

# Rancang Bangun Prototype Kelembaban Tanah

Imam Mahdi<sup>1</sup>, Dian Kasoni<sup>2</sup>

**Abstract**—*Apart from its name as a maritime country, Indonesia is also an agricultural country with fertile land and it has 2 seasons, rainy and dry. In the rainy season, food crops are usually not necessary to be watered because they already have received enough water. But in the dry season, they must be regularly watered because of the soil moisture conditions. The dependence of farmers on these seasons causes the production of farmers to decrease and become an obstacle to soil moisture. To overcome these obstacles and make farmers can still plant crops in the dry season, a product of information and communication technology-based agricultural equipment is needed. It is a programmable microcontroller chip that can automatically control plants watering based on soil moisture detected using soil moisture sensors. The development method used in this study is the prototype model and the programming language used is Arduino. The purpose of the tool is to detect whether the planting ground is dry so that the device can control watering automatically when the soil lacks water. This prototype was applied to realize a smart garden in utilizing technological advancements.*

**Intisari** - Indonesia selain sebagai negara maritim juga merupakan negara agraris dengan lahan yang subur dengan 2 musim yaitu musim penghujan dan musim kemarau. Pada musim penghujan biasanya tanaman pangan tidak perlu dilakukan penyiraman karena telah mendapatkan air yang cukup, sedangkan pada musim kemarau tanaman harus disiram dengan teratur sesuai dengan kondisi kelembaban tanah. Ketergantungan petani dengan musim menyebabkan produksi petani menurun dan menjadi kendala kelembaban tanah. Untuk mengatasi kendala tersebut agar petani tetap bisa bercocok tanam pada musim kemarau maka diperlukan suatu produk alat pertanian berbasis teknologi informasi dan komunikasi berupa chip microcontroller yang diprogram sehingga bisa mengontrol penyiraman tanaman secara otomatis berdasarkan kelembaban tanah yang dideteksi menggunakan sensor kelembaban tanah. Metode pengembangan yang digunakan pada penelitian ini dengan model *prototype* dan bahasa pemrograman yang digunakan adalah Arduino. Tujuan Alat untuk mendeteksi apakah tanah tempat bercocok tanam itu kering sehingga alat dapat mengontrol penyiraman secara otomatis saat tanah kekurangan unsur air. *Prototype* ini diaplikasikan untuk mewujudkan *smart garden* dalam memanfaatkan kemajuan teknologi.

**Kata Kunci** : Kelembaban tanah, *Prototype*, *Sensor*

## I. PENDAHULUAN

### A. Latar Belakang Masalah

Pada saat ini para petani budidaya tanaman saat ini kesulitan mengukur kelembaban tanah, padahal

kelembaban tanah sangatlah penting. Bibit tanaman tidak bisa hidup dan berkembang dengan baik jika kelembaban tanah tidak sesuai dengan kebutuhan. Untuk itu perlu dilakukan penyuburan bibit secara teratur dan terjadwal. Ketersediaan air pada masa pembibitan tanaman harus benar-benar diperhatikan. Jika kekurangan air, maka bibit akan kering dan akhirnya mati. Sebaliknya jika kelebihan air, bibit akan busuk. Dengan selalu terpenuhinya kebutuhan akan air, maka tanaman dapat tumbuh, berbuah, dan berkembang dengan baik. Berbeda dengan tanaman yang sudah tumbuh, tanaman telah memiliki akar yang banyak dan kuat sehingga mampu mencari air dengan sendiri bahkan jarang dilakukan penyuburan pun bisa dapat bertahan hidup. Masalah lainnya yaitu penyuburan dilakukan dengan metode manual sehingga menyebabkan lamanya proses penyuburan, sebagai tempat untuk melakukan pembibitan, berbeda dengan jumlah para petani yang mempunyai peran untuk melakukan pembibitan itu sendiri. Hal ini membuat para petani banyak menghabiskan waktunya untuk menyiram tanaman dan memupuk secara berulang di setiap tanaman yang akan ditanam dan hal tersebut membuat waktu mereka habis hanya untuk merawat tanaman. Keadaan tersebut membuat para petani merasa perlu melibatkan peran aktif dari teknologi terbaru terutama di bidang pertanian.

### B. Rumusan Masalah

Adapun Rumusan permasalahan dari masalah diatas, maka dirumuskan antara lain:

1. Bagaimana membangun alat *monitor* kelembaban tanah?
2. Bagaimana mempermudah proses penyiraman tanaman secara otomatis?

### C. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah :

1. Alat *monitoring* kadar kelembaban tanah berbasis *arduino* ini hanya dapat mengukur kelembaban tanah saja.
2. Alat *monitoring* kadar kelembaban tanah ini hanya dapat dibaca sensor *soil moisture*.
3. Alat *monitoring* kadar kelembaban tanah ini hanya bisa digunakan apabila ada aliran listrik.
4. Alat *monitoring* kadar kelembaban tanah ini hanya dapat mengukur kelembaban dengan skala kecil (*sample* tanah dalam pot).
5. Alat *monitoring* kadar kelembaban tanah ini sensitif pada suhu yang dingin (ber *Air Conditioner*).
6. Alat ini digunakan untuk menyiram tanaman pada saat tanah dalam keadaan kering dan berhenti apabila tanah dalam keadaan lembab.

<sup>1,2</sup>Jurusan Teknik Informatika STMIK Antar Bangsa, Kawasan Bisnis CBD Ciledug Blok A5 No.29-36, Jl.HOS Cokroaminoto Karang Tengah.,Ciledug,Tangerang (tlp: 021-73453000; e-mail : [dhekalearning@gmail.com](mailto:dhekalearning@gmail.com), [imam\\_206@yahoo.com](mailto:imam_206@yahoo.com))

#### D. Metode Penelitian

Metodologi yang digunakan penulis dalam menyelesaikan penelitian Rancang bangun Aplikasi *Monitoring* Kadar Kelembaban tanah ini menggunakan metode pengembangan dengan model *Prototype* yang terdiri dari:

1. Pengumpulan Kebutuhan  
Pelanggan atau pengembang bersama mendefinisikan format seluruh perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.
2. Membangun *Prototype*  
Membangun *Prototype* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya membuat *input Output*).
3. Evaluasi *Prototype*  
Evaluasi ini dilakukan oleh *client* apakah *Prototype* yang sudah dibangun sudah sesuai dengan keinginan *client*.
4. Mengkodekan Sistem  
Dalam tahap ini *Prototype* yang sudah di sepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.
5. Menguji Sistem  
Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites dahulu sebelum digunakan. Pengujian ini dilakukan dengan *White Box*, *Black Box*, *Basis Path*, pengujian arsitektur dan lain-lain. Pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan. Jika ya, langkah 7 dilakukan jika tidak, ulangi langkah 4 dan 5.
6. Menggunakan *system*  
Perangkat lunak yang telah diuji dan diterima pelanggan siap untuk digunakan.

