

**MONITOR DAN KONTROL PENYIRAM TANAMAN
OTOMATIS BERBASIS IOT**

PROYEK AKHIR

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan
Diploma III Politeknik Manufaktur Negeri Bangka Belitung



Disusun Oleh:

RAFI RAHMADI

NIRM : 0031522

RANGGI

NIRM : 0031523

**POLITEKNIK MANUFAKTUR NEGERI
BANGKA BELITUNG**

2018

LEMBAR PENGESAHAN

**MONITOR DAN KONTROL PENYIRAM TANAMAN OTOMATIS
BERBASIS IOT**

Oleh :

Rafi Rahmadi

NIRM : 0031522

Ranggi

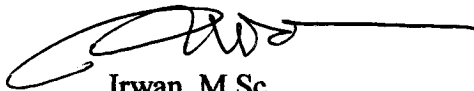
NIRM : 0031523

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan
Program Diploma III Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1

Pembimbing 2



Irwan, M.Sc.

M. Iqbal Nugraha, M.Eng.

Penguji 1

Penguji 2

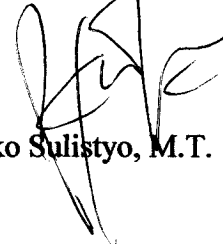
Penguji 3



Indra Dwisaputra, M.T.



Dr. Parulian Silalahi, M.Pd.



Eko Sulistyono, M.T.

PERNYATAAN BUKAN PLAGIAT

Yang bertanda tangan di bawah ini :

Nama Mahasiswa 1: Rafi Rahmadi

NIRM : 0031522

Nama Mahasiswa 2 : Ranggi

NIRM : 0031523

Dengan Judul: KONTROL DAN MONITOR PENYIRAM TANAMAN OTOMATIS BERBASIS IOT.

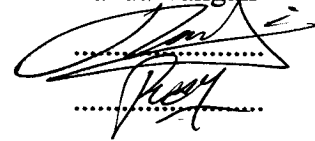
Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia diberikan sanksi yang berlaku.

Sungailiat, 31 Juli 2018

Nama Mahasiswa

1. Rafi Rahmadi
2. Ranggi

Tanda Tangan

The image shows two handwritten signatures in black ink. The first signature is larger and more stylized, while the second is smaller and more compact. Both signatures are written over a horizontal dotted line.

ABSTRAK

Pada saat ini, kebanyakan penyiraman tanaman dalam sebuah perkebunan masih menggunakan cara manual atau masih menggunakan tenaga manusia. Metode ini akan dirasa kurang maksimal karena dalam hal penyiraman memakan tenaga yang lebih dan penggunaan air yang kurang efisien. Selain itu dalam pengontrolan kelembaban kita harus selalu standby untuk memonitoring hasil dari pengukuran kelembaban agar nutrisi air yang didapatkan tanaman tercukupi. Namun hal tersebut dapat menguras waktu dan tenaga dan justru menambah biaya perawatan. Dari persoalan tersebut, penulis berinovasi untuk membuat sebuah alat penyiram tanaman otomatis yang dapat dikontrol dan monitor melalui sebuah software aplikasi handphone android melalui sistem internet. Tujuan dari penelitian ini adalah untuk memudahkan sistem penyiraman tanaman yang dapat dikontrol melalui sebuah sistem. Yaitu dengan nyalanya pompa air pada jam-jam yang ditentukan dengan jumlah air yang ditentukan melalui parameter kelembaban tanah. Hasil dari monitor dan kontrol tersebut ditampilkan pada sebuah aplikasi android dengan data logger yang tersedia dalam bentuk grafik. Dalam sebuah sistem ini menggunakan Arduino UNO dan NodeMCU sebagai komponen utama dalam hal mengendalikan pompa air dan pengiriman data. Sementara itu alat yang digunakan untuk mendeteksi nilai kelembaban tanah (soil moisture) menggunakan sensor yl39 dan sensor DHT22 sebagai pendeteksi nilai suhu lingkungan dan kelembaban udara. Hasil output dari proyek akhir ini yaitu alat dapat mengontrol dan memonitor proses penyiraman tanaman melalui android yang terhubung ke internet.

Kata kunci: IoT, Penyiram tanaman, Tanaman.

Abstract

Currently, most of the watering of plants in plantation are still using the manual way or still using human labor. This method will be less efficient because in terms of watering takes up more power and use less water efficient. In addition to controlling the humidity we have to be always standby to monitor the results of the measurement moisture so that the nutrients water obtained the plant fulfilled. However in such a case can drain the time and energy that actually adds to the cost of care. Of the issues, the author innovate to make a automatic sprinklers that can be controlled and monitored through a software application android mobile phone through the internet system. The purpose of this study is to facilitate system of watering plants that can be controlled via a system. Namely with flames water pump in the hours that have been determined by the amount of water that is specified through the parameter soil moisture. The results from the monitor and these controls will be displayed on an application on android with data logger that is available in the form of a graph. In a system of this using the Arduino UNO and NodeMCU as the main component in terms of controlling the water pump and delivery of data. While it's a tool that is used to detect the value of soil moisture using the sensor YL39 and sensor DHT22 as a detection value of the environmental temperature and air humidity. The output of this final project is a tool that can control and monitor the process of watering plants through an android connected to the internet.

Keyword: IoT, Watering plants, Plants.

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT karena atas rahmat dan kasih karunia-Nya penulis dapat menyelesaikan laporan Proyek Akhir ini dengan baik. Laporan Proyek Akhir ini disusun sebagai salah satu persyaratan dan kewajiban mahasiswa untuk menyelesaikan kurikulum program pendidikan Diploma III di Politeknik Manufaktur Negeri Bangka Belitung.

Dalam pembuatan alat dan makalah Proyek Akhir ini penulis mencoba menerapkan ilmu yang telah didapatkan selama 3 tahun menuntut ilmu di Politeknik Manufaktur Negeri Bangka Belitung serta pengalaman yang didapatkan selama melaksanakan Program Praktik Kerja Lapangan. Penulis mengakui bahwa selesainya Proyek Akhir ini tidak lepas dari bantuan banyak pihak yang membantu dan memberi dukungan dalam membuat alat maupun dalam menyelesaikan laporan Proyek Akhir ini. Untuk itu penulis mengucapkan terima kasih kepada :

1. Orang tua dan keluarga yang selalu memberikan kasih sayang, doa serta dukungan baik moril maupun materil.
2. Bapak Sugeng Ariyono, M.Eng, Ph.D selaku Direktur Politeknik Manufaktur Negeri Bangka Belitung.
3. Bapak Eko Sulistyono M.T, selaku Kepala Prodi D3 Teknik Elektronika dan Informatika Politeknik Negeri Bangka Belitung
4. Bapak Irwan, M.Sc, selaku pembimbing 1 yang telah memberikan waktu dan masukan serta solusi dalam penyelesaian Proyek Akhir ini baik pembuatan alat maupun makalah.
5. Bapak M. Iqbal Nugraha, M.Eng, selaku pembimbing 2 yang telah meluangkan waktu, tenaga dan pikiran dalam pembuatan alat Proyek Akhir ini.
6. Bapak Yudhi, M.T, selaku wali kelas III EA.
7. Seluruh staf pengajar dan karyawan di Politeknik Manufaktur Negeri Bangka Belitung.

8. Rekan-rekan mahasiswa tingkat akhir Politeknik Manufaktur Negeri Bangka Belitung.
9. Teman-teman yang telah ikut mendukung dan memberikan bantuan serta masukan dalam pembuatan Proyek Akhir ini.
10. Pihak-pihak lain yang telah memberikan bantuan secara langsung maupun tidak langsung yang tidak dapat disebutkan satu per satu.

Penulis menyadari sepenuhnya bahwa laporan Proyek Akhir ini masih terdapat banyak sekali kekurangan, untuk itu penulis sangat mengharapkan kritik dan saran yang membangun agar kedepannya penulis dapat lebih baik lagi. Demikian laporan ini dibuat semoga laporan ini dapat bermanfaat kedepannya dan menambah wawasan pembaca. Akhir kata penulis ucapkan terimakasih.

Sungailiat, 11 Juli 2018

Penulis

DAFTAR ISI

	Halaman
LEMBAR PENGESAHAN	ii
PERNYATAAN BUKAN PLAGIAT	iii
ABSTRAK	iv
KATA PENGANTAR	vi
DAFTAR ISI	viii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN	xiii
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah	2
1.4. Tujuan Proyek Akhir	3
BAB II LANDASAN TEORI	3
2.1. IOT	4
2.2. Akuator	5
2.2.1. Pompa Air	5
2.2.2. <i>Software</i> Aplikasi Blynk	5
2.3. Arduino.	7
2.3.1. Spesifikasi Arduino Uno	7
2.4. Sensor	7
2.4.1. Sensor DHT22	8
2.4.2. Sensor <i>Soil Moisture YL39</i>	8
2.5. NodeMCU	9
2.6. Tanaman	9
2.6.1. Tanaman Sawi	10
BAB III METODE PELAKSANAAN	12

3.1.	Pengumpulan Data	12
3.2.	Desain Sistem.....	13
3.3.	Pengembangan	14
3.4.	Uji Coba	14
BAB IV PEMBAHASAN.....		15
4.1.	Deskripsi Alat.....	15
4.1.1.	Diagram Blok	15
4.2.	Pembuatan Konstruksi	16
4.2.1	Pemilihan Material	17
4.2.2	Pembuatan Kerangka Penyiraman	18
4.3.	Pembuatan Rangkaian <i>Hardware</i>	19
4.3.1	Pemilihan Komponen <i>Hardware</i>	19
4.3.2.	Rangkaian Kontrol dan Komunikasi.....	21
4.3.3.	Pengujian Rangkaian <i>Hardware</i>	23
4.3.3.1.	Pengujian Sensor Kelembaban YL39	23
4.3.3.2.	Pengujian Sensor DHT 22.....	26
4.3.3.3.	Pengujian NodeMCU ESP8266 12E.....	29
4.4.	Pemrograman	32
4.4.1.	Program Penyalaaan Pompa Motor	33
4.4.2.	Program Pengiriman Data Serial Arduino Uno Ke NodeMCU	34
4.4.3.	Program NodeMCU	34
4.4.4.	<i>Setting</i> Aplikasi <i>Monitoring</i> Pada <i>Smart Phone</i> (Blynk).....	35
4.5.	Uji Coba Alat	42
4.5.1	Pengujian Kontrol Pompa Motor	42
4.5.2	Pengujian Data Monitor	44
BAB V KESIMPULAN DAN SARAN.....		48
5.1.	Kesimpulan	48
5.2.	Saran.....	48
DAFTAR PUSTAKA		49
LAMPIRAN		51

DAFTAR TABEL

Tabel	Halaman
4. 1 Tabel kriteria material yang digunakan.....	17
4.2 Tabel Perbandingan material.....	17
4.3 Tabel kriteria kontroler yang digunakan	19
4.4 Tabel kriteria pompa motor	20
4.5 Tabel kriteria modul WiFi	21
4.6 Tabel pengujian sensor kelembaban YL39	25
4. 7 Tabel pengujian sensor DHT22 pada nilai suhu	28
4. 8 Tabel pengujian sensor DHT22 pada nilai kelembaban	28
4. 9 Hasil uji coba penyalan pompa motor	43
4. 10 Uji coba data monitor	44
4. 11 Uji coba data monitor	46

DAFTAR GAMBAR

Gambar	Halaman
2.1 Logo Blynk <i>Software</i>	6
2.2 Sensor DHT 22.....	8
2.3 <i>Soil Moisture Sensor Module</i>	8
2.4 NodeMCU	9
4.1 Prinsip kerja alat penyiraman tanaman	15
4.2 Diagram blok kontrol penyiram tanaman	16
4.3 Konstruksi pemipaan penyiram tanaman.....	19
4.4 Skematik rangkaian kontrol	22
4.5 Skema pengujian sensor YL39.....	23
4.6 Pengukuran kelembaban tanah.....	24
4.7 Pengukuran tanah kering.....	24
4.8 Pengukuran tanah sedang	24
4.9 Pengukuran tanah basah.....	25
4.10 Skema pengujian sensor DHT22.....	26
4.11 Hasil pembacaan sensor DHT22	27
4.12 Skema pengujian NodeMCU	29
4.13 komunikasi antara Arduino Uno dan NodeMCU.....	31
4.14 <i>Flowchart</i> pemrograman Arduino Uno.....	32
4.15 <i>Flowchart</i> pemrograman NodeMCU	33
4.16 Pengaturan pada <i>display gauge</i>	36
4.17 Pengaturan pada <i>controller button</i>	37
4.18 Pengaturan pada <i>display value</i>	37
4.19 Pengaturan pada <i>notification email</i>	38
4.20 Pengaturan pada <i>display terminal</i>	39
4.21 Pengaturan pada <i>diplay super chart</i>	40
4.22 Pengaturan pada <i>display data stream</i> suhu.....	40
4.23 Pengaturan pada <i>display data stream</i> kelembaban tanah.....	41

4.24 Pengaturan pada <i>display</i> data <i>stream</i> kelembaban udara.....	41
4.25 Blok diagram uji coba penyalan pompa air.....	42
4.26 Blok diagram uji coba data monitor.....	44
4.27 Tampilan terminal	46
4.28 Tampilan grafik 1 <i>day</i>	47

DAFTAR LAMPIRAN

Lampiran 1 : Daftar Riwayat Hidup

Lampiran 2 : Program Penyalaan Pompa Motor

Lampiran 3 : Program Pengiriman Data Serial Arduino Uno ke NodeMCU

Lampiran 4 : Program NodeMCU

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Penyiraman tanaman merupakan suatu kegiatan penting yang perlu diperhatikan dalam melakukan pemeliharaan tanaman, hal ini tidak lepas dari pentingnya asupan air yang cukup yang dibutuhkan oleh tanaman. Asupan air yang cukup merupakan kebutuhan untuk hidup serta berguna untuk tumbuh dan berkembang tanaman. Proses penyiraman ini juga berguna untuk menjaga kondisi suhu di sekitar tanaman, karena suhu memiliki peran penting pada pertumbuhan dan perkembangan tanaman.

