

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Menurut Kementerian Kesehatan Republik Indonesia menyatakan bahwa Nilai Ambang Batas (NAB) ukuran debu yang berbahaya adalah debu yang berukuran 0,1 hingga 10 mikron. Ukuran tersebut merupakan Nilai Ambang Batas atau standar yang diperkenankan dalam lingkungan kerja. Nilai Ambang Batas tercantum pada Peraturan Menteri Tenaga Kerja Nomor 13 Tahun 2011 tentang Nilai Ambang Fisik dan Faktor Kimia di Lingkungan Kerja, Menetapkan NAB di Lingkungan Kerja adalah 10 mg/m³.

Sedangkan untuk Nilai Ambang Batas gas CO menurut Peraturan Menteri Tenaga Kerja dan Transmigrasi nomor PER.13/MEN/X/2011 tentang Nilai Ambang Batas Faktor Fisika dan Faktor Kimia di tempat kerja, gas CO memiliki Nilai Ambang Batas sebesar 25 ppm.

Disini penulis membandingkan satu jurnal ataupun tugas akhir yang diambil dari hasil penelitian jurnal tersebut. Untuk keterangan lebih lanjut bisa dibaca pada Tabel 2.1 dibawah ini.

Tabel 2.1 Penelitian Terdahulu

No	Judul Jurnal	Nama Peneliti	Alat & Bahan	Hasil Penelitian
1	Perancangan Sistem Monitoring Parameter Kualitas Udara (Suhu, CO, LPG dan Debu) Berbasis IOT	Jacson Victor Morin, Darma Santi (2021)	Arduino Uno, Sensor LM35, Sensor MQ-9, Sensor MQ-5, Sensor DSM501A, Xbee, Shield Xbee	Penelitian yang telah dilakukan berhasil menghasilkan sebuah sistem yang secara <i>realtime</i> menampilkan data tingkat pencemaran udara gas karbon monoksida

No	Judul Jurnal	Nama Peneliti	Alat & Bahan	Hasil Penelitian
				(CO), kebocoran gas LPG, Suhu serta Debu yang sangat berbahaya bagi kehidupan manusia
2	Perancangan dan Implementasi Alat Pengukur Tingkat Polusi Udara Karbon Monoksida dan Debu Berbasis Website Menggunakan Raspberry Pi	Riza Samsinar, Ichsanul Fikri, Fadliondi (2021)	Raspberry Pi, Sensor MQ-7, Sensor GP2Y1010AU 0F	Berdasarkan hasil penelitian yang telah dilakukan pada alat monitoring kualitas udara karbon monoksida dan debu hanya dapat diukur ketika diluar ruangan seperti jalan raya yang mempunyai rasio kemacetan, dan masih belum bisa diuji coba didalam ruangan
3	Rancang Bangun Alat Monitoring Radioaktivitas Kadar Polusi udara	Yunita, Asep Nurhuda, Dewi Rosita, Andi Yusika, Salmon, Rachmad Fauzi (2021)	Mikrokontroler , Arduino Uno, Sensor GP2Y1010AU 0F, Sensor DHT 11, LCD	Berdasarkan hasil dari pengujian yang telah dilakukan dari tahap pertama hingga tahap kelima dapat disimpulkan bahwa pendeteksi kadar udara ini dapat berfungsi dengan baik, tetapi pada alat ini masih

No	Judul Jurnal	Nama Peneliti	Alat & Bahan	Hasil Penelitian
				menggunakan sistem manual yang belum terhubung ke <i>Internet of Things</i> (IoT)
4	Pengembangan Alat Monitoring Kadar Gas Karbon Monoksida (CO) Berbasis <i>Internet of Things</i>	Mahmud Mustafa, Supriadi, Ainun Mutmainnah (2020)	ESP32, ATmega 328, Sensor MQ-7, Arduino Nano, <i>Power Supply</i> , Regulator <i>Stepdown</i>	Dari hasil penelitian yang menggunakan model pengembangan <i>spiral</i> ini dihasilkan sebuah sistem monitoring kadar gas karbon monoksida yang berbasis <i>Internet of Things</i> (IoT) dengan pengembangannya terletak pada aplikasi monitoring yang digunakan yang dapat mengirimkan data gas secara <i>realtime</i> untuk memudahkan pendeteksian kadar gas CO yang tak terlihat dan tak berbau di sekitar sehingga keberadaan gas dapat diketahui

