

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Untuk Untuk mendapatkan hasil penelitian yang optimal, dilakukan kajian dari penelitian penelitian terdahulu, sehingga bisa dijadikan referensi dalam penelitian dengan tujuan agar diperoleh perbandingan kelebihan pada masing-masing perancang.

Pada penelitian terdahulu sebelumnya yang dilakukan oleh (Anggoro, 2021) dalam jurnal yang berjudul “**Perancangan dan Penerapan Kendali Lampu Ruang Berbasis IoT (Internet of Things) Android** ”. Dengan dibangunnya *controller* kendali lampu ruangan berbasis IoT (*Internet of Things*) yang memanfaatkan *smartphone android* dapat membantu masyarakat dalam mengendalikan perangkat elektronik terutama lampu jarak jauh, dimanapun, dan kapanpun, tanpa harus menuju sumber saklar. Sehingga *user* dapat mengontrol perangkat yang jauh dari lokasi terutama pada bangunan bertingkat yang tentunya membutuhkan banyak waktu dan tenaga.

Pada penelitian sebelumnya yang dilakukan oleh (Artono & Putra, 2019) dalam jurnal yang berjudul “**Penerapan Internet of Things (IOT) Untuk Kontrol Lampu Menggunakan Arduino Berbasis WEB**”. *Sistem* dapat memonitor dan dapat difungsikan untuk pengendali lampu dengan *project web* yang dibuat di aplikasi *cayenne*. Dari *web* dan aplikasi *cayenne* di *smartphone* dapat diketahui apakah kondisi lampu dalam keadaan hidup atau mati. *Sensor* cahaya LDR bekerja dengan baik sebagai *input* data dari *sistem* pengendali lampu. Data dapat dikirim ke aplikasi *cayenne* melalui *wifi*. Apabila nilai resistansi LDR tinggi maka lampu dalam kondisi hidup, sebaliknya apabila nilai resistansi LDR rendah maka lampu dalam kondisi mati.

Pada penelitian sebelumnya yang dilakukan oleh (Muzawi, et al., 2018) dalam jurnal yang berjudul “**Prototype Pengendalian Lampu Jarak Jauh dengan Jaringan Internet Berbasis Internet of Things (IoT) Menggunakan Raspberry Pi 3**”. Pada penelitian ini peneliti menggunakan *rasberry* Pi 3. *Raspberry* Pi adalah Salah satu komponen *Internet of Things* (IoT) yang dapat diaplikasikan sebagai pengendali jarak jauh dengan jaringan *internet* yang dapat diaplikasi sebagai pengendali jarak jauh dengan jaringan *internet* yang dapat diterapkan pada peralatan elektronik seperti lampu. Perangkat tersebut dapat diakses dengan layanan *internet* melalui *smartphone android* dengan *Internet Protocol* sehingga tingkat efisiensi tenaga dan waktu jam kerja petugas serta dari segi penghematan energi listrik yang digunakan. Teknologi ini tepat untuk diterapkan karena untuk memudahkan petugas dalam melakukan pekerjaan tersebut.

Pada penelitian sebelumnya yang dilakukan oleh (Purnawan & Rosita, 2019) dalam jurnal yang berjudul “ **Rancang Bangun Smart Home System Menggunakan NodeMCU Esp8266 Berbasis Komunikasi Telegram Mesenger** “ Dalam uji coba pengontrolan dan pemantauan *Smart Home* menggunakan *Telegram*, semua *keyword* beroperasi dengan baik dalam jarak pengujian sejauh 151 Km. Dapat disimpulkan bahwa *sensor* suhu dan kipas angin berfungsi dengan baik sesuai dengan tujuan desainya, yaitu kipas otomatis menyala ketika suhu melebihi 30 derajat *Celcius*.

Pada penelitian sebelumnya yang dilakukan oleh (Mukti, et al., 2022) dalam jurnal yang berjudul “ **Perancangan Smart Home Menggunakan Konsep Internet of Things (IOT) Berbasis Microcontroller** “ Penerapan rangkaian *smart home* yang berhasil mencakup pengendalian lampu, air *conditioner*, pompa air, dan pemantauan suhu ruangan yang telah berhasil dilakukan. Tingkat akurasi yang tinggi dan efisiensi penggunaannya membuat *desain smart home* ini cocok untuk diimplementasikan dalam hunian. Koneksi *internet* memiliki peran penting dalam mengendalikan peralatan rumah.

2.2 Internet Of Things

Internet of Things (IoT) adalah sebuah konsep/skenario dimana suatu objek yang memiliki kemampuan untuk mentransfer data melalui jaringan tanpa memerlukan interaksi manusia ke manusia atau manusia ke komputer (Darmawan, et al., 2020) .