Selain suhu, penyiraman tanaman juga berpengaruh pada kelembaban tanah. Dari hasil peninjauan yang kami lakukan pada proses penyiraman tanaman pada taman biasanya dilakukan secara manual. Namun cara ini kurang efektif, karena:

- a. Memerlukan petugas rutin penyiraman
- b. Penyiraman dilakukan tanpa mempertimbangkan suhu dan kelembaban tanah sehingga memungkinkan penggunaan air yang berlebihan

Berdasarkan beberapa permasalahan dari kurang efektifnya proses penyiraman secara manual, pada proyek akhir ini kami mengangkat tema berjudul “Monitor & Kontrol Penyiram Tanaman Otomatis Berbasis IoT”. Dengan dikerjakannya proyek akhir ini, diharapkan sistem yang kami buat dapat meminimalisir ketidakefektifan dari sistem penyiraman tanaman secara manual yang selama ini dilakukan dan juga sebagai lanjutan inovasi proyek akhir dari Amrillah dan Anita yang berjudul “Penyiram Tanaman Otomatis Sebagai Pengatur Kelembaban Tanah Berbasis Mikrokontroler”. Alat tersebut menggunakan sensor LM35 yang dirancang untuk mendeteksi dan merespon kelembaban tanah berdasarkan suhu pada tanah itu sendiri. Maka dari itu dibuatlah inovasi tambahan pada alat tersebut, karena pada proyek akhir yang

dibuat oleh Amrillah dan Anita ini masih memiliki beberapa kekurangan diantaranya:

- Tidak memakai sensor khusus kelembaban tanah
- Data kelembaban dan suhu tidak dapat dimonitor menggunakan *smart phone*
- Tidak dapat menyimpan data logger
- Pengontrolan tidak dapat dilakukan secara manual

Maka dari itu dibuatlah proyek akhir ini sebagai inovasi lanjutan dari proyek akhir dari Amrillah dan Anita dengan sistem yang berbeda dengan beberapa penambahan teknologi dalam menutupi kekurangan dari proyek akhir yang dibuat sebelumnya.

1.2. Rumusan Masalah

Permasalahan pada proses pembuatan sistem monitor & kontrol penyiram tanaman otomatis berbasis IoT adalah sebagai berikut :

1. Bagaimana mendesain dan mengembangkan program *record* data suhu dan kelembaban yang dapat tersimpan pada android.
2. Bagaimana menentukan letak titik penyiraman tanaman yang tepat yang dapat menyiram tanaman secara merata dan baik.

1.3. Batasan Masalah

Dalam perancangan dan pembuatan proyek akhir ini terdapat batasan masalah antara lain :

1. Pembuatan program record data suhu & kelembaban yang dapat tersimpan pada android.
2. Penggunaan tanaman sawi sebagai objek dalam eksperimen penyiraman tanaman.

1.4. Tujuan Proyek Akhir

Tujuan akhir yang ingin dicapai dengan pembuatan proyek akhir dengan judul Monitor & Kontrol Penyiram Tanaman Otomatis Berbasis IoT adalah sebagai berikut :

1. Mendesain dan mengembangkan suatu sistem kontrol yang dapat menyimpan data *record* suhu dan kelembaban pada *smart phone*.
2. Mendeskripsikan hasil penentuan letak penyiram tanaman yang tepat agar merata dan baik.

BAB II

LANDASAN TEORI

2.1. IOT

IOT muncul dengan besarnya upaya masyarakat untuk mempermudah proses pekerjaan yang dulunya dilakukan secara langsung oleh manusia menjadi dapat memonitor data yang diinginkan yang terhubung langsung ke jaringan internet serta dengan bantuan teknologi seperti sensor, aktualisasi, jaringan sensor nirkabel, *real Time*, serta dapat dikerjakan tanpa harus datang langsung ditempat. Menurut burange & misalkar, 2015 yang dikutip oleh Apri Junaidi dalam jurnal ilmiah teknologi informasi terapan Universitas Widyatama dengan judul “Internet Of Things, Sejarah, Teknologi dan penerapannya: Review” Internet Of Things (IOT) adalah struktur dimana objek, orang disediakan dengan identitas eksklusif dan kemampuan untuk pindah data melalui jaringan tanpa memerlukan dua arah antara manusia ke manusia yaitu sumber ke tujuan atau interaksi manusia ke computer [2].

Pada tahun 2003 RFID mulai ditempatkan pada tingkat besar besaran di militer AS di Program Savi mereka. Pada tahun yang sama melihat raksasa ritel Walmart untuk menyebarkan RFID di semua toko-toko di seluruh dunia untuk lebih besar batas. Pada tahun 2005 arus publikasi utama seperti The Guardian, Amerika ilmiah dan Boston Globe mengutip banyak artikel tentang IOT. Pada tahun 2008 kelompok perusahaan meluncurkan IPSO Alliance untuk mempromosikan penggunaan Internet Protocol (IP) dalam jaringan dari "Smart object" dan untuk mengaktifkan Internet of Things.

Pada tahun 2008 FCC menyetujui penggunaan “white space spectrum”. Akhirnya peluncuran IPv6 ditahun 2011 memicu pertumbuhan besar di bidang Internet of Things, perkembangan ini didukung oleh perusahaan raksasa seperti Cisco, IBM, Ericson mengambil inisiatif banyak dari pendidikan dan komersial dengan IOT teknologi dapat hanya dijelaskan sebagai hubungan antara manusia dan komputer. Perkembangan Internet of Things, semua peralatan yang kita gunakan dalam kehidupan kita sehari hari dapat dikendalikan dan dipantau menggunakan IOT.

Mayoritas proses dilakukan dengan bantuan sensor di IOT. Sensor ini mengkonversi data fisik mentah menjadi sinyal digital dan mengirimkan mereka ke pusat kontrol. Dengan cara ini kita bisa memonitor perubahan lingkungan jarak jauh dari setiap bagian dari dunia melalui internet [2].

2.2. Akuator

Akuator yang digunakan pada alat proyek akhir ini adalah pompa air national GP 125 dan sebuah *display* pada *smart phone*. Yang mana pompa air sebagai akuator kontrol dan *display* pada *smart phone* sebagai akuator monitor.

2.2.1. Pompa Air

Pada proyek akhir ini penggunaan pompa air bertujuan agar dapat melakukan proses penyiraman secara otomatis dengan proses pengaktifannya melalui relay yang dikontrol dengan arduino uno yang telah dilakukan proses pemrograman. Proses pengontrolan ini dilakukan agar dapat mempermudah dalam proses penyiraman tanaman. Pompa air yang digunakan adalah pompa air national GP 125 yang memiliki spesifikasi sebagai berikut [2]:

- Kapasitas : 42 liter / menit
- Tinggi hisap : 9 meter
- Tinggi dorong : 24 meter
- Total hisap dan dorong : 33 meter
- Temperatur air : 35 celcius
- Diameter pipa : $\frac{3}{4}$ inci x $\frac{3}{4}$ inci
- Daya masuk : 350 watt
- Daya keluar : 125 watt
- Putaran : 2850 rpm

2.2.2. Software Aplikasi Blynk

BLYNK adalah platform untuk aplikasi OS Mobile (iOS dan Android) yang bertujuan untuk kendali module Arduino, Raspberry Pi, ESP8266, WEMOS D1, dan module sejenisnya melalui Internet. Aplikasi ini merupakan wadah

kreatifitas untuk membuat antarmuka grafis untuk proyek yang akan diimplementasikan hanya dengan metode drag and drop widget. Blynk tidak terikat pada papan atau module tertentu. Dari platform aplikasi inilah dapat mengontrol apapun dari jarak jauh, dimanapun kita berada dan waktu kapanpun. Semua itu dapat dilakukan dengan catatan terhubung dengan internet dengan koneksi yang stabil dan inilah yang dinamakan dengan sistem Internet of Things (IOT) [3].

Langkah-langkah awal penggunaan platform dari aplikasi blynk pada android sebagai berikut:

1. Download dan install aplikasi melalui “PlayStore“
2. Buka aplikasi, dan silahkan sign up new account atau login menggunakan “Facebook“
3. Buat new project, dan pilihlah salah satu module yang akan Anda gunakan maupun aksesoris module yang berfungsi sebagai sarana terhubung ke Internet
4. Setelah itu drag and drop rancangan proyek Anda
5. Kemudian klik Blynk untuk mengirimkan Token Auth melalui email
6. Dan terakhir cek inbox email Anda dan temukan Auth Token yang dimana ini akan digunakan untuk program yang di downloadkan ke module



Gambar 2.1 Logo Blynk Software

Untuk setiap akun blynk yang dibuat memiliki akun token tertentu agar setiap pengguna tidak salah dalam menyambungkan kemana kontrol akan terhubung ke akun Blynk.

2.3. Arduino

Arduino adalah *kit* elektrik atau papan rangkaian yang di dalamnya terdapat komponen utama yaitu sebuah chip mikrokontroler dengan jenis AVR dari perusahaan Atmel. Pengertian sederhanya, arduino merupakan computer kecil yang dapat diprogram untuk memproses masukan dan keluaran antar komponen eksternal yang dihubungkan pada arduino itu sendiri. Pemrograman arduino dilakukan dengan arduino *software* (IDE) yang merupakan *software* yang dirancang khusus untuk pemrograman arduino. Arduino yang digunakan adalah arduino Uno [4].

2.3.1. Spesifikasi Arduino Uno [5]

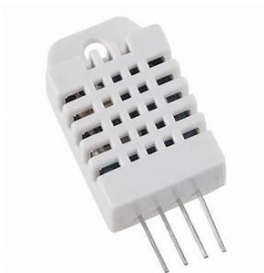
- Chip mikrokontroler : ATmega328P
- Tegangan operasi : 5 volt
- Input tegangan : 7-11 Volt
- Input tegangan batas : 6-20 Volt
- Pin I/O digital : 14 (6 PWM)
- Pin Analog : 6
- Arus DC tiap pin I/O : 50 mA
- Arus DC ketika 3,3 V : 50 mA
- Memori flash : 32 KB
- SRAM : 2 KB
- EEPROM : 1 KB
- Kecepatan clock : 16 MHz

2.4. Sensor

Sensor adalah alat yang digunakan untuk mendeteksi adanya perubahan lingkungan fisik atau kimia. Variable keluaran dari sensor yang diubah menjadi besaran listrik disebut transduser [7].

2.4.1. Sensor DHT22

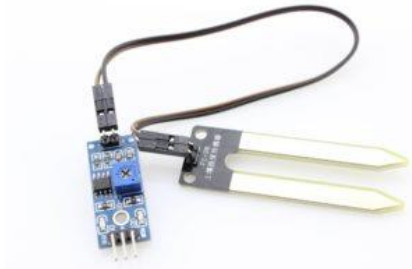
Sensor DHT 22 merupakan sensor kelembaban yang relatif dan multi sensor suhu yang terdiri dari modul yang dikalibrasi keluaran digital. Pada pengukuran suhu data yang dihasilkan 14 bit, sedangkan untuk kelembaban data yang dihasilkan 12 bit. Keluaran dari DHT-22 adalah digital sehingga untuk mengaksesnya diperlukan pemrograman dan tidak diperlukan pengkondisi sinyal atau ADC. Sensor DHT 22 dipilih karena memiliki range pengukuran yang luas yaitu 0 sampai 100% untuk kelembaban dan -40 derajat celcius sampai 125 derajat celcius untuk suhu, selain itu juga sensor DHT 22 hanya membutuhkan supply tegangan 2.4 dan 5.5 V untuk mengaktifkannya.



Gambar 2.2 Sensor DHT 22 [8]

2.4.2. Sensor *Soil Moisture* YL39

Sensor YL39 adalah sensor yang digunakan untuk mengukur kelembaban tanah dengan keluaran nilai analog yaitu 0 sampai 1023. Dalam penggunaannya sensor ini dapat dikonversikan nilainya ke dalam persen.



Gambar 2.3 *Soil Moisture* Sensor Module

2.5. NodeMCU

Ejhfajl NodeMCU bisa dibilang merupakan sebuah paket ESP8266 didalam sebuah papan yang lengkap dengan berbagai jenis fitur serta bersifat opensource. Untuk penggunaan daya sendiri NodeMCU sendiri menggunakan tegangan 3.3 v untuk dapat berfungsi. Beberapa fitur NodeMCU diantaranya [11]:

- 10 port GPIO dari D0 – D10
- Fungsionalitas PWM
- Antarmuka I2C dan SPI
- Antarmuka 1 Wire
- ADC



Gambar 2.4 NodeMCU[11]

2.6. Tanaman

Merupakan kumpulan beberapa organisme yang dibudidayakan pada suatu lahan pertanian dengan tujuan dapat dipanen pada masa ketika suatu tanaman sudah mencapai tahap pertumbuhan tertentu yang diinginkan. Pada pengertian ini sendiri dibedakan dari penggunaan secara awan bahwa tanaman juga dapat dikatakan tumbuhan. Namun dengan pengertian tanaman maka dapat tercakup didalamnya seperti fungi (jamur pangan) alga (penghasil agar-agar) yang dibudidayakan dengan tujuan dapat menambah nilai ekenomi. Selain itu menurut kelompok produknya tanaman dapat dibedakan kedalam beberapa kelompok, antara lain [1]:

1. Serealia (tanaman utama penghasil biji-bijian)
2. Kacang-kacangan
3. Tanaman buah
4. Tanaman sayur
5. Tanaman industri

6. Tanaman rempah
7. Tanaman umbi-umbian
8. Tanaman serat
9. Tanaman obat
10. Tanaman penghasil minyak atsiri

2.6.1. Tanaman Sawi

Sawi (*Brassica juncea* L.) merupakan suatu jenis sayuran daun yang tergolong dalam keluarga cruciferae atau merupakan tumbuhan berbunga yang memiliki nilai ekonomis tinggi. Tanaman sawi ini merupakan tanaman yang berasal dari Tiongkok dan Asia Timur. Untuk di Tiongkok tanaman ini telah dibudidayakan oleh masyarakat tiongkok sejak 2500 tahun lalu kemudian menyebar kewilayah Filipina dan Taiwan. Di Indonesia sawi pertama kali masuk bersamaan dengan perdagangan sayur subtropis lain pada abad XI. Tanaman sawi merupakan salah satu jenis tanaman sayur yang cukup digemari oleh semua golongan masyarakat. Berdasarkan data statistik dari BPS Sulawesi Tenggara tahun 2010 yang dikutip oleh Dedi Erawan, Wa Ode Yani dan Andi Bahrin bahwa secara nasional kemampuan produksi tanaman sawi Indonesia 8-10 ton ha. Sedangkan untuk Sulawesi tenggara produksi tanaman sawi rata-rata 3,74 ton ha dengan luas panen 165 ha.

