

BAB II

LANDASAN TEORI

2.1. Perangkat Keras

Perangkat keras pada rancang bangun alat pendeteksi kadar gas alcohol yang dibutuhkan disini berupa kumpulan komponen elektronika yang nantinya akan dirangkai secara keseluruhan dan akan diupload sebuah program kedalam IC mikrokontroller yang digunakan.

2.1.1. IC (Integrated Circuit)

(ITB 2011) menyatakan bahwa “*Integrated Circuit* (IC) juga dapat diartikan Suatu komponen elektronika yang dibuat dari bahan semikonduktor dan merupakan pengembangan dari transistor yang telah menjadi sebuah rangkaian berbentuk chip kecil”.

Sirkuit terintegrasi atau yang biasa juga disebut sebagai IC merupakan komponen elektronika yang terbuat dari kumpulan puluhan, ratusan, hingga ribuan transistor, resistor, diode, dan komponen elektronika lainnya. Kumpulan komponen-komponen tersebut dikemas dengan sedemikian rupa sehingga tidak terlalu besar.



Sumber: https://sea.banggood.com/1pc-Microcontroller-IC-ATMEGA328P-AU-TQFP-32-ATMEL-Chip-p-919916.html?cur_warehouse=CN

Gambar II.1 *Integrated Circuit* (IC)

2.1.2. Sumber Tegangan

(A Kadir 2018)mengatakan bahwa “sumber tegangan atau kadang dinamakan sumber daya adalah komponen yang memberikan pasokan listrik dalam suatu rangkaian. Sebagai contoh, sumber tegangan untuk papan Arduino diambilkan dari PC melalui kabel USB”.

Pada pembuatan alat ini dibutuhkan sumber tegangan yang tidak terlalu besar, sehingga pada alat ini hanya di perlukan sebuah baterai. Baterai merupakan salah satu contoh power supply dengan tegangan dan ukuran yang relative kecil. Baterai mengandung kutub positif dan negative sehingga pemasangannya pada rangkaian perlu diperhatikan. Kutub positif baterai ke potensial tinggi (+) rangkaian elektronika, dan kutub negative baterai dihubungkan ke potensial rendah (-).

Baterai yang digunakan pada perancangan alat ini adalah baterai dengan sumber tegangan 9 – 12 V.



Sumber: <https://www.bukalapak.com/p/elektronik/baterai-681/ebaxub-jual-baterai-kotak-9v-hw-batere-hi-watt-1604e-6f22-batre-battery-arduino>

Gambar II.2 Baterai

2.1.3. Komponen Elektronika

Komponen-komponen elektronika dibagi dalam dua jenis, yaitu komponen pasif dan komponen aktif :

1. Komponen Pasif

(Andrianto and Darmawan 2016) mengatakan bahwa "Komponen Pasif yaitu komponen yang tidak dapat menguatkan dan menyearahkan sinyal listrik serta tidak dapat mengubah energi ke bentuk lainnya".

Berikut adalah beberapa contoh Komponen Pasif:

a. Resistor

(Abdul Kadir 2019) mengatakan bahwa "Resistor adalah komponen elektronika yang digunakan untuk membatasi arus yang mengalir dalam suatu rangkaian".

Hampir bisa dipastikan semua rangkaian elektro pasti memiliki resistor sebagai salah satu elemennya.



Sumber: <https://www.codrey.com/resistor/resistor-basics/>

Gambar II.3 Resistor

b. Kapasitor

(Adi 2010) menyimpulkan bahwa: “Kapasitor adalah komponen yang menyimpan muatan listrik”. Besaran yang menunjukkan nilai kapasitor disebut kapasitansi. Satuan kapasitansi adalah Farad (F).



Sumber: <https://serviceacjogja.pro/fungsi-kapasitor/>

Gambar II.4 Kapasitor

Kapasitor dibagi menjadi dua yaitu kapasitor tetap dan kapasitor tidak tetap.

- a) Kapasitor tetap adalah kapasitor yang nilai kapasitansinya tidak dapat dirubah.
- b) Kapasitor variable adalah kapasitor yang nilai kapasitansinya bisa diubah atau tidak tetap.

Fungsi dari kapasitor dalam rangkaian yaitu:

- a) Sebagai filter dalam rangkaian *power supply*.
- b) Sebagai kopling antara rangkaian yang satu dengan rangkaian yang lain pada *power supply*.
- c) Sebagai penyimpan tegangan listrik.
- d) Sebagai pembangkit frekuensi dalam rangkaian osilator.
- e) Sebagai isolator yang menghambat arus *Direct Current*.

c. Induktor

(Adi 2010) mengemukakan bahwa: “induktor adalah komponen yang menyimpan energi dalam bentuk medan magnet”.

