

APLIKASI *INTERNET OF THINGS* (*IoT*) PADA SISTEM KENDALI POMPA AIR

Skripsi

Untuk memenuhi sebagian persyaratan
guna mencapai derajat Sarjana Teknik

Program Studi Srata 1
Jurusan Teknik elektro



Disusun Oleh:

Anung

312118052

Kepada

**INSTITUT TEKNOLOGI NASIONAL
YOGYAKARTA
2020**

HALAMAN PERSETUJUAN

Skripsi yang berjudul:

APLIKASI *INTERNET OF THINGS (IoT)* PADA SISTEM KENDALI POMPA AIR

yang diajukan oleh:

Anung

312118052

telah disetujui oleh pembimbing untuk diujikan.

Pembimbing I,

Joko Prasajo, ST., MT.

NIK: 19730069

Tanggal:

Pembimbing II,

Tugino, ST., MT.

NIK: 19730085

Tanggal:

HALAMAN PENGESAHAN

Tugas Akhir yang berjudul:

APLIKASI *INTERNET OF THINGS (IoT)* PADA SISTEM KENDALI POMPA AIR

yang dipersiapkan dan disusun oleh:

Anung
312118052

telah dipertahankan di depan Dewan Penguji
pada tanggal 06 Agustus 2020
dan dinyatakan lulus.

**Dewan Penguji,
Tandatangan:**

Tanggal:

Joko Prasajo, S.T., M.T.
Pembimbing I

1.

.....

Tugino, S.T., M.T.
Pembimbing II

2.

.....

Sudiana, S.T.
Penguji

3.

.....

Yogyakarta,.....
Program Studi Strata 1 Teknik Elektro
Ketua,

Ir. Hj. Oni Yuliani, M.Kom
NIP: 196407041991022001

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Anung
NIM : 312118052
Konsentrasi : Elektronika Kendali

Dengan ini menyatakan bahwa data yang tersaji dalam tugas akhir saya yang berjudul:

APLIKASI *INTERNET OF THINGS (IoT)* PADA SISTEM KENDALI POMPA AIR

adalah **MURNI** hasil penelitian saya pribadi.

Apabila di kemudian hari terbukti bahwa data dan judul tersebut merupakan jiplakan/plagiat dari karya tulis orang lain, maka sesuai dengan kode etik ilmiah, saya menyatakan bersedia untuk diberikan sanksi seberat-beratnya termasuk **PENCOPOTAN/PEMBATALAN** gelar akademik saya oleh pihak ITNY Yogyakarta

Demikian surat pernyataan ini dibuat agar dapat digunakan sebagaimana mestinya.

Yogyakarta, 2 Agustus 2020
Yang membuat pernyataan

Anung

HALAMAN MOTTO DAN PERSEMBAHAN

AKHIRAT HARUS DI PIKIR TAPI JANGAN LUPA DUNIA

Skripsi ini saya persembahkan pada:

1. Allah SWT, Allhamdulillah. Terima kasih atas kemudahan yang telah diberikan pada saya sehingga dapat menyelesaikan skripsi.
2. Kedua Orangtua, Kakak dan seluruh Keluarga atas dukungan biaya dan morilnya.
3. Dosen Pembimbing dan Dosen Penguji yang telah memberikan arahan dan masukan baik teknis maupun non teknis.
4. Seluruh Dosen dan Karyawan Teknik Elektro ITNY Yogyakarta atas bantuan Ilmu dan Administrasi.
5. Rekan-rekan Mahasiswa Teknik Elektro
6. Teman-teman seperjuangan kuliah.
7. -----
8. Teman-teman kantor yang selalu memberi semangat

APLIKASI *INTERNET OF THINGS (IoT)* PADA SISTEM KENDALI POMPA AIR

Anung

312118052

ABSTRAK

Penelitian ini bertujuan untuk merancang dan membuat sistem kendali pompa air melalui jarak jauh dengan memanfaatkan teknologi *Internet Of Things (IoT)* dengan mengaplikasikan *NodeMCU* sebagai otak pengendali. Prototype dari sistem diharapkan dapat diaplikasikan dan dikembangkan lebih sempurna lagi untuk sistem pengendalian jarak jauh dengan *IoT*.

Materi penelitian ini terdiri atas komponen-komponen yang mendukung perancangan perangkat-keras, seperti: *NodeMCU*, Motor DC, IC LM2596, Sensor suhu DS18B20, Sensor *flow* meter, Sensor ultrasonic HC-SR04, *Step Down Converter*, Transistor IRFZ44, Acyrlilik, Resistor, dan Capacitor, bahasa C++ dan Dart untuk mendukung perancangan perangkat-lunak dan Firebase sebagai database.

Hasil penelitian diwujudkan dengan bekerjanya alat yang dirancang dalam batasan spesifikasi yang telah dibuat. Hal ini diberikan dalam bentuk hasil pengujian perangkat-keras, yang terdiri atas, pengujian rangkaian *Step Down Converter*, pengujian rangkaian sensor ultrasonic HC-SR04, pengujian driver motor, pengujian perangkat lunak serta pengujian sistem secara keseluruhan. Pada pengujian rangkaian *Step Down Converter* diperoleh besarnya tegangan keluaran dari IC regulator sebesar 5,0 V, pada pengujian sensor ultrasonic HC-SR04 menunjukkan bahwa semakin lama sinyal echo di terima semakin panjang hasil pengukuran. Pada pengujian Sensor flow semakin cepat laju air semakin tinggi frekuensi yang dihasilkan.. Pengujian keseluruhan sistem diwujudkan dengan memonitoring dan setting level air melalui smartphone.

Kata-kunci: Internet of Thing (IoT), NodeMCU, FireBase, Pompa Air, DS18B20,

HC-SR04, Step down converter, Flutter

KATA PENGANTAR

Alhamdulillah, segala puji dan syukur penulis panjatkan ke hadirat Allah SWT yang selalu mem-berikan rahmat dan karunia-Nya sehingga penulis dapat me-nyelesaikan tugas akhir ini dengan baik, dalam rangka me-menuhi salah satu syarat untuk men-capai gelar Sarjana Teknik dari Program Studi Srata 1, Jurusan Teknik Elektro, Fakultas Teknik ITNY Yogyakarta.

Penulis menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, sangatlah sulit bagi penulis untuk me-nyelesaikan tugas akhir ini. Oleh karena itu dengan segenap ketulusan hati, penulis meng-ucapkan terimakasih kepada:

1. Bapak Ir. H. Ircham, MT. selaku Ketua ITNY
2. Ibu Ir. Hj. Oni Yuliani, M.Kom selaku Ketua Program Studi Srata 1 Teknik Elektro ITNY Yogyakarta.
3. Bapak Joko Prasajo, ST., MT. dan Tugino, ST., MT. selaku dosen pembimbing skripsi yang telah mem-berikan waktu, tenaga dan pikiran untuk meng-arahkan penulis dalam pe-nyusunan tugas akhir ini.
4. Orang tua dan keluarga yang selalu mem-beri dukungan, doa, dan motivasi.
5. Teman-teman seperjuangan dalam menyelesaikan tugas akhir yang selalu memberi bantuan, dukungan, doa dan motivasi.

Akhir kata, semoga tugas akhir ini mem-bawa manfaat bagi pengembangan ilmu pengetahuan dan teknologi Indonesia.

Yogyakarta, 02 Agustus 2020

Penulis,

Anung

DAFTAR ISI

HALAMAN PERSETUJUAN.....	ii
HALAMAN PENGESAHAN.....	iii
SURAT PERNYATAAN.....	iv
HALAMAN MOTTO DAN PERSEMBAHAN.....	v
ABSTRAK	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xii
BAB I PENDAHULUAN	13
1.1 Latar Belakang	13
1.1.1 Perumusan masalah	14
1.1.2 Batasan masalah.....	14
1.1.3 Keaslian penelitian.....	15
1.1.4 Faedah yang diharapkan	15
1.2 Tujuan Penelitian.....	16
BAB II LANDASAN TEORI	17
2.1 Tinjauan Pustaka	17
2.2 Landasan Teori	18
2.2.1 <i>NodeMCU</i>	18
2.2.2 <i>Step down converter</i> lm2596.....	19
2.2.3 Optocoupler	20
2.2.4 Sensor <i>flow</i> meter	22
2.2.5 Sensor <i>hall</i> efek	23
2.2.5 Sensor ultrasonik HC-SR04.....	25
2.2.6 Pompa air	28
2.2.7 Sensor suhu DS18B20	29
2.2.8 Platformio	30
2.2.9 Flutter.....	30
2.2.10 Dart	32
2.2.11 Firebase.....	33
2.2.12 Visual studio code.....	34
2.3 Hipotesis	35
2.4 Rencana Penelitian	36
2.4.1 Perancangan sistem.....	36
2.4.2 Perancangan perangkat keras sistem	37
2.4.3 Perancangan perangkat lunak sistem	37
2.4.4 Proses assembly	37
2.4.5 Pengujian sistem	37
BAB III CARA PENELITIAN	38
3.1 Bahan Penelitian.....	38
3.2 Alat Penelitian	39
3.3 Jalan Penelitian.....	40
3.3.1 Perancangan sistem.....	40

3.3.2 Perancangan perangkat-keras	42
3.3.2.1 Perancangan <i>step down</i> LM2596	42
3.3.2.2 Perancangan <i>driver</i> motor	43
3.3.2.3 Perancangan sensor suhu.....	45
3.3.2.4 Perancangan sensor ultrasonic	46
3.3.2.5 Perancangan sensor <i>flow</i>	48
3.3.3 Perancangan perangkat lunak	49
3.3.3.1 Perancangan perangkat lunak <i>NodeMCU</i> kendali pompa air	50
3.3.3.2 Perancangan perangkat lunak <i>NodeMCU</i> pada tangki air.....	53
3.3.3.3 Perancangan perangkat lunak untuk <i>smartphone</i>	54
3.3.4 Pengujian sistem	55
3.3.4.1 Pengujian perangkat keras sistem	55
3.3.4.2 Pengujian perangkat-lunak sistem.....	56
3.3.4.3 Pengujian sistem keseluruhan	57
3.4 Kesulitan – kesulitan	57
BAB IV HASIL PENELITIAN DAN PEMBAHASAN	58
4.1 Hasil Penelitian.....	58
4.2 Hasil Pengujian Perangkat Keras	59
4.2.1 Hasil pengujian <i>Step down Converter</i> lm2596	59
4.2.2 Pengujian <i>driver</i> motor	60
4.2.3 Pengujian sensor suhu	61
4.2.4 Pengujian sensor ultrasonic HC-SR04.	63
4.2.5 Pengujian sensor <i>flow</i>	66
4.3 Hasil Pengujian Perangkat Lunak	70
4.3.1 Hasil pengujian lunak <i>nodemcu</i> kendali pompa air	70
4.3.2 Pengujian perangkat lunak <i>NodeMCU</i> pada tangki air.....	72
4.3.3 Pengujian perangkat lunak untuk <i>smartphone Android</i>	73
BAB V SIMPULAN DAN SARAN	76
5.1 Simpulan.....	76
5.2 Saran.....	76
DAFTAR PUSTAKA	77
LAMPIRAN.....	78

DAFTAR GAMBAR

Gambar 2.1 <i>NodeMCU</i>	19
Gambar 2.2 <i>Step down Converter</i> LM2596	19
Gambar 2.3 Rangkaian <i>step down</i> converter.....	20
Gambar 2.4 Simbol dan bentuk optocoupler.....	21
Gambar 2.5 <i>Internal</i> optocoupler	21
Gambar 2.6 Sensor <i>flow</i>	22
Gambar 2.7 Bentuk sensor efek <i>hall</i>	24
Gambar 2.8 Sensor <i>ultrasonic</i> HC-SR04	26
Gambar 2.9 Pompa air.....	29
Gambar 2.10 Sensor suhu DS18B20.....	29
Gambar 2.11 Tampilan logo flutter.....	31
Gambar 2.12 Tampilan logo dart	33
Gambar 2.13 Tampilan Logo firebase	33
Gambar 2.14 Tampilan visual studio code.....	35
Gambar 2.15 <i>Flowchart</i> proses perancangan sistem	36
Gambar 3.1 Diagram blok perancangan sistem	40
Gambar 3.2 Modul <i>step down</i> lm2596.....	43
Gambar 3.3 Perancangan rangkaian <i>Driver</i> motor DC.....	43
Gambar 3.4 Perancangan rangkaian sensor suhu	46
Gambar 3.5 Perancangan rangkaian rangkaian sensor <i>ultrasonic</i>	47
Gambar 3.6 Rangkaian pembagi tegangan.....	47
Gambar 3.7 Perancangan rangkaian sensor <i>flow</i>	49
Gambar 3.8 <i>Flowchart</i> perangkat lunak <i>NodeMCU</i> untuk pengendali pompa air	50
Gambar 3.9 <i>Flowchart</i> perangkat lunak <i>NodeMCU</i> untuk <i>notifikasi</i> alarm	52
Gambar 3.10 <i>Flowchart</i> perangkat lunak <i>NodeMCU</i> untuk tangki air.....	53
Gambar 3.11 <i>Flowchart</i> perangkat lunak untuk <i>smartphone</i>	55
Gambar 4.1 Hasil perancangan perangkat-keras.....	58
Gambar 4.2 Pengujian rangkaian <i>step down converter</i>	59
Gambar 4.3 Pengujian saat motor berputar.....	60
Gambar 4.4 Proses pengujian sensor suhu DS18B20	61
Gambar 4.5 Hasil pengujian sensor suhu DS18B20	63
Gambar 4.6 Proses pengujian sensor ultrasonic HC-SR04.....	64
Gambar 4.7 Skematik pengujian sensor ultrasonic HC-SR04	64
Gambar 4.8 Hasil pengujian sensor HC-SR04.....	65
Gambar 4.9 Proses pengujian sensor <i>flow</i>	66
Gambar 4.10 Rangkaian pengujian sensor <i>flow</i>	67
Gambar 4.11 Output pengujian sensor <i>flow</i>	68
Gambar 4.12 Bentuk gelombang output sensor <i>flow</i>	69
Gambar 4.13 Merubah database <i>key</i> motor.....	70
Gambar 4.14 Database firebase dengan <i>key flow</i> saat ada aliran air.....	71
Gambar 4.15 Struktur data alarm pada database firebase	72
Gambar 4.16 Perubahan data pada database dengan <i>key</i> tangki	72
Gambar 4.17 Proses pencatatan data level air.....	73
Gambar 4.18 Tampilan <i>setting value</i> level air	74

Gambar 4.19 Tampilan monitoring level air.....	75
Gambar 4.20 Tampilan saat ada <i>notifikasi alarm</i>	75

DAFTAR TABEL

Tabel 3.1 Bahan penelitian.....	38
Tabel 3.1 lanjutan Bahan penelitian.....	39
Tabel 3.2 Alat penelitian	39
Tabel 3.2 Lanjutan Alat penelitian.....	40
Tabel 3.3 Kondisi lampu	51
Tabel 4.1 Pengujian driver motor	60
Tabel 4.2 Pengujian sensor DS18B20.....	63
Tabel 4.3 Pengujian sensor <i>ultrasonic</i>	66
Tabel 4.4 Pengujian sensor <i>flow</i>	69

BAB I

PENDAHULUAN

1.1 Latar Belakang

Air merupakan bagian kebutuhan pokok manusia yang banyak digunakan untuk me-menuhi aktivitas sehari-hari seperti minum, mandi, mencuci dan lain sebagainya. Di perkotaan, pelayanan jasa air bersih umumnya diselenggarakan oleh pemerintah melalui PDAM (Perusahaan Daerah Air Minum). Air yang disalurkan oleh PDAM ke rumah-rumah penduduk biasanya berasal dari pengunungan yang mengalir ke sungai kemudian di tampung terlebih dahulu di bak-bak pe-nampungan (*reservoir*) kemudian di saring dan di distribusikan ke rumah-rumah pelanggan

Proses pengiriman air dari sungai ke bak pe-nampungan kebanyakan menempuh jarak yang jauh. Kondisi saat ini peng-oprasian pompa PDAM masih ada yang menggunakan sistem konvensional yaitu sensor ketinggian air di bak-bak pe-nampungan di hubungkan dengan meng-gunakan kabel yang panjang untuk mengendalikan pompa yang ada di sungai. Sebagian yang lain ada yang meng-gunakan dua operator. Operator bertugas me-mantau ketinggian air di bak pe-nampungan lalu me-ngirim status ketinggian air kepada operator yang ada di sungai untuk meng-hidupkan dan me-matikan pompa.