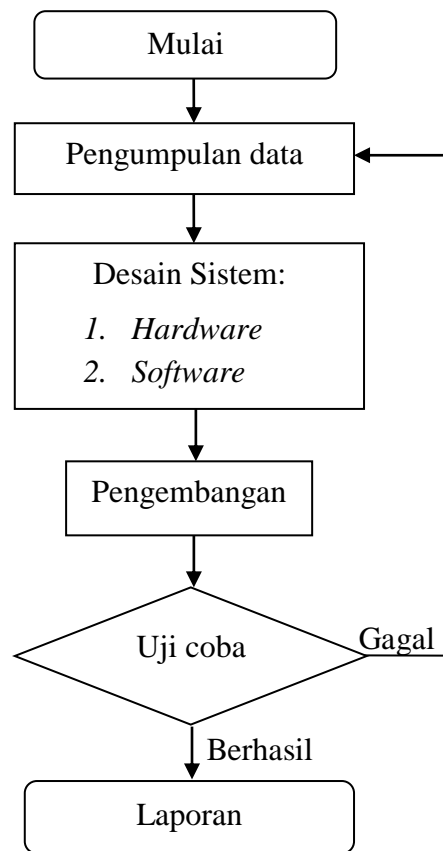
Dedi Erawan, Wa Ode Yani dan Andi Bahrin dalam Jurnal Agroteknos Maret 2013 Jurusan Agroteknologi Fakultas Pertanian Universitas Haluoleo, kendari dengan judul Pertumbuhan dan Hasil Tanaman Sawi (*Brassica Juncea* L.) pada Berbagai Dosis Pupuk Urea, yang mengutip dari pendapat Sunarjo (2005) mengatakan bahwa kondisi lingkungan yang cocok bagi pertumbuhan tanaman sawi berkisar diantaranya pada suhu 27°C - 32°C dengan kelembaban antara 60% – 87%. Selain itu dikutip juga dari pendapat Haryanto (2006) pada jurnal yang sama mengatakan bahwa kelembaban udara yang sesuai untuk pertumbuhan tanaman sawi hijau yang optimal adalah berkisar 80%-90% [1].

Rata-rata berat kering akar maupun berat kering daun tanaman sawi serta laju pertumbuhan relatif tanaman sawi tertinggi diperoleh pada perlakuan dosis

pupuk Urea 125 kg ha⁻¹. Hal ini menunjukkan bahwa dosis pupuk Urea 125 kg ha⁻¹ mampu menyuplai kebutuhan unsur hara nitrogen dalam proses pertumbuhan dan perkembangan tanaman sawi. Dimana tanaman sawi respon terhadap pemberian pupuk Urea, hal ini disebabkan karena tanaman sawi merupakan salah satu jenis tanaman sayuran hasil panen utamanya adalah daun sehingga proses pertumbuhan tanaman sawi yang harus terpenuhi suplai unsur haranya sampai pada fase vegetatif saja. Nitrogen merupakan unsur yang paling penting dalam pertumbuhan tanaman sawi karena nitrogen merupakan salah satu unsur hara esensial. Hal ini sejalan dengan pendapat Lakitan (2008) bahwa dalam jaringan tanaman, nitrogen merupakan unsur hara esensial dan unsure penyusun asam-asam amino, protein dan enzim. Selain itu, nitrogen juga terkandung dalam klorofil, hormon sitokinin dan auksin [1].

BAB III METODE PELAKSANAAN

Dalam pembuatan proyek akhir, untuk mempermudah proses pengerjaan dibuat beberapa tahapan dalam bentuk *flowchart* yang dapat dilihat pada gambar berikut ini:



3.1. Pengumpulan data

Pengumpulan data merupakan proses pencarian data referensi-referensi yang memiliki kaitan dengan data yang dibutuhkan dalam proses pengerjaan proyek akhir maupun penyusunan makalah proyek akhir ini baik melalui *internet* maupun dari buku-buku. Dalam proses pencarian referensi dilakukan dengan mencari data yang berkaitan dengan jurnal tanaman (tanaman sawi), teknik penyiraman, titik peletakan sensor, IoT. Pada tahap proses pencarian referensi ini

akan dipelajari beberapa hal-hal yang berhubungan dengan proyek akhir ini diantaranya:

1. Pencarian data referensi secara primer, yaitu melakukan diskusi dengan pembimbing dan instruktur Politeknik Manufaktur Negeri Bangka Belitung yang lain untuk memperjelas informasi yang ada dan untuk mendapatkan konsep yang akan menjadi pilihan untuk digunakan.
2. Pencarian data referensi secara sekunder, yaitu sebagai berikut:
 - a. Dari referensi-referensi buku
Mencari referensi-referensi dari buku dilakukan untuk mendapatkan rumus-rumus yang berkaitan dengan proyek akhir ini.
 - b. Dari internet
Mencari referensi dari internet dilakukan untuk mendapatkan contoh model atau bentuk dari pemrograman yang akan digunakan untuk proyek akhir ini. Namun untuk mencari referensi tersebut dapat dilakukan dengan memilih dari sumber yang terpercaya.

Setelah selesai proses pencarian data referensi kemudian dilakukan proses pengolahan data. Pengolahan data adalah tahap untuk mengolah data yang sudah dikumpulkan berupa desain, sistem control dan komponen-komponen penunjang lainnya yang akan digunakan pada pembuatan proyek akhir ini. Kemudian didiskusikan kembali bersama pembimbing untuk memperjelas informasi yang ada dan dari hasil diskusi dengan pembimbing dapat diambil kesimpulan mengenai desain sistem, teknik pengerjaan, maupun metode yang digunakan untuk menyelesaikan proyek akhir ini.

3.2. Desain Sistem

Desain sistem bertujuan agar proyek akhir dapat mencapai target yang diinginkan dan semua sistem dapat bekerja dengan baik. Proses awal yang dilakukan adalah membuat desain secara keseluruhan baik dari konstruksi, *hardware* maupun *software* yang akan digunakan agar pengerjaan proyek akhir ke tahap selanjutnya dapat berjalan lebih mudah serta teratur. Dalam proses

pembuatan *hardware* ini bertujuan untuk mendapatkan bentuk konkrit dari alat proyek akhir yang akan dibuat. *Hardware* dapat berupa rancangan hasil jadi alat proyek akhir yang mana bertujuan untuk memudahkan pelaksanaan output dari proyek akhir tersebut. Selanjutnya untuk pembuatan *software* sendiri bertujuan sebagai media penerima informasi dan pengiriman perintah. Pada hal ini menggunakan sebuah *software* yang dapat terhubung dengan sistem internet. Mengapa demikian, karena dalam unsur pembuatan proyek akhir ini menggunakan sistem internet, yang mana dari internet tersebut berfungsi untuk mengirimkan sebuah perintah serta data dan menerima sebuah informasi dengan menggunakan jaringan tersendiri.

3.3. Pengembangan

Pengembangan adalah suatu usaha untuk meningkatkan kemampuan teknis, teoritis dan konseptual sesuai dengan kebutuhan kebutuhan. Pengembangan dilakukan setelah desain sistem selesai dilakukan. Pengembangan ini berfungsi untuk memperkembang sistem pada proyek akhir yang serupa atau proyek akhir yang sudah pernah dibuat agar dapat lebih baik lagi. Pengembangan dapat dilakukan dengan penambahan-penambahan teknologi baru yang belum pernah dibuat pada proyek akhir yang akan dibuat. Selain dengan penambahan-penambahan teknologi baru, pengembangan juga dapat dilakukan dengan memodifikasi sistem lama ke sistem yang baru. Jadi pada suatu pengembangan, haruslah ada sesuatu yang baru pada proyek akhir yang akan dibuat

3.4. Uji Coba

Uji coba ini dilakukan untuk mengetahui serta mendapatkan hasil pengujian dari proyek akhir yang dibuat. Itu bertujuan agar kita dapat mengetahui bagaimana hasil *output* proyek akhir yang dibuat dan dapat dinilai apakah masih terdapat kesalahan ataupun kekurangan dalam proses kerja alat yang dibuat agar dapat segera diperbaiki. Adapun poin-poin yang diuji coba sebagai berikut:

1. Mencoba akurasi penyalan pompan air berdasarkan ketentuan yang dibuat.

BAB IV PEMBAHASAN

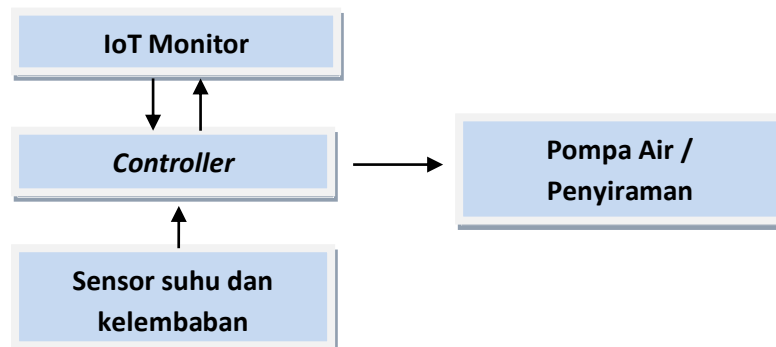
4.1. Deskripsi Alat

Penyiram tanaman otomatis berbasis IoT ini merupakan alat kontrol dan monitor yang dibuat untuk mempermudah penyiraman tanaman. Penyiraman tanaman otomatis menggunakan pompa air yang disebarkan ke seluruh permukaan tanah yang ditanami dengan pipa yang telah dibolongi. Sementara itu untuk pengontrolan dan pemonitorannya menggunakan sebuah Arduino UNO yang telah terprogram dan sebuah NodeMCU sebagai pengirim data yang akan ditampilkan pada android. Pada sistem ini menggunakan 3 buah sensor kelembaban tanah yaitu sensor YL39 yang diletakan atau ditancapkan *random* pada permukaan tanah yang akan ditanami. Dan juga pada alat ini menggunakan sensor DHT22 sebagai pendeteksi nilai suhu lingkungan.

Output pada alat ini adalah akan menghidupkan pompa air pada jam-jam tertentu yang telah ditentukan dan akan mematikan pompa dengan sendirinya jika kelembaban tanah pada tanaman sudah tercapai. Selain itu juga dapat mengetahui hasil data pada android yang telah terhubung dengan kontrol tersebut berupa status kelembaban, suhu dan kondisi tanah serta data yang tersimpan pada hari-hari sebelumnya.

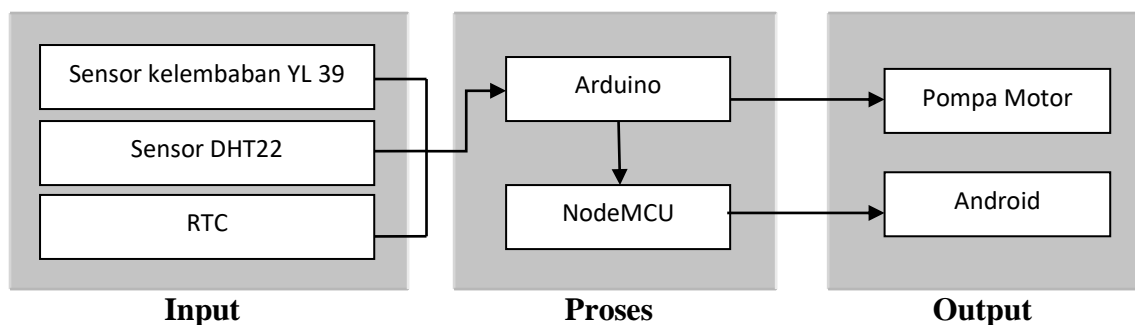
4.1.1. Diagram Blok

Gambar 4.1 menunjukkan prinsip kerja alat penyiraman otomatis secara umum.



Gambar 4.1 Prinsip kerja alat penyiraman tanaman

Pada prinsip kerja alat penyiraman tanaman ini data dapat dimonitor melalui sistem IoT yang mana data tersebut diakumulasikan melalui sebuah kontroler yang menggunakan arduino uno. Data tersebut didapatkan melalui pembacaan sensor baik sensor kelembaban tanah, kelembaban udara maupun suhu sehingga data tersebut diolah untuk menjadi syarat penyiraman tanaman. Komponen *hardware* yang digunakan telah disesuaikan dengan kebutuhan kontrol yang akan dibuat. Pada alat penyiram tanaman ini memiliki kontrol yang menggunakan sebuah NodeMCU yang difungsikan sebagai komponen pengirim data ke Android sehingga data dapat ditampilkan pada layar monitor pada sebuah *Handphone*. Gambar 4.2 menunjukkan gambar diagram blok *hardware* alat penyiram tanaman otomatis.



Gambar 4.2 Diagram blok kontrol penyiram tanaman

Berdasarkan diagram blok yang ditunjukkan pada gambar 4.1 dan gambar 4.2 Arduino akan menangkap data nilai kelembaban dan suhu sekitar tanaman yang selanjutnya akan dikirimkan ke NodeMCU yang terhubung ke jaringan *WiFi*, yang mana NodeMCU akan mengirimkan sinyal data ke android yang memiliki koneksi internet yang selanjutnya data yang diterima akan diproses dan ditampilkan pada layar *Smart Phone* melalui sebuah aplikasi blynk.

4.2. Pembuatan Konstruksi

Pada tahap pembuatan konstruksi ini, dipilihlah konstruksi alat yang dibuat sesuai dengan konsep yang telah dirancang sebelumnya agar kinerja alat penyiram

tanaman sesuai dengan apa yang diinginkan dan sesuai dengan rule yang ditetapkan agar seluruh tanaman dapat terpenuhi kebutuhan nutrisi air.

4.2.1. Pemilihan Material

Konstruksi alat penyiram tanaman dibuat dengan pertimbangan kriteria dalam pemilihan material yang diperlukan. Karena pemilihan material sangat berpengaruh terhadap kualitas hasil pengerjaannya dan juga harga yang juga menjadi pertimbangan. Selain itu, lamanya pengerjaan juga menjadi salah satu pertimbangan dari pemilihan material yang akan digunakan. Dari pertimbangan tersebut makadidapatlah criteria material yang akan digunakan dalam pembuatan konstruksi seperti pada table 4.1.

Tabel 4.1 Tabel kriteria material yang digunakan

No.	Kriteria	Alasan
1.	Kuat dan Ringan	Agar beban sama rata dengan kekuatan kontruksi
2.	Mudah dikerjakan	Agar tidak memakan waktu dalam proses pengerjaan
3.	Mudah dibentuk	Agar bentuk lebih mudah menyesuaikan
4.	Harga terjangkau	Menghemat dana tanpa menguramgi fungsi dari alat
5.	Mudah didapat	Agar proses pengerjaan tidak terhambat karena ketidak tersediaan bahan baku

Berikut beberapa material yang tersedia beserta keuntungan dan kerugiannya yang digunakan dalam pembuatan konstruksi dapat dilihat pada tabel 4.2.