No	Judul Jurnal	Nama Peneliti	Alat & Bahan	Hasil Penelitian
5	Pembangunan Sistem Monitoring Kualitas Udara dan Gas dalam Ruangan dengan Platform IoT dan Notifikasi via Android	Imam Fadli, Ery Safrianti (2020)	ESP8266, Sensor MQ-7, Sensor MQ-2, Sensor <i>Lasser Dust</i> ZH03A, Sensor PZEM-004t, Arduino Mega, <i>relay</i> , <i>fan exhaust</i> , <i>humidifier</i> , <i>air purifier</i> , <i>UPS</i>	Berdasarkan hasil monitoring kualitas udara dan gas yang dilakukan selama 24 jam, waktu yang dibutuhkan untuk mengirim data ke <i>server thingspeak.com</i> adalah sekitar 30 detik, sesuai dengan pemrograman di android. Notifikasi android memberitahu user jika keadaan listrik PLN mati maka notifikasi akan menampilkan “Listrik Padam!”. Untuk notifikasi selanjutnya adalah jika nilai bacaan dari LPG Gas (300 ppm), CO (200 ppm), dan PM2,5 (151 gr/m3) melebihi standar maka notifikasi akan memberitahu user jika nilai sensor melebihi batas aman

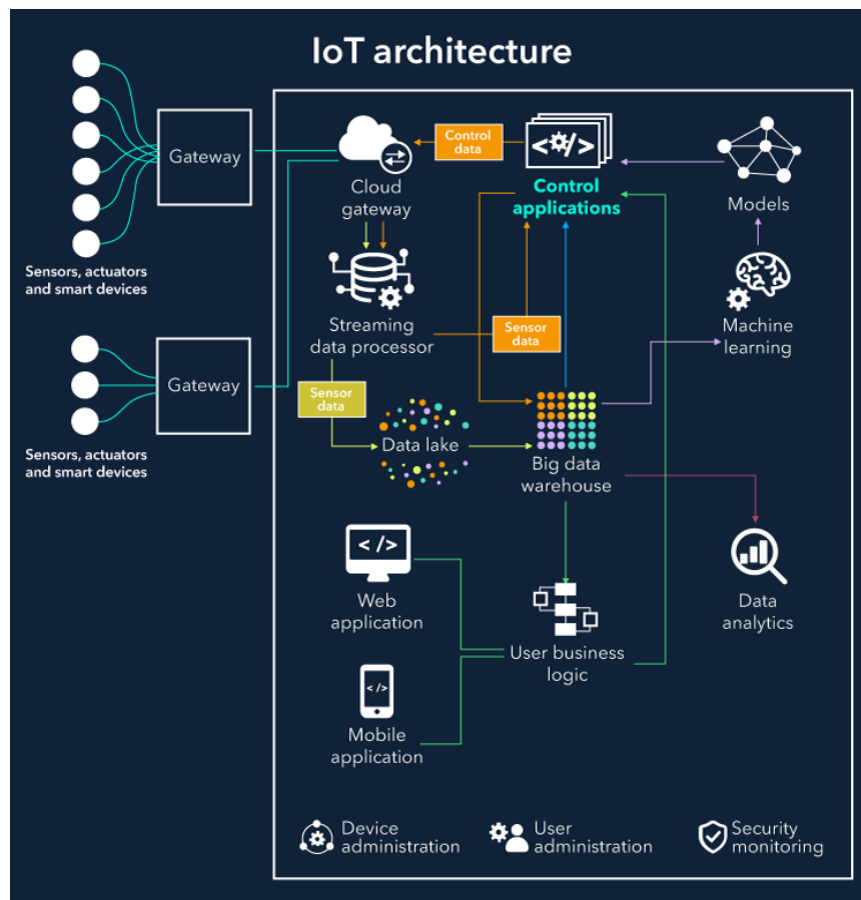
No	Judul Jurnal	Nama Peneliti	Alat & Bahan	Hasil Penelitian
6	Rancang Bangun Alat Monitoring Kualitas Udara Pada Kawasan Industri Berbasis <i>Internet of Things</i> (IoT)	Salman, Amirah (2022)	Sensor MQ-7, Sensor MQ-135, ESP32, LCD	<p>Dari data hasil pengujian pada alat yang diolah dengan menggunakan rumus ISPU maka diperoleh nilai ISPU dari perhitungan untuk senyawa CO (Karbon Monoksida) mendapatkan nilai 1452.8 kategori udara tersebut berada dilevel yang berbahaya sedangkan untuk senyawa CO₂ (Karbon Dioksida) mendapatkan nilai 27.17 ketegori udara sehat maka sesuai dengan aturan standar nilai ISPU dapat disimpulkan bahwa ketegori udara pada area parkir kendaraan Pabrik Semen Tonas dalam kondisi tidak baik karna nilai CO sangat tinggi</p>