Biasanya berbentuk kumparan dengan inti udara, besi atau ferit. Induktor memiliki karakteristik yang berlawanan dengan kapasitor yaitu menahan arus AC dan meneruskan arus DC.



Sumber: <http://www.niguru.com/2018/02/induktor-lilitan-koil-l-komponen-untuk.html>

Gambar II.5 Induktor

2. Komponen Aktif

(Andrianto and Darmawan 2016) mengatakan bahwa “Komponen aktif adalah komponen yang dapat menguatkan dan menyearahkan sinyal listrik, serta mengubah energi dari satu bentuk ke bentuk lainnya”.

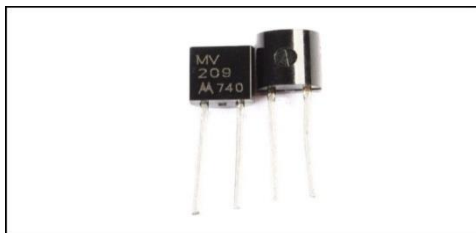
Berikut adalah beberapa contoh Komponen Aktif :

a. Dioda

(Adi 2010) mengemukakan bahwa: “Dioda adalah komponen paling sederhana yang terbuat dari bahan semikonduktor”.

Dioda atau sering juga disebut sebagai rectifier/penyearah hanya dapat

melewatkan arus pada satu arus saja



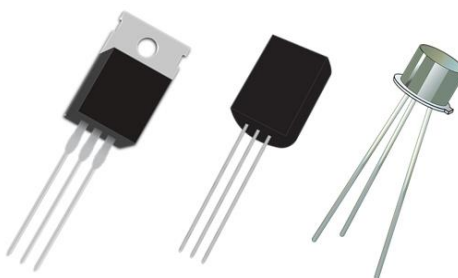
Sumber: <http://informazone.com/jenis-jenis-dioda/>

Gambar II.6 Dioda

b. Transistor

(ITB 2011) menyimpulkan bahwa: “Transistor merupakan suatu komponen dengan tiga kaki”.

Transistor muncul sebagai alternatif dari *vacuum tube*. Dengan komponen ini, arus atau tegangan kecil yang diberikan pada satu kaki dapat mengatur arus yang lebih besar melalui dua kaki lainnya. Dengan kata lain, seperti halnya *vacuum tube*, transistor dapat digunakan sebagai *amplifier* (penguat) dan *switch*.



Sumber: <https://www.circuitspedia.com/transistor-as-switch-working-how-transistor-works/>

Gambar II.7 Transistor

2.1.4. Sensor

(Adi 2010) menyatakan bahwa “ Sensor adalah suatu elemen pada sistem mekatronika atau sistem pengukuran yang menerima sinyal masukan berupa parameter/besaran fisik dan mengubahnya menjadi sinyal/besaran lain”.

Sensor yang digunakan dalam perancangan alat pendeteksi kadar gas alcohol ini adalah :

a. Sensor Gas MQ-3

Elemen sensor MQ-3 terdiri atas lapisan SnO_2 dengan konduktivitas yang kecil dalam udara bersih. Resistansi sensor akan berubah-ubah seiring dengan terdeteksinya keberadaan gas etanol oleh elemen sensor. Jika konsentrasi etanol tinggi maka resistansi sensor akan berkurang. Sensor MQ-3 cocok digunakan untuk mendeteksi kadar gas alkohol secara langsung misalnya pada saat bernafas dan pada saat botol minuman dibuka. Pada rancangan alat ini menggunakan satu buah variabel resistor.



Sumber: <https://www.indiamart.com/proddetail/mq3-gas-sensor-module-alcohol-12392978548.html>

Gambar II.8 Sensor Gas MQ-3

2.1.5. LCD (*Liquid Crystal Display*)

(Andrianto and Darmawan 2016) mengatakan bahwa “LCD (*Liquid Crystal Display*) 16 x 2 adalah suatu *display* dari bahan cairan Kristal yang pengoperasiannya menggunakan sistem dot matriks. LCD (*liquid crystal display*) 16 x 2 dapat menampilkan sebanyak 32 karakter yang terdiri dari 2 baris dan tiap baris dapat menampilkan 16 karakter”.

LCD berfungsi sebagai media penampil informasi dalam bentuk huruf atau angka yang dapat menampung 16 huruf atau angka disetiap baris. LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan 7 segmen dan lapisan elektroda pada tampilan kaca belakang.