Tabel 4.2 Tabel perbandingan material

No	Alternatif	Kelebihan	Kekurangan
1.	Pipa PVC	<ul style="list-style-type: none"> • Permukaan halus • Pemasangan mudah dan cepat 	<ul style="list-style-type: none"> • Memerlukan banyak perangkat tambahan dalam



- Bobot yang ringan
- Kokoh dan tahan lama
- Minim perawatan
- Instalasi pemipaan.
- Tidak lentur

2. Selang elastis



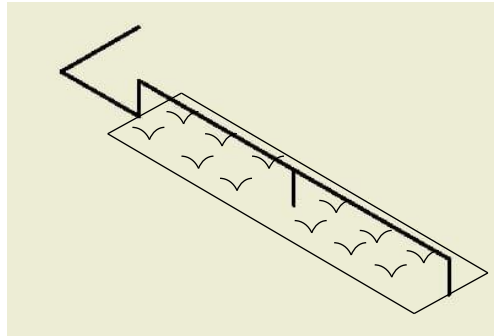
- Mudah dibentuk
- Tidak memerlukan komponen tambahan dalam instalasi
- Lebih murah dibandingkan pipa
- Tidak kokoh
- Memerlukan tempat sandaran dalam instalasi vertikal

Dari perbandingan beberapa material di atas untuk memenuhi kebutuhan seperti pada tabel 4.1 adalah pipa pvc. Pipa pvc memenuhi semua poin kriteria yang diinginkan, seperti ringan, mudah dikerjakan, mudah dibentuk, harga terjangkau, mudah didapat, serta perawatan yang mudah.

4.2.2. Pembuatan Kerangka Penyiraman

Kerangka pada bagian ini adalah bagian dari pemipaan mulai dari penyedotan sumber air sampai dengan penyiraman. Untuk mendapatkan sumber airnya didapatkan langsung dari sumur yang berada pada lingkungan Polman Babel. Alat yang digunakan untuk menyedot sumber air itu sendiri menggunakan pompa air *National GP 125* dan pipa berukuran 1 *inch* sebagai mediapenghubung air dari sumur ke pompa air. Sementara itu, untuk pipa keluaran motor menggunakan pipa berukuran 1/2 *inch*. Ukuran ini difungsikan agar keluaran air dari pompa lebih kencang. Untuk rancangan penyiramannya sendiri, alat ini tidak menggunakan *sprinkler* karena menyesuaikan dengan keadaan tanaman yang

ditanam, tetapi menggunakan pipa itu sendiri sebagai penyiram tanaman dengan cara diberikan lubang lubang kecil pada setiap sisi pipa pada jarak tertentu. Gambar 4.3 berikut menunjukkan pemipaan penyiram tanaman.



Gambar 4.3 Konstruksi pemipaan penyiram tanaman.

Pada gambar tersebut terdapat pipa saluran pembuangan air. Hal tersebut difungsikan sebagai pengendali kecepatan air jika air yang dikeluarkan terlalu besar. Dan pipa saluran pembuangan tersebut dihubungkan ke dalam tangki sumber air.

4.3. Pembuatan Rangkaian *Hardware*

Dalam pembuatan rangkaian *hardware* pertama-tama haruslah merencanakan komponen apa saja yang dibutuhkan sesuai perencanaan. Barulah setelah itu rangkaian *hardware* dapat dikerjakan.

4.3.1. Pemilihan Komponen *Hardware*

Berikut merupakan komponen hardware yang digunakan :

- Kontroler

Pemilihan kontroler tambahan harus memenuhi kriteria sebagai berikut yang ditunjukkan pada tabel 4.3 berikut.

Tabel 4.3 Tabel kriteria kontroler yang digunakan

No.	Kriteria	Alasan
1.	Berukuran sedang	Ukuran ini disesuaikan dengan kebutuhan, yaitu

		tidak terlalu besar dan juga tidak terlalu kecil.
2.	Jumlah I/O sedikit	Karena hanya digunakan untuk mengontrol sedikit output dan sedikit input.
3.	Harga terjangkau	Menghemat dana tanpa mengurangi fungsi alat

Dari tabel kriteria di atas komponen kontroler yang memenuhi semua poin kriteria yang diinginkan adalah Arduino Uno. Arduino Uno memiliki ukuran yang sedang, jumlah I/O yang tidak terlalu banyak, dan harga yang lebih murah dari kontroler lain.

- Pompa Motor

Pemilihan pompa motor untuk memompa keluaran air harus memenuhi kriteria yang ditunjukkan pada tabel 4.4 berikut.

Tabel 4.4 Tabel kriteria pompa motor

No.	Kriteria	Alasan
1.	Kapasitas yang cukup	Untuk memenuhi kapasitas kekuatan air pada tanaman.
2.	Memiliki tinggi hisap yang cukup	Agar memenuhi daya hisap yang langsung dari sumber.
3.	Memiliki tinggi dorong yang cukup	Agar dapat mendorong air sampai pada tempat tanaman di tanam.
4.	Ukuran yang pas	Agar mudah ditempatkan dimana saja.
5.	Harga terjangkau	Agar menghemat dana yang dikeluarkan

Dari tabel kriteria 4.4 di atas pompa motor yang memenuhi kriteria diatas adalah pompa motor national gp 125. Selain dengan harganya yang terjangkau dibandingkan pompa dikelasnya, pompa motor ini memiliki kekuatan yang setara dengan pompa motor yang ada dikelasnya.

- Modul *WiFi*

Pemilihan modul *WiFi* yang akan digunakan sebagai pengirim dan penerima data juga sangat diperlukan dan juga harus memenuhi kriteria yang ditunjukkan pada tabel 4.5 berikut.

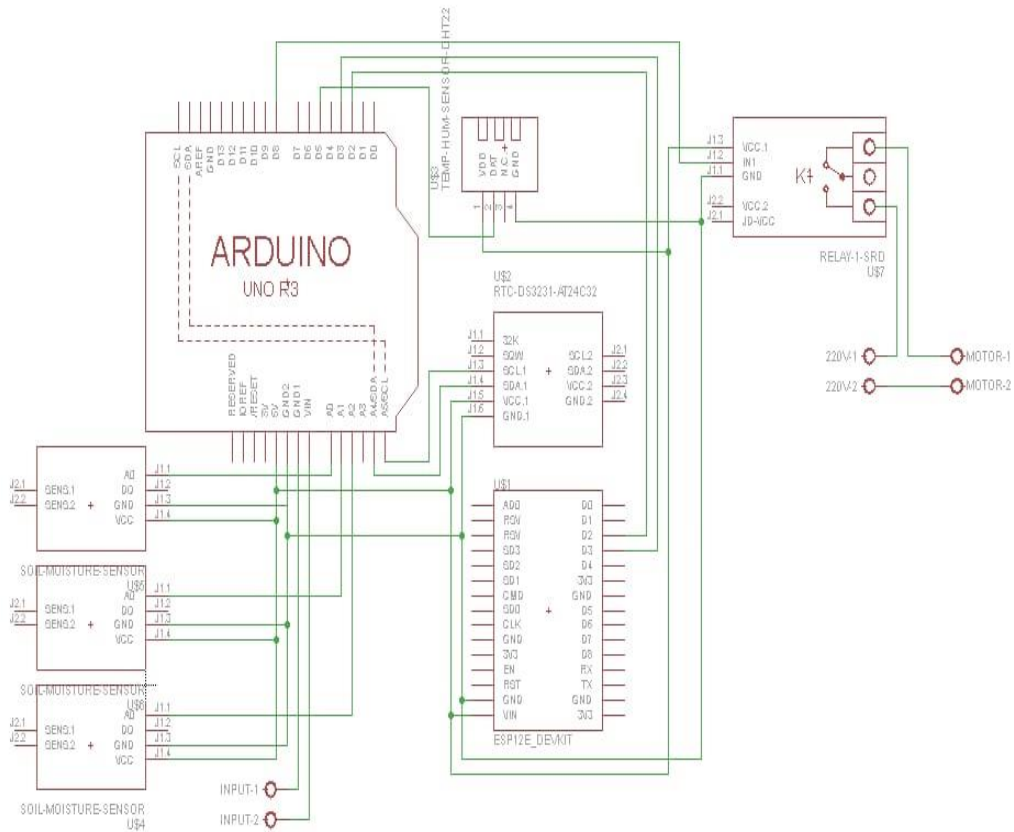
Tabel 4.5 Tabel kriteria modul *WiFi*

No.	Kriteria	Alasan
1.	Harga terjangkau	Menghemat dana yang dikeluarkan.
2.	Praktis	Mudah dalam program <i>pen-settingan</i> dan tidak memerlukan komponen tambahan.
3.	Tidak mudah <i>reset</i>	Agar tidak memerlukan pemrograman ulang jika terjadi <i>reset</i> .

Dari tabel kriteria 4.5 di atas modul *WiFi* yang memenuhi kriteria tersebut adalah NodeMCU ESP8266 atau ESP-12E. Meskipun dari segi harga lebih tinggi dibanding dengan yang lainnya, namun NodeMCU sangatlah praktis digunakan dan tidak mudah reset, tidak seperti ESP8266. Selain itu, penulis memilih NodeMCU adalah karena penulis sudah memilikinya pada sebelumnya.

4.3.2. Rangkaian Kontrol dan Komunikasi

Rangkaian kontrol dan komunikasi penyiram tanaman otomatis ini dirangkai dalam suatu *box* panel berukuran 6,5 cm x 18,5 cm x 11,5 cm dengan menggunakan *base* sebagai alas dasar dari *box*. Didalam rangkaian ini terdapat satu buah Arduino Uno untuk mengolah data nilai sensor, RTC (*Real Time Clock*), relay dan NodeMCU sebagai perantara komunikasi. Gambar 4.7 berikut adalah skematik rangkaian kontrol yang dibuat.



Gambar 4.4 Skematik rangkaian kontrol

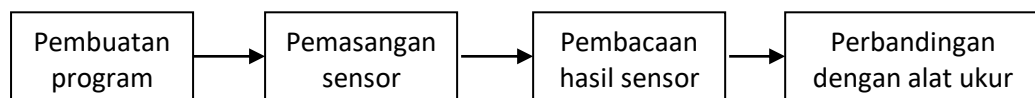
Skematik diatas menjelaskan rangkaian kontrol *hardware* yang menggunakan arduino uno. Sensor yang digunakan adalah sensor y139 sebanyak 3 buah yang masuk ke pin analog A0, A1 dan A2. Selain sensor y139, sensor DHT22 juga digunakan pada rangkaian ini yang terhubung ke pin 5 arduino. Selanjutnya pada komponen RTC (Real Time Clock) juga masuk ke dalam rangkaian kontrol, yaitu menggunakan pin SDA (Serial Data) dan SCL (Seriak Clock) yang masuk kedalam pin A4 dan A5 dengan hubungan I2C (Inter-Integrated Circuit). Pada pengontrolan motornya sendiri menggunakan relay dengan kondisi normaly open yang mana pin IN relay masuk ke pin 8 Arduino. Untuk pengiriman data ke android, kontrol ini menggunakan NodeMCU dengan hubungan serial dengan menggunakan pin D2 dan D3 NodeMCU.

4.3.3. Pengujian Rangkaian *Hardware*

Untuk hal pengujian *Hardware*, semua rangkaian haruslah selesai dibuat. Setelah itu ada beberapa aspek yang diuji untuk memastikan rangkian dan kerja sensor berjalan dengan baik. Hal-hal tersebut yang akan diuji adalah sebagai berikut:

4.3.3.1. Pengujian Sensor Kelembaban YL39

Semua sensor kelembaban dikoneksikan ke *port-port* pada rangkaian *hardware* yang mana *port* tersebut terhubung ke pin-pin Arduino. Kemudian *example* program untuk sensor kelembaban YL39 yang telah disediakan di-*download* ke dalam Arduino untuk mengecek apakah keluaran sensor sesuai dengan nilai kelembaban sebenarnya dengan membandingkan dengan alat ukur kelembaban. Berikut ini adalah skema dalam pengujian sensor YL39



Gambar 4.5 Skema pengujian sensor YL39

Berikut ini adalah *example* program yang digunakan untuk membaca nilai sensor kelembaban YL39:

```
int sensor1Pin = A0;
int sensor2Pin = A1;
int sensor3Pin = A2;

void setup ()
{
  Serial.begin(115200);
}

void loop()
{
  int nilai1Sensor = analogRead(sensor1Pin);
  int nilai2Sensor = analogRead(sensor2Pin);
  int nilai3Sensor = analogRead(sensor3Pin);

  float sensor1 = (100-(nilai1sensor*100/1023));
  float sensor2 = (100-(nilai2sensor*100/1023));
  float sensor3 = (100-(nilai3sensor*100/1023));
```

Peletakan identitas pin a0 sensor yang Akan digunakan

Setting baudrate yang akan ditampilkan pada serial monitor.

Perintah ini untuk membaca nilai analog dari pin-pin yang sudah didefinisikan

Rumus konversi nilai analog ke nilai persen

```

Serial.print(" S1 : ");
Serial.print(sensor1);
Serial.print(" S2 : ");
Serial.print(sensor2);
Serial.print(" S3 : ");
Serial.print(sensor3);
}

```

} Perintah untuk menampilkan nilai hasil sensor pada serial monitor

Dapat dilihat pemrograman pengujian sensor YL39 diatas, pemrograman tersebut difungsikan untuk membaca nilai sensor YL39. Gambar 4.6, 4.7, 4.8, 4.9 adalah gambar uji coba pengukuran sensor kelembaban terhadap alat ukur.



Gambar 4.6 Pengukuran kelembaban tanah



Gambar 4.7 Pengukuran tanah kering



Gambar 4.8 Pengukuran tanah sedang



Gambar 4.9 Pengukuran tanah basah

Setelah semua sensor kelembaban YL39 digunakan, hasil dari percobaan perbandingan dengan menggunakan alat ukur dapat dilihat pada tabel 4.6.

