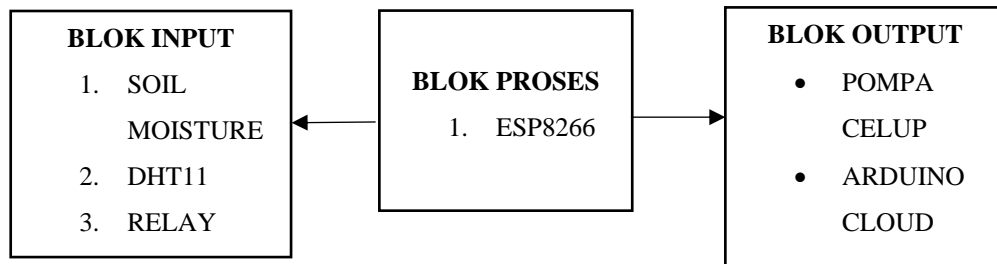


### 3 PERANCANGAN DAN ANALISA

#### 3.1 Analisa Rangkaian Secara Blok Diagram

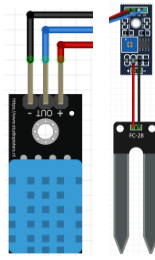


**Gambar 3. 1 Blok Diagram**

Gambar 3.1 dapat dilihat rancangan rangkaian secara blok diagram terdiri dari 3 blok yaitu, blok input, blok proses, blok output. Blok input menjelaskan masukan untuk mikrokontroller serta media masukannya yang terdiri dari *soil moisture*, DHT11 dan relay, selanjutnya untuk blok proses menggunakan ESP8266 menjelaskan pemrosesan yang didapat dari masukan agar mendapatkan output yang sesuai dan blok output menjelaskan keluaran tentang bagaimana keluaran yang dihasilkan oleh blok proses berupa pompa celup dan Arduino cloud. Secara rinci uraian pada gambar 3.1 dapat dijelaskan sebagai berikut.

##### 3.1.1 Blok Input

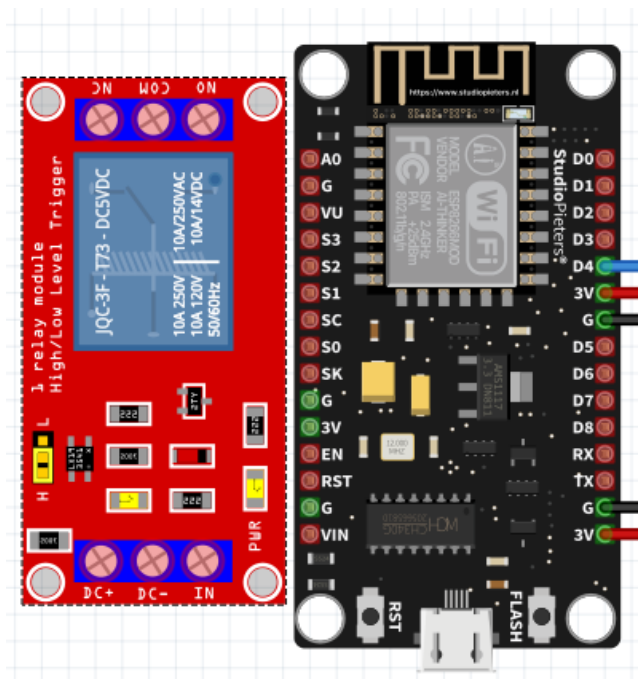
Blok *input* ini memiliki beberapa sensor yang digunakan sebagai inputan, inputan sensor yang pertama Soil Moisture digunakan untuk mendeteksi kelembapan tanah, Soil Moisture dalam *green house* disini untuk memberikan informasi kelembapan yang ada di dalam tanah secara akurat dan memberikan umpan balik ke relay untuk menghidupkan atau mematikan pompa. Sensor kedua adalah DHT11 berfungsi monitoring suhu yang ada di dalam *green house* tersebut.



**Gambar 3. 2 Blok Input**

### 3.1.2 Blok Proses

Blok proses adalah blok yang berfungsi untuk mengontrol kerja dari keseluruhan komponen yang ada pada *green house* ini. Nodemcu berperan sebagai tempat pemrosesan inputan data yang diterima dan juga sebagai penghubung alat ke Arduino Cloud melalui jaringan wifi yang berfungsi untuk mengoprasikan aplikasi. Setelah NodeMCU menerima inputan maka inputan tersebut akan langsung dikirimkan ke relay.