Adapun fungsi dari pin-pin yang terdapat pada LCD :

1. Pin 1 dan 2 Merupakan sambungan catu daya, Vss, dan Vdd. Pin Vdd dihubungkan dengan tegangan positif catu daya, sedangkan Vss pada 0 volt atau ground. Meskipun demikian, data menentukan catu 5 Vdc (hanya beberapa mA), menyediakan 6V dan 4,5V yang keduanya bekerja dengan baik, bahkan 3V cukup untuk beberapa modul.
2. Pin 3 Merupakan pin kontrol Vcc yang digunakan untuk mengatur kontras display. Idealnya pin ini dihubungkan dengan tegangan yang bisa diubah untuk memungkinkan pengaturan tingkatan kontras display sesuai kebutuhan.

3. Pin 4 Merupakan register select (RS), masukan yang pertama dari 3 command control input. Dengan membuat RS menjadi high, data karakter dapat ditransfer dari dan menuju modulnya.
4. Pin 5 Merupakan Read/Write (R/W). Cara memfungsikan perintah Write adalah R/W low atau menulis karakter ke modul. R/W high untuk membaca data karakter atau informasi status registernya.
5. Pin 6 Merupakan Enable (E). Input ini digunakan untuk transfer aktual perintah-perintah atau karakter antara modul dengan hubungan data. Ketika menulis ke display, data ditransfer hanya pada perpindahan high/low. Namun, ketika membaca dari display, data akan menjadi lebih cepat tersedia setelah perpindahan dari low ke high dan tetap tersedia hingga sinyal low kembali.
6. Pin 7 sampai 14 Pin 7 sampai 14 adalah jalur 8 jalur data (D0-D7) dimana data dapat ditransfer ke dan dari display.
7. Pin 15 dan 16 Pin 15 atau A (+) mempunyai level DC +5V dan berfungsi sebagai LED backlight +, sedangkan pin 16 atau K (-) memiliki level 0V dan berfungsi sebagai LED backlight.



Sumber: <https://encryptedtbn0.gstatic.com/images?q=tbn:ANd9GcSyX3wudQ0lJqefrL9BdlB8R1IX0UISA2i97KJSGxqvSdWkCVIg>

Gambar II.9 LCD

2.1.6. Buzzer

(A Risal 2017) Buzzer adalah komponen yang merubah besaran elektrik menjadi besaran suara. Penggunaan buzzer untuk anatarmuka mikrokontroler sama prinsipnya dengan LED. Yang diperlukan hanya menghubungkan salah satu PIN dari mikrokontroler ke kaki positif buzzer, dan kaki satunya ke GND rangkaian.



Sumber : <https://www.taydaelectronics.com/piezo-electronic-tone-buzzer-alarm-3-28v.html>

Gambar : II.10 BUZZER

2.1.7. Push Button

(A Risal 2017) Push button switch (saklar tombol tekan) adalah perangkat / saklar sederhana yang berfungsi untuk menghubungkan atau memutuskan aliran arus listrik dengan sistem kerja tekan unlock (tidak mengunci). Sistem kerja unlock disini berarti saklar akan bekerja sebagai device penghubung atau pemutus aliran arus listrik saat tombol ditekan, dan saat tombol tidak ditekan (dilepas), maka saklar akan kembali pada kondisi normal. Sebagai device penghubung atau pemutus, push button switch hanya memiliki 2 kondisi, yaitu On dan Off (1 dan 0). Istilah On dan Off ini menjadi sangat penting karena semua perangkat listrik yang memerlukan sumber energi listrik pasti membutuhkan kondisi On dan Off. Karena sistem kerjanya yang unlock dan langsung berhubungan dengan operator, push button switch menjadi device paling

utama yang biasa digunakan untuk memulai dan mengakhiri kerja mesin di industri. Secanggih apapun sebuah mesin bisa dipastikan sistem kerjanya tidak terlepas dari keberadaan sebuah saklar seperti push button switch atau perangkat lain yang sejenis yang bekerja mengatur pengkondisian On dan Off. Berdasarkan



<https://hidkitpros.com/shop/momentary-push-button-switch-red-large/>

Gambar : II.11 Prinsip Kerja Push Button

fungsi kerjanya yang menghubungkan dan memutuskan, push button switch mempunyai 2 tipe kontak yaitu NC (Normally Close) dan NO (Normally Open).