Tabel 4.6 Tabel pengujian sensor kelembaban YL39

Pin	Sensor	Jenis tanah	Pengukuran Sensor	Pengukuran <i>Moisture Meter</i>	Persentase <i>Error</i>
A0	Sensor 1	Kering	35 %	31 %	4 %
		Sedang	79 %	73 %	6 %
		Basah	95 %	93 %	2 %
A1	Sensor 2	Kering	35 %	30 %	5 %
		Sedang	41 %	39 %	2 %
		Basah	79%	75%	4 %
A2	Sensor 3	Kering	35%	31%	4 %
		Sedang	43%	38%	5 %
		Basah	79%	79%	0 %
Rata-rata error					3,5 %

Contoh perhitungan persentase *error* dengan mengambil beberapa sampel pada table yang didapatkan.

$$a. \text{ Persentase error sensor 1} = \left| \frac{35-31}{100} \right| \times 100\%$$

$$\text{Persentase error sensor 1} = 0,04 \times 100\%$$

$$\text{Persentase error sensor 1} = 4\%$$

$$b. \text{ Persentase error sensor 2} = \left| \frac{41-39}{100} \right| \times 100\%$$

$$\text{Persentase error sensor 2} = 0,02 \times 100\%$$

$$\text{Persentase error sensor 2} = 2\%$$

$$c. \text{ Persentase error sensor 3} = \left| \frac{79-79}{100} \right| \times 100\%$$

$$\text{Persentase error sensor 3} = 0 \times 100\%$$

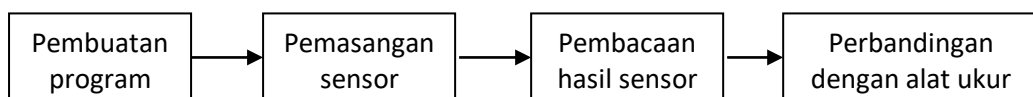
$$\text{Persentase error sensor 3} = 0\%$$

Dari tabel hasil percobaan pengukuran sensor kelembaban YL39 terhadap alat ukur, dapat dilihat bahwa terdapat beberapa sensor yang memiliki *error* lebih dari 5%. Hal ini disebabkan oleh perbedaannya sudut tanah yang di ukur karena setiap sudut tanah memiliki nilai kelembaban tanah yang berbeda.

Dalam penggunaan jumlah sensor YL39 pengujian dilakukan dengan cara mengukur kemampuan jarak bacaan sensor YL39. Pengujian ini dilakukan dengan mengukur setiap jarak penambahan 10 cm apakah kemampuan masih akurat. Maka setelah diuji didapatkan jarak baca sensor YL39 yaitu kurang lebih dengan diameter 160cm.

4.3.3.2. Pengujian sensor DHT22

Pengujian sensor suhu DHT22 difungsikan untuk mengetahui seberapa akurasi suhu yang dideteksi oleh sensor dht22 terhadap thermometer. Kemudian *example* program untuk sensor suhu yang telah disediakan di-*download* ke dalam Arduino untuk mengecek apakah keluaran sensor sesuai dengan nilai suhu lingkungan sebenarnya dengan membandingkan dengan alat ukur thermometer. Berikut ini adalah skema dari pengujian sensor DHT22:



Gambar 4.10 Skema pengujian sensor DHT22

Berikut ini adalah *example* program yang digunakan untuk membaca nilai sensor suhu:

```

#include <DHT.h>
#define DHTPIN5
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

void setup ()
{
  Serial.begin(115200);
  dht.begin();
}

void loop()
{
  delay(2000);
  h = dht.readHumidity();
  t = dht.readTemperature();
  f = dht.readTemperature(true);

  Serial.print("Humidity :");
  Serial.print(h);
  Serial.print("Temperature :");
  Serial.print(t);
  Serial.print("\n") }

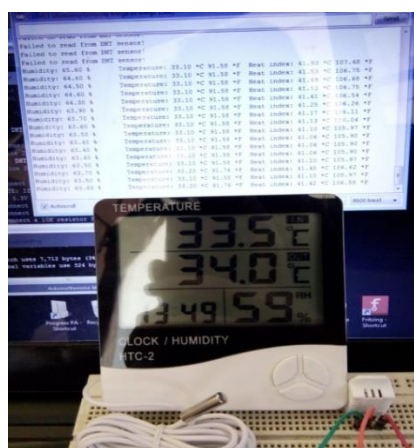
```

Mendefinisikan pin yang digunakan pada sensor dht dan tipe sensor dht yang digunakan

Membaca nilai sensor suhu dan kelembaban yang terpasang pada pin 5

Menampilkan nilai hasil bacaan sensor pada serial monitor pada software Arduino IDE.

Gambar 4.11 menunjukkan hasil pengujian sensor DHT2 yang ditampilkan pada serial monitor Arduino.



Gambar 4.11 Hasil pembacaan sensor DHT22

Dan tabel 4.7 berikut menunjukkan data yang dibaca oleh sensor DHT22 dan perbandingannya terhadap termometer.

Tabel 4.7 Tabel pengujian sensor DHT22 pada nilai suhu

Percobaan ke	Suhu DHT22 (C°)	Termometer (C°)	Error (%)
1	33,5	33,1	0,04
2	33,5	33,1	0,04
3	33,5	33,1	0,04
Rata-rata error			0,04

Tabel 4.8 Tabel pengujian sensor DHT22 pada nilai kelembaban

Percobaan ke	kelembaban DHT22 (%)	Meter Humidity (%)	Error (%)
1	59	63,70	4,7
2	59	63,50	4,5
3	59	63,60	4,6
Rata-rata error			4,6

Contoh perhitungan persentase *error* dengan mengambil beberapa sampel pada table sensor dht22 yang didapatkan.

$$d. \text{ Persentase error temperature} = \left| \frac{33,5-33,1}{100} \right| \times 100\%$$

$$\text{Persentase error temperature} = 0,004 \times 100\%$$

$$\text{Persentase error temperature} = 0,4\%$$

$$e. \text{ Persentase error humidity} = \left| \frac{59-63,7}{100} \right| \times 100\%$$

$$\text{Persentase error humidity} = 0,047 \times 100\%$$

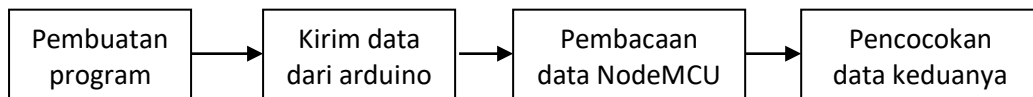
$$\text{Persentase error humidity} = 4,7\%$$

Dari tabel 4.7 dapat dilihat bahwa sensor DHT22 memiliki akurasi yang cukup untuk mendeteksi suhu lingkungan dengan akurasi 100% sementara untuk

akurasi kelembabannya sendiri memiliki error sekitar 4%, namun dengan begitu sensir ini dapat bekerja sesuai dengan semestinya.

4.3.3.3. Pengujian NodeMCU ESP8266 12E

Percobaan pengiriman data dan koneksi internet NodeMCU dilakukan dengan pengiriman nilai data sensor DHT22 dan koneksi ke *WiFi* apakah tersambung atau tidak. Dari percobaan ini, dapat dilihat apakah data yang dikirimkan sesuai dengan nilai data yang didapatkan oleh arduino. Dan dari hasil data yang diperoleh, NodeMCU dapat mengirimkan data dengan baik. Berikut ini adalah skema dari pengujian NodeMCU



Gambar 4.12 Skema pengujian NodeMCU

Berikut adalah instruksi yang digunakan pada *board* Arduino Uno dan *board* NodeMCU saat mengirimkan data serial :

```

#include <SoftwareSerial.h>
#include <DHT.h>
#define DHTPIN 5
#define DHTTYPE DHT22
SoftwareSerial ArduinoSerial(3, 2); //rx, tx
DHT dht(DHTPIN, DHTTYPE);
void setup ()
{
  Serial.begin(115200);
  ArduinoSerial.begin(4800);
  dht.begin();
}

void loop()
{
  delay(2000);
  h = dht.readHumidity();
  t = dht.readTemperature();
}
  
```

} Pendefinisian sensor yang digunakan beserta pin data. Dan juga sebagai pendefinisian pin yang digunakan untuk pengiriman data serial

} Pembacaan sensor yang akan dikirimkan

```

ArduinoSerial.print(h);
ArduinoSerial.print('e');
ArduinoSerial.print(t);
ArduinoSerial.print('f');
delay(1000);
}

```

} Data serial yang akan dikirimkan. Disini terdapat inisial, yaitu inisial e untuk mengirimkan data h, yaitu *humidity* dan inisial f untuk data t yaitu *temperature*

Pada program di atas adalah program Arduino yang digunakan untuk mengirim data serial yang akan dikirimkan ke NodeMCU. Data tersebut berupa nilai sensor DHT22 yang diserialkan ke NodeMCU melalui pin D2 dan D3 NodeMCU. Berikut ini adalah program instruksi penerimaan oleh NodeMCU:

```

#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SoftwareSerial.h>

float val, vil, vul;

char auth[] = "533ac353777e4ee3b7f83c9b51616638";
char ssid[] = "Redmi";
char pass[] = "namakamu";

BlynkTimer timer;

SoftwareSerial NodeSerial(D2, D3);

void setup()
{
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
  pinMode(D2, INPUT);
  pinMode(D3, OUTPUT);
  Serial.begin(115200);
  NodeSerial.begin(4800);
  Blynk.begin(auth, ssid, pass);
}

void loop()
{
  while (NodeSerial.available() > 0) {
    val = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'e') {
      Serial.print("Humidity: ");
      Serial.print(val);
      Serial.print("\n");
    }
    vil = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'f') {

```

Token yang digunakan untuk mengirim data ke blynk dan wifi yang digunakan beserta *password*.

Pin pada NodeMCU yang digunakan sebagai penerimaan data serial

Disini terdapat pemrograman penerimaan data yaitu jika pembacaan nodeserial diterima sesuai inisial, maka isi data dari inisial tersebut yang akan diterima NodeMCU dengan persamaan yang telah ditentukan

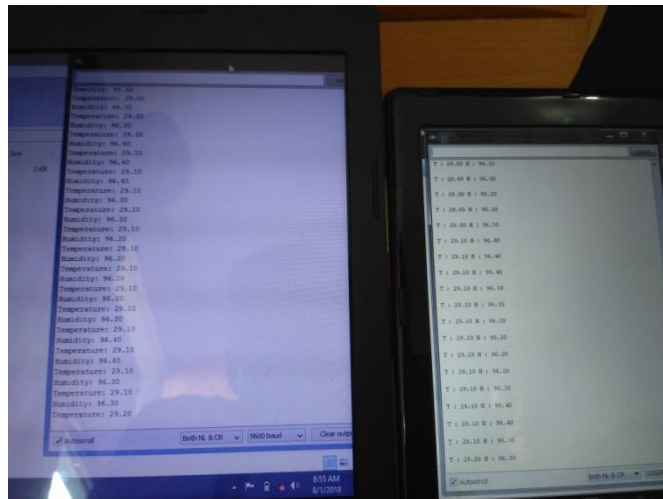
```

        Serial.print("Temperature: ");
        Serial.print(vil);
        Serial.print("\n");
    }
    Blynk.virtualWrite(V5, val);
    Blynk.virtualWrite(V6, vil);
}
delay(100);
Blynk.run();
timer.run();
}

```

Pada bagian awal program diatas pin yang digunakan oleh NodeMCU untuk menerima data adalah pin D2 dan pin D3 dengan *setting*-an baudrate serial monitor 115200 dan WiFi yang akan digunakan. Pada bagian isi program terdapat instruksi `while (NodeSerial.available() > 0)` , pada instruksi tersebut adalah setiap terjadi komunikasi serial NodeMCU maka program akan mengeksekusi perintah didalamnya. Selanjutnya terdapat perintah `NodeSerial.read()`, instruksi ini membaca data serial yang diterima dan data tersebut akan di simpan di *variable* yang telah ditentukan.

Gambar 4.13 menunjukkan hasil pengujian komunikasi antara Arduino Uno dan NodeMCU yang ditampilkan pada serial monitor Arduino.

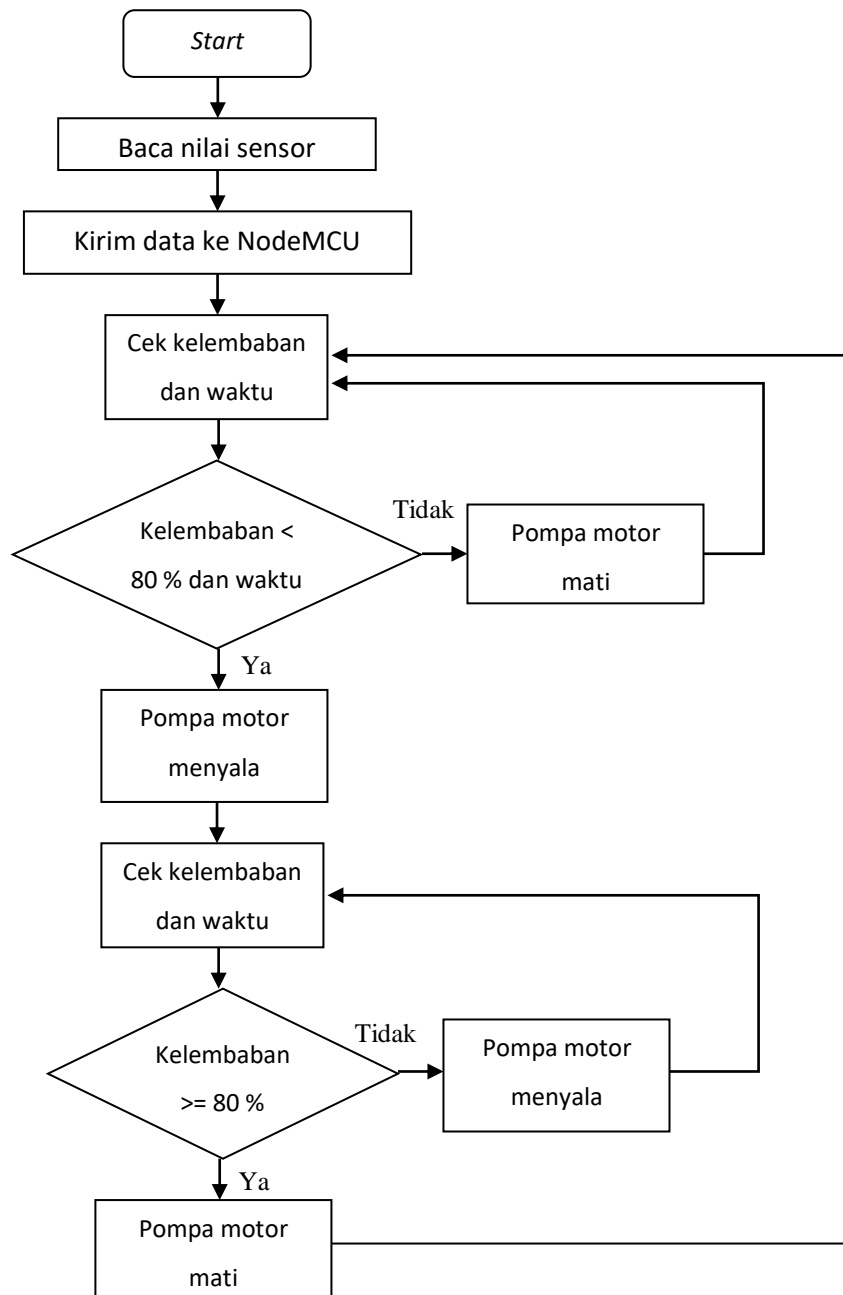


Gambar 4.13 komunikasi antara Arduino Uno dan NodeMCU

Dari hasil data di atas dapat dilihat bahwa NodeMCU menampilkan data suhu dan kelembaban yang sama dengan data yang dikirimkan, hal ini dapat disimpulkan bahwa rangkaian *hardware* berfungsi dengan baik.