2.2 **Arsitektur *Internet of Things***

Konsep IoT pertama kali dicetuskan oleh komunitas pengembang *Radio Frequency Identification* (RFID) sekitar tahun 1999. Konsep ini makin relevan dengan masa sekarang mengingat makin banyaknya pertumbuhan perangkat baik berupa telepon pintar, perangkat tertanam, sensor dan komputasi awan. Perangkat sensor dapat menangkap kondisi lingkungan dan berkomunikasi dengan menyebarkan informasi tersebut ke berbagai perangkat lain.

Kemudian, informasi tersebut bisa digunakan oleh sistem lain untuk menganalisa perilaku menentukan keputusan yang harus diambil. Dengan konsep dan skema tersebut dapat dibuat sebuah sistem yang cerdas.

Model komunikasi *Internet of Things* dapat berupa komunikasi *devices to devices*, model ini menghubungkan dua atau lebih perangkat yang saling terhubung dan berkomunikasi langsung tanpa harus melewati server atau perangkat penghubung lain (Sukaridhoto and ST Ph 2016). Model kedua adalah *devices to cloud*, pada model ini, setiap perangkat harus terhubung ke aplikasi berbasis *cloud* untuk saling mengirimkan dan mendapatkan data. Model komunikasi ketiga adalah *devices to gateway*, bedanya dengan model kedua ialah setiap perangkat atau kelompok perangkat terhubung dahulu melewati perangkat *gateway*.

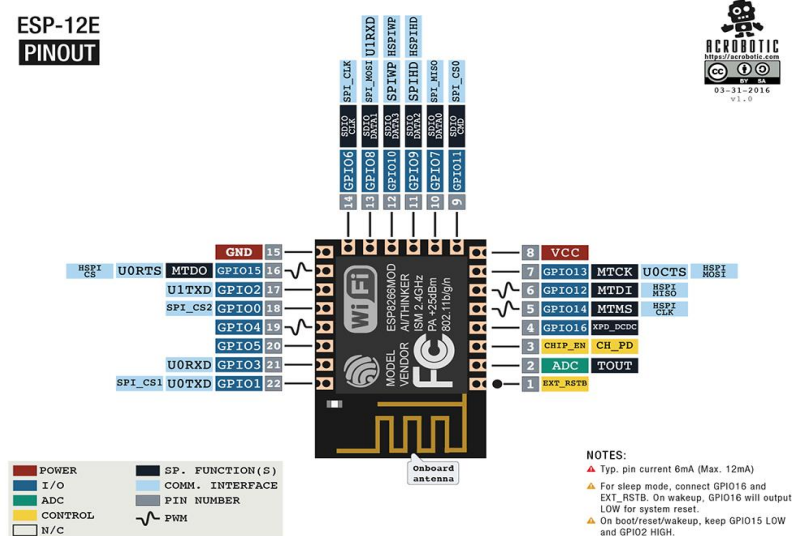


Gambar 2.1 *Arsitektur Internet of Things*

2.3 Mikrokontroler ESP8266

NodeMCU adalah sebuah *platform* IoT yang bersifat *opensource* yang terdiri dari perangkat keras berupa *System on Chip* ESP8266 buatan *Espressif System*. *Firmware* yang digunakan adalah bahasa pemrograman *scripting* Lua. Istilah *nodeMCU* secara *default* sebenarnya mengacu pada *firmware* yang digunakan daripada perangkat keras *development kit*. *NodeMCU* bisa dianalogikan sebagai *board* arduinonya ESP8266 (Wicaksono, 2017).

Pada *NodeMCU* telah menanamkan *chip* *wifi* ESP8266 menjadi sebuah *board* yang memiliki fitur seperti *board* mikrokontroler *arduino*. *Board* ini juga dilengkapi *chip* CH340 untuk komunikasi serial atau sering disebut *USB to serial* sehingga hanya membutuhkan kabel *USB type B* sebagai kabel data. Kabel ini mirip kabel *charging smartphone android*.