## II. TEORI PENDUKUNG

Pada penelitian yang dilakukan oleh Emir Nasrullah, Agus Trisanto, dan Lioy Utami [12], berisi penjelasan mengenai *system* penyiraman tanaman secara otomatis menggunakan *sensor* suhu LM35 sebagai pendeteksi suhu daerah sekitar tanaman yang akan disiram. Suhu yang dapat dideteksi oleh *sensor* suhu LM35 memiliki rentang antara -55°C sampai dengan +150°C. Sensor LM35 dapat langsung dihubungkan pada rangkaian pengendali tanpa menggunakan rangkaian pengkondisi sinyal secara terpisah.

Penelitian yang dilakukan oleh Niken Ira Widodo [13], berisi penjelasan mengenai alat pengontrol suhu dan kelembaban pada ruang budidaya jamur tiram menggunakan *sensor* suhu dan kelembaban. Alat ini berfungsi untuk mengendalikan dan mengawasi suhu serta kelembaban pada *prototype* ruangan budidaya jamur tiram agar dapat membuat ruangan seperti habitat aslinya. Cara kerjanya adalah dengan memasukkan jamur tiram pada tempat / *prototype*. *Prototype* tersebut telah dilengkapi dengan pompa, pemanas serta *sensor* suhu dan kelembaban. Pompa dan pemanas akan diaktifkan secara otomatis berdasarkan

hasil yang dideteksi oleh *sensor* suhu dan kelembaban. Pompa digunakan untuk penyemprotan air sedangkan pemanas digunakan untuk menghangatkan jamur tiram agar suhu tetap berada pada rentang 23°C -28°C serta kelembaban pada 80%-85%. Keadaan suhu dan kelembaban pada ruang tersebut akan terus diawasi dengan adanya grafik menggunakan *visual basic*.

Penelitian yang dilakukan oleh Muhamad Haidar dalam jurnal Laporan Akhir dengan judul Rancang Bangun Alat Pengendali Kelembaban. Tanah Buah Tin menggunakan Mikrokontroler AVR berisi penjelasan mengenai alat pengendali kelembaban tanah buah tin menggunakan *sensor* moisture HR202 dan mikrokontroler AVR. *Sensor* moisture HR202 diinputkan ke *system* minimum arduino yang dikelola oleh mikrokontroler ATmega 328. Hasil olahannya berupa output akan ditampilkan oleh LCD berupa nilai *sensor* HR202. Apabila tanah yang diukur oleh *sensor* HR202 kering maka mikrokontroler akan menghidupkan pompa air begitu juga sebaliknya jika tanah terasa lembab maka mesin akan mati [14].

Pada penelitian yang dilakukan oleh Viktorianus Ryan Juniardy, Dedi Triyanto, Yulrio Brianorman (2014) dengan judul *Prototype* Alat Penyemprot Air Otomatis Pada Kebun Pembibitan Sawit Berbasis *Sensor* Kelembaban dan Mikrokontroler. Jurnal ini membahas mengenai alat penyemprotan air otomatis menggunakan ATmega8 dan *sensor* kelembaban tanah untuk mendeteksi kadar kelembaban tanah. Alat penyemprot air otomatis pada kebun pembibitan sawit terdiri dari *system* minimum mikrokontroler ATmega8, *sensor* kelembaban tanah, *relay*, tiga unit tombol fungsi, kabel, dan LCD yang berfungsi sebagai media menampilkan [15].

Penelitian yang dilakukan oleh Tia Ayu Pratama pada jurnal Laporan Akhir dengan judul Rancang Bangun Alat Penyiram Air Tanaman Mawar Berbasis Android Berdasarkan Kelembaban Tanah. Jurnal ini membahas mengenai perancangan alat penyiraman air tanaman mawar berdasarkan *sensor* kelembaban tanah. Alat ini beroperasi sesuai dengan data *inputan sensor* kelembaban tanah dikirim ke mikrokontroler yang akan digunakan untuk mengaktifkan led dengan tiga keadaan, setelah itu perintah penyiraman akan dikendalikan melalui *android* [16].

#### A. Syarat Tanah

Tanah merupakan lapisan permukaan bumi yang berasal dari bebatuan yang telah mengalami serangkaian pelapukan oleh gaya-gaya alam, sehingga membentuk regolith / lapisan partikel halus [9]. Hampir semua jenis tanah yang cocok untuk budidaya tanaman pertanian. Untuk mendapatkan kuantitas dan kualitas hasil yang tinggi, tanaman menghendaki tanah yang subur, gembur, kaya akan organik, tidak mudah becek (menggenang), bebas cacing (*nematoda*) dan penyakit tular tanah.

B. Kelembaban Tanah

Kelembaban tanah adalah jumlah air yang ditahan di dalam tanah setelah kelebihan air dialirkan, apabila tanah memiliki kadar air yang tinggi maka kelebihan air tanah dikurangi melalui evaporasi, transpirasi dan *transport* air bawah tanah. Air diperlukan dalam jumlah yang cukup, tidak berlebihan atau kurang.

Kelembaban yang melebihi 80% memacu pertumbuhan cendawan yang berpotensi menyerang dan merusak tanaman. Sebaliknya, iklim yang kurang dari 60% membuat tanaman kering dan mengganggu pertumbuhan generatifnya, terutama saat pembentukan bunga, penyerbukan, dan pembentukan buah.

C. Konsep Dasar Alat

1. *Soil Moisture Sensor*(Sensor Kelembaban Tanah)

*Soil Moisture* sensor adalah sensor kelembaban yang dapat mendeteksi kelembaban dalam tanah. Sensor ini sangat sederhana, tetapi ideal untuk memantau taman kota, atau tingkat air pada tanaman pekarangan anda. Sensor ini terdiri dua *probe* untuk melewati arus melalui tanah, kemudian membaca resistansinya untuk mendapatkan nilai tingkat kelembaban. Semakin banyak air membuat tanah lebih mudah menghantarkan listrik (resistansi kecil), sedangkan tanah yang kering sangat sulit menghantarkan listrik (resistansi besar). (Dfrobot, 2014, *Moisture Sensor* (SKU:SEN0114).