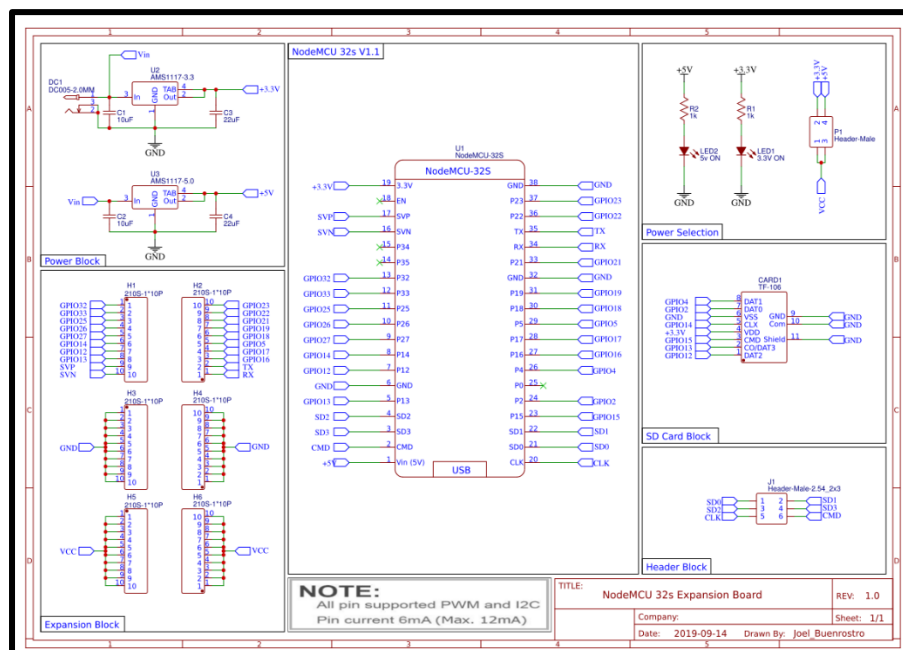
IoT telah berkembang dari konvergensi teknologi nirkabel, *micro-electromechanical systems* (MEMS), dan *Internet*. "A Things" pada *Internet of Things* dapat didefinisikan sebagai subjek misalkan orang dengan monitor implant jantung, hewan peternakan dengan transponder *biochip*, sebuah mobil yang telah dilengkapi *built-insensor* untuk memperingatkan pengemudi ketika tekanan ban rendah. Sejauh ini, IoT paling erat hubungannya dengan komunikasi *machine-to-machine* (M2M) di bidang manufaktur dan listrik, perminyakan, dan gas. Produk dibangun dengan kemampuan komunikasi M2M yang sering disebut dengan *sistem cerdas* atau "smart". (Contoh: *smart label*, *smart meter*, *smart grid sensor*). Meskipun konsep ini kurang populer hingga tahun 1999, namun IoT telah dikembangkan selama beberapa dekade. Alat Internet pertama, misalnya, adalah mesin *Coke* di *Carnegie Mellon University* di awal 1980-an. Para *programer* dapat terhubung ke mesin melalui *Internet*, memeriksa status mesin dan menentukan apakah ada atau tidak minuman dingin yang menunggu mereka, tanpa harus pergi ke mesin tersebut. Istilah IoT (*Internet of Things*) mulai dikenal tahun 1999 yang saat itu disebutkan pertama kalinya dalam sebuah presentasi oleh Kevin Ashton, *cofounder and executive director of the Auto-ID Center* di MIT (Darmawan, et al., 2020).

Internet of Things atau yang sering kita sebut IoT adalah sebuah konsep yang memiliki tujuan memperluas manfaat dari konektivitas *internet* yang tersambung secara terus-menerus. Melalui *internet* kita bisa melakukan berbagi data, *remote control*, dan berbagai hal. Sebenarnya konsep dari apa itu IoT sendiri sangat mudah dipahami oleh setiap orang.

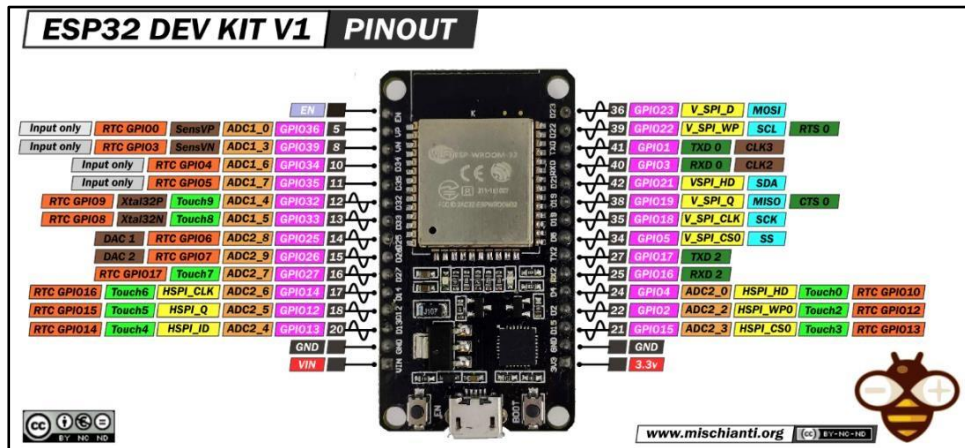
Solusi IoT diterapkan untuk dapat diperluas ke dalam aplikasi rumah pintar, termasuk penerapan kontrol meteran cerdas. IoT menyediakan kemampuan untuk mengukur dan menyimpan data dari *sensor*, berkomunikasi dengan perangkat lainnya, membuat keputusan, dan memisahkan. Penerapan konsep IoT dalam metering cerdas memiliki potensi untuk mengubah bangunan menjadi lingkungan yang sadar energi (Ontowirjo, et al., 2018).

2.3 Mikrokontroler Esp32

ESP32 adalah mikrokontroler yang diproduksi oleh *Espressif System* dan memiliki kemampuan untuk terhubung ke *internet* melalui koneksi nirkabel tanpa kartu tambahan karena *chip* tersebut sudah memiliki modul *wifi*. Secara kinerja, ESP32 memiliki kinerja yang lebih cepat karena memiliki mikroprosesor dengan dua buah inti, serta memiliki arus *Vcc* rentang 2,3-3,6V, ruang *Random Access Memory* (RAM) sebesar 512KB dan *Read-Only Memory* (ROM) yang cukup besar yaitu 4MB sebagai media penyimpanan kode *program*, sehingga dapat menyimpan *program* dengan skala yang banyak. Perbedaan mecolok antara ESP32 dengan mikrokontroler sebelumnya, ESP8266 adalah adanya fitur bluetooth (Taradhyatama, et al., 2022). Pinout ESP32 dapat dilihat pada gambar 2.1.



