ini disebabkan besarnya tekanan air dari pompa. Untuk motor *stepper* secara mekanis tidak terdapat masalah hanya saja *library* program motor *stepper* tidak dapat berkerja jika diaplikasikan ke dalam *sketch* modul *wifi*. Yang seharusnya dapat menggunakan 2 mikrokontroler menjadi 3 mikrokontroler akibat dari program motor.



BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil pembuatan dan pengujian alat yang telah dilakukan maka dapat diambil kesimpulan sebagai berikut:

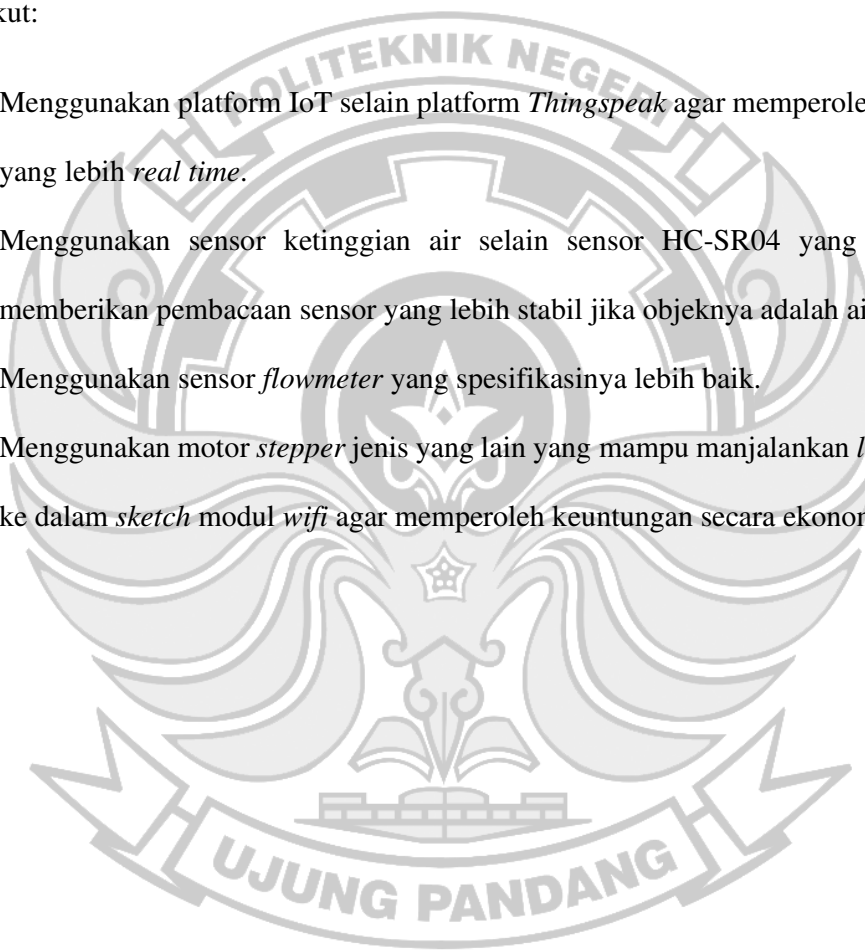
1. Sistem monitoring terhadap parameter-parameter dengan menggunakan platform *Thingspeak* dapat berjalan dengan baik hanya saja tidak dapat menampilkan data yang lebih *real time* dikarenakan *delay* pengiriman data ke *Thingspeak* membutuhkan waktu selama 15 detik.
2. Pengontrolan bukaan *gate* secara otomatis dengan bantuan sensor HC-SR04 dapat bekerja dengan dua kondisi berbeda yaitu bukaan *gate* 50% yang lebar bukaan *gate*-nya sebesar 1,865 cm dan bukaan *gate* 100% yang lebar bukaan *gate*-nya sebesar 3,73 cm. Dimana pada bukaan 50% tidak terjadi simulasi hujan dan rata-rata ketinggian air yang terjadi sebesar 7,61 cm, rata-rata debit limpasan sebesar 14,68 L/min dan rata-rata debit aliran hulu sebesar 11,89 L/min. Sedangkan pada bukaan 100% sudah terjadi simulasi hujan dan rata-rata ketinggian air yang terjadi sebesar 8,61 cm, rata-rata debit limpasan sebesar 18,87 L/min, rata-rata debit aliran hulu sebesar 17,20 L/min, rata-rata debit curah hujan sebesar 0,42 L/min dan rata-rata curah hujan sebesar 101,75 mm. Namun akibat kondisi air tidak tenang maka pembacaan sensor HC-SR04 mengalami eror yang cukup signifikan sehingga bukaan *gate* secara otomatis beroperasi walaupun belum berada di ketinggian air aktual yang diinginkan yaitu untuk bukaan *gate* 50% akan terbuka saat level air telah berada pada 7,5