Gambar 2.2 Konfigurasi ESP8266

Espressif Systems pertama kali merilis modul *wifi* ESP8266 pada 30 Desember 2013. *NodeMCU* pertama kali dirilis pada 13 Oktober 2014 oleh *Espressif Systems*, perusahaan yang juga mengembangkan ESP8266. Dua bulan kemudian proyek tersebut dikembangkan ke platform perangkat keras ketika Huang R mengunduh *file* dari papan ESP8266, yang diberi nama *devkit v.0.9*.

Pada bulan oktober 2014 juga *library client* MQTT dari *Contiki* ke *platform* ESP8266 untuk *board nodeMCU* sehingga *nodeMCU* mendukung protokol IoT MQTT. Pada Januari 2015 *library LCD* untuk *nodeMCU* sehingga memudahkan *board nodeMCU* menampilkan data melalui LCD. Saat ini *library* untuk *nodeMCU* mencapai 40 *library* yang bisa digunakan sesuai kebutuhan pengembang.

ESP8266 adalah inti dari *nodeMCU* maka fitur – fitur *nodeMCU* kurang lebih sama seperti ESP8266. Pada bahasa pemrograman *eLua* yang mirip dengan *javascript nodeMCU* telah dibungkus oleh API. Perbedaan *nodeMCU* dengan ESP8266 adalah *nodeMCU* memiliki *chip* komunikasi serial sehingga dapat langsung berkomunikasi dengan PC/laptop saat melakukan pemrograman.

Pada gambar modul ESP diatas terdapat penjelasan pin tersebut, VCC yang berfungsi sebagai sumber dengan memberikan muatan positif, GND yang harus dihubungkan dengan muatan negatif, serta pin GPI yang berjumlah 13 buah pin yang berfungsi sebagai pin *input* maupun *output*, TX yang berupa pin *transmitter*

dan RX sebagai pin *receiver* berfungsi dalam pengiriman dan penerimaan data program.

2.4 **Arduino *Integrated Development Environment (IDE)***

Integrited Development Enviroenment (IDE) merupakan program yang digunakan untuk membuat program pada Arduino Uno Program yang ditulis dengan menggunakan *Software Arduino (IDE)* disebut sebagai *sketch*. *Sketch* ditulis dalam suatu editor teks dan disimpan dalam file dengan ekstensi *ino*. Pada *Software Arduino IDE*, terdapat semacam *message box* berwarna hitam yang berfungsi menampilkan status, seperti pesan error, *compile*, dan *upload* program. Di bagian bawah paling kanan *software Arduino IDE*, menunjukkan *board* yang terkonfigurasi beserta *COM Ports* yang digunakan.

1. *Verify/Compile* berfungsi untuk mengecek apakah *sketch* yang dibuat ada kekeliruan dari segi sintaks atau tidak. Jika tidak ada kesalahan, maka sintaks yang dibuat akan *dcompile* ke dalam bahasa mesin.
2. *Upload* berfungsi mengirimkan program yang sudah dikompilasi ke *Arduino Board* (Shofiyullah, 2020).



Gambar 2.3 Tampilan Arduino IDE

Menurut Calvin (2020) Arduino Uno merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Pada tampilan awal arduino IDE terdapat tombol *verify* dapat mengkompilasi program yang ada di editor, Tombol *New* memiliki fungsi membuat program baru dengan mengosongkan isi dari jendela editor. IDE memberikan kesempatan untuk menyimpan semua perubahan yang sebelumnya belum di *save*. Ketika mengklik tombol *upload* Arduino IDE mengkompilasi program dan *upload* ke papan arduino uno yang telah dipilih di IDE menu *Tools* lalu ke *serial port*.

a. Struktur Arduino

Setiap program *Arduino* atau yang sering disebut *Sketch* mempunyai dua buah fungsi yang harus ada. Dua buah fungsi tersebut adalah:

- *Void Loop ()*

Fungsi ini dijalankan setelah fungsi *setup* selesai dijalankan. Setelah selesai dijalankan satu kali fungsi *loop* ini akan kembali menjalankan fungsi *loop* secara terus-menerus sampai catu daya atau *power* dilepaskan.