Internet of Thing (IoT) merupakan konsep yang bertujuan untuk me-manfaatkan konektifitas jaringan internet dimana suatu objek tertentu mampu me-transfer data terus me-nerus lewat jaringan tanpa me-merlu-kan adanya interaksi antar manusia maupun manusia dan perangkat komputer. Metode yang digunakan dalam *IoT* adalah nirkabel tanpa mengenal jarak. Cara kerja dari *IoT* yaitu me-

manfaatkan sebuah pemrograman dimana setiap perintah dari suatu argumen menghasilkan sebuah interaksi dan komunikasi sesama mesin yang terhubung otomatis.

Penerapan teknologi ini dapat digunakan untuk me-nyusun laporan tugas akhir dengan judul “APLIKASI *INTERNET OF THINGS (IoT)* PADA SISTEM KENDALI POMPA AIR”. *Prototype* ini dapat digunakan sebagai bahan pertimbangan untuk mem-buat kendali pompa air meng-gunakan konsep *Internet of Thing-(IoT)*-pada-kehidupan-nyata.

1.1.1 Perumusan masalah

Rumusan penelitian ini dinyatakan dengan tiga buah pertanyaan sebagai berikut:

1. Bagaimana penerapan konsep *IoT* pada pompa air.
2. Bagaimana sistem mekanis kendali pompa air dibuat dan dapat dikontrol oleh NodeMCU ESP8266.
3. Bagaimana NodeMCU ESP8266 me-ngendalikan sistem agar sistem dapat bekerja dengan baik untuk me-ngendali-kan pompa air.

1.1.2 Batasan masalah

Agar penelitian lebih fokus dan tidak meluas dari pembahasan yang dimaksud, dalam tugas akhir ini penulis mem-batasi-nya pada ruang lingkup penelitian sebagai berikut:

1. Alat pompa air yang dirancang merupakan *prototype*,
2. Menggunakan *mobile app* untuk media *user interface*,
3. Menggunakan modul NodeMCU ESP8266 untuk komunikasi antara pompa air dengan manusia melalui internet.

1.1.3 Keaslian penelitian

Penelitian mengenai penerapan konsep *IoT* untuk me-ngendali-kan pompa air melalui internet meng-guna-kan NodeMCU ESP8266 sebagai kontrol pompa air sudah pernah dilakukan. Berikut uraian singkat gagasan sebelumnya yang pernah di-lakukan dan berkaitan dengan masalah yang akan dibahas.

Menurut (Puji Suwanto, F. Trias Pontia, Bomo Wibowo Sanjaya pada jurnal penelitiannya yang berjudul “Kendali dan Monitoring Pompa Pendingin pada PLTD Siantan Berbasis *IoT (Internet of Things)*”, 2019) Aplikasi ini digunakan untuk melakukan kendali *on/off* pompa dermaga dan monitoring jarak jauh bak penampung.

Kemudian (Ulumuddin, M.Sudrajat, Rachmildha, N.Ismail, E.A.Z.Hamidi pada jurnal penelitiannya yang berjudul “Prototipe Sistem Monitoring Air Pada Tangki Berbasis *IoT (Internet of Things)* NodeMCU Esp8266 Dan Sensor Ultrasonik”, 2017). Yaitu alat yang akan mendeteksi tinggi air dalam penampungan dengan menggunakan sensor ultrasonik dan modul *WiFi* ESP8266 sebagai *transmitter*.

Pada penelitian ini penulis membuat penerapan konsep *IoT* pada pengendali pompa air menggunakan NodeMCU ESP8266. Kemudian alat pompa air ini menggunakan 2 buah sensor ultrasonik sebagai sensor ketinggian air, 1 buah pompa air mini, dan juga sensor *flow*.

1.1.4 Faedah yang diharapkan

Faedah yang diharapkan dari perancangan skripsi ini adalah sebagai berikut:

1. Dapat dijadikan sebagai penerapan pengembangan pompa air yang dapat di kendalikan melalui jaringan internet.
2. Dapat dijadikan sebagai sarana pembelajaran bagi mahasiswa program studi teknik elektro tentang *IoT* dan NodeMCU ESP8266.

1.2 Tujuan Penelitian

Penelitian ini bertujuan untuk membuat pompa air yang dapat di kendalikan melalui jaringan internet, sehingga penelitian ini dapat menjadi solusi untuk pengendalian pompa air dan *me-monitor*-ing kinerja pompa air dari jarak jauh.

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Perkembangan teknologi semakin meningkat salah satunya kendali jarak jauh. *IoT* diperuntukkan untuk monitoring dan kendali pompa air. Ada beberapa orang yang membuat alat pompa air diantaranya yang dibuat oleh :

Puji Suwanto, F. Trias Pontia, Bomo Wibowo Sanjaya (2019) dengan judul “Kendali dan Monitoring Pompa Pendingin pada PLTD Siantan Berbasis *IoT* (Internet Of Things)”. Aplikasi ini digunakan untuk melakukan kendali *on/off* Pompa Dermaga dan monitoring jarak jauh bak penampung. Alat ini menggunakan Modul Arduino Uno dan Nano sebagai pengembangan dari mikrokontroler atmel serta modul NodeMCU ESP8266 sebagai media penghubung menuju internet. Variabel yang digunakan sebagai alat monitoring adalah variabel arus sedangkan SSR (Solid State Relay) sebagai aktuatornya dengan plantnya pompa dermaga dua. Hasil diperoleh dari penelitian ini menunjukkan bahwa kendali *on/off* jarak jauh dengan perintah digital 1 adalah *on* dan perintah digital 0 adalah *off*, melalui server Cayenne. Proses pada server Cayenne memerlukan waktu 3-6 detik untuk menghidupkan atau mematikan pompa, dengan kecepatan pengiriman data dari sensor 14-16 detik. Pengukuran ini dilakukan pada CT 150/5 yang terdapat dalam panel box, adapun proteksi tambahan pada pompa adalah pemutusan arus. Pemutusan dilakukan saat kondisi arus CT terbaca lebih dari 5 A dan saat kurang dari 1,5 A. Dengan adanya alat kendali dan pemantauan jarak jauh ini dapat

membantu para pekerja dalam mengefisienkan waktu kerja dan memudahkan dalam pemantauan

Kemudian Ulumuddin, M.Sudrajat, Rachmildha, N.Ismail, E.A.Z.Hamidi pada tahun (2017) dengan judul penelitian “Prototipe Sistem Monitoring Air Pada Tangki Berbasis *Internet of Things* Menggunakan NodeMCU Esp8266 Dan Sensor Ultrasonik”. Alat yang digunakan berupa modul WiFi ESP8266 sebagai transmitter yang dipadukan dengan sensor ultrasonik. Sensor ultrasonik ini akan mendeteksi tinggi air dalam penampungan dengan memanfaatkan gelombang ultrasonik. Sistem ini diharapkan dapat memberikan kontribusi dalam menghemat air bersih dan pemantauan secara efektif dan efisien. Sistem diuji pada prototipe ground tank dengan ketinggian 300 cm. Hasil uji menunjukan respon sistem yang baik dan akurat sesuai posisi sensor.

2.2 Landasan Teori

Pada landasan teori ini akan dijelaskan beberapa sistem dan komponen Hardware maupun Software yang mendukung terbentuknya Aplikasi *Internet of Things (IoT)* Pada Sistem Kendali Pompa Air.

2.2.1 NodeMCU

NodeMCU merupakan sebuah *open source* platform *IoT* dan pengembangan kit untuk membantu dalam membuat prototype produk *IoT*. NodeMCU sudah dapat diprogram untuk banyak bahasa pemrograman seperti javascript, python, lua, C dan C++. Pengembangan kit ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (Pulse Width Modulation), IIC, 1-Wire dan ADC (*Analog to Digital Converter*) semua dalam satu board.

NodeMCU berukuran panjang 4.83cm, lebar 2.54cm, dan berat 7 gram. Board ini sudah dilengkapi dengan fitur *WiFi* dan *Firmwarena* yang bersifat *opensource*. Bentuk fisik NodeMCU ditunjukkan Gambar 2.1.



Gambar 2.1 NodeMCU

2.2.2 Step down converter lm2596

Modul *Step down* lm2596 adalah modul yang menggunakan IC LM2596 sebagai komponen utamanya. IC LM2596 adalah sirkuit terpadu / integrated circuit yang berfungsi sebagai *Step down converter* dengan arus output sebesar 3Amper. Bentuk fisik *Step down Converter* ditunjukkan Gambar 2.2.

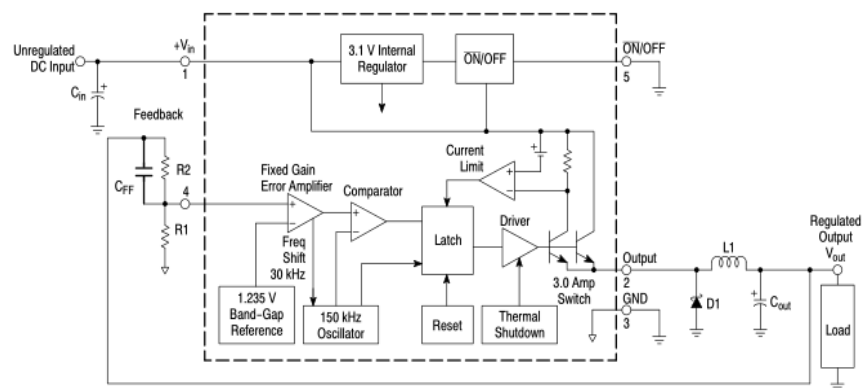


Gambar 2.2 Step down Converter LM2596

Step down converter adalah konverter penurun tegangan yang menerapkan sistem *SMPS* (*Switching Mode Power Supply*). *Step down* converter adalah konverter dengan efisiensi yang lebih tinggi jika dibandingkan dengan

power-supply penurun tegangan linier seperti IC LM7805. Efisiensinya dapat mencapai lebih dari 90%.

Didalam sebuah rangkaian *Step down converter* selalu terdapat pengendali dan pembangkit sinyal *PWM (Pulse Width Modulation)*, transistor penguat, dioda, kondensator dan induktor. Rangkaian *Step down converter LM2596* dapat dilihat pada Gambar 2.3 berikut :

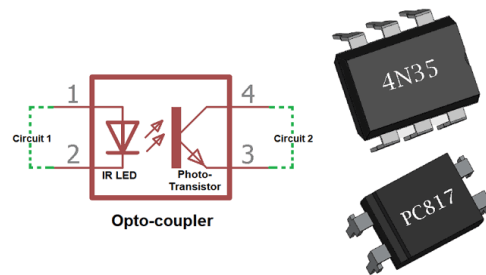


Gambar 2.3 Rangkaian *step down converter*

2.2.3 Optocoupler

Optocoupler juga dikenal dengan sebutan opto-isolator, photocoupler atau optical isolator. Optocoupler adalah komponen elektronika yang berfungsi sebagai penghubung berdasarkan cahaya *optik*. Pada dasarnya optocoupler terdiri dari 2 bagian utama yaitu *transmitter* yang berfungsi sebagai pengirim cahaya *optik* dan *receiver* yang berfungsi sebagai pendeteksi sumber cahaya.

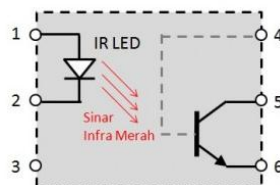
Masing-masing bagian optocoupler (*transmitter* dan *receiver*) tidak memiliki hubungan konduktif rangkaian secara langsung tetapi dibuat sedemikian rupa dalam satu kemasan komponen. Gambar 2.4 adalah contoh simbol optocoupler dan bentuk:



Gambar 2.4 Simbol dan bentuk optocoupler

Jenis-jenis optocoupler yang sering ditemukan adalah optocoupler yang terbuat dari bahan semikonduktor dan terdiri dari kombinasi LED (*Light Emitting Diode*) dan Phototransistor. Dalam Kombinasi ini, LED berfungsi sebagai pengirim sinyal cahaya *optik* (*Transmitter*) sedangkan phototransistor berfungsi sebagai penerima cahaya tersebut (*Receiver*). Jenis-jenis lain dari Optocoupler diantaranya adalah kombinasi LED-Photodiode, LED-LASCR dan juga Lamp-Photoresistor.

Pada prinsipnya, optocoupler dengan kombinasi LED dan phototransistor adalah optocoupler yang terdiri dari sebuah komponen LED yang memancarkan cahaya infra merah dan sebuah komponen semikonduktor yang peka terhadap cahaya (Phototransistor) sebagai bagian yang digunakan untuk mendeteksi cahaya infra merah yang dipancarkan oleh LED. Untuk lebih jelas mengenai prinsip kerja optocoupler lihat Gambar 2.5 rangkaian internal komponen optocoupler dibawah ini



Gambar 2.5 Internal optocoupler

Dari Gambar diatas dapat dijelaskan bahwa arus listrik yang mengalir melalui LED akan menyebabkan LED memancarkan sinyal cahaya infra merahnya. Intensitas cahaya tergantung pada jumlah arus listrik yang mengalir pada LED tersebut. Kelebihan cahaya infra merah adalah pada ketahanannya yang lebih baik jika dibandingkan dengan cahaya yang tampak. Cahaya infra merah tidak dapat dilihat dengan mata telanjang.

Cahaya Infra Merah yang dipancarkan tersebut akan dideteksi oleh phototransistor dan menyebabkan terjadinya hubungan atau Switch *ON* pada Phototransistor. Prinsip kerja phototransistor hampir sama dengan Transistor Bipolar biasa, yang membedakan adalah Terminal Basis (Base) phototransistor merupakan penerima yang peka terhadap cahaya.

2.2.4 Sensor *flow* meter

Flow meter merupakan alat ukur yang digunakan untuk mengukur laju aliran atau jumlah suatu *fluida* yang bergerak mengalir dalam suatu pipa tertutup. Jenis *fluida* yang melalui atau diukur oleh *flow* meter bisa berupa cairan, gas. Dalam aplikasinya penggunaan *flow* meter untuk mengukur aliran baik berupa kecepatan aliran, maupun jumlah aliran. Pada Gambar 2.6 adalah bentuk dari sensor *flow* meter.



Gambar 2.6 Sensor *flow*

Sensor *flow* meter terdiri dari beberapa bagian yaitu katup plastik, turbin, dan sensor *hall* efek. Cara kerja sensor ini adalah berdasarkan aliran. Ketika ada aliran fluida yang mengalir melalui sensor *flow* meter, aliran ini akan menggerakkan turbin yang ada pada sensor. Pada bagian turbin diletakkan sebuah magnet yang akan berputar ketika turbin berputar. Magnet ini nantinya akan memicu sensor hall efek. Ketika turbin berputar sensor hall efek akan mengeluarkan output berupa gelombang kotak dengan frekuensi sebanding dengan kecepatan putar turbin. Semakin cepat laju aliran semakin besar frekuensi yang akan dihasilkan oleh sensor *hall* efek. Output sensor ini yang nantinya akan dihubkan dengan mikrokontroler.