Sensor ini sangat membantu untuk mengingatkan tingkat kelembaban pada tanaman Anda atau memantau kelembaban tanah di kebun. Berikut spesifikasi dari *moisture sensor soil moisture*:

- a. *Power supply*: 3.3v or 5v
- b. *Output voltage signal*: 0~4.2v
- c. *Current*: 35mA

Pin definition:

- a. Analog output(Blue wire)
- b. GND(Black wire)
- c. Power(Red wire)
- d. Size: 60x20x5mm

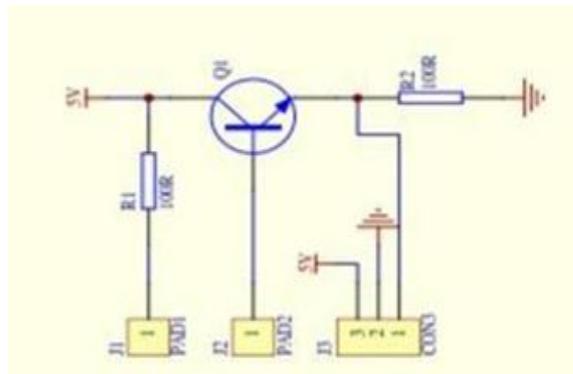
- 1) 0 ~300 bit : *dry soil*
- 2) 300~700 bit : *humid soil*
- 3) 700~950 bit : *in water*



Sumber: [www.inf-electronics.com](http://www.inf-electronics.com)

Gbr 1. *Soil Moisture Sensor*

*Moisture sensor* menggunakan *1-wire protocol*. Sensor ini mempunyai tiga pin yaitu VCC, DATA, dan GND. Skematiknya adalah seperti di Gambar 2.2



Sumber: [www.inf-electronics.com](http://www.inf-electronics.com)

Gbr 2. Schematic Sensor kelembaban tanah

Rumus untuk mengkonversi nilai keluaran sensor yang terbaca pada serial monitor menjadi satuan voltase adalah sebagai berikut :

$$\text{Voltase} = \frac{\text{Dec} \times 5 \text{ V}}{1024}$$

2. Relay

Wicaksono dan Hidayat [11] mengemukakan bahwa “Relay adalah saklar elektrik yang menggunakan elektromagnetik untuk memindahkan saklar dari posisi OFF ke posisi ON. Daya yang dibutuhkan untuk mengaktifkan relay relatif kecil. Namun, relay dapat mengendalikan sesuatu yang membutuhkan daya lebih besar”.

Terdapat beberapa jenis konfigurasi relay, misalnya *SPST* dan *SPDT* yang ditunjukkan gambar *Single Pole Single Throw* (SPST) merupakan konfigurasi yang paling sederhana, dimana relay dengan konfigurasi ini hanya memiliki dua kontak. *Single Pole Double Throw* memiliki tiga kontak. Kontak biasanya diberi label *Common* (COM), *Normally Open* (NO), *Normally Close* (NC).

Pada *Normally Close*, kontak NC akan terhubung ke kontak COM ketika coil tidak diberi daya. Pada *Normally Open*, kontak akan terputus ketika tidak ada daya yang diberikan pada *coil*. Ketika daya diberikan, makan *common* (COM) akan terhubung dengan kontak NO dan kontak NC dibiarkan mengambang (*floating*).

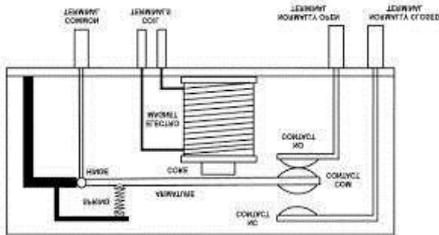


Sumber: [11]

Gbr.3. Relay

1. Kumparan elektromagnet
2. Saklar atau kontaktor
3. *Swing armature*
4. *Spring* (pegas)

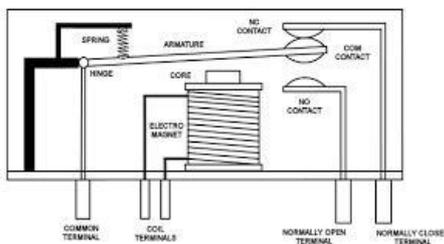
Dari konstruksi *relay* elektro mekanik diatas dapat diuraikan sistem kerja atau proses *relay* bekerja. Pada saat elektromagnet tidak diberikan sumber tegangan maka tidak ada medan magnet yang menarik *armature*, sehingga *scalar relay* tetap terhubung ke terminal NC (*Normally Close*) seperti terlihat pada gambar konstruksi diatas. Kemudian pada saat elektromagnet diberikan sumber tegangan maka terdapat medan magnet yang menarik *armature*, sehingga saklar *relay* terhubung ke terminal NO (*Normally Open*) seperti terlihat pada gambar dibawah.



Sumber : [www.elektronika-dasar.web.id](http://www.elektronika-dasar.web.id)

Gbr 4 Konstruksi *Relay* Elektro Mekanik NO

Konstruksi *relay* elektro mekanik posisi NO (*Normally Open*). *Relay* elektro mekanik memiliki kondisi saklar atau kontaktor dalam 3 posisi.



Sumber: [www.elektronika-dasar.web.id](http://www.elektronika-dasar.web.id)

Gbr 5 Konstruksi *Relay* Elektro Mekanik NC

Ketiga posisi saklar atau kontaktor *relay* ini akan berubah pada saat *relay* mendapat tegangan sumber pada elektromagnetnya. Ketiga posisi saklar *relay* tersebut adalah:

1. Posisi *Normally Open* (NO), yaitu posisi saklar *relay* yang terhubung ke terminal NO (*Normally Open*). Kondisi ini akan terjadi pada saat *relay* mendapat tegangan sumber pada elektromagnetnya.
2. Posisi *Normally Close* (NC), yaitu posisi saklar *relay* yang terhubung ke terminal NC (*Normally Close*). Kondisi ini terjadi pada saat *relay* tidak mendapat tegangan sumber pada elektromagnetnya.

3. Posisi *Change Over* (CO), yaitu kondisi perubahan *armature* saklar *relay* yang berubah dari posisi NC ke NO atau sebaliknya dari NO ke NC. Kondisi ini terjadi data sumber tegangan diberikan ke elektromagnet atau saat sumber tegangan diputus dari elektromagnet *relay*.

Dilihat dari desain saklar *relay* maka *relay* dibedakan menjadi:

1. *Single Pole Single Throw* (SPST), *relay* ini memiliki 4 terminal yaitu 2 terminal untuk *input* kumparan elektromagnet dan 2 terminal saklar. *Relay* ini hanya memiliki posisi NO (*Normally Open*) saja.
2. *Single Pole Double Throw* (SPDT), *relay* ini memiliki 5 terminal yaitu terdiri dari 2 terminal untuk *input* kumparan elektromagnetik dan 3 terminal saklar. *Relay* jenis ini memiliki 2 kondisi NO dan NC.
3. *Double Pole Single Throw* (DPST), *relay* jenis ini memiliki 6 terminal yaitu terdiri dari 2 terminal untuk *input* kumparan elektromagnetik dan 4 terminal saklar untuk 2 saklar yang masing-masing saklar hanya memiliki kondisi NO saja.
4. *Double Pole Double Throw* (DPDT), *relay* jenis ini memiliki 8 terminal yang terdiri dari 2 terminal untuk kumparan elektromagnetik dan 6 terminal untuk 2 saklar dengan 2 kondisi NC dan NO untuk masing-masing saklarnya.

### 3. Pompa Air

Pompa air adalah alat atau mesin untuk memindahkan atau menaikkan cairan dari satu tempat ketempat lainnya.

Berikut spesifikasinya:

- a. *Power*: 18 watt
- b. *Qmax.*: 1400 liter/ jam
- c. *Hmax.*: 1.50 meter

Gambar pompa air SP-3800 dapat dilihat pada gambar berikut.