- *Void Setup ()*

Fungsi ini dijalankan pertama kali ketika *Arduino* dinyalakan. Fungsi ini hanya berjalan sekali. Berbeda dengan fungsi *loop* yang berjalan berkali – kali.

b. Operasi Aritmatika

Operator yang digunakan untuk memanipulasi angka (bekerja seperti matematika sederhana). Berikut ini adalah operator matematika sederhana:

- = (sama dengan)

Digunakan untuk membuat sesuatu *variable* bernilai sama dengan *variable* yang lainnya contohnya adalah $e = 12 * 2$, e sekarang sama dengan 24.

- % (modulo)

Digunakan untuk mendapatkan sisa dari hasil pembagian. Contohnya adalah $25 \% 2$, ini akan menghasilkan angka 1.

- + (penjumlahan)

Digunakan untuk menambahkan dua buah *variable* atau lebih. Contohnya adalah $x = 12 + 13 + 24$.

- - (pengurangan)

Digunakan untuk mengurangi dua buah *variable* atau lebih. Contohnya adalah $x = 12 - 3 - 4$.

- * (perkalian)

Digunakan untuk mengalikan dua buah *variable* atau lebih. Contohnya adalah $x = 12 * 3$.

- / (pembagian)

Digunakan untuk membagi dua buah *variable* atau lebih. Contohnya adalah $x = 12 / 3$.

2.5 Sensor Gas CO (Karbon Monoksida)

Sensor MQ-7 merupakan sensor gas yang digunakan dalam peralatan untuk mendeteksi gas karbon monoksida (CO) dalam kehidupan sehari-hari, industri atau mobil. Sensor MQ-7 memiliki sensitivitas tinggi dan respon cepat terhadap gas karbon monoksida dan keluaran dari sensor MQ-7 berupa sinyal analog dan membutuhkan tegangan DC sebesar 5 Volt. Fitur dari sensor gas MQ-7 ini adalah mempunyai sensitivitas yang tinggi terhadap karbon monoksida (CO), stabil, dan berumur panjang (Wasista dkk, 2019).



Gambar 2.4 Sensor Gas CO (Karbon Monoksida)

Pada dasarnya prinsip kerja dari sensor MQ-7 adalah mendeteksi keberadaan gas-gas yang dianggap mewakili asap kendaraan yang mengandung gas karbon monoksida atau CO. Ketika sensor mendeteksi gas maka resistansi elektrik sensor akan menurun. Didalam sensor memiliki suatu penyerap keramik yang berfungsi untuk melindungi dari debu atau gas yang tidak diketahui. Heater pada sensor ini berfungsi sebagai pemicu sensor untuk dapat mendeteksi gas yang diharapkan setelah diberikan tegangan. Adapun spesifikasi pada sensor MQ-7 diantara lain:

Tabel 2.2 Spesifikasi Sensor MQ-7

Komponen	Spesifikasi
VC / (Tegangan Rangkaian)	5V±0.1
VH (H) / Tegangan Pemanas (Tinggi)	5V±0.1
VH (L) / Tegangan Pemanas (Rendah)	1.4V±0.1
RL / Resistansi Beban	Dapat disesuaikan
RH / Resistansi Pemanas	33Ω±5%
TH (H) Waktu Pemanasan (Tinggi)	60±1 <i>seconds</i>
TH (L) Waktu Pemanasan (Rendah)	90±1 <i>seconds</i>
PH Konsumen Pemanasan	Sekitar 350mW

Tabel 2.3 PIN Sensor MQ-7

PIN	Nama PIN	Deskripsi
1	Vcc	Digunakan untuk menyalakan sensor, umumnya tegangan operasi + 5 Volt
2	Ground	Digunakan untuk menghubungkan ke sistem
3	H-Pin	Terhubung dengan rangkaian pengendali heater
4	A-Pin	Terhubung dengan tegangan catu daya 5 Volt
5	B-Pin	Terhubung dengan rangkaian pengkondisi sinyal