2.2.5 Sensor *hall* efek

Sensor *efek hall* atau dalam bahasa inggris disebut dengan *hall effect* sensor adalah komponen jenis transduser yang dapat mengubah informasi magnetik menjadi sinyal listrik untuk pemrosesan rangkaian elektronik selanjutnya. Sensor *efek hall* ini sering digunakan sebagai sensor untuk mendeteksi kedekatan (*proximity*), mendeteksi posisi (*positioning*), mendeteksi kecepatan (*speed*), mendeteksi pergerakan arah (*directional*) dan mendeteksi arus listrik (*current sensing*).

Sensor *efek hall* ini merupakan perangkat atau komponen yang diaktifkan oleh medan magnet eksternal. Medan magnet memiliki dua karakteristik penting yaitu densitas flux (flux density) dan kutub (kutub selatan dan kutub utara). Sinyal masukan (input) dari sensor *efek hall* ini adalah densitas medan magnet disekitar sensor tersebut, apabila densitas medan magnet melebihi batas ambang yang ditentukan maka sensor akan mendeteksi dan menghasilkan tegangan keluaran

(output) yang disebut dengan tegangan *hall* (V_H). Pada Gambar 2.7 adalah bentuk dari sensor *efek hall*



Gambar 2.7 Bentuk sensor efek *hall*

Sensor Efek *Hall* pada dasarnya terdiri dari potongan tipis semikonduktor yang bertipe P dengan bentuk persegi panjang. Bahan semikonduktor yang digunakan biasanya adalah gallium arsenide (*GaAs*), indium antimonide (*InSb*), indium phosphide (*InP*) atau indium arsenide (*InAs*). Potongan tipis semikonduktor tersebut dilewati oleh arus listrik secara terus-menerus. Ketika didekatkan dengan medan magnet atau ditempatkan pada lokasi yang bermedan magnet, garis fluks magnetik akan menggunakan gaya pada semikonduktor tersebut untuk mengalihkan muatan pembawa ke kedua sisi pelat semikonduktor. Gerakan pembawa muatan ini merupakan hasil dari gaya magnet yang melewati semikonduktor tersebut.

Karena Elektron dan Holes bergerak masing-masing ke kedua sisi semikonduktor, maka akan timbul perbedaan potensial diantara kedua sisi tersebut. Pergerakan elektron yang melalui bahan semikonduktor ini dipengaruhi oleh adanya medan magnet eksternal pada sudut atau posisi yang benar. Bentuk yang terbaik agar mendapatkan sudut atau posisi yang tepat adalah menggunakan bentuk

persegi panjang yang pipih (Flat Rectangular) pada komponen Sensor *Hall Effect* ini.

Peristiwa berbelok atau beralihnya aliran listrik (elektron) dalam pelat konduktor karena pengaruh medan magnet ini disebut dengan efek *hall* (*hall effect*). Efek *hall* ini ditemukan oleh Dr. Edwin Hall pada tahun 1879. Untuk dapat menghasilkan perbedaan potensial diseluruh perangkat, garis fluks magnetik harus tegak lurus (90 derajat) terhadap aliran listrik dengan kutub yang benar. Nama “*hall*” ini diambil dari nama penemu efek ini yaitu Dr. Edwin *Hall*. Dasar dari prinsip kerja efek *hall* ini adalah gaya lorentz yaitu gaya yang ditimbulkan oleh muatan listrik yang bergerak dalam suatu medan magnet.

2.2.5 Sensor ultrasonik HC-SR04

Sensor Ultrasonik HC-SR04 dapat mengukur jarak antara 3 cm sampai 300 cm. Keluaran dari modul sensor ultrasonik HC-SR04 berupa pulsa yang lebarnya merepresentasikan jarak. Lebar pulsa yang dihasilkan modul sensor ultrasonik bervariasi dari 115 uS sampai 18,5 mS.

Modul sensor ultrasonik hanya akan mengirimkan suara ultrasonik ketika ada pulsa trigger dari mikrokontroler (Pulsa high selama 5μS). Suara ultrasonik dengan frekuensi sebesar 40KHz akan dipancarkan selama 200μS oleh modul sensor ultrasonik ini. Suara ini akan merambat di udara kemudian mengenai objek dan dipantulkan kembali ke modul sensor ultrasonik. Selama menunggu pantulan sinyal ultrasonik dari bagian trasmitter, modul sensor ultrasonik akan menghasilkan sebuah pulsa. Pulsa ini akan berhenti (*low*) ketika suara pantulan terdeteksi oleh modul sensor ultrasonik. Oleh karena itulah lebar pulsa tersebut dapat

merepresentasikan jarak antara modul sensor ultrasonik dengan objek. Bentuk sensor ultrasonik diperlihatkan pada Gambar 2.8 berikut.



Gambar 2.8 Sensor *ultrasonic* HC-SR04

Gelombang ultrasonik merupakan gelombang akustik yang memiliki frekuensi mulai 20 kHz hingga sekitar 20 MHz. Frekuensi kerja yang digunakan dalam gelombang ultrasonik bervariasi tergantung pada media yang dilalui, mulai dari gas, cair hingga padat. Jika gelombang ultrasonik berjalan melalui sebuah media, secara matematis besarnya jarak dapat dihitung menggunakan persamaan 2.1

$$s = \frac{v \cdot t}{2} \quad (2.1)$$

dengan:

- s = jarak dalam satuan meter
- v = kecepatan suara yaitu 344 m/detik
- t = waktu tempuh dalam satuan detik.

Ketika gelombang ultrasonik menabrak suatu penghalang maka sebagian gelombang tersebut akan dipantulkan sebagian diserap dan sebagian yang lain akan diteruskan. Sensor ultrasonik adalah sebuah sensor yang mengubah besaran fisis (bunyi) menjadi besaran listrik. Pada sensor ini gelombang ultrasonik dibangkitkan melalui sebuah benda yang disebut piezoelektrik. Piezoelektrik ini akan

menghasilkan gelombang ultrasonik dengan frekuensi 40 kHz ketika sebuah osilator diterapkan pada benda tersebut. Sensor ultrasonik secara umum digunakan untuk suatu sensor tak sentuh yang seperti pengukuran jarak. Alat ini secara umum memancarkan gelombang suara ultrasonik menuju suatu target yang memantulkan balik gelombang ke arah sensor. Kemudian sistem mengukur waktu yang diperlukan untuk pemancaran gelombang sampai kembali ke sensor dan menghitung jarak target dengan menggunakan kecepatan suara dalam medium. Rangkaian penyusun sensor ultrasonik ini terdiri dari *transmitter*, *receiver*, dan *komparator*. Selain itu, gelombang ultrasonik dibangkitkan oleh sebuah kristal tipis bersifat *piezoelektrik*. Bagian-bagian dari sensor ultrasonik adalah sebagai berikut:

1. Piezoelektrik

Peralatan piezoelektrik secara langsung mengubah energi listrik menjadi energi mekanik. Tegangan input yang digunakan menyebabkan bagian keramik meregang dan memancarkan gelombang ultrasonik. Tipe operasi transmisi elemen piezoelektrik sekitar frekuensi 32 kHz. Efisiensi lebih baik, jika frekuensi osilator diatur pada frekuensi resonansi piezoelektrik dengan sensitifitas dan efisiensi paling baik. Jika rangkaian pengukur beroperasi pada mode pulsa elemen piezoelektrik yang sama dapat digunakan sebagai transmitter dan receiver. Frekuensi yang ditimbulkan tergantung pada osilatornya yang disesuaikan frekuensi kerja dari masing-masing transduser. Karena kelebihan ini maka transduser piezoelektrik lebih sesuai digunakan untuk sensor ultrasonik.

2. Transmitter

Transmitter adalah sebuah alat yang berfungsi sebagai pemancar gelombang ultrasonik dengan frekuensi sebesar 40 kHz yang dibangkitkan dari sebuah osilator. Untuk menghasilkan frekuensi 40 KHz, harus di buat sebuah rangkaian osilator dan keluaran dari osilator dilanjutkan menuju penguat sinyal. Besarnya frekuensi ditentukan oleh komponen kalang RLC / kristal tergantung dari disain osilator yang digunakan. Penguat sinyal akan memberikan sebuah sinyal listrik yang diumpankan ke piezoelektrik dan terjadi reaksi mekanik sehingga bergetar dan memancarkan gelombang yang sesuai dengan besar frekuensi pada osilator.

3. Receiver

Receiver terdiri dari transduser ultrasonik menggunakan bahan piezoelektrik, yang berfungsi sebagai penerima gelombang pantulan yang berasal dari transmitter yang dikenakan pada permukaan suatu benda atau gelombang langsung LOS (Line of Sight) dari transmitter. Oleh karena bahan piezoelektrik memiliki reaksi yang reversible, elemen keramik akan membangkitkan tegangan listrik pada saat gelombang datang dengan frekuensi yang resonan dan akan menggetarkan bahan piezoelektrik tersebut.

2.2.6 Pompa air

Pompa air merupakan motor DC yang terdapat baling-baling dan dikemas dalam bentuk tabung. Baling-baling yang terdapat pada pompa air digunakan untuk menyedot air dan mengeluarkan melalui jalur yang disediakan. Tegangan kerja

pompa air ini 6V–12V, memiliki aliran maksimum 2 liter/menit, dan daya dorong maksimum 1 meter. Bentuk fisik pompa air ditunjukkan pada Gambar 2.9.



Gambar 2.9 Pompa air

2.2.7 Sensor suhu DS18B20

Sensor DS18B20 merupakan sensor suhu yang memiliki perlakuan khusus dalam penggunaannya. Pengoperasian sensor jenis ini berkomunikasi melalui bus 1-Wire yang menurut definisi hanya membutuhkan satu baris data untuk komunikasi dengan mikroprosesor pusat. Dengan kata lain, sensor ini menggunakan sistem 1 wire adalah standar komunikasi serial dua arah menggunakan satu saluran yang didesain khusus untuk mengirim maupun menerima data. Berikut merupakan tampilan fisik, dari sensor suhu DS18B20 ditunjukkan pada Gambar 2.10.



Gambar 2.10 Sensor suhu DS18B20

Keunggulan dari sensor suhu DS18B20 adalah selain memiliki komunikasi unik yaitu sistem 1 wire, sensor ini juga memiliki jangkauan pengukuran suhu mulai dari -55°C sampai $+125^{\circ}\text{C}$ (-67°F sampai $+257^{\circ}\text{F}$) serta memiliki tingkat akurasi sebesar $\pm 0.5^{\circ}\text{C}$ pada rentan suhu -10°C sampai $+85^{\circ}\text{C}$.

2.2.8 Platformio

Tim Arduino yang bekerja dibalik Arduino IDE telah bekerja cukup baik menjadikannya tool yang bersifat *free* dan *open source* yang lebih memudahkan dalam pemrograman, khususnya untuk *framework* Arduino. Meskipun sekarang, kini pemrograman ESP8266, NodeMCU juga STM32 dapat dilakukan dengan Arduino IDE.

Ketika membuat dan mengupload program mikrokontroler menggunakan Arduino IDE atau platform embedded yang lain terdapat beberapa kerumitan seperti memastikan semua pustaka telah terinstall dan versi yang tepat. Hal ini membutuhkan beberapa trik khusus dan cukup menguras waktu.

Platformio IDE adalah lingkungan pengembangan terintegrasi generasi berikutnya untuk *IoT*. Platformio IDE ini menyediakan ekstensi / plugin resmi untuk IDE populer seperti atom dan vscode platformio ide dibangun di atas platformio core

2.2.9 Flutter

Flutter adalah sebuah software development kit (SDK) buatan Google yang berfungsi untuk membuat aplikasi mobile phone menggunakan Bahasa pemrograman Dart, baik untuk android maupun IOS. Dengan flutter, aplikasi android dan IOS dapat dibuat menggunakan basis kode dan Bahasa pemrograman

yang sama, yaitu dart, Bahasa pemrograman yang juga diproduksi oleh google pada tahun 2011.

Sebelumnya, aplikasi murni (native) untuk android perlu dibuat menggunakan Bahasa Java atau Kotlin, sedangkan aplikasi IOS perlu dibuat menggunakan Bahasa pemrograman Objective-C atau swift. Flutter ditujukan untuk mempermudah dan mempercepat proses pengembangan aplikasi mobile yang dapat berjalan di atas android dan IOS, tanpa harus mempelajari dua Bahasa pemrograman secara terpisah.

Flutter dapat dikatakan sebagai produk google yang masih relatif baru. Rilis perdana flutter, versi Alpha (v.0.0.6), dipublikasikan pada bulan Mei 2017. Dan versi v.1.0 merupakan yang stabil saat diterbitkan oleh google (Raharjo.2019). Berikut merupakan tampilan logo Flutter ditunjukkan pada Gambar 2.11.



Gambar 2.11 Tampilan logo flutter

Salah satu keunggulan flutter dari pada teknologi multiplatform lainnya seperti react native adalah dalam hal kinerja. Flutter menjanjikan aplikasi yang dibuat akan mendapatkan tingkat sebesar 60 frame per second. Kinerja ini bisa didapatkan karena cara kerja dari flutter sedikit berbeda. Kode-kode yang ditulis dengan menggunakan bahasa dart akan diubah menjadi kode C/C++ kemudian dikompilasi secara native. Hal inilah yang menyebabkan flutter memiliki performa

yang hampir setara dengan aplikasi native. Flutter bisa berjalan pada sistem operasi android mulai dari versi 4.1 dan IOS mulai dari versi 8. Serta dijalankan di perangkat asli maupun simulator.

2.2.10 Dart

Bahasa pemrograman Dart merupakan bahasa pemrograman general-purpose yang dirancang oleh Lars Bak dan Kasper Lund. Bahasa pemrograman ini dikembangkan sebagai bahasa pemrograman aplikasi yang dapat dengan mudah untuk dipelajari dan disebar. Bahasa pemrograman buatan Google ini dapat digunakan untuk mengembangkan berbagai macam platform termasuk di dalamnya adalah web, aplikasi mobile, server, dan perangkat yang mengusung teknologi Internet of Things.

Bahasa pemrograman tersebut dapat digunakan untuk mengembangkan aplikasi untuk dijalankan pada berbagai macam *platform*. Dart juga dapat digunakan untuk mengembangkan aplikasi dari codebase tunggal menjadi aplikasi Android maupun iOS. Bahasa pemrograman Dart dapat digunakan secara bebas oleh para developer, karena bahasa ini dirilis secara open-source oleh Google di bawah lisensi BSD. Bahasa pemrograman Dart merupakan bahasa pemrograman berbasis class dan berorientasi terhadap objek.

Bahasa ini dikenalkan oleh Google sebagai pengganti bahasa pemrograman JavaScript, akan tetapi secara opsional bahasa ini dapat dikompilasi ke dalam JavaScript dengan menggunakan dart2js compiler. Sedikit berbeda dengan bahasa pemrograman JavaScript yang bertipe statis, bahasa pemrograman Dart merupakan

bahasa pemrograman bertipe dinamis. Berikut merupakan tampilan logo Dart ditunjukkan pada Gambar 2.12.



Gambar 2.12 Tampilan logo dart

2.2.11 Firebase

Firebase merupakan salah satu layanan dari Google yang memudahkan para developer dalam mengembangkan aplikasi. Firebase termasuk ke dalam kategori Baas (*Backend as a Service*). Ketika menggunakan layanan Firebase, nantinya pengembang bisa fokus *mendevlop* aplikasinya saja tanpa harus terlalu rumit berurusan dengan masalah *backend*. Berikut merupakan tampilan logo Firebase ditunjukkan pada Gambar 2.13.