Gambar 2.1 Skematik Esp32



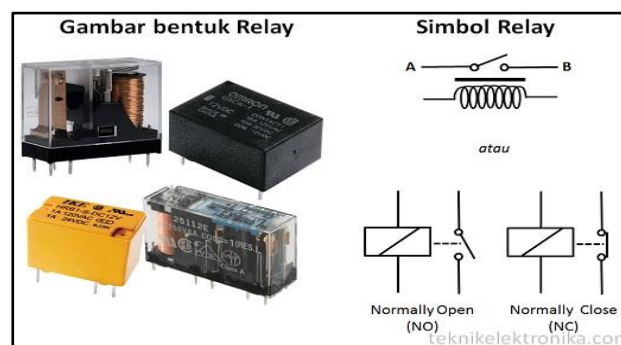
Gambar 2.2 PinOut Esp32

(Sumber : www.mischianti.org)

2.4 Relay

Relay adalah saklar () yang dioperasikan secara listrik dan merupakan komponen *electromechanical* (elektromekanikal) yang terdiri dari 2 bagian utama yakni elektromagnet (*coil*) dan mekanikal (seperangkat kontak saklar/). *Relay* menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan lebih tinggi (Ariana, 2016).

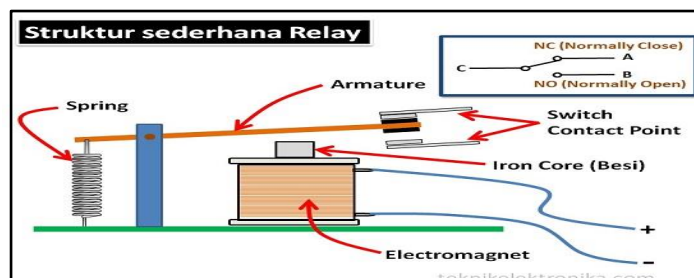
Sebagai contoh, dengan *relay* yang menggunakan elektromagnet 5V dan 50 mA mampu menggerakkan *armature relay* (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A. Bentuk dan simbol *relay* dapat dilihat pada gambar 2.2



Gambar 2.3 Bentuk dan Simbol Relay

2.4.1 Struktur Relay

Dalam *sistem* perancangan saklar otomatis ini, *relay* lah yang bertindak sebagai saklar otomatisnya, yang mana *relay* agar berposisi *normally open* ketika *sensor* mendeteksi objek kemudian sinyal tersebut akan diterima oleh esp32, dan kemudian esp32 berdasarkan *program* yang telah dimasukkan ke *arduino* akan memberikan sinyal perintah agar *relay* berposisi menjadi *normally open* dan menghantarkan listrik ke ruangan tersebut, dalam perancangan ini indikator ruangan tersebut dialiri listrik adalah dengan menggunakan Lampu 3 Watt (Al-afifi, et al., 2021). Struktur sederhana *relay* dapat dilihat pada gambar 2.4.



Gambar 2.4 Struktur Sederhana *Relay*

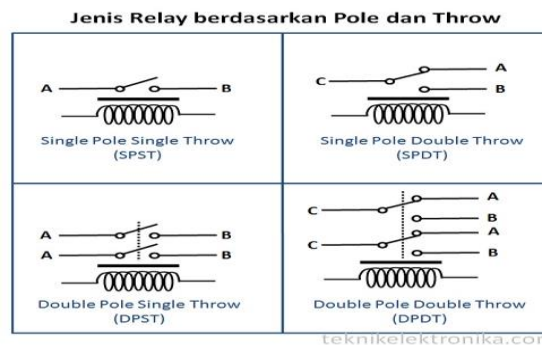
Dari gambar diatas kawat melilit besi selayaknya membentuk kumparan dan memiliki fungsi untuk mengendalikan *armature* tersebut yaitu dengan cara ketika besi dengan lilitan kumparan teraliri dengan energi listrik, maka seketika itu kemudian akan menjadikan besi elektromagnetik seketika yang dapat kemudian akan menarik *armature* yang terbuat dari logam selayaknya sebuah magnet menarik logam sehingga *relay* akan berposisi *normally open* (NO), posisi ini menandakan *relay* bertindak sebagai saklar yang dapat mengalirkan listrik dengan arus yang cukup kecil saja.

2.4.2 Jenis Relay

Berdasarkan penggolongan jumlah *pole* dan *throw*-nya sebuah *relay*, maka *relay* dapat digolongkan menjadi :

1. *Single Pole Single Throw* (SPST) : *Relay* golongan ini memiliki 4 terminal, 2 terminal untuk saklar dan 2 terminal lain untuk *coil*.
2. *Single Pole Double Throw* (SPDT) : *Relay* golongan ini memiliki 5 terminal, 3 terminal untuk saklar dan 2 terminal lagi untuk *coil*.

3. *Double Pole Single Throw (DPST)* : *Relay* golongan ini memiliki 6 terminal, diantaranya 4 terminal yang terdiri 2 pasang terminal saklar sedangkan 2 terminal lainnya untuk *coil*. *Relay* DPST dapat dijadikan 2 saklar yang dikendalikan oleh 1 *coil*.
4. *Double Pole Double Throw (DPDT)* : *Relay* golongan ini memiliki terminal sebanyak 8 terminal, diantaranya 6 terminal yang merupakan 2 pasang *relay* SPDT yang dikendalikan oleh 1 (*single*) *coil*. Sedangkan 2 terminal lainnya untuk *coil*.