2.6 Sensor Debu

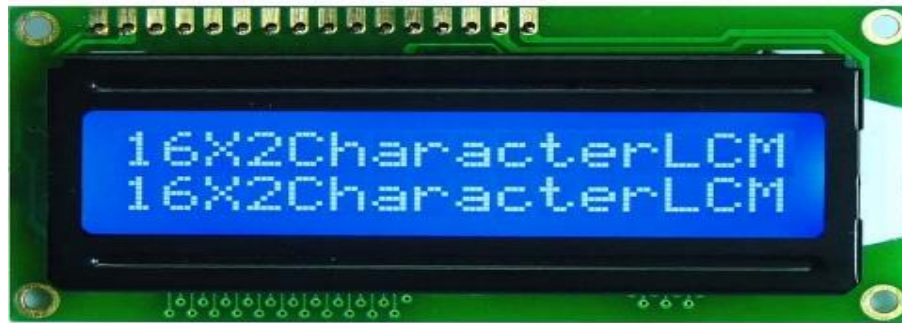
Sensor DSM501A ini dapat menghasilkan *output* ke *particulate matters* (PM) atau polusi partikel yaitu istilah untuk campuran partikel padat dan tetesan cairan yang ditemukan di udara. Prinsip kerja dari sensor ini ialah dengan memancarkan cahaya melalui debu ataupun partikel lain yang kemudian akan dipantulkan ke bagian penerima. Cahaya dicerminkan pada partikel melewati keseluruhan permukaan, kemudian oleh *fotodiode* diubah menjadi tegangan. Tegangan harus diperkuat untuk dapat membaca perubahan. *Output* dari sensor adalah tegangan analog sebanding dengan kepadatan debu yang terukur, dengan sensitivitas 0,5 V/0,1 mg/m³ (Riki Ruli A. Siregar dkk, 2017).

**Gambar 2.5** Sensor Debu

2.7 *Liquid Crystal Display*

Liquid Cristal Display (LCD) adalah salah satu jenis *display* elektronik yang dibuat dengan teknologi CMOS *logic* yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap *front-lit* atau mentransmisikan cahaya dari *back-lit*. LCD berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik. LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan *seven-segment* dan lapisan elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. Lapisan *sandwich* memiliki *polarizer* cahaya vertikal depan dan *polarizer* cahaya horizontal belakang yang diikuti dengan lapisan *reflektor*. Cahaya yang dipantulkan tidak dapat melewati molekul-molekul yang telah menyesuaikan diri dan segmen yang diaktifkan terlihat menjadi gelap dan membentuk karakter data yang ingin ditampilkan (Natsir, 2019).

Menurut (Muhammad Royhan, 2018) Layar LCD merupakan suatu media penampilan data yang sangat efektif dan efisien dalam penggunaannya. Untuk menampilkan sebuah karakter pada layar LCD diperlukan beberapa rangkaian tambahan untuk lebih memudahkan para pengguna, beberapa perusahaan elektronik menciptakan modul LCD untuk menampilkan angka atau teks. LCD yang digunakan untuk tampilan pengaturan menggunakan LCD 16x2 dibagi menjadi dua bagian yaitu bagian depan panel LCD yang terdiri dari banyak dot atau titik LCD dan mikrokontroler yang menempel pada bagian belakang panel LCD yang berfungsi untuk mengatur titik-titik LCD sehingga dapat menampilkan huruf, angka, dan simbol khusus yang dapat terbaca. Karakteristik LCD 16x2: 16 karakteristik x 2 baris, 5x7 titik matriks karakter + kursor, HD44780 *equivalent* LCD *controller/diver built-in*, 4 bit/8 bit MPU *interface* tipe standar, bekerja hampir semua mikrokontroler.



Gambar 2.6 LCD 16 x 2

Tabel 2.4 Spesifikasi LCD 16 x 2

PIN	Simbol	Fungsi
1	Vss	Ground
2	Vdd	+3 Volt atau +5 Volt
3	Vo	Pengatur kontras
4	Rs	<i>H/L Register select signal</i>
5	R/W	<i>Read/write signal</i>
6	EN	<i>Enable signal</i>
7 – 14	Data	<i>Input/Output pins</i>
15	Anoda	Tegangan positif
16	Katoda	Tegangan negatif

2.8 Buzzer

Buzzer Elektronika adalah sebuah komponen elektronika yang dapat menghasilkan getaran suara berupa gelombang bunyi. Buzzer elektronika akan menghasilkan getaran suara ketika diberikan sejumlah tegangan listrik dengan taraf tertentu sesuai dengan spesifikasi bentuk dan ukuran buzzer elektronika itu sendiri. Pada umumnya, buzzer elektronika ini sering digunakan sebagai alarm karena penggunaannya yang cukup mudah yaitu dengan memberikan tegangan input maka buzzer elektronika akan menghasilkan getaran suara berupa gelombang bunyi yang dapat didengar manusia (Hidayatullah, 2020).