Gambar 2.13 Tampilan Logo firebase

Firebase didirikan Pada tahun 2011 oleh Andrew Lee dan James Tamplin dengan nama perusahaan Envolve. Firebase pada awalnya (sebelum versi 3.0) dikenal dengan kemampuan *Database* yang real-time. Firebase *Database* sudah dilengkapi dengan kemampuan event handler, yang mana setiap perubahan yang

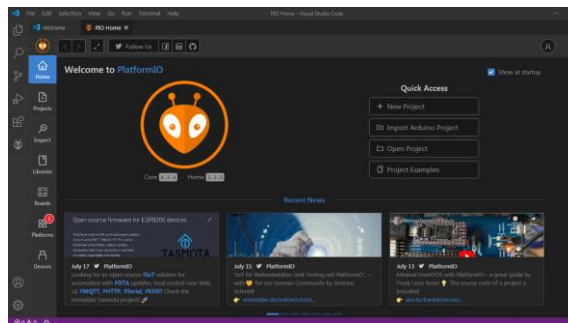
terjadi di *database* bisa dengan segera dideteksi dan ditangani, sehingga sangat cocok untuk penyimpanan data yang kolaboratif.

Firebase *Database* juga dilengkapi dengan fitur *offline* dan *online* support. Misalnya, ketika melakukan perubahan data dan di saat yang sama koneksi ke *cloud* Firebase tidak tersedia. Maka secara pintar, perubahan tersebut akan disimpan di lokal dulu, sampai dengan koneksi tersedia baru akan dilakukan perubahan di *server*. Semuanya dilakukan secara otomatis, sinkronisasi dilakukan oleh *library* Firebase.

Google menganggap layanan ini sangat bagus dan potensial sehingga pada tahun 2014 akhirnya mengakuisisi firebase. Google mengenalkan Firebase pertama kali pada 2016 tepatnya bulan mei di acara tahunan Google I/O. Layanan firebase terus dikembangkan oleh google dan saat ini cukup banyak digunakan untuk berbagai jenis kebutuhan pembuatan aplikasi android, iOS, bahkan web.

2.2.12 Visual studio code

Visual Studio Code Sering disingkat VS Code adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft dan dapat berjalan pada *operating system* Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemrograman JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via marketplace Visual Studio Code. Berikut merupakan Gambar 2.14 tampilan Visual Studio Code pada saat dibuka.



Gambar 2.14 Tampilan visual studio code

Seperti editor pada umumnya VSCode memiliki fitur *syntax coloring* dan *bracket matching*. Sampai saat ini bahasa pemrograman yang didukung adalah Batch, C++, Closure, Coffee Script, DockerFile, F#, Go, Jade, Java, HandleBars, Ini, Lua, Makefile, Markdown, Objective-C, Perl, PHP, PowerShell, Python, R, Razor, Ruby, SQL, Visual Basic, dan XML.

Banyak fitur-fitur yang disediakan oleh Visual Studio Code, diantaranya Intellisense, Git Integration, Debugging, dan fitur ekstensi yang menambah kemampuan teks editor. Fitur IntelliSense merupakan salah satu fitur yang membantu kita dalam menulis kode program melalui *popup* yang muncul secara otomatis pada saat kita sedang mengetik dan menampilkan saran sintaks yang bisa kita pilih. Fitur-fitur tersebut akan terus bertambah seiring dengan bertambahnya versi Visual Studio Code. Pembaruan versi Visual Studio Code ini juga dilakukan berkala setiap bulan. Teks editor visual studio code bersifat *open source*, di mana kode sumbernya dapat di lihat dan diubah secara bebas.

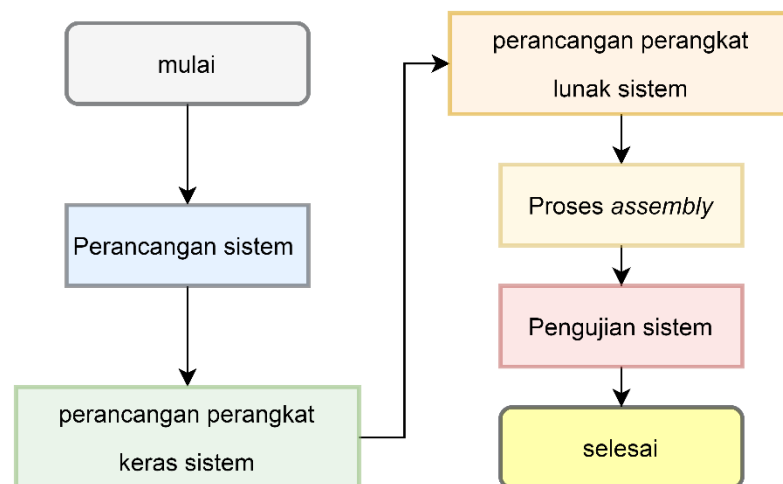
2.3 Hipotesis

Hipotesis pada penelitian ini adalah membuat prototip membuat penerapan konsep *IoT* pada pengendali pompa air. Serta dapat dijadikan sebuah referensi

untuk mengembangkan teknologi ini yang nantinya bisa diterapkan pada media lainnya untuk membantu dan mempermudah kegiatan dalam pekerjaan yang menggunakan sistem kendali jarak jauh melalui aplikasi pada *smartphone*.

2.4 Rencana Penelitian

Untuk mewujudkan apa yang terangkum dalam hipotesis perlu adanya rencana tersrtuktur. Tahapan awal proses yang dilakukan pada proses penelitian yang dilakukan adalah dengan mempersiapkan rencana penelitian. Pada tahapan ini dilakukan proses *listing* proses perancangan dan prioritas tahapan yang akan dilakukan seperti yang ditunjukkan pada diagram alir proses perancangan pada Gambar 2.15 dibawah ini. 4



Gambar 2.15 Flowchart proses perancangan sistem

Adapun rencana tahapan yang hendak dilakukan adalah sebagai berikut.

2.4.1 Perancangan sistem

Perancangan sistem adalah tahap awal dalam penelitian. Pada tahap perancangan sistem dibuat konsep dan kinerja alat untuk menentukan langkah dan

hal apa saja yang harus dilakukan dalam penelitian agar sistem yang dibuat dapat berjalan sebagaimana mestinya.

2.4.2 Perancangan perangkat keras sistem

Langkah kedua yaitu perancangan perangkat-keras sistem. Perancangan perangkat-keras dilakukan untuk merancang perangkat elektronis dan mekanik agar sesuai dengan perancangan sistem.

2.4.3 Perancangan perangkat lunak sistem

Tahap selanjutnya yaitu perancangan perangkat lunak sistem. Pada tahap ini dilakukan perancangan perangkat-lunak sebagai penunjang perangkat-keras sistem sehingga sistem dapat berjalan sesuai dengan sebagaimana mestinya dan selanjutnya dilakukan uji coba secara simulasi pada aplikasi perangkat-lunak sebelum diuji coba pada sistem yang telah dirancang.

2.4.4 Proses assembly

Proses *assembly* dilakukan untuk menyatukan perangkat-keras dan perangkat-lunak yang sudah dirancang pada proses sebelumnya, pada proses ini juga dilakukan *troubleshooting* pada sistem apabila terdapat kesalahan dalam perancangan yang sudah dilakukan sebelumnya sehingga dapat segera diperbaiki.

2.4.5 Pengujian sistem

Setelah semua proses dilakukan, tahapan terakhir adalah melakukan pengujian *running process* pada sistem baik secara auto maupun secara manual agar sistem yang dirancang dapat berjalan sesuai dengan tujuan perancangan.

BAB III

CARA PENELITIAN

Cara penelitian yang dilakukan untuk perancangan penerapan konsep *IoT* untuk mengendalikan pompa air menggunakan NodeMCU melalui beberapa tahapan proses yang dilakukan diantaranya adalah melakukan pengumpulan alat dan bahan, perancangan sistem baik sistem pada bagian perangkat keras maupun perangkat-lunak, perakitan (assembly), penyelesaian masalah (troubleshooting), dan pengujian sistem pada setiap bagian dan secara keseluruhan sistem sehingga sistem yang dirancang mampu berjalan optimal sesuai dengan perencanaan perancangan.

3.1 Bahan Penelitian

Bahan yang digunakan dalam perancangan sistem merupakan bahan yang berfungsi sebagai bahan elektronis dan mekanik. Pada Tabel 3.1 ditunjukkan bahan yang digunakan dalam penelitian sebagai komponen perancangan sistem.

Tabel 3.1 Bahan penelitian

No	Nama Bahan	Jumlah	Keterangan
1	NodeMCU	2	Unit Pengendali
2	Modul Srhc-04	2	Sensor Ultrasonic
3	<i>Flow</i> Meter	1	Sensor Kecepatan Air
4	Ds18b20	1	Sensor Suhu
5	Pompa Air Dc 12v	1	Pompa Air
6	Toples Plastik	2	Tempat Air
7	Mur Dan Baut	6	Baut Motor Pompa
8	Soket Dc	1	Input Catu Daya
9	Power Supply 12v/2a	1	Catu Daya
10	Terminal Block 2p	3	Terminal Kabel
11	<i>Step Down</i> 2576	1	Penurun Tegangan Dc
12	Spacer	8	Penyangga
13	Modem	1	Penghubung Internet
14	Transistor IRFZ44	1	<i>Driver</i> Pompa Air

Tabel 3.1 lanjutan Bahan penelitian

No	Nama Bahan	Jumlah	Keterangan
15	Saklar toggle	1	Simulasi Overload
16	Optocoupler Pc817	1	Driver Pompa Air
17	Acrylic	Secukupnya	Tempat Peletakkan Hardware
18	Selang	Secukupnya	Saluran Aliran Air
19	Kabel	Secukupnya	Pemhubung Komponen
20	Terminal Block 2p	1	Input 5v
21	Elco 100uf/16v	1	Filter
22	LED	3	Indikator Power Dan <i>WiFi</i>
23	Alumunium	1 Meter	Penyangga <i>Hardware</i>
24	Penggaris 30cm	2	Indikator Level Air
25	Resistor 10k	3	Pull Up Input
26	Resistor 330 Ohm	2	Pembagi Tegangan
27	Resistor 680 Ohm	2	Pembagi Tegangan
28	Resistor 220 Ohm	3	Pembatas Arus LED
29	Resistor 100 Ohm	1	Pembatas Arus Transistor
30	Dioda 1n4002	1	Pengaman Transistor

3.2 Alat Penelitian

Dalam perancangan sistem membutuhkan berbagai macam peralatan untuk membantu dan mendukung peneliti dalam melakukan perancangan sistem baik peralatan perangkat-lunak maupun perangkat-keras. Pada Tabel 3.2 ditunjukkan alat-alat yang digunakan dalam penelitian.

Tabel 3.2 Alat penelitian

No	Nama Bahan	Jumlah	Keterangan
Perangkat Lunak			
1	Eagle	1	Perancangan Pcb
2	Microsoft Office 2010	1	Pembuatan Laporan
3	Visual Studio Code	1	<i>Text Editor</i>
4	Flutter	1	Pemrograman Android
5	Platform Io	1	Pemrograman NodeMCU
Perangkat Keras			
1	Pc/Laptop	1	Media Perangkat Lunak
2	Multimeter	1	Pengukur Nilai Elektronis
3	Solder	1	Pemanas Timah
4	Timah	1	Perekat Komponen Elektronis

Tabel 3.2 Lanjutan Alat penelitian

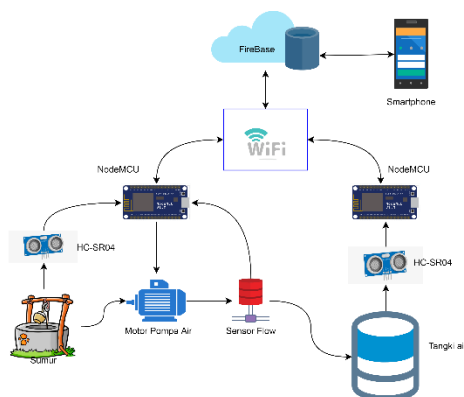
No	Nama Bahan	Jumlah	Keterangan
11	Osilloscope	1	Melihat Bentuk Sinyal
12	Gerindra	1	Pemotong
13	Mesin Amplas	1	Penghalus
14	Bolpoin	1	Menggambar Garis
15	Penggaris	1	Mengukur
16	Kabel Usb	1	Upload Program
5	Attractor	1	Penghilang Timah
6	Bor	1	Pelubang
7	Obeng	1	Pengencang Baut
8	Tang Potong	1	Pemotong Kabel
9	Tang Kupas	1	Pengupas Kabel
10	Tang Buaya	1	Penjepit

3.3 Jalan Penelitian

Proses yang dilakukan untuk mencapai keberhasilan dalam penelitian ini adalah meliputi perancangan sistem, pengujian sistem, analisis hasil, dan kemudian menarik kesimpulan dari hasil penelitian.

3.3.1 Perancangan sistem

Perancangan aplikasi *internet of things (IoT)* pada sistem kendali pompa air dimulai dengan membuat blok *diagram* sistem secara keseluruhan yang ditunjukkan pada Gambar 3.1 untuk mengetahui *input* dan *output* yang akan digunakan.



Gambar 3.1 Diagram blok perancangan sistem

Penjelasan Proses Alur Perancangan Sistem

1. Sistem kendali dihidupkan melalui aplikasi yang telah terinstall di smartphone berbasis operasi system android
2. Kemudian *database* merekap data perintah kendali tersebut
3. Lalu *NodeMCU* merespon dan menangkap sinyal untuk memberikan perintah ke *driver* motor untuk menggerakkan motor listrik
4. Motor listrik akan menyalurkan air dari sumur kedalam tangki penampungan
5. Kemudian sensor pada masing masing alat dan media (Sensor level air pada sumur dan tangki penampungan, sensor arus pada motor listrik, sensor *flow* penghubung anantara motor listrik ke tangki penampungan) akan memberikan data ke *NodeMCU* untuk disimpan pada database
6. *Database* merekap setiap data yang masuk dari *NodeMCU* dan menampilkan informasi ke aplikasi yang telah di install pada smartphone berbasis operasi system android tersebut.

Proses perancangan aplikasi *internet of things (IoT)* pada sistem kendali pompa air dimulai dari mengumpulkan bahan-bahan yang digunakan untuk membuat perangkat keras, kemudian dilanjutkan dengan merancang dan membuat perangkat keras. Setelah perancangan selesai kemudian dilanjutkan dengan mengecek tiap-tiap bagian apabila tidak sesuai rencana maka mengevaluasi dan memperbaiki proses perancangan perangkat keras jika proses perancangan perangkat selesai selanjutnya dimulai perancangan perangkat lunak. Perancangan perangkat lunak meliputi pembuatan aplikasi mobile dan program untuk menjalankan modul *NodeMCU*, jika perangkat lunak selesai selanjutnya

melakukan pengujian terhadap perangkat lunak jika proses pengujian tidak sesuai rencana maka proses dilanjutkan dengan memperbaiki perancangan perangkat lunak jika sudah sesuai rencana dilanjutkan dengan analisa dan kesimpulan.