Gambar 2.5 Jenis *Relay* Berdasarkan *Pole* dan *Throw*

2.5 Smartphone

Smartphone adalah sebuah benda (alat atau barang elektronik) teknologi kecil yang memiliki fungsi khusus, tetapi sering diasosiasikan sebagai sebuah inovasi atau barang baru. *Smartphone* merupakan ponsel multimedia yang menggabungkan fungsionalitas PC dan *handset* sehingga menghasilkan *gadget* yang mewah, di mana terdapat pesan teks, kamera, pemutar musik, video, *game*, akses *email*, tv digital, *search engine*, pengelola informasi pribadi, fitur GPS, jasa telepon *internet* dan sebagainya (Paridawati, et al., 2021).

Ponsel pintar (*smartphone*) adalah telepon gengam yang mempunyai kemampuan dengan penggunaan dan fungsi yang menyerupai komputer dan contoh manfaat *smartphone* dari sisi *software* adalah tersedianya layanan akses data .

Layanan ini dapat dimanfaatkan oleh setiap *smartphone* untuk memungkinkan penggunaanya terhubung dengan konektivitas *internet* setiap saat dimanapun mereka berada. Untuk segi arsitektur *device* sudah dibekali dengan *inputan* seperti QWERTY miniatur *keyboard* dan *touchscreen* (Paridawati, et al., 2021). *Smartphone* dapat dilihat pada gambar 2.6.



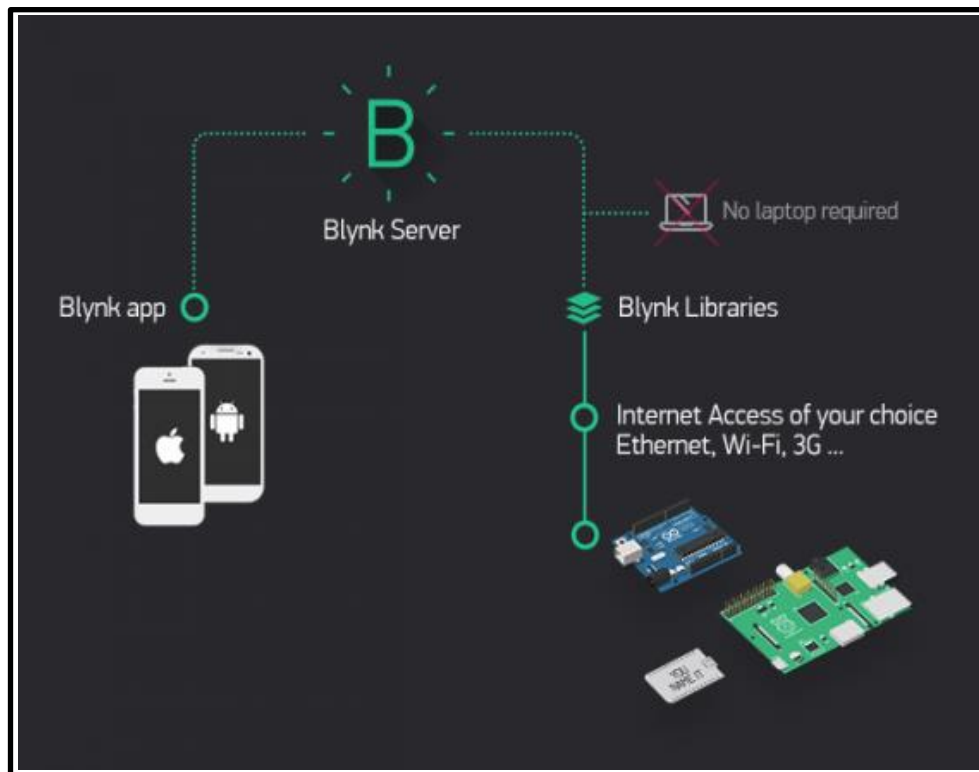
Gambar 2.6 *Smartphone*

2.6 Blynk

Blynk adalah aplikasi yang dibuat sebagai layanan *sistem Internet of Things* (IoT) yang dapat digunakan dalam memberi notifikasi, mengendalikan dan *memonitoring* perangkat *Arduino*, ESP32, ESP8266, dan sejenis lainnya menggunakan *smartphone*. *Blynk* dapat diinstal di *smartphone* secara gratis. Banyak fitur yang ada didalam aplikasi *Blynk* memungkinkan pengguna untuk melakukan pembuatan proyek - proyek seperti *control*, *notifikasi*, *monitoring*, tampilan grafik, dan lain lain (Rofii, et al., 2022).

Blynk adalah *platform* yang mempermudah dalam pembuatan *interface* untuk melakukan *controlling* dan *monitoring* melalui *android*. *Blynk* merupakan *framework* yang berupa aplikasi *android* dan di *desain* untuk *Internet of Things* yang dapat digunakan untuk melakukan *control hardware* secara *remote*, dapat menampilkan data *sensor*, menyimpan data, dan mengvisualisasikannya. Terdapat 3 komponen utama di *platform Blynk* yaitu *Blynk App* yang digunakan untuk membuat *interface* dengan *widget* yang disediakan, *Blynk Server* yang bertanggung jawab tentang semua komunikasi antara *smartphone* dan *hardware*,

dan *Blynk Libraries* yang digunakan untuk komunikasi antara *server* dengan proses *input* dan *output* (Gozal, et al., 2020). Logo *Blynk* dapat dilihat pada gambar 2.7.