Data Digital Mikrokontroler AVR 45

1. NO (Normally Open), merupakan kontak terminal dimana kondisi normalnya terbuka (aliran arus listrik tidak mengalir). Dan ketika tombol saklar ditekan, kontak yang NO ini akan menjadi menutup (Close) dan mengalirkan atau menghubungkan arus listrik. Kontak NO digunakan sebagai penghubung atau menyalakan sistem circuit (Push Button ON).
2. NC (Normally Close), merupakan kontak terminal dimana kondisi normalnya tertutup (mengalirkan arus listrik). Dan ketika tombol saklar push button ditekan, kontak NC ini akan menjadi membuka (Open), sehingga memutus aliran arus

listrik. Kontak NC digunakan sebagai pemutus atau mematikan sistem circuit (Push Button Off).

2.1.8. Mikrokontroler

(Andrianto and Darmawan 2016) mengatakan bahwa “mikrokontroller adalah mikrokontroller (pengendali mikro) pada suatu rangkaian elektronik berfungsi sebagai pengendali yang mengatur jalannya proses kerja dari rangkaian elektronik. Didalam sebuah IC mikrokontroller terdapat CPU, memori, timer, saluran komunikasi serial dan paralel, port *input/output*, ADC, dll”.

2.1.9. Arduino UNO

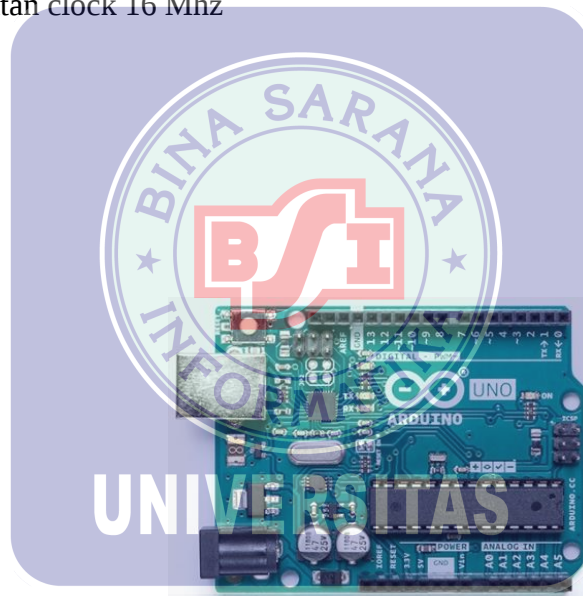
(Andrianto and Darmawan 2016) mengatakan bahwa “Arduino adalah sebuah *broad* mikrokontroler yang bersifat *open source*, dimana desain skematik dan PCB bersifat *open-source*, sehingga kita dapat menggunkannya maupun melakukan modifikasi”.

Arduino merupakan papan rangkaian system minimum mikrokontroler yang memang dirancang untuk bisa digunakan dengan mudah oleh para pemula yang memang baru belajar membuat rangkaian elaktronika. Dengan demikian, tanpa mengetahui bahasa pemrograman, Arduino bisa digunakan untuk menghasilkan karya yang sederhana dan canggih.

1. Spesifikasi Arduino UNO

- a. Mikrokontroler ATmega 328.
- b. Beroperasi pada tegangan 5 volt.
- c. Tegangan input (rekomendasi) 7 - 12 V.

- d. Batas tegangan input 6 - 20V
- e. Pin digital input/output 14 (6 mendukung output PWM).
- f. Pin analog 6
- g. Arus pin per input/output 40 mA.
- h. Arus untuk pin 3.3V adalah 50mA.
- i. Flash memory 32 KB (ATmega 328), 2 KB digunakan oleh bootloader.
- j. SRAM 2 KB (ATmega 328).
- k. EEPROM 1 KB (ATmega 328).
- l. Kecepatan clock 16 Mhz



Sumber: <https://store.arduino.cc/usa/arduino-uno-rev3>

Gambar II.12 Arduino UNO

2. Bagian-bagian Arduino UNO

a. Pin Digital I/O (0 – 13)

14 pin digital ini berfungsi sebagai *input* atau *output* dan dapat diatur oleh program. Khusus untuk pin 3, 5, 6, 9, 10, dan 11 dapat juga berfungsi sebagai pin *analog output* yang tegangan *outputnya* dapat diatur.

b. USB

USB digunakan untuk mengupload program dari komputer ke dalam Board Arduino, dan berfungsi sebagai komunikasi serial antara komputer dengan Board Arduino.

c. Kristal (*quartz crystal oscillator*)

Kristal merupakan jantung dari Board Arduino itu sendiri, dikarenakan kristal menghasilkan detak-detak yang dikirim kepada mikrokontroller agar melakukan sebuah operasi untuk setiap detaknya, yang biasanya berdetak 16 juta kali per detik (16MHz).