cm hingga 7,8 cm sedangkan bukaan *gate* 100% akan terbuka saat level air berada pada 9 cm hingga 9,5 cm.

5.2 Saran

Adapun saran untuk pengembangan lebih lanjut dari Tugas Akhir ini sebagai berikut:

1. Menggunakan platform IoT selain platform *Thingspeak* agar memperoleh data yang lebih *real time*.
2. Menggunakan sensor ketinggian air selain sensor HC-SR04 yang dapat memberikan pembacaan sensor yang lebih stabil jika objeknya adalah air.
3. Menggunakan sensor *flowmeter* yang spesifikasinya lebih baik.
4. Menggunakan motor *stepper* jenis yang lain yang mampu menjalankan *library* ke dalam *sketch* modul *wifi* agar memperoleh keuntungan secara ekonomis.



DAFTAR PUSTAKA

- Aji, Bagus Widoto. 2017. Modelling dan Analisis Berdasarkan Studi Eksperimental Pengaruh Profil Gigi Terhadap Regangan Dinamis Pada Roda Gigi Kerucut Lurs Dengan Metode Elemen Hingga. Skripsi. Surabaya: Institut Teknologi Sepuluh Nopember (ITS).
- Arisandi, Akhmad. 2019. Prototipe Alat Pengukur Intensitas Hujan Otomatis Tipe Tipping Bucket Berbasis IoT. Skripsi. Jember: Universitas Jember.
- Chwalisz, M. (2019). ThingSpeak Documentation. Mathworks, 1–2.(*Online*), (https://www.mathworks.com/help/thingspeak/index.html?s_tid=CRUX_lftnav) diakses 4 September 2023).
- Cahaya Ardhitamara, R. (2021). Perancangan Robot Pemotong Rumput Berbasis Android Dengan Kontrol Pwm Dan Variasi Pisau Potong (Doctoral dissertation, Universitas Muhammadiyah Ponorogo).
- Hardianto, R. R. (2017). Perancangan Web Monitoring Tempat Sampah Menggunakan Raspberry Pi Berbasis NRF24L01 (Doctoral dissertation, Institut Teknologi Nasional Malang).
- Kusuma, T., & Mulis, M. T. (2018). Perancangan Sistem Monitoring Infus Berbasis Mikrokontroler Wemos D1 R2. 1422-1425
- Kusuma, N. A. A. (2018). Rancang bangun smart home menggunakan wemos d1 r2 arduino compatible berbasis esp8266 esp-12f (Bachelor's thesis, Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta).
- Otomasi.ugm.com. Pengertian Internet of Things (IoT). Diakses dari <https://otomasi.sv.ugm.ac.id/2018/06/02/pengertian-internet-of-things-IoT>, (*Online*), (diakses pada 10 Januari 2023).
- Prabowo, R. R. (2023). *Rancang Bangun Monitoring Pergerakan Sudut Pada Gedung Untuk Simulasi Gempa Dengan Tuned Mass Damper Berbasis Arduino Mega 2560 (Doctoral dissertation, Universitas Diponegoro)*.
- Prasetya, Ardi Dwi. (2020). Rancang Bangun Sistem Monitoring dan Pendetejsu Lokasi Kebocoran Pipa Berdasarkan Analisis Debit Air Berbasis IoT, (*Online*), 12(1), 39-47, Universitas Trunojoyo Madura.
- Prastyo, E. A. 2019. Arduino Nano. (*Online*) www.arduinoindonesia.id: (<https://www.arduinoindonesia.id/2019/01/arduino-nano.html>), diakses 4 September 2023).