Sumber : [www.eprints.polsri.ac.id](http://www.eprints.polsri.ac.id)

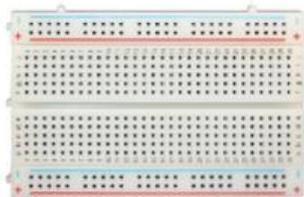
Gbr.6 Pompa Air

### 4. Breadboard

*Breadboard* adalah papan kecil yang mengandung sejumlah lubang yang dirancang untuk memudahkan dalam menyusun rangkaian elektronika tanpa melakukan penyolderan. Kadir (2016:4)

Hubungan antar lubang yang secara internal tidak terlihat secara fisik di *breadboard*, sebenarnya memiliki hubungan, sederet lubang pada garis pertama sebenarnya saling terhubung. Sesuai dengan tanda garis berwarna merah

dan simbol + yang diberikan, baris tersebut berguna untuk saluran positif. artinya lubang-lubang tersebut dapat dihubungkan ke sumber tegangan positif dan semua bagian komponen yang memerlukan tegangan positif. Adapun baris kedua yang ditandai dengan garis biru dan symbol digunakan untuk pentanahan (*ground*).



Sumber : Kadir (2016:5)

Gbr.7. Bentuk fisik Breadbord

### 5. Kabel

Kabel digunakan untuk menghubungkan satu lubang ke lubang lain di *breadboard* yang secara internal tidak terhubung ke komponen. Kabel dapat dibuat sendiri dari kabel tembaga berserat tunggal. Kadir (2016:9)



Sumber : kadir (2016:9)

Gbr .8. Bentuk fisik Kabel

### 6. LCD Display

Penampil kristal cair (Inggris: *liquid crystal display*; LCD) adalah suatu jenis media tampilan yang menggunakan kristal cair sebagai penampil utama. LCD sudah digunakan di berbagai bidang misalnya dalam alat-alat elektronik seperti televisi, kalkulator ataupun layar komputer. Kini LCD mendominasi jenis tampilan untuk komputer meja maupun *notebook* karena membutuhkan daya listrik yang rendah, bentuknya tipis, mengeluarkan sedikit panas, dan memiliki resolusi tinggi.

Pada LCD berwarna semacam monitor, terdapat banyak sekali titik cahaya (piksel) yang terdiri dari satu buah kristal cair sebagai sebuah titik cahaya. Walau disebut sebagai titik cahaya, kristal cair ini tidak memancarkan cahaya sendiri. Sumber cahaya di dalam sebuah perangkat LCD adalah lampu neon berwarna putih di bagian belakang susunan kristal cair.

Titik cahaya yang jumlahnya puluhan ribu bahkan jutaan inilah yang membentuk tampilan citra. Kutub kristal

cair yang dilewati arus listrik akan berubah karena pengaruh polarisasi medan magnetik yang timbul dan oleh karenanya akan hanya membiarkan beberapa warna diteruskan sedangkan warna lainnya tersaring.



Sumber : elektronika-dasar.web.id

Gbr.9. Bentuk LCD Display

### 7. Bahasa Pemrograman Arduino

Bahasa pemrograman *Arduino* dapat dibagi dalam tiga bagian utama: struktur, nilai (*variabel dan konstanta*), dan fungsi.

#### a. Stuktur Program

##### 1. Loop ()

Setelah membuat fungsi *setup ()*, yang menginisialisasi dan menetapkan nilai awal, fungsi *loop ()* melakukan tepatnya apa yang disarankan namanya, dan loop secara berurutan, memungkinkan program untuk berubah dan merespons. Gunakan untuk secara aktif mengendalikan papan *Arduino*.

##### 2. Setup ()

Fungsi *setup ()* dipanggil ketika sketsa dimulai. Berfungsi untuk menginisialisasi *variabel, mode pin*, mulai menggunakan pustaka dan sebagainya. Fungsi *setup ()* hanya akan berjalan sekali, setelah setiap *powerup* atau *reset board Arduino*.

##### 3. If .... Else

Pernyataan *if* memeriksa suatu kondisi dan mengeksekusi pernyataan atau serangkaian pernyataan jika kondisi tersebut 'benar'.

```
if (kondisi)
{
    // pernyataan (s)
}
```

#### b. Nilai Variabel atau Konstanta

##### 1. Int ()

Mengonversi nilai ke tipe data *int*. Dengan sintaksis *int (x)*, dimana nilai x, nilai jenis apapun.

##### 2. Const ()

Kata kunci *const* adalah singkatan dari konstanta. Ini adalah kualifikasi variabel yang memodifikasi perilaku variabel, membuat variabel "*read-only*". Ini berarti bahwa *variabel* dapat digunakan sama seperti *variabel* lain dari jenisnya, tetapi nilainya tidak dapat diubah. *Programmer* akan mendapatkan kesalahan *kompiletor* jika mencoba menetapkan nilai ke *variabel const*.

Konstanta didefinisikan dengan kata kunci *const* mematuhi aturan pengelompokan *variabel* yang mengatur *variabel* lain. Ini, dan jebakan menggunakan *#define*, menjadikan kata kunci *const* sebagai metode superior untuk mendefinisikan konstanta dan lebih disukai daripada menggunakan *#define*.

### c. Fungsi

#### 1. *pinMode()*

Mengkonfigurasi pin yang ditentukan untuk berperilaku sebagai *input* atau *output*. Pada *Arduino 1.0.1*, dimungkinkan untuk mengaktifkan *resistor pullup internal* dengan mode *INPUT\_PULLUP*. Selain itu, mode *INPUT* secara eksplisit menonaktifkan *pullup internal*. Sintaksis dari fungsi ini yaitu, *pinMode (pin, mode)*.

#### 2. *digitalRead ()*

Membaca nilai dari pin digital tertentu, baik tinggi atau rendah. Dengan Sintaksis *digitalRead (pin)*. Parameter pin adalah jumlah pin digital yang ingin programmer baca.

#### 3. *digitalWrite ()*

*digitalWrite ()* berfungsi untuk menulis nilai tinggi atau rendah ke pin digital. Jika pin telah dikonfigurasi sebagai *output* dengan *pinMode ()*, tegangannya akan diatur ke nilai yang sesuai: 5V (atau 3.3V pada papan 3.3V) untuk *high*, 0V (*ground*) untuk *low*.

Jika pin dikonfigurasi sebagai *input*, *digitalwrite ()* akan mengaktifkan (*high*) atau menonaktifkan (*low*) *pullup internal* pada pin *input*. Disarankan untuk mengatur *pinMode ()* ke *input\_pullup* untuk mengaktifkan *resistor pull-up internal*.

Jika tidak mengatur *pinMode ()* ke *output*, dan menghubungkan *led* ke pin, saat memanggil *digitalWrite (high)*, *led* mungkin tampak redup. Tanpa secara eksplisit mengatur *pinMode ()*, *digitalWrite ()* akan mengaktifkan *resistor pull-up internal*, yang bertindak seperti resistor pembatas arus besar. Sintaksis *digitalWrite (pin, nilai)*.