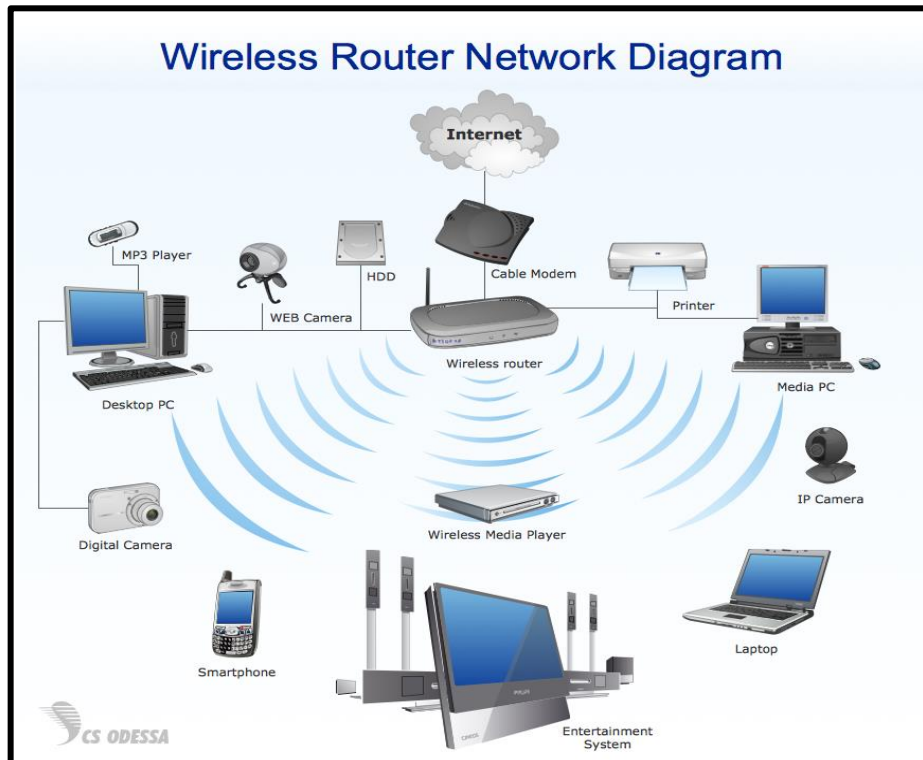


Gambar 2.7 Konsep Dasar *Blynk*

2.7 Wi-Fi

Hotspot (*Wi-Fi*) adalah satu standar *Wireless Networking* tanpa kabel, hanya dengan komponen yang sesuai dapat terkoneksi ke jaringan. *Wi-Fi* merupakan singkatan dari *Wireless Fidelity* yaitu sebuah media penghantar komunikasi data tanpa kabel yang bisa digunakan untuk komunikasi atau mentransfer *program* dan data dengan kemampuan yang sangat cepat (Pratama, et al., 2021).

Teknologi *Wi-Fi* ini merupakan teknologi yang berbasis pada standar IEEE 802. 11. Pemegang merek dagang *Wi-Fi* yaitu *Wi-Fi Alliance* mendefinisikan *Wi-Fi* sebagai “produk jaringan wilayah lokal *nirkabel* (WLAN) apapun yang didasarkan pada standar *Institute of Electrical and Electronics Engineers* (IEEE) 802. 11“. Karena kemampuannya yang memperbolehkan Jaringan Area Lokal (*Local Area Network* atau LAN) untuk beroperasi tanpa memerlukan kabel (*nirkabel*). Logo *Wi-Fi* dapat dilihat pada gambar 2.8.



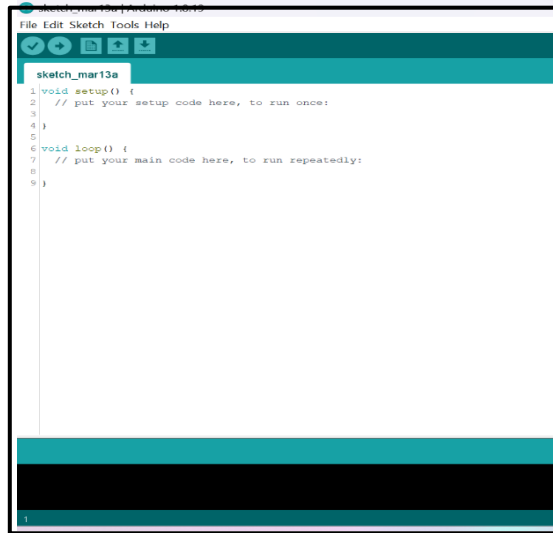
Gambar 2.8 Konsep Dasar *Wi-Fi*

2.8 Arduino IDE

IDE merupakan *akronim* dari (*Integrated Development Environment*) yang dapat diartikan sebagai lingkungan terintegrasi untuk melakukan pengembangan. *Software* ini dikembangkan untuk melakukan bahasa *pemrograman* pada papan *Arduino Uno*. Memanfaatkan *Arduino IDE* selain memiliki perangkat keras juga memiliki perangkat lunak, untuk *hardware* sendiri sudah dijelaskan mengenai karakteristik, jenisnya dan konfigurasi *pin Arduino* yang tersedia. Untuk *software*, *Arduino* memanfaatkan *software* sendiri dan bahasa *pemrograman* sendiri dan dinamakan *Arduino IDE*, bahasa *pemrograman Arduino* yang dipakai yaitu C / C++ (Fathulrohman & Saepuloh, 2018).