- Prayogo, Bambang dkk. 2019. Rancang Bangun Prototipe Sistem Monitoring Mini Stasiun Cuaca BMKG Provinsi Lampung. Skripsi. Bandar Lampung: Universitas Lampung.
- Purwanto, Heru dkk. 2019. Komparasi Sensor Ultrasonik HC-SR04 dan JSN-SR04T Untuk Aplikasi Sistem Deteksi Ketinggian Air. *Jurnal SIMETRIS*, (Online), 10 (2): 71-72, diakses 28 Agustus 2023).
- Ridwan, R., Swistiawan, M. H., & Bhaskoro, S. B. 2021. Otomatisasi Sistem Bendung menggunakan Metode Backpropagation untuk Mengatur Debit Air berbasis Internet of Thing. *JTRM (Jurnal Teknologi dan Rekayasa Manufaktur)*, 3(2), 73-86, Bandung: Teknik Otomasi Manufaktur dan Mekatronika Politeknik Manufaktur Bandung.
- Solih, A., & Jamaaluddin, J. (2017). Rancang Bangun Pengaman Panel Distribusi Tenaga Listrik Di Lippo Plaza Sidoarjo Dari Kebakaran Berbasis Arduino Nano. *JEEE-U (Journal of Electrical and Electronic Engineering-UMSIDA)*, (Online), 1(2), 61-68, Universitas Muhammadiyah Sidoarjo,
- Suharjono, A., Rahayu, L. N., & Afwah, R. (2016). Aplikasi Sensor Flow Water Untuk Mengukur Penggunaan Air Pelanggan Secara Digital Serta Pengiriman Data Secara Otomatis Pada PDAM Kota Semarang. *TELE*, 13(1), Jurusan Teknik Elektro, Politeknik Negeri Semarang, Indonesia
- Warriornux.com. Pengertian Modul Wifi ESP8266. 22 Oktober 2022. Diakses dari <https://www.warriornux.com/pengertian-modul-wifi-esp8266/>, (Online), (diakses 17 Januari 2023).
- Wibisono Darmawan, C., U A Sompie, S. R., & Kambey, F. D. (2020). Implementasi Internet of Things pada Monitoring Kecepatan Kendaraan Bermotor. *Jurnal Teknik Elektro Dan Komputer*, (Online), 9(2), 91-100, diakses 4 September 2023).
- Widiarto, Yosua D. dkk. 2018. Sistem Penggerak Robot Beroda Vacuum Cleaner Berbasis Mini Computer Raspberry pi. *Jurnal Teknik Elektro dan Komputer*, (Online), 7 (1): 25-32, diakses 5 September 2023).
- Yana, K. L., Dantes, K. R., & Wigraha, N. A. (2017). Rancang bangun mesin pompa air dengan sistem recharging. *Jurnal Pendidikan Teknik Mesin Undiksha*, (Online), 5(2), diakses 5 September 2023).
- Zahrok, R. F., Sakti, S. P., & Anggraeni, D. Rancang Bangun Pengontrol Jarak Menggunakan Motor Stepper Nema 17 Berbasis Mega 2560 Pro pada Ultrasonic Atomizer Spray Coating, diakses 5 september 2023).