d. Tombol Reset

Tombol reset digunakan untuk me-reset Board Arduino sehingga program akan memulai dari awal kembali. Tombol ini tidak untuk menghapus program atau mengosongkan mikrokontroller.

e. *In-Circuit Serial Programming (ICSP)*

Port ICSP digunakan untuk memprogram mikrokontroller tanpa melalui bootloader. ICSP menggunakan jalur *Serial Peripheral Interface (SPI)* untuk transfer data. Namun pada umumnya para pengguna tidak melakukan hal ini.

f. IC – mikrokontroller atmega

IC merupakan komponen utama dari Board Arduino yang didalamnya terdapat CPU, ROM, dan RAM.

g. *Eksternal Power Supply*

Sumber tenaga eksternal adalah penyuplai tegangan DC sebesar 9–12 Volt.

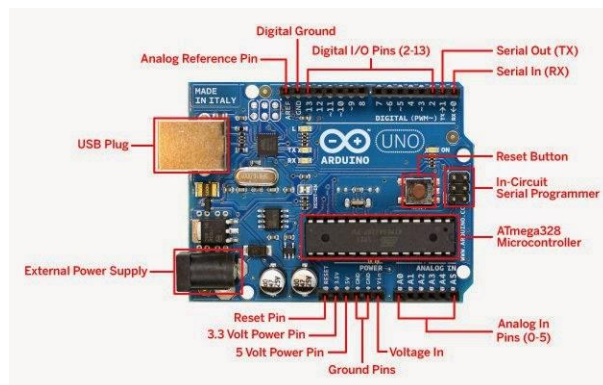
h. Pin Analog *Input* (0 – 5)

Pin analog digunakan untuk membaca tegangan yang dihasilkan oleh sensor analog. Nilai pin *output* analog dapat diprogram dari 0 – 255 dan dapat

membaca nilai analog dari 0 – 1023, yang keduanya mewakili nilai tegangan 0-5 Volt.

i. Pin Power yang terdapat pada Arduino Uno adalah sebagai berikut:

- 1) Vin = pin yang digunakan jika ingin memberikan power langsung ke board Arduino dengan tegangan 7V – 12V
- 2) 5V = pin *output* 5V yang telah diatur oleh regulator board Arduino. Board dapat diaktifkan dengan daya dari Power Supply (7 - 12V), USB (5V), atau pin Vin board (7-12V). Jika tegangan diinput melalui pin 5V atau 3.3V secara langsung tanpa melewati regulator, maka dapat merusak board Arduino.
- 3) 3.3V = dihasilkan oleh regulator on-board. Menyediakan arus maksimum 50 mA.
- 4) GND = Pin Ground
- 5) IOREF = memberikan tegangan referensi ketika mikrokontroler beroperasi. Biasa digunakan pada board shield untuk dapat memilih sumber daya yang tepat agar dapat bekerja dengan 5V atau 3.3V.



Sumber: <https://www.arduinoindonesia.id/2017/02/arduino-uno.html>

Gambar II.13 Bagian - Bagian Arduino UNO

2.2. Perangkat Lunak

Perangkat lunak atau *software* yang digunakan dalam “rancang bangun alat pendeteksi kadar gas alcohol” ini adalah Arduino IDE (*Integrated Development Environment*) yang menggunakan bahasa C.

2.2.1. Bahasa Pemrograman

(Ahmad Risal 2017) mengatakan bahwa “bahasa C adalah bahasa pemrograman yang dapat dikatakan berada di bahasa beraras tinggi. Bahasa beraras rendah artinya bahasa yang berorientasi pada mesin, sedangkan beruas tinggi berorientasi pada manusia. Bahasa beraras rendah, misalnya bahasa assembler. Ditulis dengan sandi hanya dimengerti oleh mesin sehingga hanya digunakan bagi yang memprogram mikrokontroler. Bahasa beruas rendah merupakan bahasa yang membutuhkan kecermatan tinggi bagi pemrogram karena perintahnya harus rinci, ditambah lagi masing-masing pabrik mempunyai sandi perintah sendiri. Bahasa tinggi relatif mudah digunakan karena ditulis dengan bahasa manusia sehingga mudah dimengerti dan tidak tergantung mesinnya. Bahasa beraras tinggi umumnya digunakan pada komputer”.

Pencipta bahasa C adalah Brian W. Kernighan dan Denis M. Ritchi sekitar 1972. Penulisan program dalam bahasa C dilakukan dengan membagi dalam blok – blok sehingga bahasa C disebut bahasa terstruktur. Bahasa C dapat digunakan di berbagai mesin dengan mudah, mulai dari PC sampai mainframe, serta menggunakan berbagai sistem operasi misalnya DOS, UNIX, VMS, dan lain-lain.