3.3.2 Perancangan perangkat-keras

Perancangan perangkat keras dilakukan dengan urutan-urutan sebagai berikut :

1. Perancangan *Step Down* LM2596
2. Perancangan *Driver* Motor
3. Perancangan Sensor Suhu
4. Perancangan Sensor *Ultrasonic*
5. Perancangan Sensor *Flow*

3.3.2.1 Perancangan *step down* LM2596

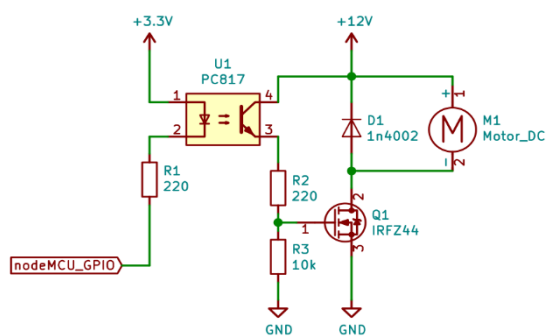
Step down converter lm2596 digunakan untuk menurunkan tegangan dari power supply 12 volt menjadi tegangan 5 volt, tegangan 5 volt ini digunakan untuk memberikan sumber tegangan NodeMCU dan sensor HC-SR04. Pemilihan *Step down* converter ini karena memiliki efisiensi yang lebih baik dibandingkan dengan penurun tegangan seperti ic lm7805. *Step down* converter ini menggunakan sistem Switching Mode Power Supply (SMPS) sehingga efisiensi dari *Step down* converter ini dapat mencapai lebih dari 90 persen. Pada Gambar 3.2 menunjukkan bentuk fisik modul *step down* lm2596.



Gambar 3.2 Modul *step down* lm2596

3.3.2.2 Perancangan *driver* motor

Driver motor yang digunakan berfungsi sebagai pemutus dan penyambung sumber arus listrik yang menuju pompa air yang dikendalikan oleh NodeMCU. Output dari port gpio NodeMCU menurut data sheet maksimal hanya boleh mengalirkan arus sebesar 12mA. Sedangkan arus yang digunakan untuk menghidupkan berkisar antara 200mA sampai 250mA sehingga agar NodeMCU dapat mengendalikan motor pompa air maka harus ditambahkan rangkaian *driver* motor. Rangkaian *driver* motor ini terdiri dari komponen resistor, dioda, optocoupler dan transistor FET. Rangkaian *driver* motor pompa air ditunjukkan Gambar 3.3



Gambar 3.3 Perancangan rangkaian *Driver* motor DC

Optocoupler digunakan untuk mengisolasi tegangan NodeMCU 3,3volt dari tegangan 12v pada motor dan transistor. Pada bagian input terdapat resistor yang di rangkai secara seri dengan LED optocoupler. Resistor ini digunakan untuk

membatasi arus yang lewat pada LED optocoupler dan GPIO NodeMCU. Tegangan kerja LED pada optocoupler adalah 1,2 volt dengan arus 10mA. Pemilihan arus 10 mA dikarenakan arus arus maksimal GPIO NodeMCU adalah 12mA dan arus LED optocoupler bekerja antara 1-20mA sehingga dipilih arus kerja optocoupler sebesar 10mA.

Untuk menghitung nilai resistor R1 yang digunakan, harus diketahui tegangan pada R1. Dibawah ini adalah perhitungan untuk menentukan nilai R1

$$VR1 = V_{cc} - V_d = 3,3\text{volt} - 1,2\text{volt} = 2,1\text{ volt} \quad (3.1)$$

Keterangan :

$VR1$ = tegangan pada resistor R1
 V_{cc} = tegangan sumber
 V_d = tegangan dioda pada optocoupler

$$R1 = \frac{VR1}{i} = \frac{2,1\text{ volt}}{10\text{ mA}} = 210\ \Omega \approx 220\ \Omega \quad (3.2)$$

Keterangan :

$R1$ = hambatan pada R1
 $VR1$ = tegangan pada resistor R1
 i = arus yang mengalir pada R1

R2 digunakan untuk membatasi arus listrik yang mengalir pada foto transistor optocoupler. arus maksimal yang di izinkan phototransistor adalah 50mA. sehingga nilai R2 adalah.

$$R2 = \frac{VR2}{i} = \frac{12\text{volt}}{50\text{mA}} = 240\ \Omega \approx 220\ \Omega \quad (3.3)$$

Keterangan :

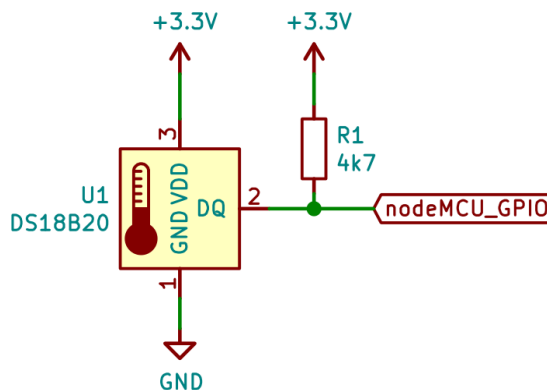
$VR2$ = tegangan pada resistor R2
 i = arus yang mengalir pada R2

R3 digunakan untuk pengosongan kapasitor CGS agar transistor mati saat tidak ada arus yang mengalir dari optocoupler. dioda D1 digunakan untuk pengaman transistor dari tegangan balik yang timbul saat motor dimatikan.

Perancangan ini menggunakan transistor jenis fet dengan type IRFZ44. Menurut data sheet transistor IRFZ44 mampu mengalirkan arus sampai 36 amper, sehingga transistor ini sudah cukup untuk *driver* motor dc dengan arus 200mA sampai 250mA. Transistor akan dihubungkan pada kaki negatif motor pompa air, dan kaki positif motor pompa air langsung dihubungkan ke sumber tegangan 12 volt. Ketika kaki gate transistor dipicu akan menyebabkan aktifnya transistor, sehingga negatif motor listrik akan terhubung dengan negatif sumber tegangan. Selanjutnya arus akan mengalir dari positif 12v melalui motor dc dan transistor menuju negatif sumber. Adanya arus yang mengalir melalui motor dc akan menyebabkan berputarnya motor pompa air yang akan mengalirkan air dari sumur ke tangki penampungan.

3.3.2.3 Perancangan sensor suhu

Sensor ds18b20 adalah sensor suhu dengan keluaran output digital. Pembacaan nilai suhu dilakukan melalui komunikasi serial 1 kabel. Sensor ini hanya membutuhkan komponen external berupa 1 buah resistor untuk pull up dengan nilai 4,7 k ohm. Sensor ini memiliki 3 kaki, dari 3 tersebut 2 diantaranya digunakan untuk sumber tegangan dan 1 kaki digunakan untuk komunikasi serial. Sensor ini diberi tegangan sebesar 3,3 volt agar se-level dengan tegangan NodeMCU sehingga tidak perlu lagi menambah konverter tegangan. Gambar perancangan sensor suhu terlihat pada Gambar 3.4 berikut

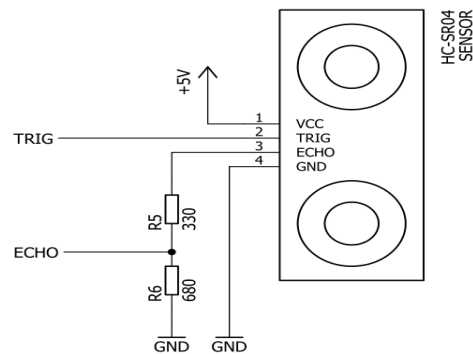


Gambar 3.4 Perancangan rangkaian sensor suhu

Sensor ini akan digunakan untuk mengukur suhu motor pompa air. Jika suhu motor pompa air lebih dari 100°C maka NodeMCU akan mematikan sistem dan mengirimkan *notifikasi* alarm overhear motor pada aplikasi android melalui *database* firebase. Hal ini dimaksudkan agar tidak terjadi kerusakan yang parah pada motor pompa air.

3.3.2.4 Perancangan sensor ultrasonic

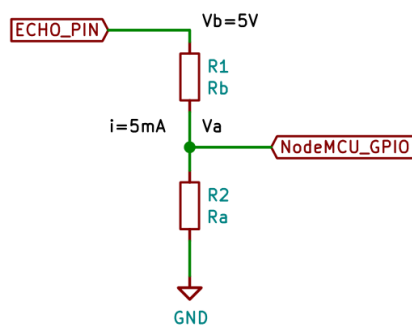
Perancangan sensor ultrasonic dilakukan dengan cara menghubungkan pin Trig dengan output pin NodeMCU. NodeMCU akan memberikan kondisi *LOW* menjadi High dan di tahan selama 2 uS (micro seconds) kemudian di rubah menjadi kondisi High dan ditahan selama 10uS. Setelah itu di kembalikan kondisi *Low*. Setelah itu pin echo pada sensor HC-SR04 akan mengeluarkan sinyal yang akan di baca oleh NodeMCU. Rangkaian sensor ultrasonic dapat di lihat pada Gambar berikut.



Gambar 3.5 Perancangan rangkaian rangkaian sensor *ultrasonic*

Output dari pin echo sensor HC-SR04 beranda pada level 0-5 volt. Sedangkan level yang diizinkan pada GPIO (*general purpose input output*) NodeMCU adalah 0-3,3 volt. GPIO NodeMCU tidak toleransi terhadap tegangan diatas 3,6 volt, jika tegangan GPIO melebihi 3,6 volt dapat menyebabkan kerusakan pada chip NodeMCU. Untuk menghindari hal tersebut maka output dari pin echo sensor HC-SR04 harus di beri pembagian tegangan dari level 0 - 5 volt menjadi level 0 - 3,3 volt.

Untuk menentukan nilai resistor pembagi tegangan, pertama-tama adalah menentukan arus output sensor. Setelah itu baru bisa menemukan nilai masing-masing resistor. Arus yang akan digunakan untuk output sensor adalah 5mA



Gambar 3.6 Rangkaian pembagi tegangan

Untuk menentukan nilai resistor Ra dan Rb dapat menggunakan hukum Ohm, dengan diketahui Vb=5volt, Va=3,3volt dan arus yang mengalir adalah 5mA

$$Ra = \frac{Va}{i} = \frac{3,3 \text{ volt}}{5 \text{ mA}} = 660 \Omega \approx 680 \Omega \quad (3.4)$$

$$VRb = Vb - VRa = 5 \text{ volt} - 3,3 \text{ volt} = 1,7 \text{ volt} \quad (3.5)$$

$$Rb = \frac{VRb}{i} = \frac{1,7}{5 \text{ mA}} = 340 \Omega \approx 330 \Omega \quad (3.6)$$

Keterangan :

Ra = hambatan pada Ra

Rb = hambatan pada Rb

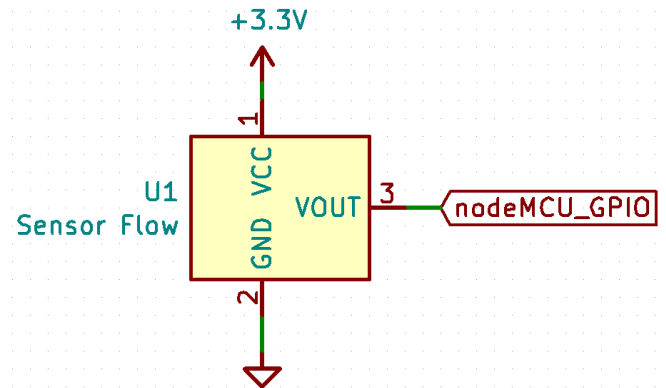
VRb = tegangan pada resistor Rb

i = arus yang mengalir pada Ra dan Rb

Jadi untuk menurunkan tegangan dari 5 volt menjadi 3,3 volt digunakan kombinasi resistor Ra = 680 ohm dan Rb = 330 ohm

3.3.2.5 Perancangan sensor *flow*

Penggunaan sensor *flow* pada sistem ini adalah untuk mendeteksi ada dan tidak aliran air saat motor pompa air hidup. Jika motor pompa air hidup dan tidak ada aliran air maka hal ini bisa disebabkan oleh beberapa faktor. Faktor penyebab tidak ada aliran air diantaranya adalah air sumur habis, pipa atau selang air bocor, motor pompa air rusak, jalur tersumbat, dan pompa air masuk angin. Saat pompa air hidup dan sensor tidak mendeteksi adanya aliran air, NodeMCU sebagai pengendali akan mematikan sistem dan mengirimkan *notifikasi* alarm *low pressure* ke aplikasi android melalui *database* firebase. Rangkaian sensor *flow* dapat dilihat pada Gambar 3.7 di bawah ini.



Gambar 3.7 Perancangan rangkaian sensor *flow*

Sensor *flow* meter ini tidak membutuhkan komponen external bila dihubungkan dengan pin NodeMCU. Sensor membutuhkan tegangan 3,3 volt yang diambil dari pin 3,3 volt yang ada pada modul NodeMCU. Output dari modul ini berupa gelombang kotak dengan jumlah pulsa berbanding lurus dengan jumlah aliran air.

Satuan yang akan digunakan pada sistem ini adalah ml. Untuk menentukan jumlah pulsa per 1ml harus dilakukan percobaan. Percobaan akan dilakukan dengan cara mengalirkan air sebanyak 600ml dan 350ml kemudian menghitung jumlah pulsa yang dihasilkan oleh sensor *flow* meter. Setelah didapatkan hasil percobaan kemudian dimasukkan kedalam persamaan berikut :

$$Pulsa/1mL = \frac{Jumlah\ Pulsa}{Jumlah\ Air\ (ml)} \quad (3.7)$$

3.3.3 Perancangan perangkat lunak

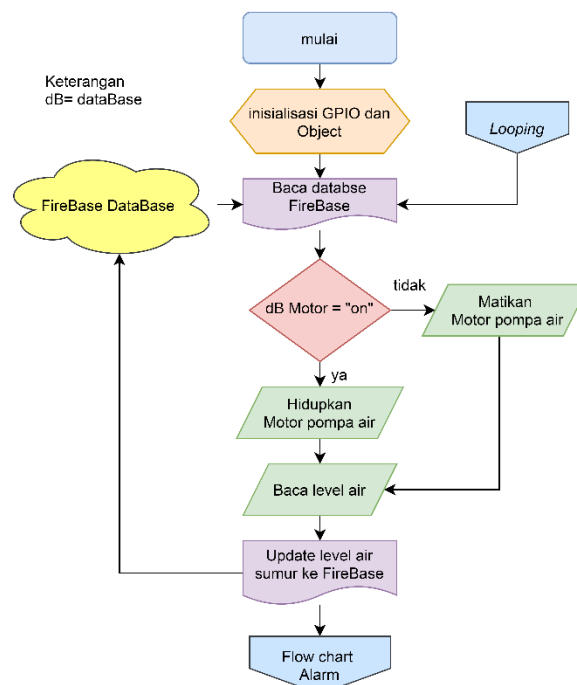
Perancangan perangkat lunak dibuat setelah perancangan perangkat keras selesai dibuat. Terdapat 3 perangkat lunak yang akan dibuat yaitu perangkat lunak *NodeMCU* kendali pompa air, perangkat lunak *NodeMCU* monitor tangki air, dan

perangkat lunak *android*. Proses perancangan perangkat lunak dalam penelitian ini dibagi menjadi 3 tahap.

1. Perancangan perangkat lunak *NodeMCU* kendali pompa air
2. Perancangan perangkat lunak *NodeMCU* monitor tangki air
3. Perancangan perangkat lunak *android*

3.3.3.1 Perancangan perangkat lunak *NodeMCU* kendali pompa air

Perancangan perangkat lunak *NodeMCU* kendali pompa air berfungsi untuk membaca perintah dari firebase untuk menghidupkan dan mematikan motor pompa air. Selain itu *NodeMCU* juga bertugas membaca sensor level air, sensor *flow* , sensor suhu dan sensor *overload*. Adapun diagram alir perangkat lunak *NodeMCU* kendali untuk mengendalikan pompa air ditunjukkan pada Gambar 3.8.