LAMPIRAN

Lampiran 1 Sketch Arduino Nano

```
//Library Arduino ke NodeMCU
#include <SoftwareSerial.h>
#include <ArduinoJson.h>

//Library Motor Stepper
#include <Stepper.h>

//Inisialisasi Arduino ke NodeMCU (Rx=6, Tx=7)
SoftwareSerial nodemcu(6,7);
StaticJsonDocument<1000> doc;

#define in1      8
#define in2      9
#define in3      10
#define in4      11
#define echopin  12
#define trigpin  13

Stepper myStepper(200,in1,in2,in3,in4);

unsigned long timerDelay = 1500;

//Sensor Ketinggian Air
long durasi;
float jarak;
float tinggi_air;

void setup() {
  Serial.begin(9600);
  nodemcu.begin(9600);

  myStepper.setSpeed(70);
  pinMode(trigpin, OUTPUT);
  pinMode(echopin, INPUT);
}

void loop() {
  if ((millis() % timerDelay) == 0)
  {
    Serial.println(millis());
  }
}
```



```

doc["X"]      = Ketinggian_Air();

Serial.print("Ketinggian Air = ");
Serial.print(doc["X"].as<float>(),1);
Serial.println(" cm");

Serial.println("Sending to NodeMCU");
serializeJson(doc, nodemcu);
nodemcu.flush();
Serial.println();
}
}

float Ketinggian_Air(){
digitalWrite(trigpin, HIGH);
delayMicroseconds(5);
digitalWrite(trigpin, LOW);

durasi = pulseIn(echopin, HIGH);
jarak = (durasi*0.034)/2;
tinggi_air = 41.3 - jarak;

//Bukaan Gate 100%
while (tinggi_air >= 9 && tinggi_air <= 9.5)
{
Serial.print("Ketinggian Air : ");
Serial.print(tinggi_air,1);
Serial.println(" cm");
Serial.println("Gate Terbuka");
myStepper.step(-6000);
delay(15000);
Serial.println("Gate Tertutup");
myStepper.step(6000);
break;
}

//Bukaan Gate 50%
while (tinggi_air >= 7.5 && tinggi_air <= 7.8)
{
Serial.print("Ketinggian Air : ");
Serial.print(tinggi_air,1);
Serial.println(" cm");
Serial.println("Gate Terbuka");
myStepper.step(-3000);
delay(15000);
}

```

```
Serial.println("Gate Tertutup");  
myStepper.step(3000);  
break;  
}  
return tinggi_air;  
}
```



Lampiran 2 Sketch Wemos D1 R2

```
//recieve
#include <SoftwareSerial.h>
#include <ArduinoJson.h>

#include <ESP8266WiFi.h>
WiFiClient client;
const char* ssid = ".....";
const char* password = ".....";

//Initialise Arduino to NodeMCU (7=Rx & 6=Tx)
SoftwareSerial nodemcu(D7, D6);

#include "ThingSpeak.h"
unsigned long myChannelNumber = 2240864;
const char * myWriteAPIKey = "IQX64BUQCH41WCYQ";

// Timer variables
unsigned long lastTime = 0;

unsigned long timerDelay = 15000;

//pembacaan sensor
String message;

float Ketinggian_Air;

//json document
StaticJsonDocument<1000> doc;

void setup() {
  Serial.begin(9600); //Initialize serial
  nodemcu.begin(9600);

  ThingSpeak.begin(client); // Initialize ThingSpeak
}

void loop() {

  if (nodemcu.available() > 0) {

    DeserializationError error = deserializeJson(doc, nodemcu);
```

```

// Test if parsing succeeds
if (error) {
  Serial.println("deserializeJson error");
  return;
}
else
{
  Ketinggian_Air = doc["X"];
}
Serial.print("Ketinggian Air = ");
Serial.print(Ketinggian_Air,1);
Serial.println(" cm");
}

if ((millis() - lastTime) > timerDelay) {

  lastTime = (millis() / timerDelay) * timerDelay;
  Serial.println(lastTime);

  // Connect or reconnect to WiFi
  if (WiFi.status() != WL_CONNECTED) {
    Serial.print("Attempting to connect");
    while (WiFi.status() != WL_CONNECTED) {
      WiFi.begin(ssid, password);
      delay(10000);
    }
    Serial.println("\nConnected.");
  }
  ThingSpeak.setField(1, Ketinggian_Air);

  // upload ke thingspeak
  int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);

  if (x == 200) {
    Serial.println("Channel update successful.");
  }
  else {
    Serial.println("Problem updating channel. HTTP error code " + String(x));
  }
  serialFlush();
}

```

```
}  
  
void serialFlush() {  
  while (nodemcu.available() > 0) {  
    char t = nodemcu.read();  
  }  
}
```