1. Syntax dalam bahasa C

Syntax atau penulisan *statement* (pertanyaan) dalam bahasa C meyakini *case sensitive* artinya mengenal perbedaan huruf kecil dan huruf besar kecuali dalam penulisan heksadesimal.

Penulisan dalam bahasa C dibagi menjadi beberapa bagian yaitu:

a. Konstanta

Konstanta adalah nilai yang tidak pernah berubah.

b. Variable

Variable adalah nilai yang dapat berubah-ubah saat programnya dijalankan.

c. Komentar

Komentar berfungsi untuk memberikan catatan dalam program kita.

d. Pernyataan

Pernyataan adalah sebuah instruksi lengkap yang sudah siap dieksekusi compiler untuk diubah menjadi bahasa mesin.

e. Blok pernyataan

Blok pernyataan adalah sekelompok pernyataan yang diawali “{” dan diakhiri “}”. Blok pernyataan digunakan untuk mengumpulkan instruksi-instruksi yang merupakan satu kesatuan pernyataan.

f. Ekspresi

Ekspresi adalah kombinasi antara konstanta, variable, dan operator untuk membentuk sebuah operasi yang wajib. Operator adalah suatu fungsi untuk melakukan operasi tertentu dan melibatkan satu atau lebih *operand*.

2. Struktur Program C

a. Deklarasi file *header* diperlukan bila program membutuhkan konstanta atau fungsi yang prototipenya ada di dalam file *header* tersebut. Format deklarasi file *header* akan dibahas berikutnya di *preprocessor* compiler.

b. Prototipe dari fungsi-fungsi yang digunakan program ditulis di awal program.

- c. Definisi konstanta digunakan untuk memberi nama suatu konstanta yang digunakan oleh program. Format definisi konstanta akan dibahas di *preprocessor compiler*.
- d. Deklarasi konstanta dan deklarasi variable global diletakkan di awal program dan di luar fungsi manapun. Variable ini bisa diakses dalam fungsi **main** ataupun fungsi lainnya. Fungsi yang mengakses variable global harus ada di bawah deklarasi variable global.
- e. Berikutnya bisa dituliskan fungsi-fungsi yang diperlukan program. Eksekusi program dimulai dari fungsi **main**. Oleh karena itu, untuk mengetahui alur suatu program bisa ditelusuri mulai dari fungsi main. Bila program terbagi menjadi beberapa file, tiap file dengan ekstensi “*.c” tetap dapat menggunakan struktur diatas tetapi tanpa fungsi **main**. Fungsi **main** sebagai fungsi utama hanya ada satu dalam tiap program c.

3. Preprocessor Compiler

Dalam membuat program bahasa C adalah fasilitas preprocessor compiler. Preprocessor compiler mempersiapkan tahap awal sebelum kompilasi. Preprocessor diawali dengan simbol ‘#’ dan di ikuti kata khusus untuk mengarahkan compiler saat menerjemahkan program.

4. Macam – macam tipe data dasar dalam bahasa C

Tabel II.1 Tipe Data

Tipe Data	Ukuran (Byte)	Range
Byte	1	0 – 1
Char	8	-128 – 127

Unsigned Char	8	0 – 255
Signed Char	8	-128 – 127
Int	16	-32768 – 32767
Short Int	16	-32768 – 32767
Unsigned Int	16	0 – 65535
Signed Int	16	-32768 – 32767
Long Int	32	-2147483648 – 2147483647
Unsigned Long Int	32	0 – 4294967295

Sumber: (Ahmad Risal 2017)

5. Operator Aritmatika

Operator aritmatika digunakan untuk operasi yang tergolong sebagai operator *binary*.

Tabel II.2 Operator Aritmatika

Operator	Contoh	Fungsi
+	10+2	Penjumlahan
-	15-10	Pengurangan dan Negasi
*	5*5	Perkalian
/	10/2	Pembagian
%	30%10	Modulus (siswa pembagian)

Sumber: (Abdul Kadir 2018)

6. Operator Relasi

Operator relasi atau bisa disebut operator pembandingan biasa digunakan untuk membandingkan dua nilai.