Gambar 3.8 *Flowchart* perangkat lunak *NodeMCU* untuk pengendali pompa air

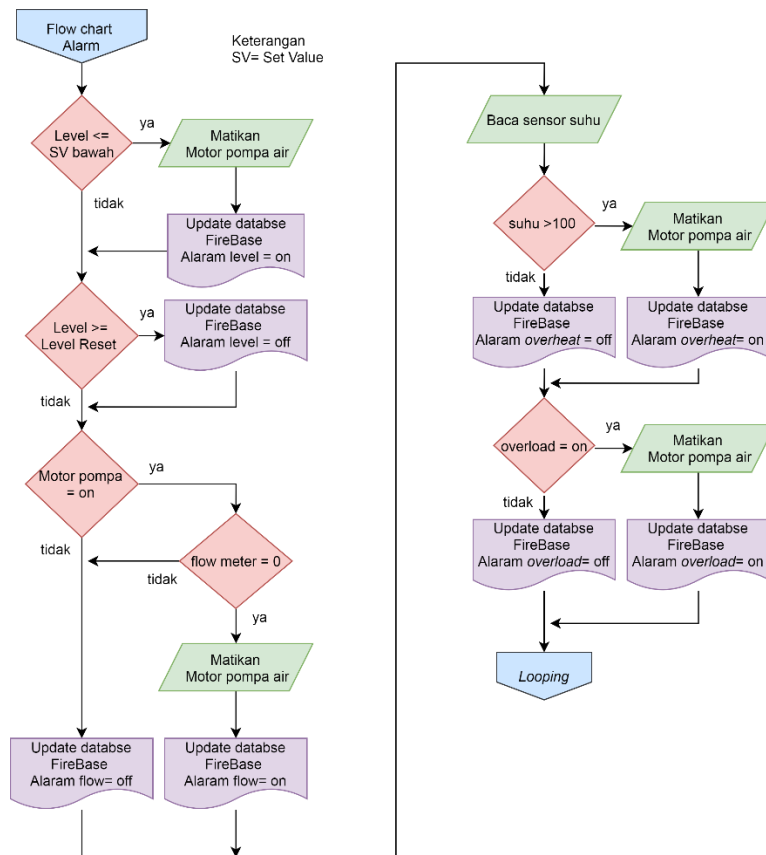
Program dimulai dengan inisialisasi GPIO dan *object* yang akan digunakan pada program. Setelah itu program akan membaca database firebase dengan *key* motor. Jika isi dari database itu berisi *true* maka *NodeMCU* akan memerintahkan motor pompa air untuk hidup tetapi jika bukan *true* maka *NodeMCU* akan mematikan motor pompa air.

Selanjutnya *NodeMCU* akan membaca level air sumur melalui sensor ultrasonic HC-SR04. Data level air akan disimpan pada database firebase dengan *key* sumur. Program selanjutnya adalah membaca kerusakan yang mungkin terjadi melalui sensor level air, sensor *flow*, sensor suhu dan sensor *overload* lalu mengirimkan *notifikasi* alarm ke smartphone yang terhubung melalui firebase. Dari sensor tersebut *NodeMCU* dapat mengetahui kerusakan yang mungkin terjadi sesuai tabel 3.3 berikut ini.

Tabel 3.3 Kondisi lampu

No	Motor	Sensor <i>Flow</i>	Sensor Suhu	Level air	<i>Overload</i>	Keterangan
1	X	X	X	<SV	X	<i>Low Level</i>
2	ON	> 0	X	X	X	<i>Low Preassure</i>
3	X	X	> 100	X	X	<i>Over Heat</i>
4	X	X	X	X	0	<i>Over Current</i>

Dari tabel di atas dapat dilihat bahwa, lima kondisi sensor dan motor dapat menghasilkan empat jenis kemungkinan kerusakan. Tanda X menyatakan kondisi yang tidak diketahui. Melalui tabel diatas lalu dibuat alur program seperti yang digambarkan pada Gambar 3.9 di bawah ini.



Gambar 3.9 Flowchart perangkat lunak NodeMCU untuk notifikasi alarm

Program akan membaca level air sumur, jika level air kurang dari atau sama dengan set value bawah yang sudah di tentukan, maka nodeMCU akan mematikan motor pompa air jika dalam kondisi hidup lalu akan mengirimkan *notifikasi* alarm *low level* ke firebase. Selanjutnya jika level air sumur sudah lebih dari atau sama dengan level reset yang sudah di tentukan, maka *NodeMCU* akan menghapus *notifikasi* alarm *low level* di firebase.

Saat pompa air dalam kondisi hidup sensor *flow* akan dibaca oleh *NodeMCU*, jika nilai sensor *flow* sama dengan nol maka *NodeMCU* akan mematikan motor pompa air dan akan mengirimkan *notifikasi* alarm *low preassure* ke firebase. Jika

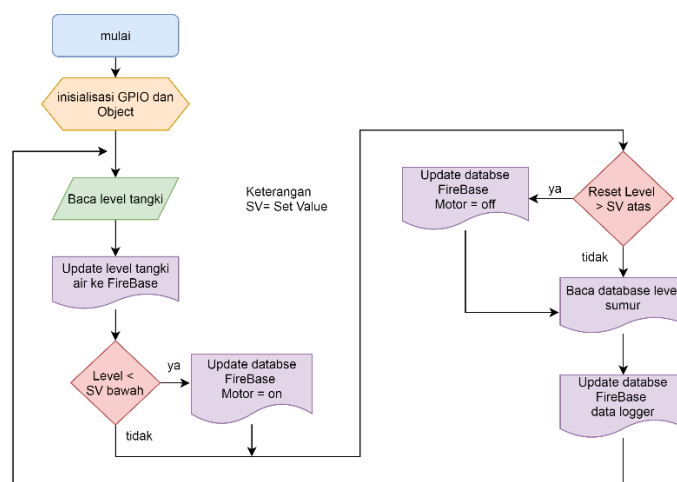
nilai sensor *flow* lebih dari nol maka *NodeMCU* akan menghapus *notifikasi low preassure* di firebase.

Sensor suhu digunakan untuk memeriksa suhu pada motor pompa air. Jika suhu yang dibaca oleh *NodeMCU* melebihi 100°C maka *NodeMCU* akan mematikan motor pompa air agar motor tidak terbakar karena suhu yang berlebih. Lalu *NodeMCU* akan mengirimkan *notifikasi alarm overheat* ke firebase.

Proses terakhir adalah memeriksa sensor overload. Sensor overload bersifat kontak normaly open, jika terjadi overload karena kelebihan arus pada motor pompa air *NodeMCU* akan mematikan motor dan mengirimkan *notifikasi alarm overcurrent* ke firebase.

3.3.3.2 Perancangan perangkat lunak *NodeMCU* pada tangki air

Perancangan perangkat lunak *NodeMCU* pada tangki air berfungsi untuk membaca level tangki air lalu menentukan hidup atau matinya pompa air melalui perintah yang dikirim ke firebase. Selain itu *NodeMCU* juga bertugas mencatat level air lalu disimpan di firebase. Adapun diagram alir perangkat lunak *NodeMCU* pada tangki air ditunjukkan pada Gambar 3.10.



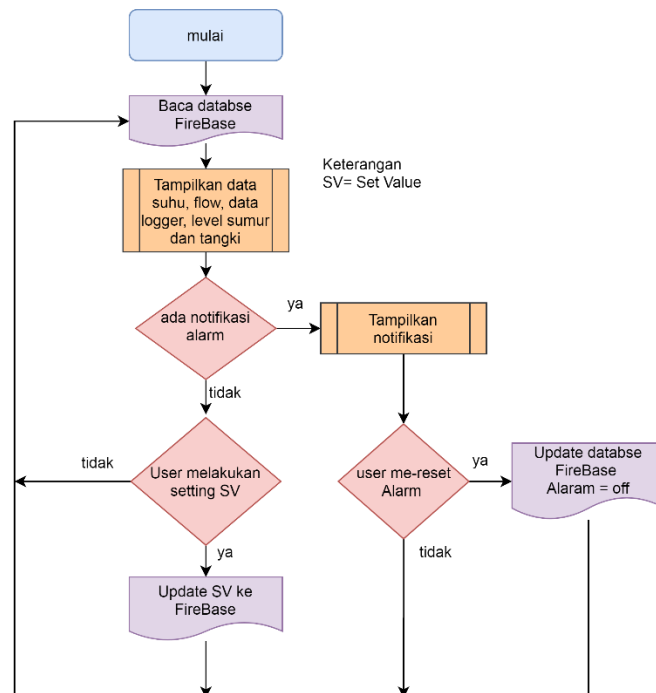
Gambar 3.10 Flowchart perangkat lunak *NodeMCU* untuk tangki air

Program dimulai dengan inisialisasi GPIO dan *object* yang akan digunakan pada program. Selanjutnya *NodeMCU* akan membaca level air tangki melalui sensor ultrasonic HC-SR04 lalu meng-*update* data level air tangki yang ada di firebase. Jika level air kurang dari atau sama dengan set value bawah yang sudah ditentukan *NodeMCU* akan meng-*update* data firebase dengan *key* motor dengan nilai *true* (motor on). Jika level air lebih dari atau sama dengan set value atas yang sudah ditentukan *NodeMCU* akan meng-*update* data firebase dengan *key* motor dengan nilai *false* (motor off).

Setelah itu program akan membaca database firebase dengan *level sumur*, kemudian menyimpan data level sumur dan level tangki ke database firebase setiap 10 detik untuk data logger. Setelah itu program kembali dari awal membaca level air pada tangki.

3.3.3.3 Perancangan perangkat lunak untuk *smartphone*

Perancangan perangkat lunak untuk *smartphone* berfungsi untuk memantau sistem aplikasi *IoT* pada sistem kendali pompa air. *Smartphone* akan menampilkan data sensor, *notifikasi* dan data logger pada sistem aplikasi *IoT* pada sistem kendali pompa air. *Smartphone* juga digunakan untuk memasukan nilai *set value* pada sistem aplikasi *IoT* pada sistem kendali pompa air. diagram alir perangkat lunak lunak untuk *smartphone* ditunjukan pada Gambar 3.11.



Gambar 3.11 Flowchart perangkat lunak untuk *smartphone*

Program dimulai dengan membaca data pada database firebase lalu menampilkan data-data tersebut pada layar *smartphone*. Bila ada *notifikasi smartphone* akan menampilkan *notifikasi* yang aktif lalu *user* akan diberi pilihan untuk me-*reset* alarm apabila diperlukan.

Jika tidak ada *notifikasi* program akan mengecek apakah *user* melakukan *set value* untuk parameter yang ada pada sistem. Apabila ada parameter yang di ubah program akan meng-*update* data parameter *set value* pada database firebase.

3.3.4 Pengujian sistem

Dalam proses pengujian sistem terdapat beberapa proses pengujian yang akan dilakukan untuk mengetahui kemampuan sistem yang dirancang, pengujian yang dilakukan diantaranya yaitu:

3.3.4.1 Pengujian perangkat keras sistem

1. Pengujian *step down* Lm2596

Pengujian dilakukan untuk memastikan rangkaian *step down* converter sudah bekerja dengan baik dan tegangan output sesuai.

Pengujian dilakukan dengan cara mengukur tegangan input dan output

2. Pengujian *Driver* Motor

Pengujian *driver* motor dilakukan dengan cara memberikan logika 0 dan 1 pada bagian input dan mengamati kondisi motor dalam keadaan *on* atau *off*.

3. Pengujian sensor suhu

Pengujian sensor suhu dilakukan dengan cara menaikkan suhu sensor menggunakan api lalu mengamati hasil pembacaan sensor

4. Pengujian sensor ultrasonic

Pengujian sensor ultrasonic dilakukan dengan cara memberi benda yang bisa memantulkan gelombang ultrasonic, lalu membaca hasil output dari sensor ultrasonic

5. Pengujian *flow* sensor

Pengujian sensor *flow* dilakukan dengan cara mengalirkan air pada sensor dengan jumlah yang sudah ditentukan, lalu menghitung jumlah pulsa keluaran sensor *flow*.

3.3.4.2 Pengujian perangkat-lunak sistem

Perangkat-lunak yang digunakan pada sistem ini ada 2, sebagai berikut.

1. Pengujian perangkat-lunak NodeMCU

Pengujian perangkat-lunak NodeMCU dilakukan dengan cara melakukan *run* program pada NodeMCU untuk mengetahui laju aliran

program yang dirancang dengan disesuaikan pada perancangan kerja sistem.

2. Pengujian aplikasi android

Pengujian aplikasi android dilakukan dengan cara mengkoneksikan antara aplikasi dengan NodeMCU dan menjalankan aplikasi dengan menggunakan masing-masing fungsi pengendali yang terdapat pada aplikasi pengendali tersebut.

3.3.4.3 Pengujian sistem keseluruhan

Pengujian sistem secara keseluruhan dilakukan dengan cara menghubungkan seluruh sistem NodeMCU dengan aplikasi android menjadi satu dan mengujinya sesuai dengan kinerja sistem secara manual melalui aplikasi pengendali.

3.4 Kesulitan – kesulitan

Dalam penelitian yang dilakukan peneliti menemui kesulitan - kesulitan yang ditemui selama proses perancangan sistem. Kesulitan-kesulitan yang ditemui selama proses perancangan sebagai berikut.

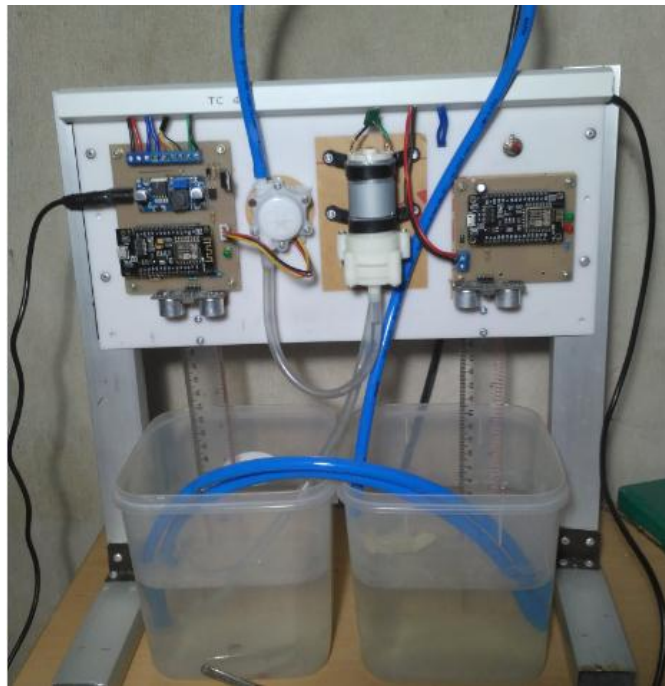
1. Proses building aplikasi android yang memakan waktu cukup lama
2. Pembacaan sensor ultrasonic yang tidak setabil
3. Pengaksesan *database* firebase menggunakan NodeMCU yang menurut penulis lambat, hal ini karena keterbatasan dari modul NodeMCU

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1 Hasil Penelitian

Perangkat-keras aplikasi *internet of things (IoT)* pada sistem kendali pompa air dirancang menggunakan PCB yang diletakan pada sebuah papan akrilik kemudian papan tersebut disangga menggunakan alumunium. Di bawah papan terdapat dua wadah air untuk simulasi sumber air dan tangki air. Gambar hasil perancangan Perangkat-keras ditunjukkan pada Gambar 4.1.



Gambar 4.1 Hasil perancangan perangkat-keras

Setelah perancangan selesai, diperoleh data dari melakukan pengukuran dengan menggunakan alat ukur dan dengan pengamatan secara langsung. Hasil pengujian dalam hal ini adalah meliputi hasil pengujian pada perangkat keras dan pengujian perangkat lunak. Pengujian perangkat lunak hanya dilakukan apabila perangkat keras yang dibuat sudah selesai, hal ini dapat dikatakan juga bahwa

pengujian perangkat lunak adalah pengujian alat secara keseluruhan. Berikut adalah uraian hasil pengujian dari masing masing bagian perangkat keras tersebut.