Lampiran 3 Sketch NodeMCU ESP8266

```
#include <ESP8266WiFi.h>
WiFiClient client;
const char* ssid = ".....";
const char* password = ".....";

#define sensorflow1 D2
#define sensorflow2 D6
#define sensorflow3 D4
#define pin_hujan D5

#include "ThingSpeak.h"
unsigned long myChannelNumber = 1;
const char * myWriteAPIKey = "HHOLQWIXMFG4JZOJ";

// Timer variables
unsigned long lastTime = 0;
unsigned long timerDelay = 15000;

//pembacaan sensor
String message;

//Deklarasi Sensor Flowrate1
long previousMillis1 = 0;
float calibrationFactor1 = 1.34;//1 1/2 inch
volatile byte pulseCount1;
byte pulse1Sec1 = 0;
float debit_air1;
float flowLitres1;
float totalLitres1;

//Dekralasi Sensor Flowrate2
long previousMillis2 = 0;
float calibrationFactor2 = 4;
volatile byte pulseCount2;
byte pulse1Sec2 = 0;
float debit_air2;
float flowLitres2;
float totalLitres2;

//Dekralasi Sensor Flowrate3
long previousMillis3 = 0;
float calibrationFactor3 = 4,5;
```

```

volatile byte pulseCount3;
byte pulse1Sec3 = 0;
float debit_air3;
float flowLitres3;
float totalLitres3;

//Deklarasi Sensor Curah Hujan
float milimeter_per_tip = 0.92;
float curah_hujan = 0.00;
long int jumlah_tip = 0;
long int temp_jumlah_tip = 0;
volatile boolean flag = false;

void IRAM_ATTR pulseCounter1()
{
  pulseCount1++;
}

void IRAM_ATTR pulseCounter2()
{
  pulseCount2++;
}

void IRAM_ATTR pulseCounter3()
{
  pulseCount3++;
}

void ICACHE_RAM_ATTR hitung_curah_hujan()
{
  flag = true;
}

void setup() {
  Serial.begin(9600); //Initialize serial

  pinMode(pin_hujan, INPUT);
  pinMode(sensorflow1, INPUT);
  pinMode(sensorflow2, INPUT);
  pinMode(sensorflow3, INPUT);
  attachInterrupt(digitalPinToInterrupt(pin_hujan), hitung_curah_hujan, FALLING);
  attachInterrupt(digitalPinToInterrupt (sensorflow1), pulseCounter1, FALLING);
  attachInterrupt(digitalPinToInterrupt (sensorflow2), pulseCounter2, FALLING);
  attachInterrupt(digitalPinToInterrupt (sensorflow3), pulseCounter3, FALLING);
}

```

```

ThingSpeak.begin(client); // Initialize ThingSpeak

}

void loop() {
  if ((millis() - lastTime) > timerDelay) {

    lastTime = (millis() / timerDelay) * timerDelay;
    Serial.println(lastTime);

    // Connect or reconnect to WiFi
    if (WiFi.status() != WL_CONNECTED) {
      Serial.print("Attempting to connect");
      while (WiFi.status() != WL_CONNECTED) {
        WiFi.begin(ssid, password);
        delay(10000);
      }
      Serial.println("\nConnected.");
    }

    ThingSpeak.setField(1, debit_air1);
    ThingSpeak.setField(2, debit_air2);
    ThingSpeak.setField(3, debit_air3);
    ThingSpeak.setField(4, curah_hujan);

    // upload ke thingspeak
    int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);

    if (x == 200) {
      Serial.println("Channel update successful.");
    }
    else {
      Serial.println("Problem updating channel. HTTP error code " + String(x));
    }
  }
}

//Curah Hujan
if (flag == true)
{
  curah_hujan += milimeter_per_tip;
  jumlah_tip++;
}