**Gambar 3. 3 Blok Proses**

### 3.1.3 Blok Output

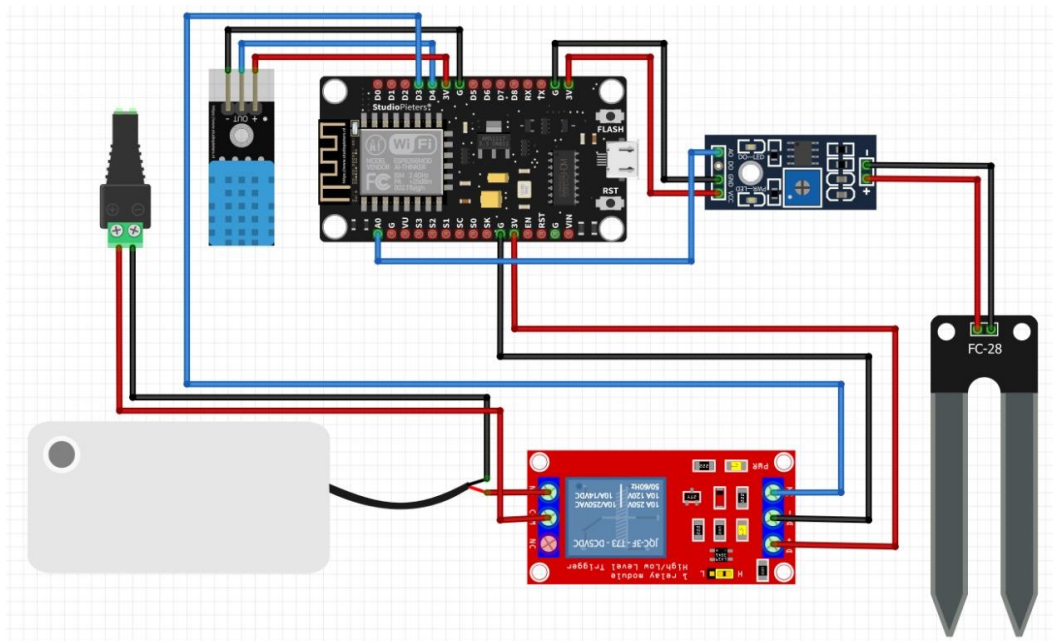
Blok output memiliki dua komponen sebagai outputan, yang pertama adalah pompa celup berfungsi sebagai komponen yang di gunakan untuk penyaluran dari

tempat penampungan air ke tanaman, Selanjutnya, Arduino cloud adalah software yang berfungsi untuk menampilkan informasi kelembapan tanah dan juga suhu yang ada di *green house* tersebut.



Gambar 3. 4 Blok Output

### 3.2 Perancangan Alat



Gambar 3. 5 Analisa Diagram Secara Detail

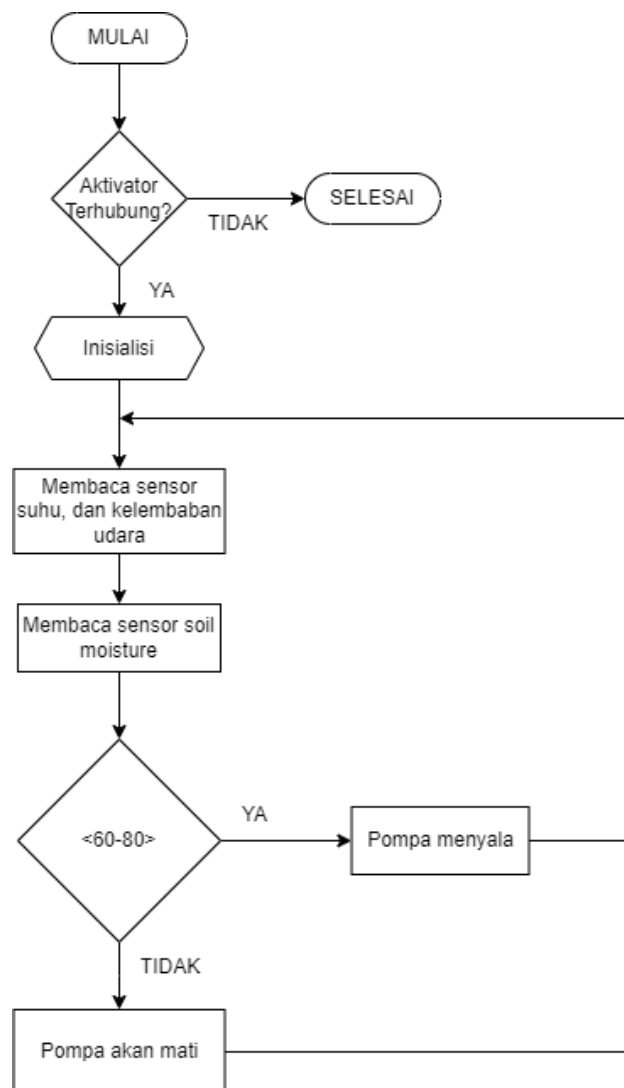
Rangkaian ini menggunakan tegangan +5V untuk mengaktifkan NodeMCU yang merupakan tempat pemrosesan dari alat ini, dan tegangan 12V dialiri ke komponen seperti pompa celup. Media input dari alat ini adalah Soil Moisture dengan pin A0 pada NodeMCU. Sensor ini adalah sensor kelembapan tanah yang mengukur kadar air dalam tanah, kemudian setelah data dari pin analog yang sudah

di kalibrasi dan sudah di konversi ke dalam persentase, jika kondisi tanah kurang dari 60% maka NodeMCU memproses dan relay akan memberikan perintah untuk menyalakan pompa sampai 80% terpenuhi maka relay akan memberikan perintah untuk mematikan pompa, dan alat ini juga menggunakan sensor DHT11 untuk mengukur suhu pada *green house*. Kemudian Aplikasi Arduino Cloud berfungsi sebagai software IoT untuk membuat codingan dan juga di lengkapi dengan monitoring system.

### **3.3 Analisa Rangkaian Secara Diagram Alur (Flowchart)**

Pada rangkaian flowchart ini terbagi menjadi 2 bagian yaitu *flowchart* perancangan alat dan *flowchart* program aplikasi. Pada *flowchart* perancangan alat menjelaskan alur kerja dari alat yang dibuat, dan *flowchart* program aplikasi menjelaskan alur kerja dari aplikasi yang digunakan yaitu Arduino Cloud.

### 3.3.1 Flowchart Perancangan Alat



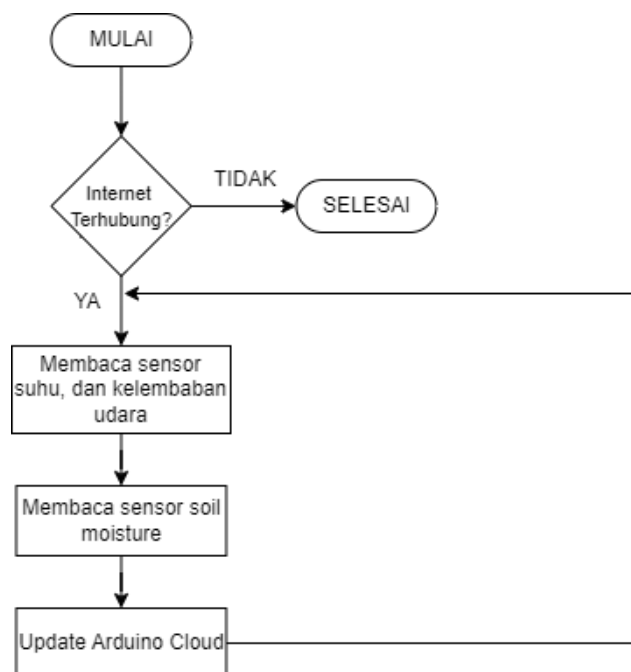
**Gambar 3. 6 Flowchart Perancangan Alat**

Diagram alur (*flowchart*) perancangan alat menjelaskan alur kerja untuk menentukan input dan output pada alat. Berikut adalah penjelasan dari diagram alur (*flowchart*) perancangan alat untuk *green house* pada tanaman cabai dengan menggunakan metode *smart farming* berbasis arduino cloud:

1. Pengecekan keterhubungan aktivator 5V – 12V untuk NodeMcu dan komponen , jika “Tidak” maka proses selesai dan jika “Ya” maka akan melanjutkan ke proses menginisialisasi komponen.
2. Proses menghubungkan NodeMCU ke internet.

3. Membaca sensor soil moisture menunggu data yang diterima dari sensor , lalu data tersebut akan di proses oleh NodeMCU, selanjutnya akan di teruskan ke relay.
4. Membaca sensor DHT11 menunggu data yang diterima dari sensor yaitu suhu dan kelembaban udara, lalu data tersebut akan di proses oleh NodeMCU, selanjutnya ke pembacaan soil moisture.
5. Pengecekan kelembapan tanah kurang dari 60%?, jika “Ya” maka akan diproses oleh NodeMCU dan relay memberikan perintah untuk menyalakan pompa, jika sudah lebih dari 80% maka pompa akan mati.