4.2 Hasil Pengujian Perangkat Keras

Pada hasil pengujian perangkat keras aplikasi *internet of things (IoT)* pada sistem kendali pompa air meliputi hasil pengujian *step down converter* lm2596, pengujian *driver* motor, pengujian sensor suhu, Pengujian sensor ultrasonic HC-SR04 dan pengujian sensor *flow*.

4.2.1 Hasil pengujian *Step down Converter* lm2596

Rangkaian *Step down Converter* digunakan sebagai penurun tegangan dari 12 Volt menjadi 5 Volt DC. Rangkaian *Step down Converter* digunakan untuk memberi tegangan ke NodeMCU, sensor ultrasonic HC-SR04. Pada Gambar 4.2 menunjukkan gambar pengujian rangkaian *Step down Converter* lm2596.



Gambar 4.2 Pengujian rangkaian *step down converter*

Pengujian dilakukan dengan cara mengukur output rangkaian rangkaian *Step down Converter* lm2596 menggunakan multimeter. Hasil pengukuran rangkaian rangkaian *Step down Converter* lm2596 harus menunjukkan angka 5 volt. Apabila pengukuran belum sesuai yang di inginkan maka trimpot yang ada pada rangkaian rangkaian *Step down Converter* lm2596 harus diputar sampai output menunjukkan angka 5 volt.

4.2.2 Pengujian *driver* motor

Driver motor digunakan sebagai saklar untuk memutus dan menghubungkan sumber tegangan pada motor pompa air. Pengujian dilakukan dengan memberikan tegangan 3.3 volt pada *input driver* motor untuk mengetahui apakah *driver* motor dapat berfungsi atau tidak, setelah itu mengukur tegangan pada motor pompa air dengan menggunakan multimeter. Hasil pengujian ditunjukkan pada Gambar 4.3.



Gambar 4.3 Pengujian saat motor berputar

Saat input driver motor diberi logika *low* (tegangan 0 volt) lalu melakukan pengukuran tegangan pada motor pompa, multimeter menunjukkan pengukuran sebesar 12 volt. Kemudian saat input driver motor diberi logika *high* (tegangan 3,3 volt) lalu melakukan pengukuran tegangan pada motor pompa, multimeter menunjukkan pengukuran sebesar 0 volt. hasil pengujian *driver* motor ditunjukkan pada Tabel 4.1.

Tabel 4.1 Pengujian driver motor

No	Input	Output	Kondisi Motor Poma Air
1	<i>Low</i> (0 volt)	12 volt	Berputar
2	<i>High</i> (3,3 volt)	0 volt	Berhenti

Dari pengujian yang telah dilakukan, disimpulkan bahwa *driver* motor ini telah bekerja dengan baik. *Driver* motor ini bersifat *inverting* yaitu ketika diberi lokika *high* output mati dan ketika diberi logika *low* output hidup.

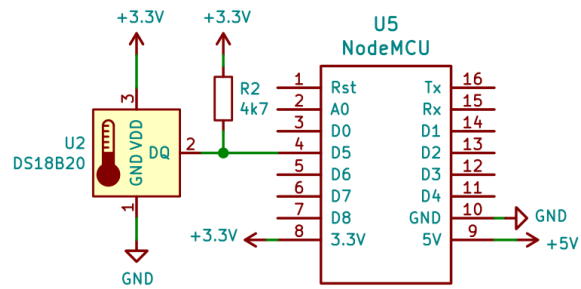
4.2.3 Pengujian sensor suhu

Pengujian sensor suhu DS18B20 bertujuan untuk mengetahui tingkat akurasi dari data hasil pengukuran sensor suhu. Pengukuran dilakukan dengan membandingkan termometer digital dan sensor suhu DS18B20 dengan perlakuan yang sama. Pengukuran dilakukan menggunakan air panas. Sensor suhu DS18B20 dan termometer digital diletakkan didalam air panas. Pada Gambar 4.4 di bawah ini adalah gambar proses pengujian sensor suhu DS18B20.



Gambar 4.4 Proses pengujian sensor suhu DS18B20

Sensor suhu DS18B20 dihubungkan dengan modul *NodeMCU*. Sensor suhu DS18B20 yang digunakan adalah type *waterproof* yang mempunyai 3 kabel, kabel tersebut berwarna merah, kuning dan hitam. Kabel merah sensor dihubungkan dengan +3,3volt, kabel hitam dihubungkan dengan GND dan kabel kuning dihubungkan dengan pin D5 pada *NodeMCU*. Pada kabel yang berwarna kuning ditambahkan resistor 4,7K Ω di hubungkan dengan +3,3volt yang berfungsi sebagai *pull-up*. Skematik rangkaian antara *NodeMCU* dengan sensor suhu DS18B20 dapat dilihat pada Gambar 4.5 berikut ini.



Langkah selanjutnya adalah proses *upload* listing program ke modul *NodeMCU* untuk pengujian sensor suhu DS18B20. Di bawah ini adalah listing program untuk pengujian sensor suhu DS18B20.

```
#include <Arduino.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS D5
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
void setup(void){
  Serial.begin(9600);
  sensors.begin();
}
void loop(void){
  sensors.requestTemperatures();
  float tempC = sensors.getTempCByIndex(0);
  Serial.print("Suhu sensor : ");
  Serial.println(tempC);
  delay(2000);
}
```

Ketika dijalankan Program diatas akan menampilkan hasil pembacaan sensor ultrasonic suhu DS18B20 yang akan ditampilkan di serial monitor pada komputer. Hasil pengujian sensor ini dapat dilihat pada Gambar 4.5.

```

OUTPUT  TERMINAL  DEBUG CONSOLE  PROBLEMS
Suhu sensor : 62.25
Suhu sensor : 62.19
Suhu sensor : 62.06
Suhu sensor : 62.00
Suhu sensor : 62.00
Suhu sensor : 61.94
Suhu sensor : 61.88
Suhu sensor : 61.81
Suhu sensor : 61.75

```

Gambar 4.5 Hasil pengujian sensor suhu DS18B20

Sensor suhu digital dan suhu DS18B20 diletakan dalam air panas, kemudian ditunggu beberapa saat sampai hasil pembacaan stabil. Pada saat sensor suhu digital menunjukan suhu 62,8 °C, pada serial monitor di komputer menunjukan suhu 62,63 °C lalu ketika sensor suhu digital menunjukan suhu 65°C serial monitor di komputer menunjukan suhu 65,4°C. hasil pengujian dari sensor suhu DS18B20 ditunjukkan pada Tabel 4.3

Tabel 4.2 Pengujian sensor DS18B20

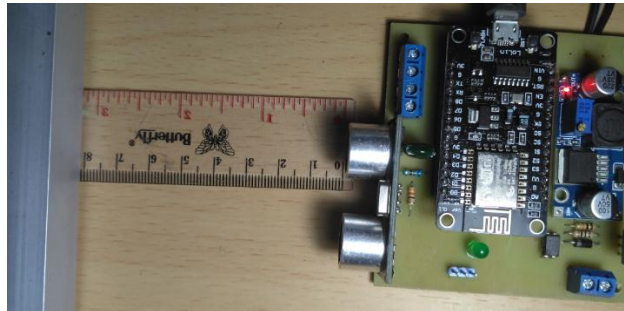
No	Suhu thermometer	Suhu sensor ds18b20
1	30,7 °C	30,12 °C
2	35,3 °C	34,86 °C
3	41,8 °C	40,83 °C
4	45,3 °C	45,16 °C
5	50,6 °C	50,00 °C
6	60,1 °C	59,69 °C
7	65,8 °C	65,63 °C

Dari hasil pengujian yang ditujukan pada Table 4.3. terlihat jarak hasil pengujian pada alat terdapat selisih sekitar 1°C , hal ini terjadi karena tingkat kepresisian sensor yang berbeda.

4.2.4 Pengujian sensor ultrasonic HC-SR04.

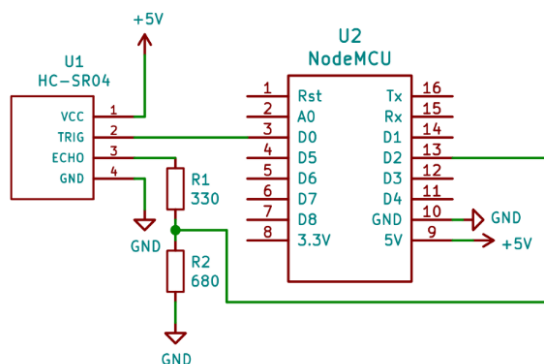
Pengujian sensor ultrasonic dikalukan dengan cara memberi benda yang bisa memantulkan gelombang ultrasonic, lalu membaca hasil output dari sensor

ultrasonic. Pada Gambar 4.6 di bawah ini adalah gambar proses pengujian sensor ultrasonic HC-SR04.



Gambar 4.6 Proses pengujian sensor ultrasonic HC-SR04

Sensor ultrasonic HC-SR04 dihubungkan dengan modul *NodeMCU*. Pin TRIG sensor ultrasonic HC-SR04 dihubungkan dengan pin D0 *NodeMCU* dan pin ECHO sensor ultrasonic HC-SR04 dihubungkan dengan resistor pembagi tegangan lalu output resistor pembagi tegangan dihubungkan dengan pin D2 *NodeMCU*. Resistor pembagi tegangan yang digunakan adalah 330Ω dan 680Ω . Skematik rangkaian antara *NodeMCU* dengan sensor HC-SR04 dapat dilihat pada Gambar 4.7 berikut ini.



Gambar 4.7 Skematik pengujian sensor ultrasonic HC-SR04

Setelah pengkabelan selesai dilakukan, kemudian dilakukan proses *upload* listing program ke modul *NodeMCU*. Di bawah ini adalah listing program untuk pengujian sensor ultrasonic HC-SR04.


```

#include <Arduino.h>
#include "NewPING.h"
#define TRIGGER_PIN D0 // trigger pin sensor ultrasonic .
#define ECHO_PIN D2 // echo pin sensor ultrasonic .
#define MAX_DISTANCE 200 // panjang maksimal dalam centimeters.

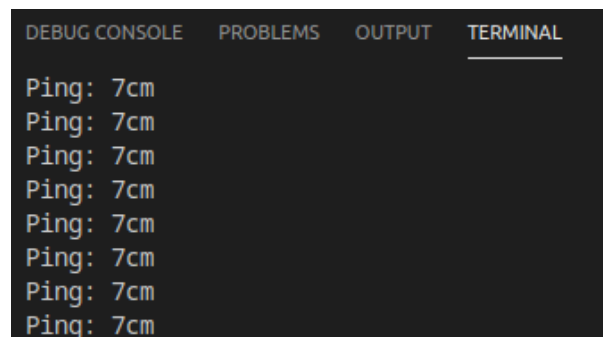
NewPING *sonar = new NewPING (TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);

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

void loop() {
  Serial.print("PING: ");
  Serial.print(sonar->PING_cm());
  Serial.println("cm");
  delay(2000);
}

```

Ketika dijalankan Program diatas akan menampilkan hasil pembacaan sensor ultrasonic HC-SR04 yang akan ditampilkan di serial monitor pada komputer. Hasil pengujian sensor ini dapat dilihat pada Gambar 4.8.



Gambar 4.8 Hasil pengujian sensor HC-SR04

Saat benda diletakan dengan jarak 7 cm dari sensor ultrasonic HC-SR04, pada serial monitor di komputer menunjukan tulisan Ping: 7 cm, lalu ketika jarak sensor diubah menjadi 8 cm serial monitor di komputer menunjukan tulisan Ping: 8 cm. hasil pengujian dari sensor ultrasonic HC-SR04 ditunjukkan pada Tabel 4.3.

Tabel 4.3 Pengujian sensor *ultrasonic*

No	Panjang pengukuran (penggaris)	Output sensor (serial monitor)
1	5 cm	5 cm
2	6 cm	6 cm
3	7 cm	7 cm
4	8 cm	8 cm
5	9 cm	9 cm
6	10 cm	10 cm
7	11 cm	11 cm

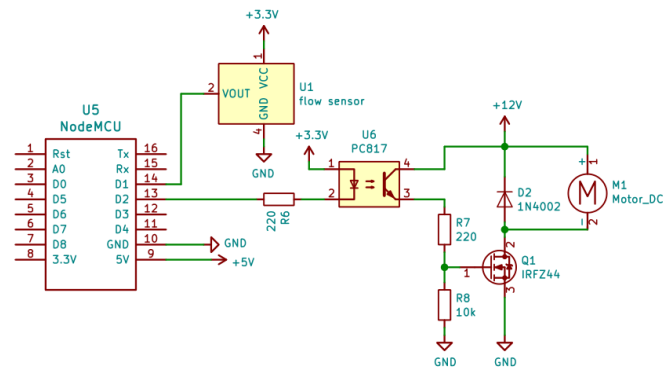
4.2.5 Pengujian sensor *flow*

Pengujian sensor *flow* dilakukan dengan cara mengalirkan air pada sensor dengan jumlah yang sudah ditentukan, lalu menghitung jumlah pulsa keluaran sensor *flow*. Pada Gambar 4.9 di bawah ini adalah gambar proses pengujian sensor *flow*.



Gambar 4.9 Proses pengujian sensor *flow*

Output dari sensor *flow* dihubungkan dengan pin D1 *NodeMCU* dan pin D2 *nodemcu* dihubungkan dengan rangkaian *driver* motor lalu output dari *driver* motor dihubungkan dengan motor pompa air. Sambungan skematik rangkaian antara *NodeMCU* dengan sensor *flow*, *driver* motor dan motor pompa air dapat dilihat pada Gambar 4.10 berikut ini.



Gambar 4.10 Rangkaian pengujian sensor *flow*

Selanjutnya proses *upload* listing program ke modul *NodeMCU* untuk pengujian sensor *flow*. Di bawah ini adalah listing program untuk pengujian sensor *flow*.

```
#include <Arduino.h>
int pin = D1;
int tambah;
unsigned long counter;
void setup() {
  Serial.begin(9600);
  Serial.setTimeout(20);
  pinMode(pin, INPUT);
  pinMode(D4, OUTPUT);
}
void loop() {
  if (digitalRead(pin)==HIGH){
    if (tambah == 1){
      counter++;
      tambah = 0;
    }
  }
  else{
    tambah=1;
  }
  if (Serial.available() > 0) {
    String baca =Serial.readString();
    if (baca == "C"){
      Serial.println("Stop Counter");
      Serial.print("Jumlah Pulsa = ");
      Serial.println(counter, DEC);
      digitalWrite(D4,HIGH);
    }
  }
}
```

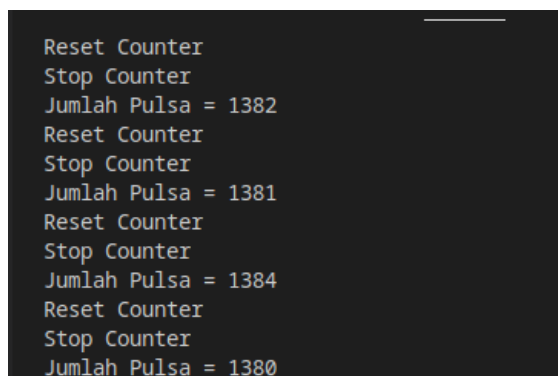
```

    }
    if (baca == "R"){
        Serial.println("Reset Counter");
        counter=0;
        digitalWrite(D4,LOW);
    }
}
}
}

```

Program diatas digunakan untuk menghidupkan dan mematikan pompa air. ketika pompa air hidup, air akan mengalir melalui sensor *flow* lalu output dari sensor ini akan dibaca oleh *NodeMCU* lalu ditampilkan di serial monitor.