```



```

    delay(500);
    flag = false;
}
curah_hujan = jumlah_tip * milimeter_per_tip;
if ((jumlah_tip != temp_jumlah_tip))
// Print serial setiap 1 menit atau ketika jumlah_tip berubah
{
Serial.print("Curah hujan=");
Serial.print(curah_hujan, 2);
Serial.println(" mm");
Serial.println();
}
temp_jumlah_tip = jumlah_tip;

//Flowmeter1
if (millis() - previousMillis1 > 1000)
{
    pulse1Sec1 = pulseCount1;
    pulseCount1 = 0;
    debit_air1 = ((1000.0 / (millis() - previousMillis1)) * pulse1Sec1) /
calibrationFactor1;
    previousMillis1 = millis();

    Serial.print("Flowrate 1 : ");
    Serial.print(debit_air1);
    Serial.println(" L/min");
    Serial.println();
}

//Flowmeter2
if (millis() - previousMillis2 > 1000)
{
    pulse1Sec2 = pulseCount2;
    pulseCount2 = 0;
    debit_air2 = ((1000.0 / (millis() - previousMillis2)) * pulse1Sec2) /
calibrationFactor2;
    previousMillis2 = millis();

    Serial.print("Flowrate2 : ");
    Serial.print(debit_air2);
    Serial.println(" L/min");
    Serial.println();
}

//Flowmeter3

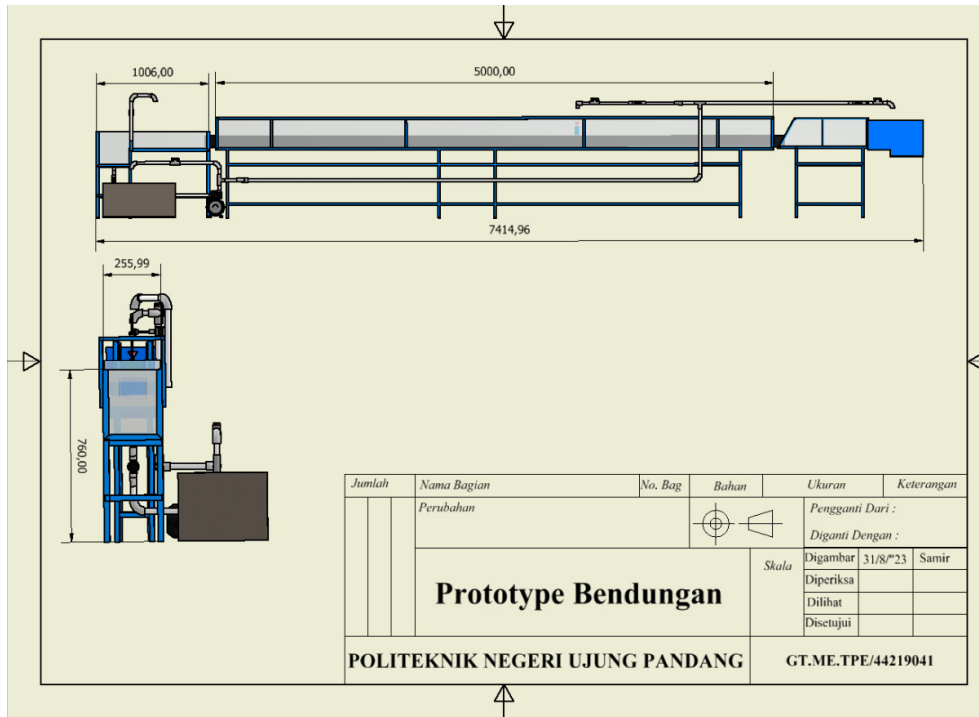
```

```
if (millis() - previousMillis3 > 1000)
{
  pulse1Sec3 = pulseCount3;
  pulseCount3 = 0;
  debit_air3 = ((1000.0 / (millis() - previousMillis3)) * pulse1Sec3) /
calibrationFactor3;
  previousMillis3 = millis();

  Serial.print("Flowrate3 : ");
  Serial.print(debit_air3);
  Serial.println(" L/min");
  Serial.println();
}
}
```



Lampiran 4 Gambar Teknik Rancangan Prototype Bendungan



Lampiran 5 Foto Kegiatan



(a)



(b)

Gambar 1. (a) Proses Pemotongan Akrilik (b) Pemasangan Potongan-potongan Akrilik



Gambar 2 Proses Pembuatan Rangka



Gambar 3 Proses Uji Coba Alat



Gambar 4 Tampilan Data Pada Platform Thingspeak Pada Android