Pada dasarnya Buzzer Elektronika menyerupai *loudspeaker* namun memiliki fungsi-fungsi yang lebih sederhana. Berikut adalah beberapa fungsi buzzer elektronika:

- 1) Sebagai bel rumah
- 2) Alarm pada berbagai peralatan
- 3) Peringatan mundur pada truk
- 4) Komponen rangkaian anti maling
- 5) Indikator suara sebagai tanda bahaya atau yang lainnya
- 6) Timer



Gambar 2.7 Buzzer

2.9 Blynk

Blynk adalah platform untuk IOS atau ANDROID yang digunakan untuk mengendalikan *module* arduino, Raspberry Pi, Wemos dan *module* sejenisnya melalui *internet*. Aplikasi ini sangat mudah digunakan bagi orang yang masih awam. Aplikasi ini memiliki banyak fitur yang memudahkan pengguna dalam memakainya. Cara membuat projek di aplikasi ini sangat mudah, tidak sampai 5 menit yaitu dengan cara *drag and drop*. Blynk tidak terkait dengan *module* atau papan tertentu. Dari aplikasi inilah kita dapat mengontrol apapun dari jarak jauh

dimana pun kita berada dengan catatan terhubung dengan *internet*. Hal inilah yang disebut dengan IoT (*Internet of Things*).



Gambar 2.8 Aplikasi Blynk

2.10 *Flowchart*

Flowchart adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan *flowchart* akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu *flowchart* juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek. *Flowchart* membantu memahami urutan-urutan logika yang rumit dan panjang. *Flowchart* membantu mengkomunikasikan jalannya program ke orang lain (bukan pemrogram) akan lebih mudah (Santoso, 2017).


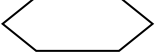

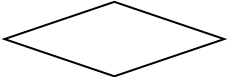

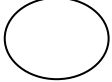
Menurut Wicaksono (2017), *flowchart* adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program. Diagram alur dapat menunjukkan secara jelas, arus pengendalian suatu algoritma

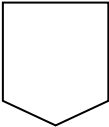
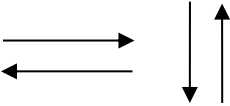



yakni bagaimana melaksanakan suatu rangkaian kegiatan secara logis dan sistematis.

Simbol-simbol yang di pakai dalam *flowchart* dibagi menjadi 3 kelompok:

- a) *Flow direction symbol*, digunakan untuk menghubungkan simbol satu dengan yang lain, disebut juga *connecting line*.
- b) *Processing symbols*, menunjukkan jenis operasi pengolahan dalam suatu proses/prosedur.
- c) *Input/Output symbol*, menampilkan jenis peralatan yang digunakan sebagai media *input* atau *output*.

Tabel 2.5 Simbol-Simbol *Flowchart*

Simbol	Keterangan
Terminal 	Digunakan untuk menunjukkan awal dan akhir dari program
Persiapan 	Digunakan untuk memberikan nilai pada awal suatu variabel atau <i>counter</i>
Proses 	Digunakan untuk mengolah aritmatika dan pemindahan data
Keputusan 	Digunakan untuk mewakili operasi perbandingan logika
Proses 	Digunakan untuk proses yang detailnya dijelaskan terpisah, misalnya dalam bentuk <i>subroutine</i>
Connector 	Digunakan untuk menunjukan hubungan arus proses yang terputus masih dalam halaman yang sama

Simbol	Keterangan
<p data-bbox="389 421 552 450">Penghubung</p> 	<p data-bbox="651 427 1358 568">Digunakan untuk menunjukkan hubungan arus dari suatu proses yang terputus dalam halaman yang berbeda</p>
<p data-bbox="437 645 504 674">Arus</p> 	<p data-bbox="651 689 1114 719">Penghubung antar prosedur / proses</p>
<p data-bbox="402 842 539 871"><i>Document</i></p> 	<p data-bbox="651 860 1358 943">Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> di cetak dikertas</p>
<p data-bbox="386 1048 555 1077"><i>Input-Output</i></p> 	<p data-bbox="651 1061 1358 1144">Simbol yang menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya</p>
<p data-bbox="386 1249 555 1279"><i>Disk Storage</i></p> 	<p data-bbox="651 1263 1358 1346">Simbol untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i>.</p>