Cara pengujianya adalah mengirimkan karakter “R” dari serial monitor ke *NodeMCU*, Karakter “R” ini akan memerintahkan motor pompa air untuk hidup sekaligus me-reset variabel *counter*. Pompa air yang hidup akan mengisi botol dengan ukuran 350ml dan 600ml. Ketika air dalam botol penuh, kirim karakter “C” dari serial monitor. Karakter “C” ini akan memerintahkan motor pompa air untuk mati sekaligus mengirimkan jumlah variabel *counter* ke serial monitor. Poses pengujian sensor ini dapat dilihat pada Gambar 4.11.



```

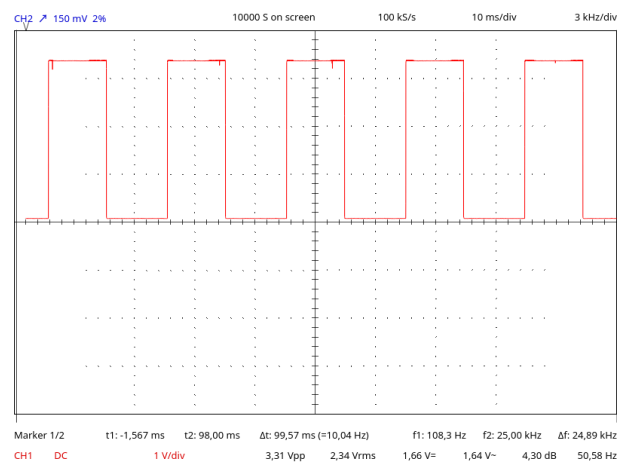
Reset Counter
Stop Counter
Jumlah Pulsa = 1382
Reset Counter
Stop Counter
Jumlah Pulsa = 1381
Reset Counter
Stop Counter
Jumlah Pulsa = 1384
Reset Counter
Stop Counter
Jumlah Pulsa = 1380

```

Gambar 4.11 Output pengujian sensor *flow*

Output dari sensor *flow* adalah gelombang kotak dengan tegangan 3,3 volt (sama dengan suber sensor *flow*). Frekuensi yari gelombang ini bergantung pada

kecepatan aliran air, semakin cepat aliran air semakin tinggi frekuensi yang di hasilkan. Bentuk gelombang kotak yang di hasilkan oleh sensor *flow* dapat dilihat pada Gambar 4.12 dibawah ini



Gambar 4.12 Bentuk gelombang output sensor *flow*

Proses pengujian sensor *flow* ini menggunakan air dengan volume 350ml dan 600ml. setiap volume dilakukan pengujian sebanyak tiga kali. Setelah semua data didapatkan kemudian dihitung jumlah pulsa per 1ml menggunakan persamaan berikut ini.

$$Pulsa/1mL = \frac{Jumlah\ Pulsa}{Jumlah\ Air\ (ml)} \quad (4.1)$$

Berikut ini adalah hasil pengujian sensor *flow* yang ditunjukkan pada Tabel

4.4

Tabel 4.4 Pengujian sensor *flow*

No	Jumlah air	pengujian	Output (pulsa)	Pulsa / ml
1	350ml	Pengujian ke-1	1382	3,94
2	350ml	Pengujian ke-2	1381	3,94
3	350ml	Pengujian ke-3	1384	3,95
4	600ml	Pengujian ke-1	2342	3,90
5	600ml	Pengujian ke-2	2347	3,91
6	600ml	Pengujian ke-3	2357	3,92

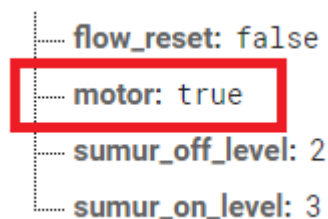
Dari hasil perhitungan didapat angka rata-rata jumlah pulsa yang dihasilkan setiap 1mL air adalah 3,9 pulsa. Angka ini nantinya dapat digunakan untuk menghitung kecepatan laju air.

4.3 Hasil Pengujian Perangkat Lunak

Pada hasil pengujian perangkat lunak aplikasi *internet of things (IoT)* pada sistem kendali pompa air meliputi hasil pengujian lunak *NodeMCU* kendali pompa air, pengujian perangkat lunak *NodeMCU* pada tangki air, dan pengujian perangkat lunak untuk *smartphone* .

4.3.1 Hasil pengujian lunak *nodemcu* kendali pompa air

Pengujian perangkat lunak *nodemcu* dilakukan dengan cara mengamati dan merubah database firebase melalui browser, lalu melihat respon dari hardware *NodeMCU*. Pengujian pertama adalah mengubah data pada firebase dengan *key* motor, data di rubah dari *false* ke *true* dan sebaliknya. Pada Gambar 4.13 di bawah ini adalah gambar saat proses merubah database dari *false* ke *true*. Saat data dari firebase diubah dari *false* ke *true* motor pompa air hidup, lalu ketika data dari firebase diubah dari *true* ke *false* motor pompa air mati.



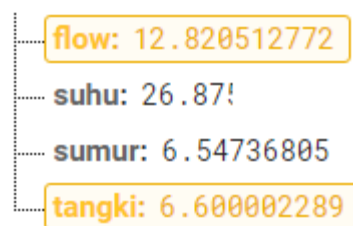
```
----- flow_reset: false
----- motor: true
----- sumur_off_level: 2
----- sumur_on_level: 3
```

Gambar 4.13 Merubah database *key* motor

Pengujian selanjutnya yaitu menghidupkan motor pompa air, lalu memicu terjadinya alarm dengan cara menurunkan level air, menghentikan aliran air,

menaikkan suhu sensor sampai lebih dari 100 °C dan menghubungkan kontak *overload*.

Pada saat level air diturunkan sampai batas bawah level alarm, pada saat yang sama database firebase dengan *key level* berubah menjadi *true*. Kemudian saat level air dinaikan sampai batas level reset, pada database firebase dengan *key level* berubah menjadi *false*. Selanjutnya pada sensor flow diberi aliran air untuk memutar turbin sensor, pada saat yang sama dilakukan pengamatan terhadap database firebase dengan *key flow*. Pada Gambar 4.14 di bawah ini adalah gambar struktur database firebase dengan *key flow* saat ada aliran air.

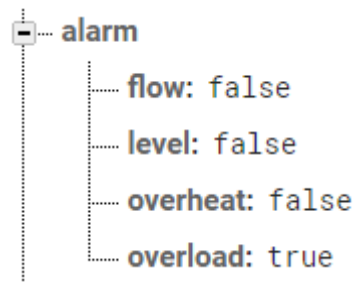


Gambar 4.14 Database firebase dengan *key flow* saat ada aliran air

Pengamatan terhadap database firebase dengan *key flow* menunjukkan ada perubahan nilai dari 0 menjadi 12,82. Angka ini adalah kecepatan laju air dalam satuan ml/detik. Kemudian saat aliran air dihentikan pada *key alarm/flow* terjadi perubahan dari *false* ke *true*.

Proses berikutnya adalah memicu sensor suhu. Sensor suhu dipicu dengan cara dipanaskan menggunakan api sampai suhu lebih dari 100 °C. Saat suhu sudah melebihi 100 °C, pada database firebase dengan *key overheat* menunjukkan ada perubahan dari *false* menjadi *true*. Pengujian terakhir pada perangkat lunak nodemcu kendali pompa air adalah pengujian *overload*. Proses pengujianya adalah menghubungkan kontak *overload* lalu mengamati database firebase. Saat kontak

sensor aktif, pada database firebase dengan *key overload* menunjukkan ada perubahan dari *false* menjadi *true*. Pada Gambar 4.15 di bawah ini adalah gambar struktur data alarm pada database firebase.

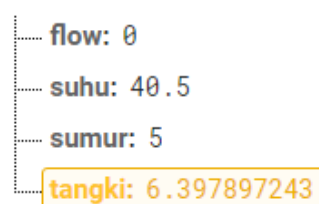


Gambar 4.15 Struktur data alarm pada database firebase

Hasil pengujian aplikasi dengan realtime database Firebase menunjukkan bahwa pengiriman data dari aplikasi ke database sukses, dapat dipantau secara realtime. Selain itu, data yang dikirim dari aplikasi dan data yang diterima oleh database sudah sama. Berdasarkan Hasil pengujian perangkat lunak nodemcu menunjukkan bahwa pembuatan perangkat lunak telah berjalan sesuai rencana

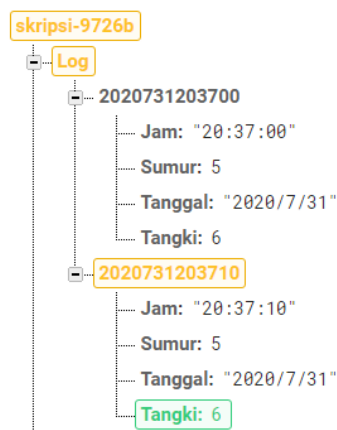
4.3.2 Pengujian perangkat lunak *NodeMCU* pada tangki air

Pengujian perangkat lunak *NodeMCU* pada tangki air dibagi menjadi dua proses pengujian. Pengujian pertama adalah mengamati perubahan level air terhadap database, yang ke dua adalah mengamati pencatatan data level di database firebase. Saat level air pada tangki air dirubah pada database firebase dengan *key* tangki data juga ikut berubah. Pada Gambar 4.16 menunjukan saat terjadi perubahan data tersebut.



Gambar 4.16 Perubahan data pada database dengan *key* tangki

Pengujian pada pencatatan level air adalah mengamati database dengan *key* Log. Proses pencatatan ini terjadi setiap 10 detik sekali. Saat proses pengamatan terjadi penambah data setiap 10 detik sekali, dengan isi data adalah waktu dan tanggal pencatatan dan data level air pada sumur dan tangki air. Gambar 4.17 di bawah adalah saat proses pencatatan terjadi.



Gambar 4.17 Proses pencatatan data level air

4.3.3 Pengujian perangkat lunak untuk *smartphone Android*

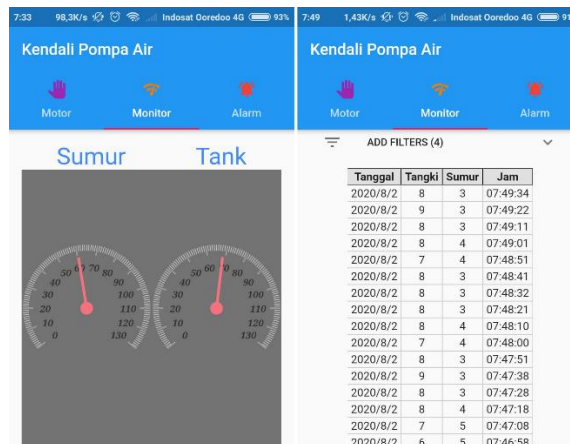
Perangkat lunak *smartphone android* ini di bagi menjadi tiga buat tampilan. Tampilan pertama adalah tampilan *setting value* level air, tampilan kedua adalah *monitoring* level air dan monitoring data *logger*, tampilan ketiga adalah tampilan alarm. Pengujian pertama adalah pengujian setting level air . Pada bagian setting level air terdapat empat variabel untuk diubah. Variabel itu adalah sumur alarm, sumur *reset*, tangki min, tangki max. Variabel sumur alarm adalah nilai minimal level air pada sumur untuk memicu level alarm. Sumur *reset* alarm adalah nilai level air pada sumur untuk me-*reset* level alarm. Tangki min dan tangki max adalah level air minimal dan maksimal pada tangki untuk menentukan motor hidup atau mati.



Gambar 4.18 Tampilan *setting value* level air

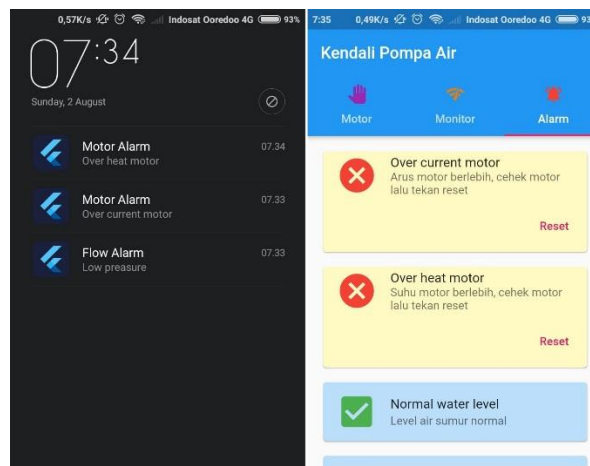
Gambar 418 di atas adalah tampilan *setting value* level air, pada bagian atas terdapat dua angka yang menunjukkan suhu motor dan kecepatan laju air. Kemudian pada bagian bawah terdapat kotak untuk merubah variabel *set value* dan tombol *save*. Pengujian dilakukan dengan cara merubah nilai *set value* lalu mengamati database firebase. Saat nilai diubah lalu menekan tombol save pada database firebase menunjukkan perubahan dengan nilai yang sama. Hal ini menunjukkan bahwa perangkat lunak *smartphone* untuk *setting value* level air sudah bekerja dengan baik.

Pengujian selanjutnya adalah pengujian monitoring level air dan data logger. Pengujian ini dilakukan dengan mengamati level air dan tampilan aplikasi perangkat lunak. Saat level air berubah tampilan pada *smartphone* juga ikut berubah kemudian pada bagian bawah terdapat tampilan datalogger level air setiap 10 detik. Hal ini menunjukkan bahwa perangkat lunak *smartphone* untuk monitoring level air dan data logger sudah bekerja dengan baik. Pada Gambar 4.19 dibawah ini adalah tampilan monitoring level air.



Gambar 4.19 Tampilan monitoring level air

Pengujian terakhir adalah pengujian tampilan alarm. Pengujian dilakukan dengan pemicu alarm pada perangkat keras *NodeMCU* lalu mengamati tampilan pada aplikasi *smartphone*. Saat alarm terjadi pada *smartphone* muncul *popup notifikasi* kemudian jika diklik akan menuju ke tampilan alarm. Pada tampilan ini jika ada alarm akan muncul tanda silang dengan warna merah, namun jika kondisi normal tanda centang berwarna hijau. Pada Gambar 4.20 di bawah ini adalah tampilan saat terjadi alarm.



Gambar 4.20 Tampilan saat ada *notifikasi alarm*

BAB V

SIMPULAN DAN SARAN

5.1 Simpulan

Kesimpulan yang dapat diambil setelah melakukan perancangan dan melakukan ujicoba pada sistem yang dirancang adalah sebagai berikut.

1. Perangkat keras dan perangkat lunak pada pompa air telah berhasil di rancang dan dapat berjalan sesuai dengan fungsi.
2. Sistem yang dirancang dapat di kendalikan melalui aplikasi android.

5.2 Saran

Setelah melakukan penelitian sistem yang dirancang dirasa masih terdapat kekurangan yang dapat dikembangkan lebih lanjut di antaranya adalah sebagai berikut.

1. Sensor ultrasonic yang digunakan kurang stabil, sehingga kedepannya dapat menggunakan sensor yang mempunyai kualitas yang bagus.
2. Pompa air hanya bisa *on off*, untuk kedepannya dapat di buat kendali PID
3. Tampilan pada aplikasi android agar bisa dibuat lebih menarik.

DAFTAR PUSTAKA

- Arief, U. M. (2011). *Pengujian Sensor Ultrasonik PING untuk Pengukuran Level Ketinggian dan Volume Air*. Semarang: UNHAS.
- Raharjo, B. (2019). *Pemrograman Android dengan Flutter*. Bandung: Informatika.
- Suwanto, P. (2019). *Kendali dan Monitoring Pompa Pendingin pada PLTD Siantan Berbasis IoT (Internet Of Things)*. Pontianak: Fakultas Teknik, Universitas Tanjungpura.
- Ulumuddin, S. T. (2017). *Prototipe Sistem Monitoring Air Pada Tangki Berbasis Internet of Things Menggunakan NodeMCU Esp8266 Dan Sensor Ultrasonik*. Bandung: Teknik Elektro UIN Sunan Gunung Djati.

LAMPIRAN