Tabel II.3 Operator Relasi

Operator Kondisi	Fungsi
<code>==</code>	Persamaan
<code>></code>	Lebih Besar
<code><</code>	Lebih Kecil
<code>>=</code>	Lebih Besar Sama Dengan
<code><=</code>	Lebih Kecil Sama Dengan
<code>!=</code>	Pertidaksamaan

Sumber: (Abdul Kadir 2019)

7. Operator Manipulasi Bit

Tabel II.4 Operator Manipulasi Bit

Operator	Fungsi
<code>&</code>	Operator bit “dan”
<code> </code>	Operator bit “atau”
<code>^</code>	Operator bit “atau eksklusif” atau
<code>~</code>	Operator bit “bukan”
<code><<</code>	Operator geser bit ke kiri
<code>>></code>	Operator geser bit ke kanan

Sumber: (Abdul Kadir 2018)

8. Operator Logika

Operator logika adalah operator yang biasa digunakan untuk menghubungkan dua ungkapan *boolean* menjadi satu ungkapan *boolean*.

Tabel II.5 Operator Logika

Operator	Keterangan
&&	Logika AND akan menghasilkan True apabila kondisi A dan B sesuai (True), jika tidak, maka akan menghasilkan False.
!	Logika NOT akan menghasilkan True jika kondisi tidak sesuai (False), karena NOT adalah kondisi sebaliknya.
	Logika OR akan menghasilkan True apabila salah satu dari A, atau B (True), tetapi jika kondisi keduanya False, maka akan menghasilkan nilai False.

Sumber: (Abdul Kadir 2018)

9. Dasar Pemrograman bahasa C

a. Aritmatika dan Logika

Aritmatika dan logika adalah operasi yang dilakukan dalam program.

Penulisan pernyataan aritmatika dan logika ke dalam subbab sebelumnya menjadi dasar penting untuk membuat program dalam bahasa C.

b. Percabangan dan seleksi kondisi

Alur program seringkali membutuhkan pencabangan baik itu bersyarat ataupun tanpa syarat. Pencabangan bersyarat meliputi proses seleksi atau pengujian kondisi dari suatu masukan untuk menentukan alur program. dalam C ada beberapa kata kunci yang digunakan untuk melakukan seleksi kondisi yaitu:

1) perintah: *if - else*

format penulisannya:

if (ekspresi logika)

```

{
    pernyataan 1
else
    pernyataan 2
}

```

penjelasannya:

a) Ekspresi logika yaitu seleksi kondisi untuk menemukan pernyataan mana yang akan di eksekusi berikutnya.

b) Pernyataan1 yaitu pernyataan yang dikerjakan apabila hasil ekspresi logikanya bernilai benar.

c) Pernyataan2 yaitu pernyataan yang dikerjakan apabila hasil ekspresi logikanya bernilai salah.

2) Perintah Switch – case:

Format penulisannya:

Switch (var)

```
{
```

Case konstanta1 :

Pernyataan 1

Break;

Case konstanta2 :

Pernyataan 2

Break;

Case konstanta3 :

Pernyataan 3

Case konstanta4 :

Case konstanta5 :

Pernyataan 4

Break;

Default :

pernyataanX

break;

}

penjelasannya:

- a) *Case* digunakan untuk konstanta pengujian di awal.
- b) *Var* yaitu berisi nilai yang akan dibandingkan dengan konstanta dan berupa ekspresi aritmatika.
- c) Konstanta 1-5 yaitu nilai pembandingan variable. Dalam sebuah pernyataan *switch*, dan konstanta pembandingan tidak boleh berulang.
- d) Pernyataan 1-5 yaitu pernyataan yang dikerjakan apabila nilai variable sesuai dengan konstantanya.
- e) *pernyataanX* yaitu pernyataan yang dikerjakan apabila tidak ada nilai yang sesuai antara variable dengan konstanta pembandingannya.
- f) *Default* bersifat optional.
- g) *Break* digunakan sebagai akhir dari seleksi kondisi dengan *switch*.

c. Perulangan

Tugas yang biasa digunakan dalam pemrograman adalah mengulang suatu proses. Dalam bahasa C tersedia kata kunci untuk melakukan perulangan sehingga tidak perlu menuliskan proses yang diulang berkali-kali.

Kata kunci yang digunakan sebagai berikut:

1) Perintah *for*

Format penulisannya:

For (start; syarat perulangan; ekspresi perulangan)

{

Pernyataan perulangan;

}

Penjelasannya:

- a) Start adalah kondisi pada saat awal perulangan digunakan untuk membuat dan memberikan nilai.
- b) Syarat perulangan yaitu digunakan untuk menentukan apakah perulangan masih terus dilakukan atau tidak. Syarat perulangan berupa ekspresi logika, jika bernilai benar perulangan masih berlaku sedangkan jika bernilai salah maka perulangan selesai.
- c) Ekspresi perulangan yaitu berisi pernyataan yang dapat mempengaruhi hasil dari pengujian syarat perulangan sehingga perulangan dapat diakhiri.
- d) Pernyataan perulangan yaitu pernyataan atau blok pernyataan yang di ulang.

2) Perintah *while*

Format penulisannya:

While (syarat perulangan)

{

Pernyataan perulangan

}

Penjelasannya:

Syarat perulangan yaitu ekspresi logika yang diuji nilainya untuk menentukan. Jika ekspresi logika bernilai benar maka pernyataan perulangan akan dilaksanakan tapi jika bernilai salah maka eksekusi akan dilanjutkan ke pernyataan setelah *while*.

3) Perintah *do – while*

Format penulisannya:

Do

{

Pernyataan perulangan

}

While (syarat perulangan);

Penjelasannya:

Jika perulangan menggunakan *do – while* maka perulangan menguji syarat perulangan diakhir, sehingga pernyataan perulangan selalu dieksekusi minimal satu kali.

d. Interupsi

Interupsi adalah kejadian *event* tertentu yang disediakan oleh mikrokontroler untuk menyela eksekusi program untuk mengerjakan fungsi ISR (*Interrupt Service Routine*).

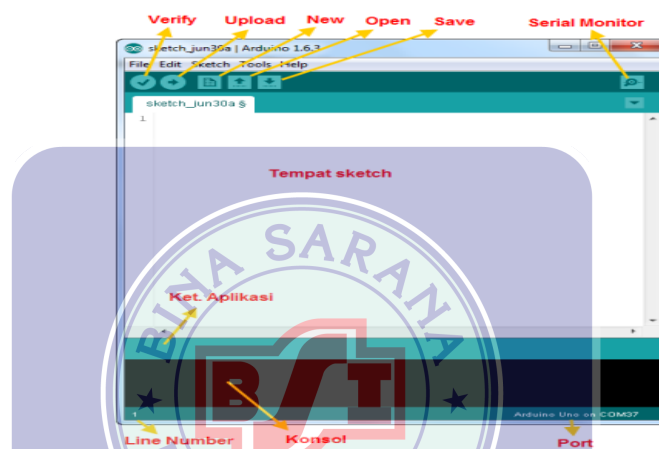
e. Algoritma Program

Algoritma program adalah alur atau urutan penyelesaian masalah dalam program. Algoritma program digambarkan dengan diagram alur (*flowchat*).

2.2.2. Aplikasi Program ArduinoIDE

(Andrianto and Darmawan 2016) mengatakan bahwa “Software IDE Arduino adalah pengendalian mikro *single – broad* yang bersifat *open – source*, diturunkan dalam *platform Wiring*, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang”.

1. Bagian – bagian arduino IDE



Sumber: <https://www.pusatkomponen.com/blog/mengenal-tampilan-bagian-arduino-ide>

Gambar II.14 Bagian - Bagian Arduino IDE.

- a. **Verify** : pada versi sebelumnya dikenal dengan istilah *Compile*. Sebelum aplikasi diupload ke *board* Arduino, biasanya untuk memverifikasi terlebih dahulu *sketch* yang dibuat. Jika ada kesalahan pada *sketch*, nanti akan muncul error. Proses Verify / *Compile* mengubah *sketch* ke *binary code* untuk diupload ke mikrokontroler.
- b. **upload** : tombol ini berfungsi untuk mengupload *sketch* ke *board* Arduino. Walaupun kita tidak mengklik tombol *verify*, maka *sketch* akan di-*compile*, kemudian langsung diupload ke *board*. Berbeda dengan tombol *verify* yang hanya berfungsi untuk memverifikasi *source code* saja.
- c. **New Sketch** : Membuka window dan membuat *sketch* baru

- d. **Open Sketch** : Membuka *sketch* yang sudah pernah dibuat. *Sketch* yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi file **.ino**
- e. **Save Sketch** : menyimpan *sketch*, tapi tidak disertai mengcompile.
- f. **Serial Monitor** : Membuka *interface* untuk komunikasi serial, nanti akan kita diskusikan lebih lanjut pada bagian selanjutnya
- g. **Keterangan Aplikasi** : pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal "*Compiling*" dan "*Done Uploading*" ketika kita mengcompile dan mengupload *sketch* ke *board* Arduino
- h. **Konsol** : Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang *sketch* akan muncul pada bagian ini. Misal, ketika aplikasi mengcompile atau ketika ada kesalahan pada *sketch* yang kita buat, maka informasi *error* dan baris akan diinformasikan di bagian ini.
- i. **Baris Sketch** : bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada *sketch*.
- j. **Informasi Port** : bagian ini menginformasikan *port* yang dipakai oleh *board* Arduino.