Arduino adalah sebuah *platform* komputasi fisik *open source* berbasiskan Rangkain *input /output* sederhana (I/O) dan lingkungan pengembangan yang mengimplementasikan bahasa *Processing*. *Arduino* dapat digunakan untuk mengembangkan *obyek interaktif* mandiri atau dapat dihubungkan ke perangkat lunak pada komputer anda (seperti *Flash*, *Pengolahan*, *VVVV*, atau *Max / MSP*). Rangkaiannya dapat dirakit dengan tangan atau dibeli. IDE

(*Integrated Development Environment*) *Arduino* bersifat *open source* (Fathulrohman & Saepuloh, 2018). *Arduino* IDE dapat dilihat pada gambar 2.9.



Gambar 2.9 *Arduino* IDE

2.9 Lampu Pijar

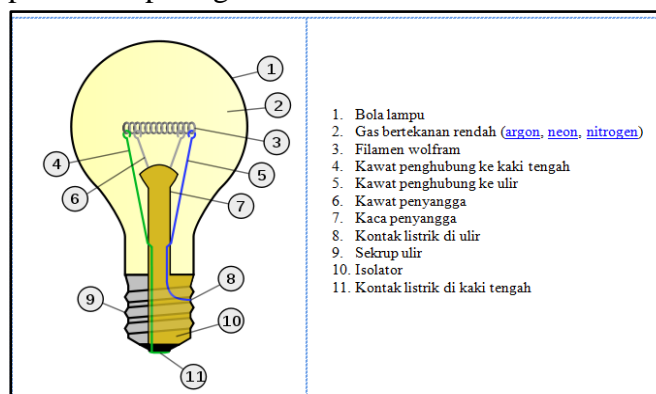
Lampu pijar adalah sumber cahaya buatan yang dihasilkan melalui penyaluran arus listrik melalui *filamen* yang kemudian memanaskan dan menghasilkan cahaya. *Filamen* panas dilindungi dari oksidasi di udara dengan pelindung yang terbuat dari kaca yang diisi dengan gas *inert* atau dievakuasi. Kaca yang menyelubungi *filamen* panas tersebut menghalangi udara untuk berhubungan dengannya sehingga *filamen* tidak akan langsung rusak akibat teroksidasi. Lampu pijar dipasarkan dalam berbagai macam bentuk dan tersedia untuk tegangan (*voltase*) kerja yang bervariasi dari mulai 1,25 *volt* hingga 300 *volt*. Energi listrik yang diperlukan lampu pijar untuk menghasilkan cahaya yang terang lebih besar dibandingkan dengan sumber cahaya buatan lainnya seperti lampu pendar dan *diode* cahaya, maka secara bertahap pada beberapa negara peredaran lampu pijar mulai dibatasi (Wulandani & Aksioma, 2018).

Disamping memanfaatkan cahaya yang dihasilkan, beberapa penggunaan lampu pijar lebih memanfaatkan panas yang dihasilkan, contohnya adalah pemanas kandang ayam, dan pemanas inframerah dalam proses pemanasan di bidang industri. Produksi lampu pijar dilakukan menggunakan lima mesin utama yakni mesin *Flare*, *Stem*, *Mouting*, *Sealex*, dan *Basing*.

Tahap pertama dalam pembuatan lampu pijar berada pada mesin *Flare*. Pada tahap ini, *tube flare* dimasukkan pada mesin *Flare* lalu dilakukan pemotongan. Hasil dari proses ini disebut *flare*.

Tahap selanjutnya dilakukan pada mesin *Stem*. Proses stem merupakan proses penggabungan antara *flare*, *exhaust tube* dan *lead-in wire*. *Mounting* merupakan mesin produksi tahap ketiga. Proses ini diawali dengan memasukkan *stem* ke dalam *head* berupa *konveyor* yang menuju ke *head mounting*. Selanjutnya *lead-in wire* diluruskan ke samping dan ujungnya ditekuk sebagai tempat penjepit *filament*. Setelah *filament* dipasangkan pada kaitan tersebut, kemudian kaitan tersebut ditutup. Akhir dari proses *mounting* ini adalah proses perapatan *lead-in wire* seperti posisi semula.

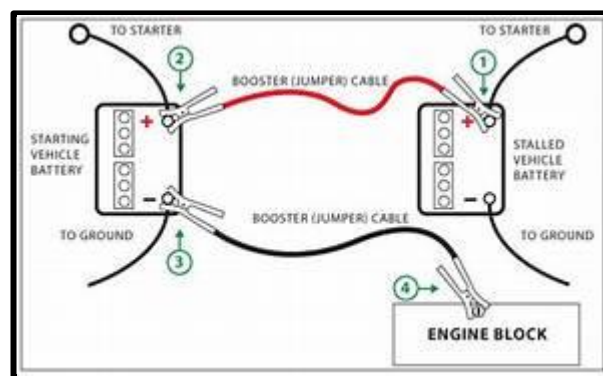
Tahap keempat berada pada mesin *Sealex* yang merupakan gabungan antara mesin *sealing* dan mesin *exhausting*. Pada proses *sealing* lakukan pemanasan untuk menggabungkan hasil *mounting* dengan *glass bulb*. Kemudian hasil *sealing* masuk ke mesin *exhausting*. Pada bagian *exhausting* ini, dilakukan proses pemvakumkan dengan cara menghisap udara yang berada dalam *bulb* dengan menggunakan N₂ melalui bagian tengah *exhaust tube* yang berlubang. Tahap terakhir berada pada mesin *Basing* dimana dilakukan penggabungan *base* yang dinding dalamnya telah diberi *cement* dengan *bulb* lampu setengah jadi (Wulandani & Aksioma, 2018). Lampu Pijar dapat dilihat pada gambar 2.910