### 3.3.2 Flowchart Aplikasi IoT Remote



**Gambar 3. 7 Flowchart Aplikasi**

Diagram alur *flowchart* Aplikasi IoT Remote menjelaskan alur kerja alat dengan menggunakan Arduino Cloud. Berikut adalah penjelasan dari diagram alur (*flowchart*) aplikasi IoT Remote untuk perancangan *green house* pada tanaman cabai dengan menggunakan metode *smart farming* berbasis arduino cloud:

1. Proses menghubungkan NodeMCU ke internet.

2. Monitoring pada sensor suhu pada aplikasi IoT Remote yang berfungsi untuk mengetahui apakah suhu ruangan tersebut sudah ideal untuk tanaman cabai.
3. Membaca sensor soil moisture menunggu data yang diterima dari sensor , lalu data tersebut akan di proses oleh NodeMCU,
4. Update Arduino Cloud sinkronisasi waktu dan board yang telah di pasang, dilanjutkan dengan proses monitoring sensor.

### 3.4 Analisis Program

Analisa Program Pada NodeMCU ESP8266 ini dilengkapi dengan contoh dari program isi tabel berupa *source code* dari program yang di buat, program lengkap ada pada lampiran.

**Tabel 3. 1 Listing Program**

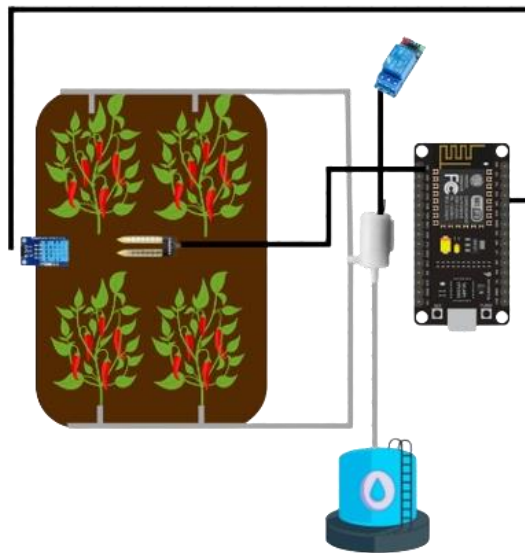
Program	Keterangan
<pre>void loop() {   ArduinoCloud.update();   onMsgChange();   onKelembabanUdaraChange();   onSuhuChange();   onRelayChange() ;   onKelembabanTanahChange(); }</pre>	Void loop berfungsi melakukan pengulangan perintah yang diberikan pada alat dan pada kondisi ini perintah yang dapat diulang terus menerus adalah kondisi pemberian perintah suara pada alat
<pre>void onKelembabanUdaraChange() {   // Add your code here to act upon   KelembabanUdara change   float hm = dht.readHumidity();   Serial.print("kelembaban_udara ");   Serial.println(hm);   kelembaban_udara = hm; }</pre>	Disini adalah codingan untuk membaca kelembaban udara yang ada di sensor DHT11
<pre>void onSuhuChange() {   float temp = dht.readTemperature();   Serial.print("suhu ");</pre>	Disini adalah codingan untuk membaca suhu yang ada di sensor DHT11

Program	Keterangan
<pre>Serial.println(temp); suhu = temp; }</pre>	
<pre>void onMsgChange() {   // Do something   msg = "suhu = " + String (suhu) + " kelembaban_udara      =      "      + String(kelembaban_udara)      +      " kelembaban_tanah      =      "      + String(kelembaban_tanah); }</pre>	<p>Disini saya menggunakan fitur message yang ada pada Arduino cloud.</p>
<pre>void onRelayChange() {   if (kelembaban_tanah &lt; 60) {     digitalWrite(relayPin, HIGH);   }   else if (kelembaban_tanah &gt; 80) {     digitalWrite(relayPin, LOW);   } }</pre>	<p>Mengaktifkan kondisi relay untuk kelembaban tanah</p>
<pre>void onKelembabanTanahChange() {   // Add your code here to act upon   KelembabanTanah change   kelembaban_tanah = analogRead(A0);   kelembaban_tanah      =   map(kelembaban_tanah, 476, 1023, 100, 0);   Serial.print("Persentase      Kelembaban Tanah: ");   Serial.print(kelembaban_tanah);   Serial.println("%");   delay(100); }</pre>	<p>Disini adalah codingan untuk membaca kelembaban tanah yang ada pada sensor soil moisture.</p>



### 3.5 Perancangan Hardware

Pada bagian ini membahas tentang perancangan hardware secara keseluruhan yang sudah terintegrasi dengan Arduino Cloud dan aplikasi IoT Remote sebagai tempat monitoring melalui Handphone.



**Gambar 3. 8 Perancangan Hardware**

### 3.6 Perancangan Software

Perancangan perangkat lunak atau software menggunakan aplikasi Arduino Cloud untuk mengelolah program sistem pada penelitian ini, aplikasi ini disediakan gratis oleh pihak Arduino Cloud termasuk juga library.



**Gambar 3. 9 Perancangan Software**