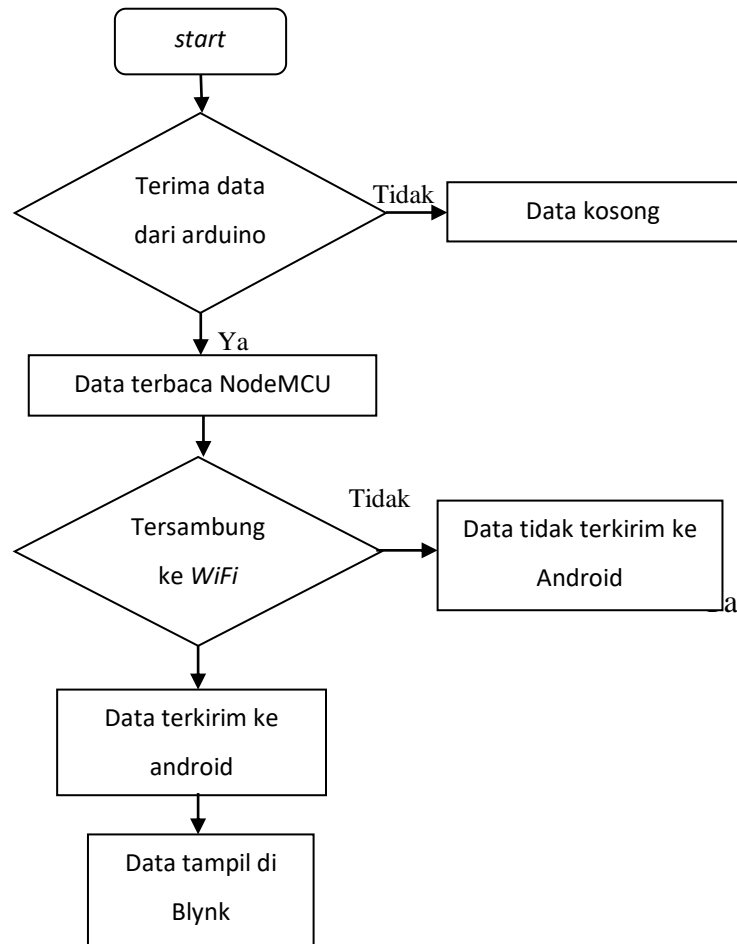
4.4. Pemrograman

Pada pemrograman penyiraman tanaman ini terdapat dua pemrograman yang digunakan yaitu pemrograman Arduino Uno dan pemrograman NodeMCU. Berikut ini penjabaran tentang pemrograman alat penyiram tanaman otomatis. Berikut merupakan *flowchart* pemrograman Arduino Uno.



Gambar 4.14 *Flowchart* pemrograman Arduino Uno

Berikut merupakan *flowchart* pemrograman NodeMCU.



Gambar 4.15 *Flowchart* pemrograman NodeMCU

Pada *flowchart* NodeMCU data didapatkan melalui pengiriman data serial dari arduino. Kemudian jika NodeMCU terhubung dengan internet maka NodeMCU akan mengirim data ke android yang akan tampil pada *software* yang telah didesain.

4.4.1. Program Penyalan pompa motor

Pemrograman utama ini terdapat pada Arduino yaitu menangkap nilai sensor, pembacaan rtc dan pengontrolan motor. Pembacaan nilai sensor ini digunakan untuk memenuhi kriteria untuk menyalakan pompa motor pada jam-jam yang telah ditentukan. Dalam program ini juga terdapat pemrograman fungsi-

fungsi lain seperti fungsi pengiriman data serial, konversi nilai rata-rata sensor ke persen dan lain-lain. Pemrograman yang digunakan untuk menyalakan pompa motor dapat dilihat pada daftar lampiran 2.

Pada program penyalakan pompa motor untuk penyiraman tanaman diatas dapat dijelaskan bahwa dalam persyaratan untuk menyiram tanaman diatur dengan waktu yang tertera yaitu jam 6.30 dan jam 17.00. Selain itu untuk persyaratan untuk penyalakan pompa penyiraman tanaman menggunakan nilai kelembaban dibawah 80% dengan mengkalkulasikan nilai rata-rata ketiga sensor kelembaban yang digunakan. Jika nilai kelembaban tanah dibawah 80 % maka pompa akan menyala, dan jika nilai kelembaban tanah diatas 80% maka pompa tidak menyala.

4.4.2. Program Pengiriman Data Serial Arduino Uno ke NodeMCU.

Pemrograman ini difungsikan untuk mengirimkan data apa saja yang akan dibaca oleh NodeMCU yang mana akan dikirimkan ke *Smart Phone*. Pemrograman ini bertujuan untuk menentukan nilai apa saja yang akan di-*monitoring* . Pemrograman data serial arduino dapat dilihat pada daftar lampiran 3

Dari program tersebut dapat dilihat bahwa *coding* dengan instruksi `ArduinoSerial.print()`; adalah intruksi yang digunakan dalam mengirim data ke NodeMCU. Dapat dilihat pada program tersebut bahwa data yang dikirimkan adalah data nilai kelembaban udara, kelembaban tanah, suhu lingkungan serta waktu dan tanggal yang terprogram oleh rtc. Untuk pemrograman Arduino secara keseluruhan maka dapat dengan menggabungkan program pengiriman data serial oleh Arduino Uno beserta program penyalakan pompa motor penyiram tanaman.

4.4.3. Program NodeMCU.

Data yang dikirimkan oleh Arduino akan diterima oleh NodeMCU yang selanjutnya akan diolah oleh NodeMCU untuk dikirimkan ke aplikasi Blynk pada *Smartphone* agar dapat di-*monitoring* oleh pembaca. Proses penerimaan data yang

dikirimkan oleh Arduino dapat dilakukan dengan proses *sampling* berikut pada lampiran 4.

Program tersebut merupakan program *sampling* penerimaan data serial dari Arduino. Data yang telah diterima kemudian akan dikirimkan ke *Smartphone*. Saat data diterima oleh NodeMCU maka data tersebut akan dijadikan acuan dalam menentukan data yang akan dikirimkan ke *Smart Phone*. Pada instruksi `char auth[] = "533ac353777e4ee3b7f83c9b51616638";` adalah instruksi untuk meletakkan token akun Blynk agar tersambung pada akun yang kita buat. Dan juga disusul dengan ID dan *password* WiFi yang akan kita sambungkan. Kemudian untuk pengiriman data yang akan tampil pada aplikasi Blynk adalah dengan menambahkan instruksi `Blynk.virtualWrite(V5, val);`. Variable V5 adalah pin *input* dengan tipe virtual yaitu virtual V5, jadi data yang dikirimkan akan masuk ke dalam komponen dengan identitas pin virtual V5. Sementara itu untuk pengiriman data berupa karakter maka menggunakan perintah instruksi `WidgetTerminal terminal (V8);` dan `BLYNK_WRITE(V8);` kemudian tuliskan karakter yang kita ingin kirimkan dengan instruksi `terminal.print(val);`.

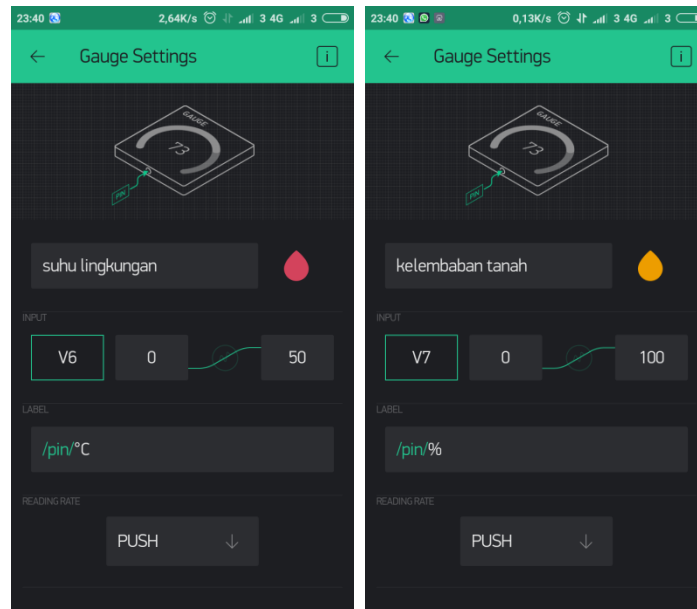
4.4.4. Setting Aplikasi Monitoring Pada Smart Phone (Blynk)

Untuk mendapatkan aplikasi Blynk dapat dengan mengunduh di *play store* pada *handphone* android dan dapat mendaftarkan dengan menggunakan akun *facebook* atau email. Setelah kita mendaftarkan, maka kita akan mendapatkan token yang akan dikirimkan ke email yang kita daftarkan, selain itu kita akan mendapatkan *power* sebanyak 2000 *power* secara gratis yang berfungsi untuk membeli komponen yang akan kita gunakan. Berikut ini adalah *setting-an* monitor pada aplikasi Blynk:

- Setting komponen Gauge

Pada *setting-an* komponen *monitor gauge* dapat mengisi dengan identitas *tittle* dan *input*. Disini data yang di monitor menggunakan *gauge* adalah suhu lingkungan dan kelembaban tanah. Jadi isi *tittle* dengan yang kita inginkan. Sementara itu untuk mengisi nilai *input* kita menggunakan pin virtual Vx dengan

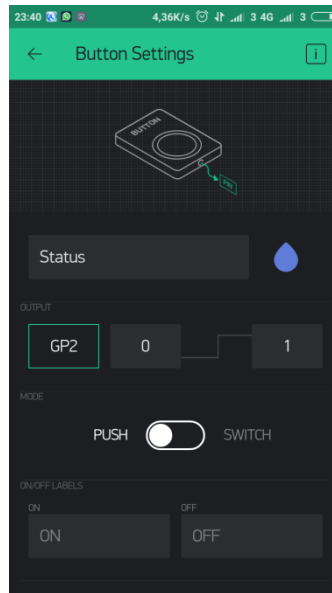
mengisi nilai minimal dan maksimal pada komponen *gauge*. Berikut ini adalah gambar *setting*-an pada koomponen *gauge* suhu lingkungan.



Gambar 4.16 Pengaturan pada *display gauge*

- *Setting* komponen *button*

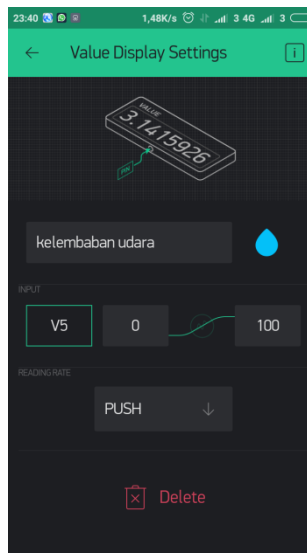
Untuk pengaturan komponen *button* yang perlu diatur adalah *output* dari *push button*. Yaitu dengan memilih pin digital dengan pin gpX, dan nilai minimal dan maksimal yang kita inginkan. Berikut ini adalah gambar pengaturan *button*.



Gambar 4.17 Pengaturan pada *controller button*

- *Value Display Setting.*

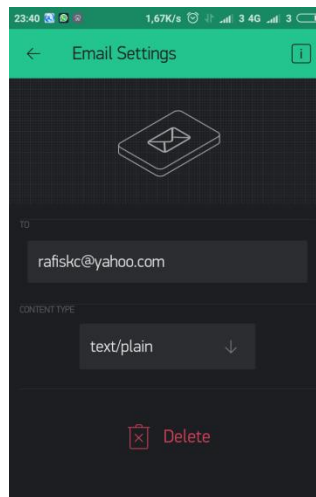
Pada bagian ini hampir sama dengan *setting* pada komponen *gauge*, yaitu dengan menambahkan pin input yang digunakan dengan pin virtual VX. Berikut ini adalah *setting-an value display*



Gambar 4.18 Pengaturan pada *display value*

- Email *setting*.

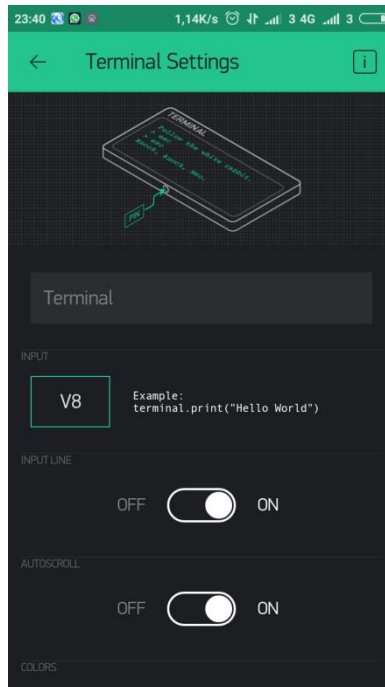
Berbeda dengan *setting* komponen lain, email hanya membutuhkan nama email yang akan kita kirimkan data. Dengan begitu data yang diinginkan akan terkirim pada email yang kita tulis. Berikut ini adalah *setting* email pada Blynk.



Gambar 4.19 Pengaturan pada *notification* email

- *Terminal Setting*.