Gambar 2.10 Lampu Pijar

2.10 Kabel Jumper

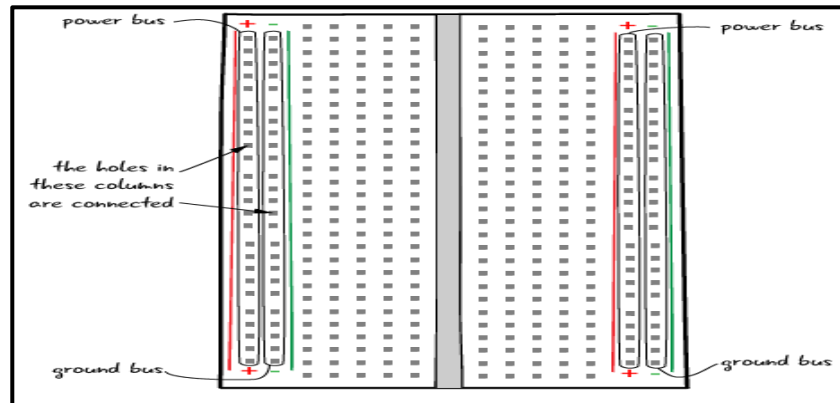
Menurut (Santoso & Kurnia, 2021) “Kabel jumper adalah kabel elektrik yang memiliki pin konektor di setiap ujungnya dan memungkinkan untuk menghubungkan dua komponen yang melibatkan Arduino tanpa melakukan solder. Kabel jumper juga berfungsi sebagai penghubung antara kabel dan PCB dan juga elemen elektronik pada proyek breadboard kabel jumper ialah sebuah kabel yang dipakai untuk mengkonstruksikan alat elektronika serta sejumlah port di elemen elektronika lainnya sesuai dengan fungsinya”. Kemudian kabel tersebut terdapat 2 jenis kabel, yaitu female dan male. Kabel jumper dapat dilihat pada gambar 2.10.



Gambar 2.11 Skematik *Kabel Jumper*

2.11 Breadboard

Breadboard merupakan sebuah papan *board* atau papan yang berfungsi untuk merancang sebuah rangkaian elektronik sederhana. *Breadboard* tersebut nantinya akan dilakukan *prototype* atau uji coba tanpa harus melakukan *solder*. Salah satu keuntungan menggunakan *breadboard* adalah komponen-komponen yang dirakit tersebut tidak akan mengalami kerusakan. Komponen tersebut juga masih bisa dirangkai kembali untuk membentuk rangkaian yang lainnya (Siregar, et al., 2021). *Breadboard* dapat dilihat pada gambar 2.12.

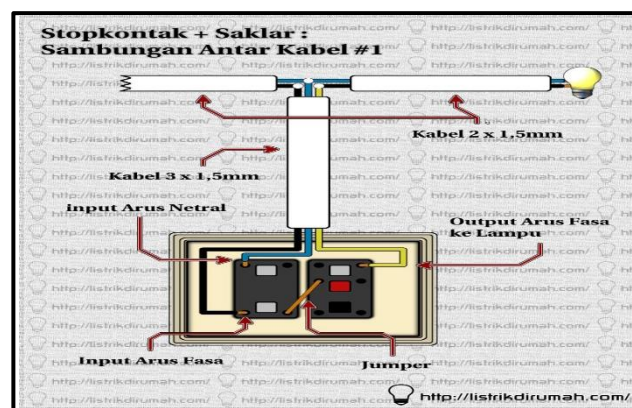


Gambar 2.12 Breadboard

2.12 Stop Kontak

Stop kontak adalah sebuah perangkat yang berfungsi untuk memutus aliran listrik ketika terjadi kontak antara arus *positif*, arus *negatif*, dan *grounding* dalam sistem listrik. Yang lebih penting, ELCB (*Earth Leakage Circuit Breaker*) dapat memutuskan aliran listrik ketika terjadi kontak antara listrik dan tubuh manusia. Biasanya, jika peralatan listrik berfungsi normal, arus total yang mengalir melalui kabel *positif* dan *netral* sama, sehingga tidak ada perbedaan arus. Namun, jika seseorang tersengat listrik, kabel *positif* akan mengalirkan arus tambahan melalui tubuh orang yang tersengat menuju ke tanah (Hesti & Marniati, 2018).

Menurut (Wicaksana, 2016) “Stop kontak merupakan material instalasi listrik yang berfungsi sebagai muara penghubung antara arus listrik dengan peralatan listrik. Agar alat listrik terhubung dengan stop kontak, maka diperlukan kabel dan *stekker* atau colokan yang nantinya akan ditancapkan pada stop kontak”. Stop kontak dapat dilihat pada gambar 2.13.