#### 4. *analogRead ()*

*analogRead ()* berfungsi untuk membaca nilai dari *pin analog* yang ditentukan. Papan *Arduino* berisi 6 saluran (8 saluran pada *Mini* dan *Nano*, 16 di *Mega*), 10-bit *analog* ke *digital converter*. Ini berarti bahwa ia akan memetakan tegangan input antara 0 dan 5 volt menjadi nilai *integer* antara 0 dan 1023. Ini menghasilkan resolusi antara pembacaan: 5 volt / 1024 unit atau, .0049 volt (4,9 mV) per unit. Kisaran input dan resolusi dapat diubah menggunakan *analogReference ()*.

Diperlukan sekitar 100 *microsecond* (0,0001 s) untuk membaca input *analog*, jadi tingkat pembacaan maksimum sekitar 10.000 kali per detik. Dengan sintaksis *analogRead (pin)*.

#### 5. *analogWrite ()*

*analogWrite()* berfungsi untuk menulis nilai *analog (PWM wave)* ke pin. Dapat digunakan untuk menyalakan *led* pada berbagai tingkat kecerahan atau

menggerakkan motor dengan berbagai kecepatan. Setelah panggilan ke *analogWrite ()*, pin akan menghasilkan gelombang persegi yang bagus dari siklus tugas yang ditentukan sampai panggilan berikutnya ke *analogWrite()* (atau panggilan ke *digitalRead ()* atau *digitalWrite()*) pada pin yang sama. Frekuensi sinyal *PWM* pada sebagian besar pin kira-kira 490 Hz. Pada papan *Uno* dan yang serupa, pin 5 dan 6 memiliki frekuensi sekitar 980 Hz.

Pada kebanyakan papan *Arduino* (yang memiliki *ATmega168* atau *ATmega328P*), fungsi ini berfungsi pada pin 3, 5, 6, 9, 10, dan 11. Pada *Arduino Mega*, fungsi ini bekerja pada pin 2 - 13 dan 44 - 46. *Arduino* yang lebih tua papan dengan *ATmega8* hanya mendukung *analogWrite ()* pada pin 9, 10, dan 11

*The Arduino DUE* mendukung *analogWrite()* pada pin 2 hingga 13, ditambah pin *DAC0* dan *DAC1*. Tidak seperti pin *PWM*, *DAC0* dan *DAC1* adalah konverter *Digital* ke *Analog*, dan berfungsi sebagai *output analog* yang benar. *Programmer* tidak perlu memanggil *pinMode ()* untuk mengatur pin sebagai output sebelum memanggil *analogWrite ()*. Fungsi *analogWrite* tidak ada hubungannya dengan pin *analog* atau fungsi *analogRead*. Sintaksis *analogWrite (pin, nilai)*

#### 6. *delay()*

Menjeda program untuk jumlah waktu (dalam milidetik) yang ditentukan sebagai parameter. Ada 1000 *milidetik* dalam satu detik. Penundaan (*ms*) *ms*: jumlah *milidetik* untuk dijeda (*unsigned long*).

## 8. Bahasa Pemrograman C

Bahasa pemrograman C merupakan salah satu bahasa pemrograman komputer. Dibuat pada tahun 1972 oleh Dennis Ritchie untuk Sistem Operasi Unix di Bell Telephone Laboratories. Meskipun C dibuat untuk memprogram sistem dan jaringan komputer namun bahasa ini juga sering digunakan dalam mengembangkan *software* aplikasi. C juga banyak dipakai oleh berbagai jenis *platform* sistem operasi dan arsitektur komputer, bahkan terdapat beberapa *compiler* yang sangat populer telah tersedia. C secara luar biasa memengaruhi bahasa populer lainnya, terutama C++ yang merupakan *ekstensi* dari C.

Dalam pembuatan program yang menggunakan fungsi atau aritmatika, Bahasa C menawarkan kemudahan dengan menyediakan fungsi-fungsi khusus, seperti: pembuatan konstanta, operator aritmatika, operator logika, operator *bitwise* dan operator *Assignment*. Selain itu bahasa C menyediakan program kontrol seperti: Percabangan (*if* dan *if...else*), Percabangan *switch*, *looping* (*for*, *while*, dan *do...while*), *Array*, serta fungsi-fungsi lainnya.

Struktur dari program C dapat dilihat sebagai kumpulan dari sebuah atau lebih fungsi-fungsi. Fungsi pertama yang harus ada di program C sudah ditentukan namanya, yaitu bernama *main()*. Suatu fungsi di program C dibuka dengan kurung kurawal (*{*) dan ditutup dengan kurung kurawal tutup (*}*). Diantara kurung-kurung kurawal

dapat dituliskan pernyataan-pernyataan program C. struktur bahasa pemrograman C, antara lain:

- Header File* adalah berkas yang berisi *prototype* fungsi definisi konstanta dan definisi *variable*. Fungsi adalah kumpulan kode C yang diberi nama dan ketika nama tersebut dipanggil maka kumpulan kode tersebut dijalankan.
- Preprocessor Directive (#include)* adalah bagian yang berisi pengikut sertaan *file* atau berkas-berkas fungsi maupun pendefinisian konstanta.
- Void* artinya fungsi yang mengikutinya tidak memiliki nilai kembalian (*return*).
- Main ()* adalah fungsi yang pertama kali dijalankan ketika program dieksekusi, tanpa fungsi main suatu program tidak dapat dieksekusi namun dapat dikompilasi.
- Statement* adalah instruksi atau perintah kepada suatu program ketika program itu dieksekusi untuk menjalankan suatu aksi. Setiap *statement* diakhiri dengan titik-koma.

Bahasa C dikatakan sebagai bahasa pemrograman terstruktur, karena strukturnya menggunakan fungsi-fungsi sebagai program-program bagian (*subroutine*). Fungsi-fungsi selain fungsi utama merupakan program-program bagian. Fungsi-fungsi ini dapat ditulis setelah fungsi utama atau diletakkan di *file* pustaka (*library*). Jika fungsi-fungsi diletakkan di *file* pustaka dan akan dipakai disuatu program, maka nama *file* judulnya (*header file*) harus dilibatkan di dalam program yang menggunakannya dengan *preprocessor directive #include* (Hartono, 1992).

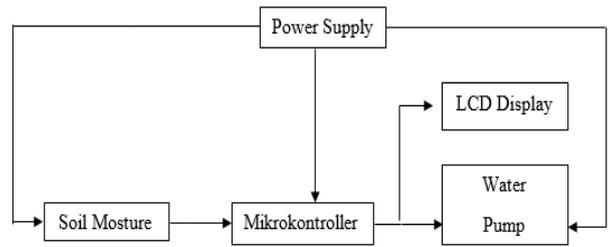
### 9. Flowchart

*Flowchart* atau bagan alir, awal mulanya memang berkembang dari industri komputer yaitu untuk menggambarkan urutan proses penyelesaian masalah. Namun seperti kata pepatah lama bahwa „satu gambar adalah sejuta kata” maka suatu *flowchart* dapat dengan mudah menjelaskan suatu urutan proses yang relatif rumit bila diuraikan dalam kata-kata. Dengan visualisasi maka adanya *bottleneck* (penumpukan) atau ketidak-efisienan dari suatu proses dapat terdeteksi untuk melakukan perbaikan. Oleh karena itu *flowchart* juga diterima di kalangan lain (manufaktur, sains, militer, manajemen, dsb.). *Flowchart* sangat berguna khususnya untuk menjelaskan urutan proses yang pelaksanaannya mempunyai banyak *option* pilihan atau percabangan.

## III. PEMBAHASAN

### Blok Diagram

Diagram blok adalah suatu pernyataan gambar yang ringkas, dari gabungan sebab dan akibat antara masukan dan keluaran dari suatu sistem.



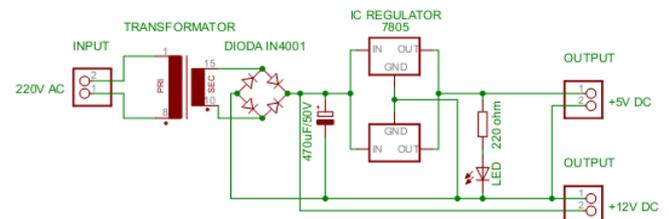
Sumber: Rangkaian Penelitian

Gbr 10. Blok Diagram

- Power supply* sebagai pemberi daya bagi semua komponen, dari mulai bagian input, proses dan output.
- Sensor *Soil Moisture* berfungsi sebagai komponen pengukur kadar air didalam tanah.
- Dibagian proses menggunakan *Arduinouno*, dimana *Arduino uno* ini merupakan komponen pemrosesan logika. Hasil data dari sensor *Soil Moisture* akan diterima dan diproses oleh *Arduino uno* sesuai dengan script yang ditulis
- Bagian terakhir yaitu *output*, water pump. *output* ini akan aktif setelah mikrokontroler menerima input dari sensor dengan data yang sesuai dengan nilai yang telah ditentukan dibaris pemrograman.

### Perencanaan Catu Daya

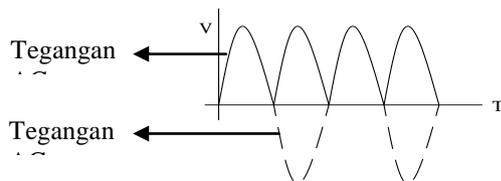
Rangkaian *power supply* atau rangkaian catu daya merupakan rangkaian utama dalam suatu rangkaian elektronika. Untuk mempermudah pembelajaran rangkaian ini terbagi menjadi empat bagian blok sistem catu daya seperti gambar dibawah ini.



Sumber : Aprianto (2015)

Gbr 11. Rangkaian Power Supply

- Blok pertama**  
Pada blok pertama ini sebuah transformator step down yang berfungsi untuk menurunkan tegangan AC 220V menjadi tegangan AC yang lebih kecil.
- Blok kedua**  
Pada blok ini sebuah diode penyearah jembatan atau diode bridge rectifier yang berfungsi sebagai penyearah tegangan AC menjadi DC. *Diode* yang digunakan adalah jenis *diode silicon*. Pada gambar dibawah ini adalah hasil output dari *diode bridge rectifier*.



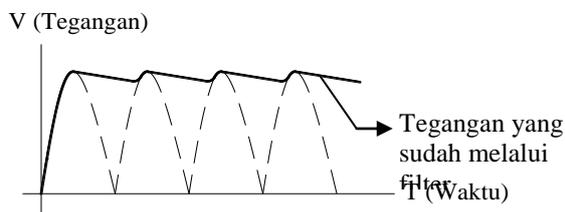
Sumber : Aprianto (2015)

Gbr 12. Sinyal Penyearah Dioda

Dari gambar di atas sinyal yang dihasilkan dari output dioda bridge terbentuk saat gelombang AC dari output transformator dioda bridge terbias *forward* dan *reverse* sesuai polaritas yang terhubung. Sehingga pada saat output dari transformator negatif akan tertahan dan tegangan negatif akan dicerminkan ke atas menjadi tegangan positif. Pencerminkan tegangan ini disebabkan oleh dioda-dioda yang selalu membukakan untuk arus *forward* sehingga mengakibatkan tegangan output dioda menjadi positif. Besarnya tegangan output juga dikurangi sebesar tegangan pada 2 dioda karena masing-masing memiliki tegangan *forward* bias sebesar 0,7 volt.

### 3. Blok ketiga

Pada blok ini terdapat kapasitor elektrolit atau *elco* yang digunakan sebagai filter atau penyaring tegangan DC. Semakin besar nilai kapasitannya maka tegangan DC akan semakin mendekati tegangan DC murni dan menghilangkan *ripple* dari output penyearah. Bentuk sinyal output dari kapasitor seperti gambar di bawah ini.



Sumber : Aprianto (2015)

Gbr 13. Sinyal Penyearah Dioda

### 4. Blok keempat

Pada blok ini adalah rangkaian penstabil tegangan yang berfungsi untuk menstabilkan tegangan DC. Karena pada *power supply*, terutama *power supply* buatan sendiri walaupun sudah disearahkan dengan *diode* pasti tegangan output kurang stabil, maka dari itu perlu suatu komponen yang berfungsi untuk menstabilkan tegangan output tersebut. Pada umumnya rangkaian penstabil tegangan menggunakan IC *regulator* LM 78xx dan LM 79xx. Pada perancangan rangkaian catu daya ini menggunakan IC *regulator* 7805 untuk mensuplay rangkaian *driver* motor serta *relay* dc dan juga untuk memberikan input pada *RS232 converter*.

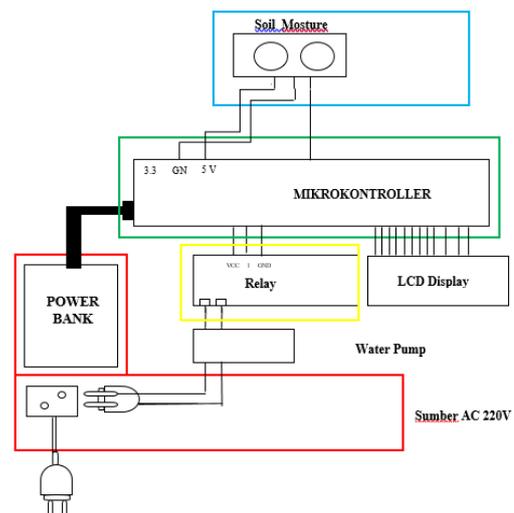
## Perencanaan *Input*, *Sensor* dan *Output*

Perencanaan *sensor* yang digunakan, terdiri dari *sensor Soil Moisture*. *Soil Moisture* merupakan *sensor* untuk mendeteksi tingkat kelembaban tanah dan juga dapat digunakan untuk menentukan apakah ada kandungan air di tanah/sekitar *sensor*. *Sensor soil moisture* ditancapkan pada tanah untuk mengukur kelembaban suatu tanah. *Sensor soil moisture* digunakan untuk membaca data sebuah objek tanah yang bertujuan untuk mengaktifkan perangkat *output water pump* apabila nilai kandungan air dalam tanah kurang/dalam keadaan kering. *Sensor soil moisture* berfungsi juga untuk menonaktifkan perangkat *output water pump* apabila nilai kandungan air dalam tanah tercukupi/dalam keadaan lembab. Adapun bentuk rangkaian *sensor* pada gambar 3.5. dengan blok berwarna biru.

Pada bagian proses, mikrokontroler berfungsi sebagai penerima data *sensor Soil Moisture*. Mikrokontroler juga menyimpan list-list program yang didalamnya terdapat pengkondisian data yang masuk melalui sensor. Bentuk rangkaian bagian proses pada gambar 3.5. dengan blok berwarna hijau.

Relay berfungsi menjadi penengah antara tegangan 5 volt dan 220 volt. *Relay* dapat menggerakkan saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi. Bentuk rangkaiannya pada gambar 3.5. dengan blok berwarna kuning.

Bagian *output* menggunakan *water pump* dan LCD display sebagai indikator. Ketika *sensor* membaca keadaan kemudian mendapatkan sebuah *value* atau nilai yang dikirimkan ke mikrokontroler untuk diproses sesuai dengan logika pemrograman. *Water pump* sebagai indikator akan menyala ketika logika pemrograman terpenuhi dan LCD display akan menampilkan nilai *sensor soil moisture*. Bentuk rangkaian *output* pada gambar III.5. dengan blok berwarna hitam.

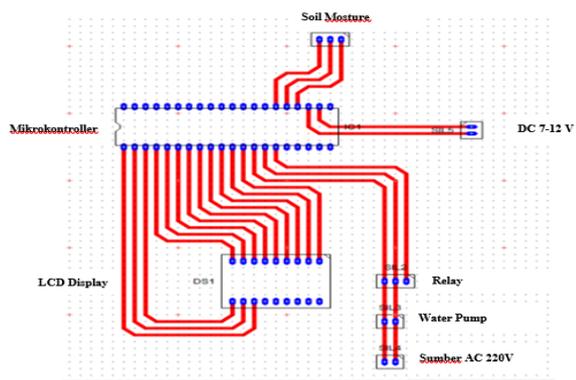


Sumber : Rancangan Penelitian

Gbr 14. Skema Diagram Perencanaan *Input*, Proses dan *Output*

IV PENGUJIAN DAN ANALISA

Skematik Diagram



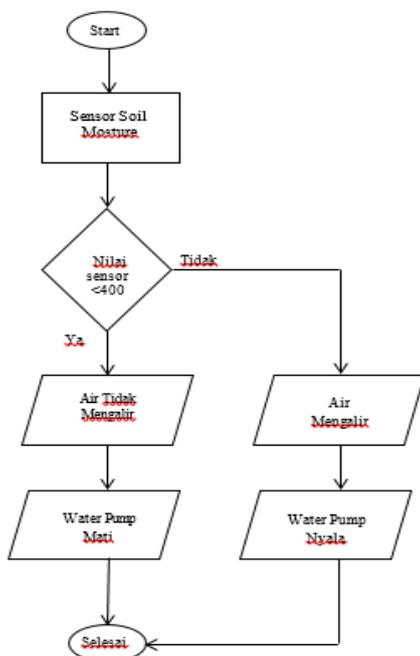
Sumber : Rancangan Penelitian

Gbr 15. Skema Diagram

Cara Kerja Alat

Prototype kelembaban tanah menggunakan sensor soil moisture sebagai pendeteksi tingkat kelembaban tanah. Setelah sensor mendapatkan value atau nilai kemudian dikirimkan ke mikrokontroler yang didalamnya sudah terdapat logika pemrograman. Logika pemrograman yang digunakan adalah ketika value atau nilai sensor lebih dari value yang telah ditetapkan maka output akan aktif sedangkan ketika nilai sensor kurang atau cukup dari value yang telah ditetapkan maka output tidak aktif. Output yang digunakan sebagai indikator air mengalir atau tidak adalah water pump.

Flowchart



Sumber : Rancangan Penelitian

Gbr 16. Flowchart Kelembaban Tanah

Tujuan dari pengujian alat prototype kelembaban tanah adalah sebagai berikut:

- Untuk mengetahui value atau nilai dari sensor ketika kondisi tanah kering dan kondisi tanah lembab.
- Untuk mengetahui pengaruh dari prototype kelembaban tanah ini dalam memanfaatkan efisiensi waktu.
- Untuk mengetahui apakah prototype kelembaban tanah ini berkontribusi dalam mengurangi pemanasan global

Hasil Pengujian

a. Pengujian Catu Daya

Sensor, mikrokontroler dan output yang digunakan membutuhkan catu daya yang berbeda-beda. Karena didalam sensor dan mikrokontroler menggunakan komponen-komponen dalam kategori arus lemah. Sensor Soil Moisture, pendeteksi kandungan air dalam tanah membutuhkan daya 3,3V, sensor ini mendapatkan supply daya dari mikrokontroler yang sudah tertera dikaki pin mikrokontroler. Mikrokontroler membutuhkan daya sebesar 5V dan mendapatkan supply daya dari powerbank. Dibagian output mendapatkan supply daya dari sumber listrik Alternating current (AC), 220V. Hasil dari semua supply daya tersebut berjalan dengan baik selama pengujian alat berlangsung.

b. Pengujian Input

Hasil dari pengujian input didapatkan kesimpulan bahwa sensor Soil Moisture atau pendeteksi kandungan air dalam tanah membaca value atau nilai sensor sebesar 600 ketika posisi tanah dalam keadaan kering, nilai sensor sebesar 400 ketika posisi tanah dalam keadaan lembab. yang perlu diperhatikan adalah supply tegangan untuk sensor harus sesuai dengan datasheet sensor. Gambar 17 dan Gambar 18 memperlihatkan hasil pembacaan sensor ketika tanah dalam keadaan kering dan dalam keadaan lembab.

c. Pengujian Proses

Hasil dari pengujian proses didapatkan kesimpulan bahwa mikrokontroler berjalan dengan baik. Selama pengujian tidak terjadi masalah terhadap kinerja mikrokontroler. Hal yang menjadi kekurangan selama pengujian proses mikrokontroler adalah ketika baris coding tidak sesuai dengan tujuan alat yang dibuat diantaranya penulisan nilai yang didapat dari pembacaan sensor tidak sesuai, dari hal itu perlu percobaan berulang-ulang untuk mendapatkan nilai sensor yang tepat. Faktor lain yang menjadi kekurangan selama pengujian proses adalah penggunaan operator pembanding seperti lebih kecil, lebih besar, lebih kecil sama dengan, lebih besar sama dengan, perlu juga percobaan yang berulang-ulang untuk mendapatkan hasil yang tepay sesuai tujuan output.



Untuk memperoleh hasil yang optimal dari pengujian *prototype* ini adalah yang pertama memastikan setiap kabel terhubung dari satu komponen ke komponen yang lain sesuai dengan rangkaian yang telah dibuat dan pemasangan kabel ke pin project board, relay dan sensor harus pas. Kedua, memperhatikan keadaan tanah yang mempengaruhi dalam mendapatkan nilai sensor untuk hasil output yang maksimal. Terakhir adalah usia dari setiap komponen yang mempengaruhi kinerja alat secara keseluruhan.

#### IV. KESIMPULAN

Dari hasil perancangan, pembuatan, pengujian dan analisa dapat disimpulkan sebagai berikut:

1. *Prototype kelembaban tanah* ini dapat diaplikasikan untuk mewujudkan *smart garden* dalam memanfaatkan kemajuan teknologi.
2. *Prototype kelembaban tanah* ini dapat memaksimalkan efisiensi waktu.
3. Dan dengan adanya alat ini, kita dapat mengatur dan mengukur kelembaban tanah pada tanaman

#### UCAPAN TERIMA KASIH

Terima kasih yang sebesar-besarnya kehadirat Allah SWT, rekan-rekan dosen STMIK Antar Bangsa yang telah membantu. Peneliti berharap hasil dari penulisan bermanfaat.

#### REFERENSI

- [1] Adam Vrileuis. 2013. Pemantauan Lalu Lintas dengan Sensor LDR Berbasis Mikrokontroler ATmega16. ISSN: 1412-4785. Banda Aceh : Jurnal Rekayasa Elekrika Vol. 10, No. 3, April 2013, Hal 115-159. Diambil dari : [http://jurnal.unsyiah.ac.id/JRE/article/view/1016/10\\_3\\_5](http://jurnal.unsyiah.ac.id/JRE/article/view/1016/10_3_5). (20 Juli 2018)
- [2] Alfa Saleh. 2015. Implementasi Metode Klasifikasi Naïve Bayes Dalam Memprediksi Besarnya Penggunaan Listrik Rumah Tangga. ISSN: 2354-5771. Citec Journal, Vol. 2, No. 3, Mei 2015 – Juli 2015. Diambil dari : <http://citec.amikom.ac.id/main/index.php/citec/article/view/49/49>. (04 April 2018)
- [3] Artanto, Dian, 2012, Interaksi Arduino dan Labview. Jakarta: PT Elex Media Komputindo.
- [4] Deny Siswanto, Slamet Winardi. 2015. Jemuran Pakaian Otomatis menggunakan Sensor Hujan dan Sensor LDR Berbasis Arduino Uno. E-ISSN: 2407-7712. Narodroid, Vol. 1 No. 2 Juli 2015. Diambil dari : <http://jurnal.narotama.ac.id/index.php/narodroid/article/view/69/59>. (20 Juli 2018)
- [5] Decy Nataliana, Iqbal Syamsu, Galih Giantara. 2014. Sistem Monitoring Parkir Mobil Menggunakan Sensor Infrarad Berbasis Raspberry Pi. Bandung: Teknik Elektro Itenas | No.1 | Vol. 2. Diambil dari : <https://jurnalonline.itenas.ac.id/index.php/elkomika/article/view/819/1018>. (20 Juli 2018)
- [6] Espressif Systems. ESP32 Datasheet Version 2.3. Diambil dari : <https://www.espressif.com/en/products/hardware/esp32-devkitc/resources>. (20 Juli 2018)
- [7] Heru Dibyo Laksono, M. Nasir Sonni. 2007. Perancangan dan Implementasi Relay Arus Lebih Sesaat Berbasis Microcontroller. 70 Gematek Jurnal Teknik Komputer, Volume 9 Nomor 2, September 2007. Diambil dari :

<http://ced.petra.ac.id/index.php/tek/article/view/16919/16905>. (20 Juli 2018)

- [8] Jazi Eko Istiyanto, 2014, Pengantar Elektronika & Instrumentasi Pendekatan Project Arduino & Android. Yogyakarta: C.V Andi Offset.
- [9] Kadir, Abdul. 2016. Simulasi Arduino. Jakarta: PT Elex Media Komputindo.
- [10] Prianto, E. 2007. Rumah Tropis Hemat Energi Bentuk Kepedulian Global Warming. Riptek, Vol. 1 No. 1, November 2007, Hal:1-10. Diambil dari : [http://bappeda.semarangkota.go.id/v2/wp-content/uploads/2013/12/1.-Rumah\\_Tropis\\_Hemat\\_Energi-PRianto.pdf/](http://bappeda.semarangkota.go.id/v2/wp-content/uploads/2013/12/1.-Rumah_Tropis_Hemat_Energi-PRianto.pdf/). (20 Juli 2018)
- [11] Wicaksono, Mochamad Fajar dan Hidayat. 2017. Mudah Belajar Mikrokontroler Arduino. Bandung: Informatika Bandung.
- [12] Nasrullah, Emir. Trisanto, Agus. Utami, Lioty. 2011. Rancang Bangun Sistem Penyiraman Tanaman Secara Otomatis Menggunakan Sensor Suhu LM35 BerbasisMikrokontroler Atmega8535. ELECTRICIAN – Jurnal Rekayasa dan Teknologi Elektro. Volume 5, No. 3, September 2011. <http://jurnal.fp.unila.ac.id/index.php/thp/article/view/1/1>
- [13] Widodo, Niken Ira 2013. “Prototipe Alat Pengontrol Dan Monitoring Suhu Serta Kelembaban Pada Ruang Budidaya Jamur Tiram Melalui Media Wireless”.Palembang: Politeknik Negeri Sriwijaya.
- [14] Haidar, Muhammad. 2014. Laporan Akhir : Rancang Bangun Alat Pengendali Kelembapan Tanah Pada Tanaman Buah Tin Menggunakan Mikrokontroler AVR. Politeknik Negeri Sriwijaya. Palembang
- [15] Viktorianus Ryan Juniardy, Dedi Triyanto, Yulrio Brianor. 2014 Prototype Alat Penyemprot Air Otomatis Pada Kebun Pembibitan Sawit Berbasis Sensor Kelembaban Dan Mikrokontroler AVR ATMEGA8Jurnal Coding Sistem Komputer Universitas Tanjungpura Volume 02 No. 3 2014, hal 1 – 10. <http://jurnal.untan.ac.id/index.php/jcskommipa/article/download/7614/7733>
- [16] Tia Ayu Pratama. 2015. Rancang Bangun Alat Peyiram Air Tanaman Mawar Berbasis Android Berdasarkan Kelembaban Tanah.

<http://repositori.usu.ac.id/bitstream/handle/123456789/6882/152408027.pdf?sequence=1&isAllowed=y>



Imam Mahdi. Lahir di Jakarta 15 Februari 1996. Pada Tahun 2018 lulus Program Sarjana (S1) Teknik Informatika di STMIK Antar Bangsa. Saat ini aktif sebagai staff Tata Usaha (TU) di sekolah SD Negeri Joglo 10



Dian Kasoni. Lahir di Tegal, 03 Mei 1986. Lulus Program Pasca Sarjana (S2) Magister Komputer di STMIK Nusa Mandiri Jakarta. Saat ini Aktif sebagai Dosen Tetap di STMIK Antar Bangsa