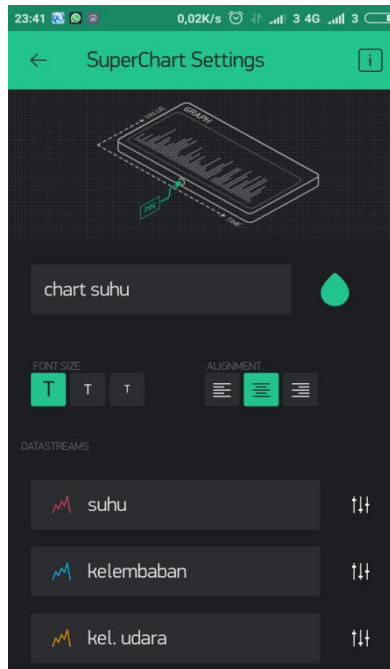
Terminal adalah komponen *display* yang digunakan untuk menampilkan data berupa tulisan, yang mana *display* yang akan ditampilkan adalah status dari bacaan sensor berupa suhu, kelembaban dan status tanah. Dalam pengaturannya cukup dengan mengatur input dengan pin virtual VX serta *input line* dan *auto scroll* apakah perlu diaktifkan atau tidak. Berikut ini adalah tampilan dari pengaturan terminal



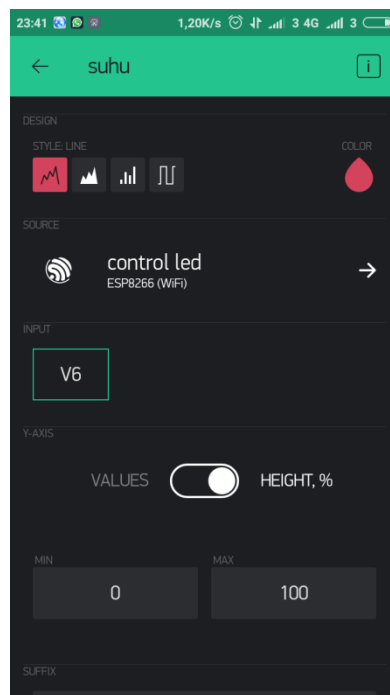
Gambar 4.20 Pengaturan pada *display terminal*

- *Supper Chart Setting*

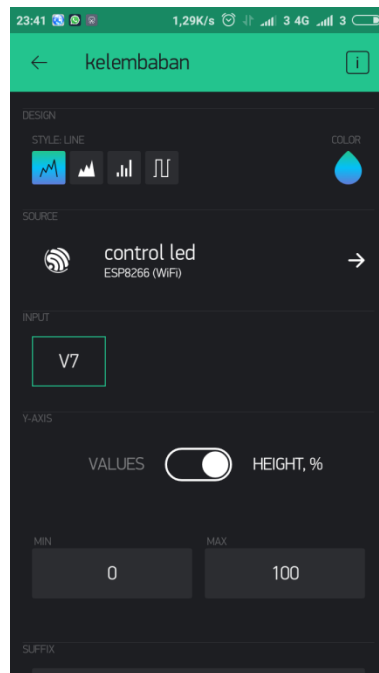
Pada salah satu komponen *display* ini adalah berfungsi untuk menampilkan rekaman data selama proses pengambilan data berlangsung. *Setting*-nya cukup sederhana yaitu dengan menambahkan dan meng-*input*-kan *data stream* yang akan kita gunakan. Pada gambar dibawah ini menggunakan 3 *data stream* yaitu suhu, kelembaban udara dan kelembaban tanah. Masing-masing *data stream* dapat diatur dengan jenis grafik yang diinginkan. Sementara itu untuk acuan waktu rekaman pada grafik sudah ditentukan dalam komponen *display* ini,yaitu live,per 1 jam, per 6 jam, per 1 hari, per 1 minggu, per 1 bulan dan per 3 bulan. Berikut ini adalah gambar pengaturann *super chart* yang kita gunakan.



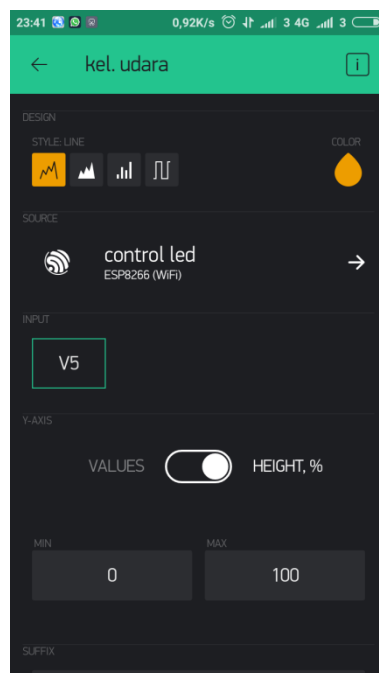
Gambar 4.21 Pengaturan pada *diplay super chart*



Gambar 4.22 Pengaturan pada *display data stream suhu*



Gambar 4.23 Pengaturan pada *display data stream* kelembaban tanah



Gambar 4.24 Pengaturan pada *display data stream* kelembaban udara

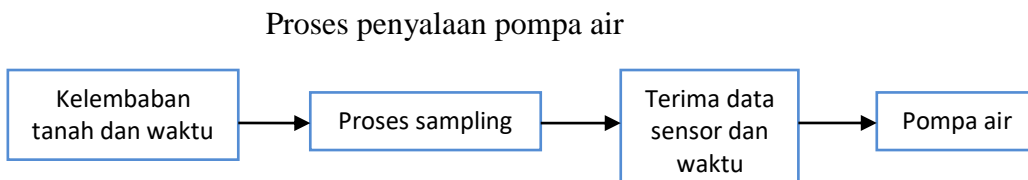
Seperti yang tampak pada pengaturan data *stream, input* nilai yang akan dikirim sesuai dengan virtual data yang dikirimkan pada aplikasi blynk, yaitu suhu V6, kelembaban tanah V7 dan kelembaban udara V5.

4.5. Uji Coba Alat

Pada bagian pengujian alat, dapat dilakukan setelah semua kontrol dan kontruksi selesai dan sudah terpasang pada kebun yang akan diuji. Pengujian dilakukan adalah sebagai berikut.

4.5.1. Pengujian kontrol Pompa Motor

Uji coba yang dilakukan adalah dengan menguji pompa motor apakah akan menyala ketika waktu dan kelembaban yang sudah ditetapkan memenuhi syarat dan akan kembali padam ketika kelembaban yang ditentukan sudah terpenuhi. Gambar 4.25 berikut merupakan blok diagram uji coba penyalan pompa motor.



Gambar 4.25 Blok diagram uji coba penyalan pompa air

Uji coba dilakukan dengan memeberikan nilai sensor dan waktu yang berbeda beda dengan variasi tertentu. Variasi ini dilakukan untuk menganalisa apakah pompa air bekerja sesuai dengan syarat yang telah ditentukan. Pada uji coba kali ini, ditetapkan bahwasanya waktu penyiraman jam 6.30 dan 17.00 dengan rentan waktu 3 menit dan dengan syarat kelembaban $\leq 80\%$ dan pompa air kembali mati ketika kelembaban menyentuh angka 80% . Setelah dilakukan uji coba, tabel 4.9 berikut merupakan hasil dari uji coba.

Tabel 4.9 Hasil uji coba penyalaaan pompa motor

Kelembaban tanah	Waktu	Kondisi pompa air
49 %	17.00	Nyala
55 %	16.30	Padam
82 %	17.00	Padam
70 %	06.30	Nyala
81 %	06.33	Padam
81 %	17.02	Padam
53 %	13.30	Padam
78 %	06.30	Nyala
79 %	17.00	Nyala
60 %	20.30	Padam
67 %	09.30	Padam

Dari tabel 4.9 dan gambar pengujian pompa air di atas didapat bahwa pompa tidak akan menyala jika persyaratan yang sudah diberikan tidak terpenuhi. Sebaliknya, pompa akan menyala jika persyaratan yang sudah ditentukan terpenuhi.

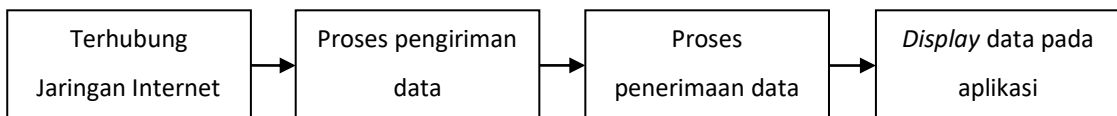
Dalam pengujian ini salah satunya dilakukan penyiraman manual ketika mendekati jam penyiraman. Contohnya, ketika pukul 16.55 tanaman disiram manual dengan menggunakan tenaga manusia sampai melampaui batas kelembaban yang telah ditentukan yaitu 80%. Kemudian ketika pukul 17.00 yang mana waktu tersebut adalah waktu untuk penyalaaan pompa air , kelembaban masih diatas persyaratan yang ditentukan. Maka kondisi pompa air yang sedang di uji adalah padam. Selain itu pengujian yang sama dilakukan pada waktu pagi hari, yaitu waktu yang mendekati dengan jam-jam penyiraman. Maka hasil yang didapatpun serupa dengan pengujian yang sudah dilakukan sebelumnya.

Dari pengujian penyalaaan pompa air tersebut, dapat disimpulkan bahwa pompa air bekerja sesuai dengan smestinya. Namun yang perlu dipertimbangkan

adalah dalam menentukan peletakan sensor-sensor kelembaban YL39, karena peletakan ini sangat berpengaruh pada program kerja penyalaan pompa air. Maka dari itu, peletakan sensor diletakan secara *random* dengan mempertimbangkan posisi yang pas agar posisi tanah yang terukur dapat mewakili posisi dan kelembaban tanah disekitarnya.

4.5.2. Pengujian data *monitor*.

pada pengujian data monitor ini dilakukan dengan cara memastikan apakah data yang dikirimkan terkirim dan tampil pada aplikasi android yang telah digunakan yaitu Blynk. Selain itu dalam pengujian data monitor juga perlu mencocokkan apakah data yang dikirimkan sama dengan data yang diterima guna memastikan kesesuaian terhadap data yang dikirimkan. . Gambar 4.23 berikut merupakan blok diagram uji coba data monitor.





Gambar 4.26 Blok diagram uji coba data monitor

Proses awal dimulai dengan terhubungnya sistem dengan jaringan internet kemudian dilakukan proses pengiriman data, setelah itu proses penerimaan data yang kemudian ditampilkan pada aplikasi pada android yang kita gunakan.

- Uji coba *value display*

Tabel 4.10 Uji coba data monitor

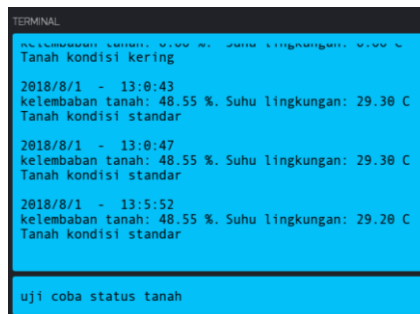
NO	Data terkirim	Display pada android
1	Suhu : 29,2 Kelembaban tanah : 48,66 Kelembaban udara : 95,3	

<p>2 Suhu : 29,2 Kelembaban tanah : 48,66 Kelembaban udara : 99,9</p>	
<p>3 Suhu : 30,2 Kelembaban tanah : 48,66 Kelembaban udara : 99,9</p>	
<p>4 Suhu : 30,9 Kelembaban tanah : 48,77 Kelembaban udara : 99,9</p>	

Dari tabel tersebut dapat disimpulkan bahwa dalam pengiriman data terkirim sesuai dengan data yang tampil pada *display* aplikasi Blynk. Adapun waktu pengiriman data yang berhasil ditampilkan adalah < 1 detik, dan perubahan data dapat berubah-ubah langsung sesuai kondisi pada sebenarnya atau terjadi secara *real time*.

- Uji coba status pada terminal

Pada pengujian ini dilakukan untuk mengetahui status kondisi alat dengan mengirimkan data berupa tulisan diantaranya data suhu lingkungan, kelembaban tanah dan status tanah. Selain itu, data yang tampil juga harus sesuai dengan sensor yang terukur. Dalam pengujian ini akan melibatkan satu buah tombol yang berfungsi untuk menampilkan status yang terjadi pada saat tombol itu ditekan. Berikut ini adalah hasil dari uji coba dari status data pada terminal:



Gambar 4.27 Tampilan terminal

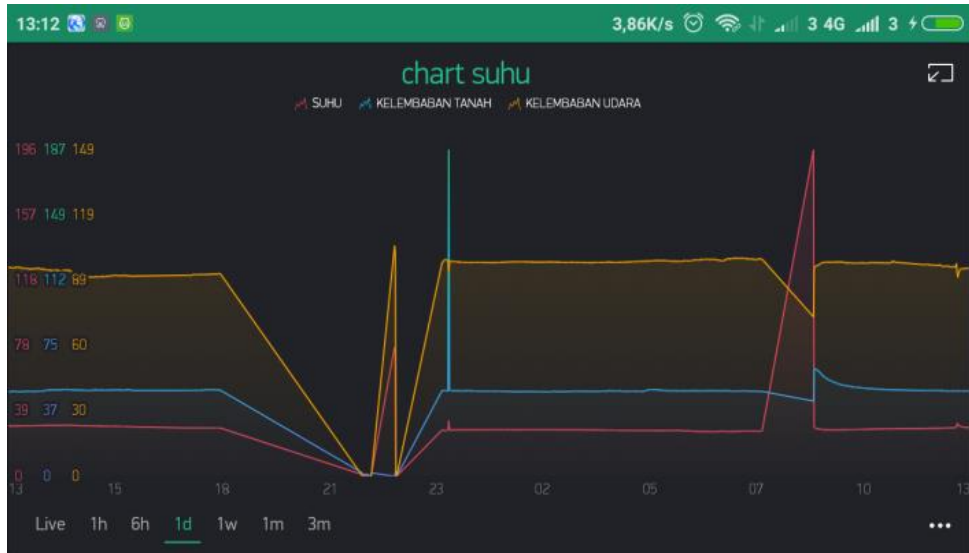
Dapat dilihat pada gambar *display* diatas, bahwa status dari suhu, kelembaban tanah dan kondisi tanah tampil dengan baik pada *display* terminal.

- Uji coba data *logger*

Data *logger* ini berfungsi untuk menyimpan semua data rekaman yang berisikan suhu lingkungan, kelembaban tanah dan kelembaban udara. Dengan menyimpan data tersebut, maka pengguna dapat melihat data pada hari-hari atau jam-jam sebelumnya. Pada uji coba data *logger* ini memanfaatkan penggunaan grafik pada aplikasi Blynk agar mempermudah *user* dalam melihat data *logger*. Berikut ini adalah gambar hasil dari uji coba data *logger*. Berikut ini adalah table dari hasil data *logger* yang terekam dalam 1 hari.

Tabel 4.11 Uji coba data monitor

jam	Kelembaban tanah	Kelembaban udara	suhu
15.00	52 %	90 %	33° C
18.00	55 %	92 %	31° C
21.00	18 %	30 %	28° C
23.00	55 %	95 %	30° C
02.00	55 %	95 %	30° C
05.00	55 %	95 %	30° C
07.00	60 %	97 %	30° C



Gambar 4.28 Tampilan grafik 1 day

Dapat dilihat pada grafik yang terekam pada gambar-gambar diatas, data logger yang dibuat sudah dapat menyimpan data dengan baik. Baik pada tampilan *live, 1 hour, 6 hours, 1 day, 1 week, 1 month* dan *3 month*.

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil pengujian dan analisa fungsi terhadap proyek akhir kami dengan judul “Monitor & Kontrol Penyiraman Tanaman Otomatis Berbasis IoT” ini maka dapat ditarik kesimpulan sebagai berikut :

1. Dari hasil uji coba yang kami lakukan, alat penyiram tanaman ini dapat memonitor suhu, kelembaban tanah dan kelembaban udara dengan error pembacaan sensor 1 %.
2. Alat proyek akhir penyiram tanaman ini dapat menyiram tanaman pada jam 07.00 pagi dan jam 16.30 sore dengan syarat kelembaban dibawah 70 %.
3. Data monitoring dapat dilihat apabila kedua perangkat tersambung jaringan internet, namun dalam pengontrolannya tetap berjalan meskipun tidak terhubung ke jaringan internet.

5.2. Saran

Dari keseluruhan proyek akhir yang dikerjakan ini terdapat beberapa saran untuk kedepannya dalam pembuatan proyek akhir ini, antara lain :

1. Untuk mendapatkan nilai keakurasian pembacaan sensor, peletakan sensor dianjurkan dengan jarak 160 cm.
2. Untuk penggunaan dalam skala besar, katup valve disarankan untuk digunakan pada pipa agar dapat mengatur lak mana yang sudah terpenuhi nutrisi air.
3. Penggunaan alat pada tanaman lain diharuskan untuk merubah syarat kelembaban dari nilai 70 % ke nilai yang lebih cocok sesuai kebutuhan tanaman yang digunakan

DAFTAR PUSTAKA

- [1]. Dedi Erawan, Wa Ode Yani, Andi Bahrin, "Pertumbuhan dan Hasil Tanaman Sawi (*Brassica juncea* L.) pada Berbagai Dosis Pupuk Urea", *Jurnal Agroteknologi*, Vol.3 No.1 Hal 19 – 25, 2013.
- [2]. IoT, [Online], diakses pada 8 agustus 2018, Available: <http://jitter.widyatama.ac.id/index.php/jitter/article/viewFile/51/35>
- [3]. Blynk, [Online], diakses pada 25 juli 2018, Available: <https://www.nyebarilmu.com/mengenal-aplikasi-blynk-untuk-fungsi-iot/>
- [4]. Robotis, Controller CM-530[Online], diakses pada 18 April 2017, Available:<http://support.robotis.com/en/product/controller/cm530>
- [5] Arduino, Arduino Uno [Online], diakses pada 25 juli 2018, Available: <http://ecadio.com/beda-dan-perbandingan-arduino-uno-dengan-nano-pro-mini-mega-2560>
- [6] Arduino, Arduino Uno [Online], diakses pada 25 juli 2018, Available: <https://uae.souq.com/ae-en/arduino-uno-r3-6186780/i/>
- [7] Tneutron, Pengertian Sensor[Online], diakses pada 2 Agustus 2018, Available:<https://www.tneutron.net/mikro/pengertian-sensor/>
- [8] Anodas, Temperature Humidity Sensor [Online], diakses pada 2 Agustus 2018, Available: <https://anodas.lt/en/dht22-temperature-humidity-sensor>
- [9] Nexiot, Soil Moisture Sensor [Online], diakses pada 2 Agustus 2018, Available: <https://nexiot.com/product/soil-moisture-sensor/>

- [10] Embeddednesia, Mengenal NodeMCU [Online], diakses pada 2 Agustus 2018, Available: <https://embeddednesia.com/v1/?p=2050>
- [11] Embeddednesia, Mengenal NodeMCU [Online], diakses pada 2 Agustus 2018, Available: <https://embeddednesia.com/v1/?p=2050>

DAFTAR RIWAYAT HIDUP



1. Data Pribadi

Nama lengkap : Rafi Rahmadi
Tempat dan tanggal lahir : Koba, 18 Agustus 1997
Alamat rumah : Jl. Raya Arung Dalam, Kec. Koba, Bangka
Tengah, Bangka Belitung.
Telp : -
Hp : 081368444575
Email : rafiskc@yahoo.com
Jenis kelamin : Laki-laki
Agama : Islam

2. Riwayat Pendidikan

SD N 1 Koba 2003-2009
SMP N 1 Koba 2009-2012
SMK N 1 Koba 2012-2015

3. Pendidikan Non Formal

-

Sungailiat, 1 Agustus 2017

A handwritten signature in blue ink, appearing to be 'Rafi Rahmadi', written over a white background.

Rafi Rahmadi

DAFTAR RIWAYAT HIDUP



4. Data Pribadi

Nama lengkap : Ranggi
Tempat dan tanggal lahir : Baturusa, 30 Juli 1997
Alamat rumah : Jl. Perkuburan, Desa Baturusa, Kecamatan Merawang, Kabupaten Bangka, Provinsi Bangka Belitung

Telp : -
Hp : 0895605512515
Email : ranggiismail@gmail.com

Jenis kelamin : laki-laki
Agama : islam

5. Riwayat Pendidikan

SD N 1	Baturusa Bangka	2003-2009
SMP N 1	Merawang Bangka	2009-2012
SMK N 1	Merawang Bangka	2012-2015

6. Pendidikan Non Formal

-

Sungailiat, 1 Agustus 2018

Ranggi

LAMPIRAN 2

PROGRAM PENYALAN POMPA

MOTOR

```

#include <SoftwareSerial.h>
#include <Wire.h>
#include "RTClib.h"
#define RELAY_ON 0
#define RELAY_OFF 1
#define RELAY_1 9
RTC_DS3231 rtc;

int sensor1Pin = A0;
int sensor2Pin = A1;
int sensor3Pin = A2;

void soil()
{
S1 = analogRead(A0);
S2 = analogRead(A1);
S3 = analogRead(A2);
}

void setup ()
{
Serial.begin(115200);
Wire.begin();
rtc.begin();
pinMode (RELAY_1, OUTPUT);
pinMode (RELAY_1, OUTPUT);
digitalWrite (RELAY_1, RELAY_OFF);
}

void loop()
int nilai1Sensor = analogRead(sensor1Pin);
int nilai2Sensor = analogRead(sensor2Pin);
int nilai3Sensor = analogRead(sensor3Pin);

DateTime now = rtc.now();
float rata=((nilai1Sensor + nilai2Sensor + nilai3Sensor)/3);
float persen=((100-((rata*100)/1023))/100)*100;

if(now.hour()>=6 && now.minute() >= 30 && persen <=80)
{
digitalWrite (RELAY_1, RELAY_ON);
Serial.print("pagi rafi, tanaman sedang disiram");
}

```

```
Serial.print("\n");
}

if(now.hour()>=6 && now.minute() >= 35 && persen >=80 )
{
digitalWrite(RELAY_1, RELAY_OFF);
Serial.print("pompa mati");
Serial.print("\n");
}

if(now.hour()>=17 && now.minute() >= 00 && persen <=80)
{
digitalWrite (RELAY_1, RELAY_ON);
Serial.print("sore rafi, tanaman sedang disiram");
Serial.print("\n");
}

if(now.hour()>=17 && now.minute() >= 5 && persen >=80 )
{
digitalWrite(RELAY_1, RELAY_OFF);
Serial.print("pompa mati");
Serial.print("\n");
}

}
```

LAMPIRAN 3

PROGRAM PENGIRIMAN DATA SERIAL

ARDUINO UNO KE NODEMCU

```

#include <SoftwareSerial.h>
#include <Wire.h>
#include "RTClib.h"
#include <DHT.h>
#define DHTPIN 5
#define DHTTYPE DHT22
SoftwareSerial ArduinoSerial(3, 2); //rx, tx
DHT dht(DHTPIN, DHTTYPE);
RTC_DS3231 rtc;

int sensor1Pin = A0;
int sensor2Pin = A1;
int sensor3Pin = A2;
float S1, S2, S3, h, t;

void soil()
{
S1 = analogRead(A0);
S2 = analogRead(A1);
S3 = analogRead(A2);
}

void setup ()
{
Serial.begin(115200);
ArduinoSerial.begin(4800);
dht.begin();
Wire.begin();
rtc.begin();
}

void loop()
{
delay(2000);
h = dht.readHumidity();
t = dht.readTemperature();
float f = dht.readTemperature(true);

int nilai1Sensor = analogRead(sensor1Pin);
int nilai2Sensor = analogRead(sensor2Pin);
int nilai3Sensor = analogRead(sensor3Pin);

```

```

DateTime now = rtc.now();
float rata=((nilai1Sensor + nilai2Sensor + nilai3Sensor)/3);
float persen=((100-((rata*100)/1023))/100)*100;

if (isnan(h) || isnan(t) || isnan(f)) {
Serial.println("Failed to read from DHT sensor!");
return;
}

Serial.print(now.year(), DEC);
Serial.print('/');
Serial.print(now.month(), DEC);
Serial.print('/');
Serial.print(now.day(), DEC);
Serial.print(" ");
Serial.print(" ");
Serial.print(now.hour(), DEC);
Serial.print(':');
Serial.print(now.minute(), DEC);
Serial.print(':');
Serial.print(now.second(), DEC);
Serial.println();
Serial.print(" S1 : ");
Serial.print(nilai1Sensor);
Serial.print(" S2 : ");
Serial.print(nilai2Sensor);
Serial.print(" S3 : ");
Serial.print(nilai3Sensor);
Serial.print("\n");
Serial.print(" T : ");
Serial.print(t);
Serial.print(" H : ");
Serial.print(k);
Serial.print("\n");
Serial.print("rata-rata kelembaban: ");
Serial.print(persen);
Serial.print("\n");
Serial.print("\n");

ArduinoSerial.print(h);
ArduinoSerial.print("\n");
ArduinoSerial.print(t);

```

```
ArduinoSerial.print("c");
ArduinoSerial.print(persen);
ArduinoSerial.print("d");
ArduinoSerial.print(now.year(), DEC);
ArduinoSerial.print("e");
ArduinoSerial.print(now.month(), DEC);
ArduinoSerial.print("f");
ArduinoSerial.print(now.day(), DEC);
ArduinoSerial.print("g");
ArduinoSerial.print(now.hour(), DEC);
ArduinoSerial.print("h");
ArduinoSerial.print(now.minute(), DEC);
ArduinoSerial.print("i");
ArduinoSerial.print(now.second(), DEC);
ArduinoSerial.print("j");
delay(1000);
}
```


LAMPIRAN 4
PROGRAM NODEMCU

```

#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SoftwareSerial.h>

//float untuk pemanggilan data suhu, kelembaban udara dan kelembaban
tanah

float val, vil, vul;
int vel, f, g, h, i, j;

//identitas token blynk dan nama wifi beserta password

char auth[] = "533ac353777e4ee3b7f83c9b51616638";
char ssid[] = "Redmi";
char pass[] = "namakamu";

BlynkTimer timer;

// pin yang digunakan untuk mode serial pada nodeMCU

SoftwareSerial NodeSerial(D2, D3);

void loop()
{
  while (NodeSerial.available() > 0) {
    val = NodeSerial.parseFloat();
    if (NodeSerial.read() == '\n') {
      Serial.print("Humidity: ");
      Serial.print(val);
      Serial.print("\n");
    }
    vil = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'c') {
      Serial.print("Temperature: ");
      Serial.print(vil);
      Serial.print("\n");
    }
    vul = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'd') {
      Serial.print("Soil: ");
      Serial.print(vul);
    }
  }
}

```

```
        Serial.print("\n");
    }
    vel = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'e') {
        Serial.print("haha");
        Serial.print(vel);
        Serial.print("\n");
    }
    f = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'f') {
        Serial.print("haha");
        Serial.print(f);
        Serial.print("\n");
    }
    g = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'g') {
        Serial.print("haha");
        Serial.print(g);
        Serial.print("\n");
    }
    h = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'h') {
        Serial.print("haha");
        Serial.print(h);
        Serial.print("\n");
    }
    i = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'i') {
        Serial.print("haha");
        Serial.print(i);
        Serial.print("\n");
    }
    j = NodeSerial.parseFloat();
    if (NodeSerial.read() == 'j') {
        Serial.print("haha");
        Serial.print(j);
        Serial.print("\n");
        Serial.print("\n");
        Serial.print("\n");
    }
    Blynk.virtualWrite(V5, val);
    Blynk.virtualWrite(V6, vil);
```

```

        Blynk.virtualWrite(V7, vul);
    }
    delay(100);
    Blynk.run();
    timer.run();
}

void emailOnButtonPress()
{

int isButtonPressed = !digitalRead(2); // Invert state, since button is
"Active LOW"

if (isButtonPressed) {
Serial.println("Button is pressed."); // This can be seen in the Serial
Monitor
String str = "Kelembaban tanah : " + String(vul, 1) + " %" + " ,    suhu
udara : " + String(vil, 2) + " C";
Blynk.email("rafiskc@yahoo.com", "Subject: Button Logger", str) ;

    WidgetTerminal terminal(V8);
    BLYNK_WRITE(V8);
    {

        terminal.print(v);
        terminal.print("/");
        terminal.print(f);
        terminal.print("/");
        terminal.print(g);
        terminal.print(" - ");
        terminal.print(h);
        terminal.print(":");
        terminal.print(i);
        terminal.print(":");
        terminal.print(j);
        terminal.print("\n");
        terminal.print("kelembaban tanah: ");
        terminal.print(vul);
        terminal.print(" %.");
        terminal.print("\t");
        terminal.print("Suhu lingkungan: ");
    }
}

```

```

        terminal.print(vil);
        terminal.print(" C");
        terminal.print("\n");
        if (vul>=0 && vul<=33)
        {
            terminal.print("Tanah kondisi kering");
        }
        if (vul>33 && vul<=66)
        {
            terminal.print("Tanah kondisi standar");
        }
        if (vul>66 && vul<=100)
        {
            terminal.print("Tanah kondisi basah");
        }
        terminal.print("\n");
        terminal.print("\n");
    }
    terminal.flush();
}

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

    Blynk.begin(auth, ssid, pass);

    Blynk.email("rafiskc@yahoo.com", "Subject", "My Blynk project is
online.");
    pinMode(2, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(2), emailOnButtonPress, CHANGE);

    pinMode(D2, INPUT);
    pinMode(D3, OUTPUT);
    Serial.begin(115200);
    NodeSerial.begin(4800);
    Blynk.begin(auth, ssid, pass);
}

```