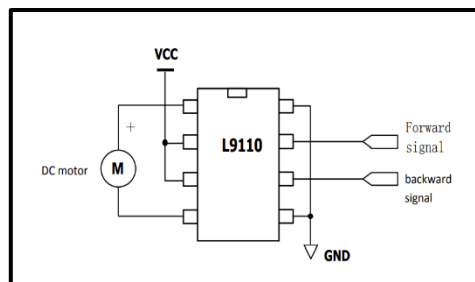


Gambar 2.13 Konsep Dasar Stop Kontak

2.13 Kipas

Kipas adalah alat elektronik yang berfungsi untuk menghasilkan aliran udara dengan cara menggerakkan baling-baling atau bilah-bilah yang terpasang pada poros pusat (Kara & Agargun, 2018). Kipas dapat digunakan untuk meningkatkan sirkulasi udara di dalam ruangan, menghilangkan panas dan kelembaban, serta memberikan kesejukan bagi penggunanya. Kipas dapat dioperasikan secara *manual* atau otomatis dengan menggunakan tombol, *remote control*, atau aplikasi pada *smartphone*. Kipas umumnya digunakan di rumah, kantor, toko, atau tempat-tempat lain yang membutuhkan sirkulasi udara yang lebih baik.

Fungsi umum kipas adalah untuk mendinginkan udara, meyejarkan udara, ventilasi (*exhaust fan*), dan pengering (umumnya memakai komponen penghasil panas). Cara kerja kipas adalah alat yang mengubah energi listrik menjadi energi gerak. Dengan menggunakan motor listrik yang berguna untuk mengubah energi listrik menjadi energi gerak. Dalam motor listrik tersebut, ada kumparan besi yang bergerak dan sepasang *magnet* U pada bagian yang diam. Saat listrik mengalir pada lilitan kawat dalam kumparan besi, peristiwa ini mengubah kumparan besi menjadi *magnet* (Kara & Agargun, 2018). Kipas dapat dilihat pada gambar 2.14.



Gambar 2.14 Skematik Fan Modul L9110

2.14 Flowchart

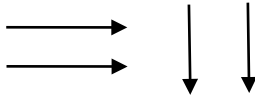
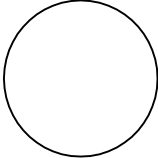
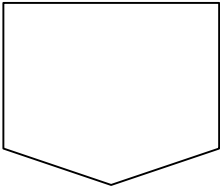
Menurut (Rosaly & Prasetyo, 2019) “*Flowchart* atau sering disebut dengan *diagram alir* merupakan suatu jenis *diagram* yang merepresentasikan *algoritma* atau langkah- langkah instruksi yang berurutan dalam *sistem*. *Flowchart* digambarkan dengan menggunakan simbol-simbol setiap simbol mewakili suatu proses tertentu. Sedangkan untuk menghubungkan satu proses ke proses selanjutnya digambarkan dengan menggunakan garis penghubung.

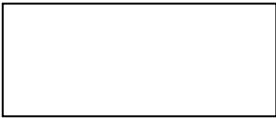
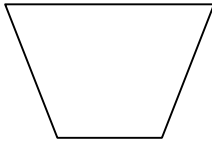
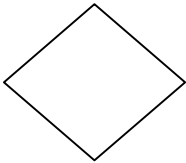
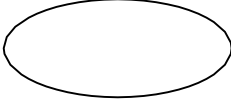
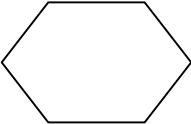

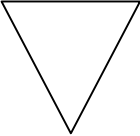
Flowchart dibuat dengan tujuan untuk menunjukkan setiap proses yang harus dilalui dalam suatu *sistem*.”


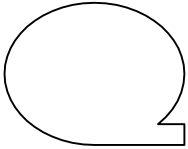



Flowchart atau bagan alur adalah diagram yang menampilkan langkah-langkah dan keputusan untuk melakukan sebuah proses dari suatu *program*. Setiap langkah digambarkan dalam bentuk *diagram* dan dihubungkan dengan garis atau arah panah. *Flowchart* dibuat dengan tujuan untuk menunjukkan setiap proses yang harus dilalui dalam suatu *sistem*.

Agar dapat menjelaskan alur kerja secara maksimal, satu set simbol disepakati secara global. Semua dimulai dari satu set simbol standar yang diresmikan oleh *American National Standard Institute* (ANSI) pada tahun 1960. Setiap simbol memiliki fungsi tersendiri dan harus ditempatkan sesuai praktiknya. *Programmer* tidak perlu lagi menjelaskan panjang lebar tentang bagaimana *program* buatan mereka akan bekerja. Cukup tampilkan *flowchart* maka *audiens* akan dengan mudah memahaminya. Simbol-simbol yang umum dipakai untuk menggambarkan *workflow program* adalah:

Tabel 2.1 Simbol-Simbol Flowchart

NO	SIMBOL	KETERANGAN
1.		Simbol arus/ <i>flow</i> , berfungsi untuk menyatakan jalannya arus suatu proses
2.		Simbol <i>connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
3.		Simbol <i>offline connector</i> , berfungsi untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda

NO	SIMBOL	KETERANGAN
4.		Simbol <i>process</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang dilakukan oleh computer
5.		Simbol <i>manual</i> , berfungsi untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh computer
6.		Simbol <i>decision</i> , berfungsi untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya/tidak
7.		Simbol <i>terminal</i> , berfungsi untuk menyatakan permulaan atau akhir suatu program
8.		Simbol <i>predefined process</i> , berfungsi untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
9.		Simbol <i>keying operation</i> , berfungsi untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i>
10.		Simbol <i>offline-storage</i> , berfungsi untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu

NO	SIMBOL	KETERANGAN
11.		Simbol <i>input/output</i> , berfungsi untuk menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
12.		Simbol <i>magnetic tape</i> , berfungsi untuk menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis
13.		Simbol <i>disk storage</i> , berfungsi untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i>
14.		Simbol <i>document</i> , berfungsi untuk mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i>)
15.		Simbol <i>punched card</i> , berfungsi untuk menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu