BAB II

LANDASAN TEORI

2.1. Perangkat Keras

> Menurut (Susilawati dan Risanto Darmawan, 2017) menyimpulkan bahwa: *Hardware* atau perangkat keras adalah segala piranti atau komponen dari sebuah rangkaian elektronika yang sifatnya bisa dilihat secara kasat mata dan bisa diraba secara langsung. *Hardware* berfungsi untuk mendukung segala proses dalam sebuah rangkaian elektronika dan bekerja sesuai perintah yang sudah ditentukan penggunanya. Dengan adanya komponen *hardware*, maka bentuk *output* dari setiap proses bisa direalisasikan.

Pada tahap perancangan alat ini ada beberapa perangkat keras yang digunakan yaitu: mikrokontroler Arduino Uno, modul bluetooth HC-06, Motor Servo, *LCD* 16x2, LED, Konektor, Kabel Jumper, Modul *I2C*.

2.1.1. Teori *IC* Digital

Menurut (Maulana dan Rachmat Adi Purnama, 2017) mengatakan bahwa "*Integrated Circuit* (*IC*) adalah "Sirkuit terintegrasi atau yang biasa juga disebut sebagai *IC* merupakan komponen elektronika yang terbuat dari kumpulan puluhan, ratusan, hingga ribuan transistor, resistor, dioda dan komponen elektronika lainnya".

IC mempunyai fungsi dan tugas khusus yang telah ditentukan oleh pembuatnya, tugas khusus ini ditentukan dengan bagaimana cara merangkai komponen yang ada karena pada dasarnya adalah sebuah rangkaian tertentu dengan fungsi tertentu pula. *IC* Digital beroperasi pada tegangan 0 *volt* (*low*) dan 5 *volt* (*high*). *IC* digital tersusun dari beberapa rangkaian logika *AND*, *OR*, *NOT*, *NAND*, *NOR*, dan *XOR*.

UNIVERSITAS BSI INFORMATIKA

2.1.2. Sumber Tegangan

Menurut (Fadlilah dan Ahmad Arifudin, 2018) mengatakan bahwa "Sumber tegangan atau catu daya atau sering disebut dengan *power supply* adalah sebuah piranti yang berguna sebagai sumber listrik untuk piranti lain".

Daya untuk menjalankan peralatan elektronik dapat diperoleh dari berbagai sumber. Pada intinya semua *power supply* atau catu daya mempunyai fungsi yang sama yaitu sebagai penyearah dari *AC* ke *DC*.

Berikut Jenis-jenis *powersupply*:

1. Sumber Arus Bolak Balik (*AlternatingCurrent/AC*)

   Arus listrik bolak balik adalah arus listrik dengan besar dan arah yang berubah-ubah secara bolak balik. Arus *AC* mengalir bolak balik dari potensial tinggi (+) ke potensial rendah (-) dan dari potensial rendah (-) ke potensial tinggi (+), Gelombang listrik pada arus *AC* berbentuk *sinusoidal*, gelombang segi empat atau gelombang segitiga, contoh penggunaan arus listrik bolak-balik adalah pada jaringan PLN dan generator *AC*, jika menggunakan tegangan listrik PLN, besar arusnya berkisar antara 110 *volt* hingga 220 *volt* dengan frekuensi 50 *Hertz*.

2. Sumber Arus Searah (*DirectCurrent/ DC*)

   Arus listrik searah adalah arus listrik yang bernilai konstan dan mengalir dari potensial tinggi (+) ke potensial rendah (-). Besar arus listrik yang sering ditemukan berkisar antara 1,5 *volt* hingga 24 *volt*. Arus listrik searah biasa digunakan pada baterai, dinamo arus searah atau aki, sumber tegangan ini tidak mengalami perubahan terhadap waktu.

BINA SARANA INFORMATIKA UNIVERSITAS

Sumber: http//:panduanteknisi.com

Gambar II.1.

Baterai

2.1.3. Komponen Elektronika

Secara umum komponen elektronika dibagi dalam dua jenis, yaitu:

1. Komponen Pasif

   Menurut (Listiyarini, 2018) mengatakan bahwa “Komponen Pasif merupakan komponen yang dapat bekerja tanpa sumber tegangan”.

   Berikut adalah beberapa contoh Komponen Pasif:

   a. Resistor

      Menurut (Listiyarini, 2018) memberikan batasan bahwa “Resistor disebut juga dengan tahanan atau hambatan, merupakan komponen pasif elektronika yang berfungsi untuk membatasi arus listrik yang mengalir pada suatu rangkaian listrik”.

Sumber : Listiyarini (2018:49)

Gambar II.2

Resistor

BINA SARANA INFORMATIKA UNIVERSITAS

1) Resistor Tetap (*Fixed Resistor*)

Resistor tetap merupakan suatu resistor yang nilai resistansinya tidak dapat diubah. Resistor tetap memiliki nilai resistansi yang tertulis pada badan resistor dengan menggunakan kode warna dan kode angka. Resistor jenis ini sering digunakan sebagai penghambat arus listrik secara permanen dalam rangkaian elektronika.

Sumber : Listiyarini (2018:49)

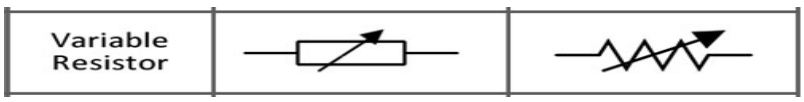
Gambar II.3

Simbol *Fixed Resistor*

2) Resistor Tidak Tetap (*Variable*)

Resistor *Variable* adalah resistor yang nilai resistansinya dapat diubah secara langsung baik dengan tuas yang telah tersedia atau menggunakan obeng.

Ada 2 jenis Resistor *Variabel* yang ada dipasaran yaitu Trimpot dan Potensiometer.



Sumber : https://skemaku.com/simbol-resistor-yang-terdapat-pada-rangkaian-elektronika/

Gambar II.4

Simbol *Variable Resistor*

a) Trimpot (*Trimer Potensio*)

BINA SARANA INFORMATIKA
UNIVERSITAS

Resistor jenis ini merupakan resistor yang nilai resistansinya dapat diubah dengan memutar porosnya menggunakan obeng. Nilai resistansi dari trimpot tertulis pada badan trimpot menggunakan kode angka, nilai yang tertulis pada badan trimpot merupakan nilai maksimum dari resistansi trimpot tersebut.

Sumber : https://skemaku.com/jenis-jenis-resistor/

Gambar II.5

Trimpot

b) Potensiometer

Resistor jenis ini merupakan resistor yang jenis resistansinya dapat diubah-ubah dengan cara memutar porosnya melalui tuas yang telah tersedia. Nilai resistansi potensiometer tertulis pada bagian potensio menggunakan kode angka. Nilai resistansinya yang dapat diubah secara logaritma dan nilai resistansi yang dapat diubah secara *linear.*

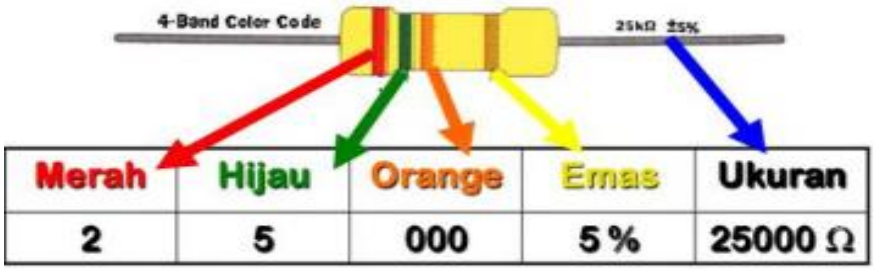
Sumber : https://skemaku.com/jenis-jenis-resistor/

Gambar II.6

Potensiometer

Nilai sebuah resistor dapat dilihat dari kode baik itu angka maupun

warna yang ada pada badan resistor. Kode warna pada resistor menyatakan harga resistansi dan toleransinya. Semakin kecil harga toleransi suatu resistor semakin baik, karena harga sebenarnya adalah harga yang tertera.



Sumber : Listiyarini (2018:52)

Gambar II.7

Membaca Nilai Resistor

BINA SARANA INFORMATIKA
UNIVERSITAS

Tabel II.1.
Tabel Kode Warna Pada Resistor

Warna	Nilai Angka	Nilai Pengali	Toleransi
Hitam	0	X 1	
Coklat	1	X 10	± 1 %
Merah	2	X 100	± 2 %
Jingga (Oranye)	3	X 1.000	
Kuning	4	X 10.000	
Hijau	5	X 100.000	
Biru	6	X 1.000.000	
Ungu (violet)	7	X 10.000.000	
Abu-Abu	8	X 100.000.000	
Putih	9	X 1.000.000.000	
Emas		X 0,1	± 5%
Perak		X 0,01	± 10 %
Tanpa Warna			± 20%

Sumber : Listiyarini (2018:52)

b. Kapasitor

Menurut (Listiyarini, 2018) memberikan batasan bahwa “Kapasitor

adalah komponen elektronika yang dapat menyimpan muatan listrik".

Struktur sebuah kapasitor terbuat dari 2 buah plat metal yang dipisahkan oleh suatu bahan dielektrik. Bahan dielektrik yang umum dikenal misalnya udara vakum, keramik, gelas dan lain-lain. Jika Kedua ujung plat metal diberi tegangan listrik, maka muatan–muatan positif akan mengumpul pada salah satu kaki (*elektroda*) metalnya pada saat yang sama muatan–muatan negatif terkumpul pada ujung metal yang satu lagi. Muatan positif tidak dapat mengalir menuju ujung kutub negatif dan begitu juga sebaliknya karena terpisah oleh bahan dielektrik yang non-konduktif. Muatan elektrik ini tersimpan selama tidak ada kondiksi pada ujung-ujung kakinya.

BINA SARANA INFORMATIKA
UNIVERSITAS

Sumber : Listiyarini (2018:58)

Gambar II.8

Jenis-jenis Kapasitor

Ada dua metode yang digunakan sebagai penanda nilai kapasitor yaitu:

1) Metode pertama adalah dengan menggunakan pita warna seperti halnya yang diterapkan pada resistor cincin warna.
2) Metode kedua adalah dengan penanda alfabet-numerik (huruf dan angka)

Tabel II.2.
Tabel Kode Warna Pada Kapasitor

Warna	Gelang1 (angka)	Gelang2 (angka)	Gelang 3 (Pengali)	Gelang 4 (toleransi)	Gelang 5 (tegangan Kerja)	
Hitam	-	0	1	-	-	-
Coklat	1	1	10	1	-	-
Merah	2	2	100	2	250V	160V
Jingga	3	3	1.000	3	-	-
Kuning	4	4	10.000	4	400V	200V
Hijau	5	5	100.000	5	-	-
Biru	6	6	1.000.000	6	630V	220V
Ungu	7	7	10.000.000	7	-	-
Abu-Abu	8	8	100.000.000	8	-	-
Putih	9	9	1.000.000.000	9	-	-

Sumber : Listiyarini (2018:60)

c. Induktor

Menurut (Listiyarini, 2018) memberikan batasan bahwa “Induktor adalah komponen yang tersusun dari lilitan kawat, induktor termasuk juga komponen yang dapat menyimpan muatan listrik”.

Bersama kapasitor induktor dapat berfungsi sebagai rangkaian resonator yang dapat beresonansi pada frekuensi tertentu.

Sumber : https://skemaku.com/fungsi-induktor-dan-jenis-jenis-induktor-pada-komponen-elektronika/

Gambar II.9

Jenis-jenis Induktor

Membaca nilai induktansi dari kode warna yang ditetapkan oleh *RMA* ini menentukan besarnya nilai induktansi induktor dalam *micro henry* (*μH*)**.**

Tabel II.3.
Tabel Kode Warna Pada Induktor

Warna	Cincin Pertama	Cincin Kedua	Cincin ketiga (pengali)	Cincin Keempat (Toleransi)
Hitam	0	0	1	20%
Coklat	1	1	10	1%
Merah	2	2	100	2%
Orange	3	3	1000	3%
Kuning	4	4	10000	4%
Hijau	5	5	----------	----------
Biru	6	6	----------	----------
Ungu	7	7	----------	----------
Abu-abu	8	8	----------	----------
Putih	9	9	----------	----------
Emas	----------	---------	0,1	5%
Perak	----------	---------	0,01	10%
Tanpa warna	----------	---------	----------	20%

Sumber : Listiyarini (2018:64)

2. Komponen Aktif

Menurut (Listiyarini, 2018) mengatakan bahwa "Komponen Aktif merupakan komponen yang tidak dapat bekerja tanpa adanya sumber tegangan".

Berikut adalah beberapa contoh Komponen Aktif :

a. Dioda

Menurut (Maulana dan Rachmat Adi Purnama, 2017) memberikan batasan bahwa "Dioda adalah komponen elektronika yang membuat arus listrik mengalir hanya dalam satu arah, sehingga biasa disebut juga sebagai penyearah".

Dioda terbuat dari semikonduktor jenis silikon dan germanium. Dioda disusun menggunakan semikonduktor jenis P sebagai kutub positif (+) dan semikonduktor jenis N sebagai kutub negatif (-).

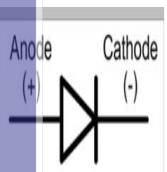
Dioda dibagi dalam beberapa jenis yaitu dioda penyearah dan dioda zener:

1) Dioda Penyearah

Jika arus listrik yang lewat searah dengan arah dioda yaitu dari

BINA SARANA INFORMATIKA UNIVERSITAS

potensial tinggi ke potensial rendah dan tegangan lebih besar dari tegangan minimum tegangan dioda maka arus akan dilewatkan. Namun jika dioda dipasang berkebalikan dengan arah arus listrik maka dioda berfungsi untuk menghambat arus listrik yang lewat. Kapasitas dioda memiliki batas, sehingga jika tegangan disambungkan N jauh lebih besar lebih dari puluhan atau ratusan *volt* dari pada tegangan disambungkan P, kemungkinan dioda akan *breakdown* karena tidak mampu menahan aliran arus listrik. Dioda penyearah antara lain digunakan untuk menyearahkan arus listrik bolak-balik pada transformator dan mencegah arus berbalik arah dalam rangkaian elektronika.



Sumber : https://skemaku.com/jenis-jenis-dioda-dan-fungsi-dioda-pada-rangkaian-elektronika/

Gambar II.10

Dioda Penyearah

2) Dioda Zener

Dioda Zener memiliki *breakdown voltage* terbalik yang terdefinisi dengan baik, di mana ia mulai menghantarkan arus, dan beroperasi terus menerus dalam mode *bias* tanpa mengalami kerusakan. Selain itu, penurunan voltase dioda tetap konstan pada berbagai voltase, fitur yang membuat dioda zener cocok untuk digunakan dalam

BINA SARANA INFORMATIKA
UNIVERSITAS

regulasivoltase.

Sumber : https://skemaku.com/jenis-jenis-dioda-dan-fungsi-dioda-pada-rangkaian-elektronika/

Gambar II.11

Dioda Zener

b. *LED* (*Light Emiting Diode*)

Menurut (Maulana dan Rachmat Adi Purnama, 2017) memberikan batasan bahwa "*LED* (*Light Emiting Diode*) merupakan jenis dioda yang jika diberikan tegangan *forward bias* akan menimbulkan cahaya dengan warna-warna tertentu, seperti merah, hijau, dan kuning".

Sumber : https://teknikelektronika.com/pengertian-led-light-emittingdiode-cara-kerja/

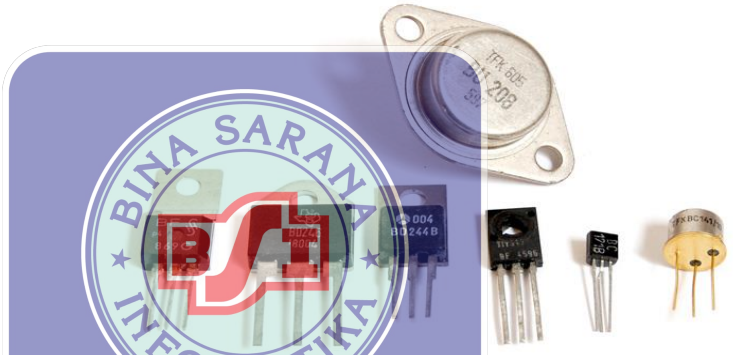
Gambar II.12

LED (*Light Emiting Diode*)

c. Transistor

Menurut (Maulana dan Rachmat Adi Purnama, 2017) memberikan batasan bahwa “Transistor memiliki 3 terminal komponen semikonduktor pada satu terminal adalah berfungsi sebagai pembuka (*open*) atau rangkaian”.

Transistor biasanya lebih banyak dibuat dari bahan silikon. Kaki yang berlainan membentuk transistor bipolar adalah emitor, basis dan kolektor. Mereka dapat dikombinasikan menjadi jenis NPN (*Negative Positive Negative*) atau PNP (*Positive Negative Positive*).



Sumber : https://skemaku.com/pengertian-transistor-fungsi-transistor-dan-cara-kerja-transistor-pada-rangkaian-elektronika/

Gambar II.13

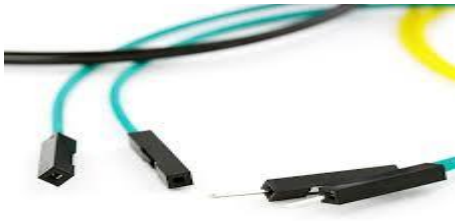
Jenis-jenis Transistor

3. Konektor

Menurut (Maulana dan Rachmat Adi Purnama, 2017) memberikan batasan bahwa “Konektor yang digunakan dalam rangkaian elektronika, untuk memudahkan melepas pasang rangkaian konektor ini memiliki lubang pin beragam, dan disesuaikan dengan kebutuhan”.

Header atau biasa dikenal dengan nama pin *head* berguna sebagai soket tempat menghubungkan kabel-kabel konektor, ada dua jenis pin14 *header* yaitu pin

header male dan pin *header female*, walaupun berbeda namun memiliki fungsi yang sama yaitu sebagai konektor.



Sumber : https://electronics.stackexchange.com/questions/37783/how-can-i-create-a-female-jumper-wire-connector

Gambar II.14

Konektor

4. Kabel Jumper

Menurut (Maulana dan Rachmat Adi Purnama, 2017) memberikan batasan bahwa "Kabel Jumper digunakan untuk menyalurkan energi listrik. Sebuah kabel listrik terdiri dari isolator dan konduktor isolator adalah bahan pembungkus kabel yang biasa terbuat dari plastik atau karet, sedangkan konduktor terbuat dari tembaga".

Sumber : https://klinikkonstruksi.jogjaprov.go.id/artikel_detail.php?id=99

Gambar II.15

Kabel Jumper

5. Modul *I2C*

Menurut (Sari, Cucu Suhery, dan Yudha Arman, 2015) memberikan batasan bahwa "*I2C* merupakan singkatan dari *Inter Integrated Circuit*, yaitu sebuah

protokol untuk komunikasi *serial* antar *IC* dan sering disebut juga *Two Wire Interface* (*TWI*)".

Sumber : www.miniinthebox.com

Gambar II.16

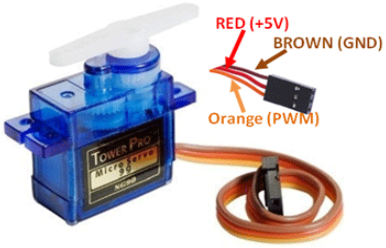
Modul *I2C*

2.1.4. *Output*

1. Motor Servo

Menurut (Iswanto, 2016) mengatakan bahwa "Motor Servo adalah sebuah motor dengan sistem *closed feedback* di mana posisi dari motor akan dikonfirmasikan kembali ke rangkaian kontrol yang ada di dalam motor servo". Motor servo ini terdiri dari sebuah motor, serangkaian gear, potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran servo . Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor. Semakin lebar pulsa *OFF* maka akan semakin besar gerakan sumbu ke arah jarum jam dan semakin kecil pulsa *OFF* maka akan semakin besar gerakan sumbu ke arah yang berlawanan dengan jarum jam.

BINA SARANA INFORMATIKA UNIVERSITAS



Sumber : https://components101.com/servo-motor-basics-pinout-datasheet

Gambar II.17

Motor Servo

Lebih dalam dapat digambarkan bahwa motor servo memiliki :

a. 3 Jalur kabel : *power*, *ground*, dan *control*.
b. Sinyal *control* mengendalikan posisi.
c. Operasional dari servo motor dikendalikan oleh sebuah pulsa selebar 20 ± ms, dimana lebar pulsa antara 0.5 ms dan 2 ms menyatakan akhir dari *range* sudut maksimum.
d. Konstruksi didalamnya meliputi internal *gear*, potensiometer, dan *fedback control*.

2. LCD (*Liquid Crystal Display)*

Menurut (Dinata dan Wahri Sunanda, 2015) memberikan batasan bahwa “LCD berfungsi untuk menampilkan suatu nilai hasil sensor, menampilkan teks, atau menampilkan menu pada aplikasi mikrokontroler”.

Adapun fitur yang disajikan dalam LCD ini adalah:

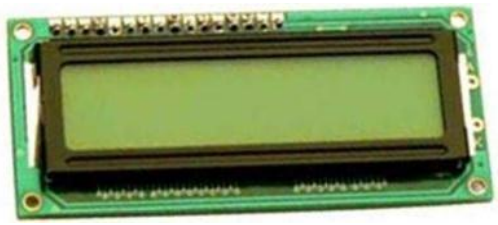
a. Terdiri dari 16 karakter dan 2 baris.

b. Mempunyai 192 karakter tersimpan.

c. Terdapat karakter generator terprogram.

d. Dapat dialamati dengan mode 4-bit dan 8-bit.

e. Dilengkapi dengan *back light.*



Sumber : Dinata dan Wahri Sunanda (2015:86)

Gambar II.18

LCD (Liquid Crystal Display)

BINA SARANA INFORMATIKA UNIVERSITAS

2.1.5. Mikrokontroler

Menurut (Sujarwata, 2018) mengatakan bahwa "Mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran (*I/O*) serta pengendali (kontrol) dengan suatu program yang dapat ditulis dan dihapus dengan cara khusus".

Mikrokontroler berfungsi sebagai pengontrol rangkaian elektronika dan pada umumnya dapat menyimpan program yang pada dasarnya menggunakan bahasa *assembler*. Saat ini mikrokontroler dapat diprogram dengan menggunakan bahasa tingkat tinggi seperti *BASIC, PASCAL* atau *C*. Agar semua mikrokontroler dapat berfungsi, maka mikrokontroler tersebut memerlukan komponen eksternal yang kemudian disebut dengan *system minimum*. Untuk membuat sistem minimal paling tidak dibutuhkan sistem *clock* dan *reset*, walaupun pada beberapa mikrokontroler sudah menyediakan sistem *clock* internal, sehingga tanpa rangakaian eksternal pun mikrokontroler sudah beroperasi. Mikrokontroler berbeda dari mikroprosesor serba

guna yang digunakan dalam sebuah *personal computer (PC),* karena sebuah mikrokontroler pada umumnya telah berisi komponen pendukung sistem minimal mikroprosesor, yakni memori dan antarmuka *I/O*. Mikrokontroler cenderung beroperasi pada manipulasi bit, sedangkan mikroprosesor cenderung beroperasi pada operasi *byte* (8bit).

Pada alat tempat sampah otomatis ini menggunakan mikrokontroler ATmega328P, dan implementasi yang digunakan dari ATmega328P adalah *platform* pengembangan Arduino yaitu model Arduino Uno.

Sedangkan Arduino menurut (Ahyadi, 2018) mengatakan bahwa "Arduino adalah suatu *open-source platform* elektronik yang berbasis kemudahan penggunaan (*easy to use*) baik *hardware* maupun *software*".

1. Arduino Uno

   Menurut (Kadir, 2017) mengatakan bahwa "Arduino Uno adalah papan yang mengandung sebuah mikrokontroler buatan *Atmel* (Atmega328p) yang menjadi pusat pengendali perangkat keras dan sejumlah pin untuk kepentingan operasi masukan (*input*) dan keluaran (*output*)".

   Hardware mikrokontroler Arduino diprogram dengan menggunakan bahasa pemrograman *wiring-based* yang berbasiskan *syntax* dan *library*. Pemrograman *wiring-based* ini tidak jauh berbeda dengan C/C++, tetapi dengan beberapa penyederhanaan dan modifikasi. Untuk memudahkan dalam pengembangan aplikasinya, mikrokontroler Arduino juga menggunakan *Integerated Development Environment (IDE)* berbasis *processing*. Mikrokontroler Arduino dapat dipasangkan dengan bermacam-macam sensor. Adapun sensor yang dapat dipasangkan pada Arduino seperti sensor gerak, ultrasonik, panas, suara, asap, gas, dan lain-lain.

Sumber : Akhiruddin (2017:36)

Gambar II.19

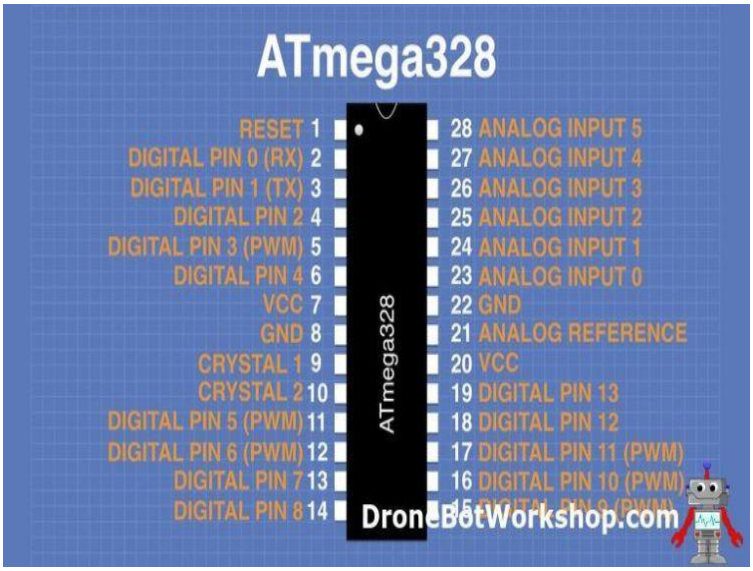
Arduino Uno

Konfigurasi yang terdapat pada Arduino Uno menurut (Sokop, Jendri Steven, Dringhuzen J. Mamahit, 2016) adalah:

a. Mikrokontroler : ATmega328
b. Tegangan pengoperasian : 5V
c. Tegangan *input* yang disarankan: 7-12V.
d. Batas tegangan *input* : 6-20V.
e. Jumlah pin *I/O* digital : 14.
f. Jumlah pin *input* analog : 6.
g. Arus *DC* tiap pin *I/O* : 40 mA.
h. Arus *DC* untuk pin 3.3V : 50 mA.
i. Memori : 32 KB (ATmega328).

   sekitar 0.5 KB digunakan oleh *bootloader*.
j. SRAM : 2 KB (ATmega328).
k. EEPROM : 1 KB (ATmega328).
l. Clock Speed : 16 MHz.

BINA SARANA INFORMATIKA UNIVERSITAS



Sumber: https://dronebotworkshop.com/

Gambar II.20

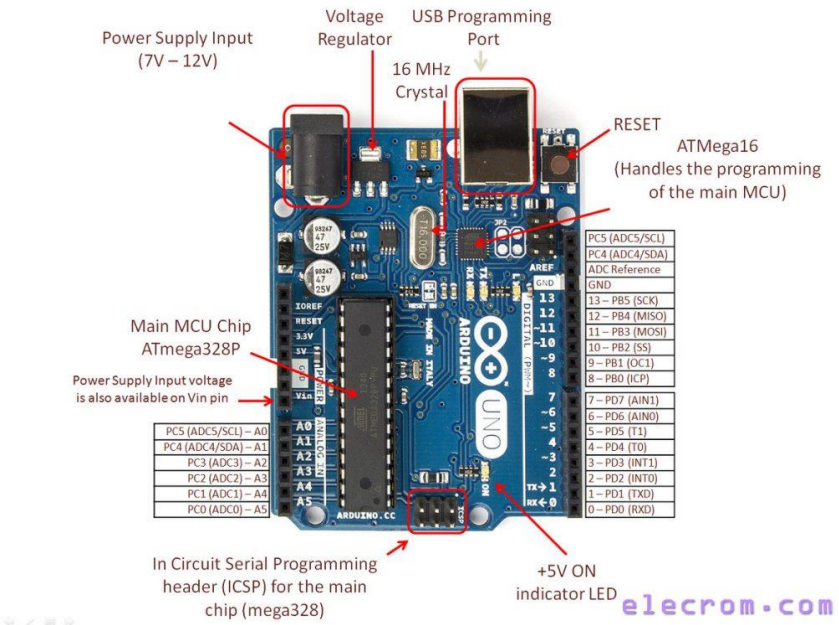
Kaki Pin Atmega328

2. Berikut ini bagian-bagian Arduino Uno yang terdapat pada gambar:

Sumber : Dinata dan Wahri Sunanda (2015:85)

Gambar II.21

Bagian-bagian Arduino Uno



Sumber : https://www.masodha.com

Gambar II.22

Skema Bagian-bagian Arduino Uno

a. *USB*

USB digunakan untuk meng*upload* program dari komputer ke dalam *Board* Arduino dan digunakan untuk memberikan catu daya ke Papan Arduino menggunakan kabel *USB* dari komputer.

b. Kristal (*quartz crystal oscillator*)

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

c. Tombol *Reset*

BINA SARANA INFORMATIKA UNIVERSITAS

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

d. 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*.

e. 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 *output*nya dapat diatur.

f. Eksternal *Power Supply*

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

g. *In Circuit Serial Programming* (*ICSP*)

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

h. *IC* – mikrokontroler atmega

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

BINA SARANA INFORMATIKA UNIVERSITAS

i. Pin *Power*

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 di *input* 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.

BINA SARANA INFORMATIKA UNIVERSITAS

2.2. Perangkat Lunak

Menurut (Utami, 2015) mengatakan bahwa “Perangkat lunak *(software)* adalah program yang berisi kumpulan instruksi untuk melakukan proses pengolahan data”.

Perangkat lunak atau *software* yang digunakan dalam pembuatan alat tempat sampah otomatis menggunakan sensor sonar dan sensor asap ini adalah *Software Proteus*, *Flowchart*, *Software* Arduino *IDE* (*Integrated Development Environtment*) yang menggunakan bahasa C.

1. *Software Proteus*

Menurut (Yahya, 2017) menyimpulkan bahwa:
Proteus merupakan gabungan dari Program *ISIS* dan *ARES*. Dengan penggabungan kedua program ini maka skematik rangkaian elektronika dapat dirancang serta disimulasikan dan dibuat jadi *layout PCB*. *ISIS* singkatan dari *Intelligent Schematic Input System* dan merupakan salah satu program simulasi yang terintegrasi dengan *Proteus* dan jadi program utamanya.

Aplikasi *ISIS* dapat digunakan untuk membuat sebuah simulasi rangkaian elektronika yang bersifat animasi dari hasil rangkaian yang akan dibuat. Penggunaan aplikasi ini dapat memudahkan kita untuk membuat sebuah rangkaian elektronika sampai dengan simulasi hasil jadi rangkaian, sehingga kita tidak perlu membuat rangkaian langsung jadi yang bertujuan untuk mengurangi kesalahan-kesalahan saat membuat rangkaian elektronika.

Sumber : Yahya (2017:59)

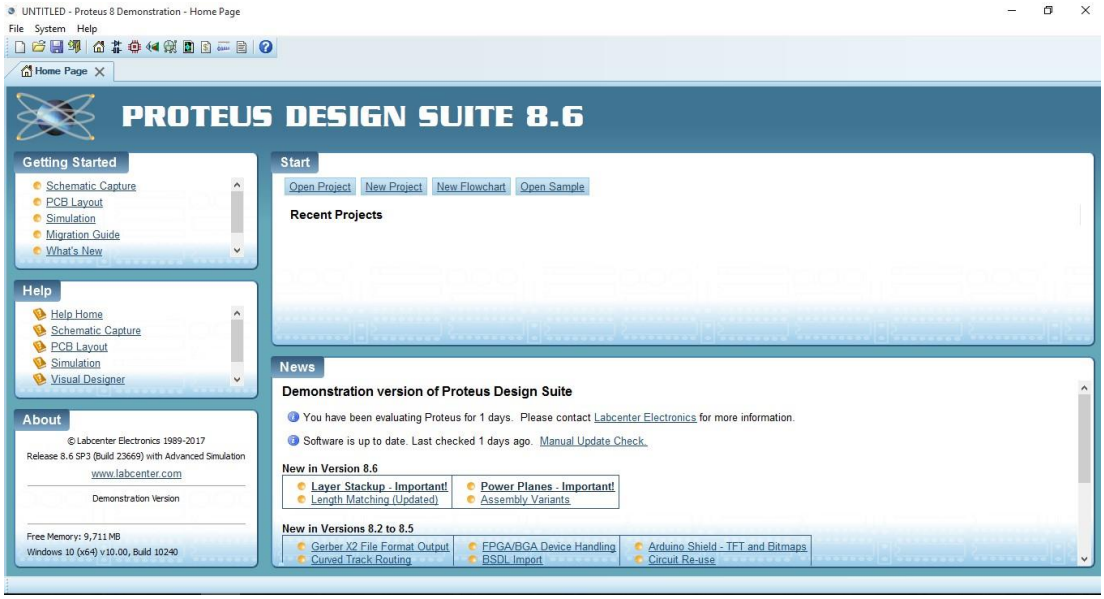
Gambar II.23

Software Proteus

Berikut ini adalah cara membuat skema rangkaian pada *software proteus:*

a. Buka aplikasi *Proteus* pada desktop/*PC*.

BINA SARANA INFORMATIKA
UNIVERSITAS



Sumber : http://idebelajar.com/cara-menggunakan-aplikasi-isis-proteus/

Gambar II.24

Tampilan Awal *Software Proteus*

b. Klik pada "*Schematic Capture*" seperti pada gambar berikut ini:

c. Lalu akan terbuka layout untuk membuat rangkaian yang ingin kita buat seperti gambar berikut:

d. Untuk mengetahui komponen-komponen yang akan digunakan, klik pada *Component Mode→Pick device*, lihat petunjuk pada gambar berikut:

BINA SARANA INFORMATIKA UNIVERSITAS

2.2.1. Bahasa Pemprograman

Menurut (Iswanto dan Nia Maharani Raharja, 2015) menyimpulkan bahwa:
Bahasa C pertama kali digunakan di komputer Digital *Equipment Corporation* PDP-11 yang menggunakan sistem operasi *UNIX* C adalah bahasa yang standar, artinya suatu program yang ditulis dengan bahasa C tertentu akan dapat dikonversi dengan bahasa C yang lain dengan sedikit modifikasi, Standar bahasa C yang asli adalah standar dari *UNIX*.
Berikut ini adalah fungsi-fungsi dari bahasa C :

1. Struktur

Struktur dalam penulisan program bahasa C mencakup tiga hal yaitu File *Header*, Fungsi Utama, Fungsi Lain. Pemrograman bahasa C biasa menggunakan fungsi utama Main() yang akan dijalankan pertama kali pada saat eksekusi program, meskipun didalam fungsi utama ini terdapat statement yang memanggil fungsi lain. Sedangkan didalam program Arduino (*Sketch*) memiliki dua fungsi yang diperlukan yang setiap fungsinya melampirkan blok-blok pernyataan.

BINA SARANA INFORMATIKA UNIVERSITAS

```
void setup()
{
statements;
}
void loop()
{
statements;
}
```

a. *Void setup*() { }

Fungsi *setup*() akan dipanggil ketika *sketch* dijalankan pertama kali dan hanya akan berjalan sekali, yaitu setiap *power-up* atau *restart*

board Arduino. Struktur fungsi ini untuk menginisialisasi mode pin, *variabel*, memulai penggunaan *library*, dan lain-lain.

b. *Void Loop*() { }

Setelah fungsi *setup*() selesai, fungsi *loop*() akan dijalankan secara terus menerus sampai catu daya (*power supply*) dilepas. Fungsi ini secara aktif mengontrol *board* Arduino baik secara *input* maupun *output*.

2. *Further Syntax*

a. ; (titik koma) digunakan pada akhir setiap baris kode.

b. {} (kurung kurawal) untuk mendefinisikan awal fungsi dan akhir fungsi.

c. // (baris komentar) digunakan untuk memberi satu baris catatan dari arti kode-kode yang dituliskan.

d. /**/ (blok komentar) digunakan untuk memberi dua atau lebih baris catatan dari arti kode-kode yang dituliskan.

e. #*include* yaitu satu jenis pengarah praprosessor yang dipakai untuk membaca file yang dinamakan file judul (*header file*).

3. *Variabel*

Variabel adalah cara penamaan dan menyimpan nilai numerik yang nantinya digunakan oleh program. Sebuah *variabel* perlu dinyatakan dan diberikan secara opsional ke nilai yang perlu disimpan. *int inputVariable = 0;* // menjelaskan sebuah *variabel* dan memberi nilai 0 *inputVariable* = analog*Read(2);* // menetapkan *variabel* ke nilai pin analog 2 *input Variable* adalah *variabel* itu sendiri.

4. Tipe Data

a. *Char* : Memiliki ukuran memori 1 *Byte* dengan jangkauan nilai -128 s/d 127 dan menyimpan 1 karakter menggunakan kode *ASCII*.

b. *Byte* : Menyimpan nilai numerik 8-bit tanpa titik desimal dan memiliki jangkauan nilai 0-255.

c. *Int* : Menyimpan bilangan tanpa titik desimal dan memiliki ukuran memori 2 *Byte*(16-bit) dengan jangkauan nilai -32,768 s/d 32,767.

d. *Long* : Tipe data *extended* untuk bilangan *long integer* dan memiliki ukuran memori 4 *Byte* dengan jangkauan nilai -2,147,435,648 s/d 2,147,435,647.

e. *Float* : Tipe data untuk angka yang memiliki titik desimal dan memiliki ukuran memori 4 *Byte* dengan jangkauan nilai -3.4028235E+38 s/d 3.4028235E+38.

f. *Double* : Memiliki ukuran memori 8 *Byte* dengan jangkauan nilai -1,7x10^(-308) s/d 1,7x10(+308).

g. Boolean : Digunakan untuk menyimpan nilai *TRUE* (benar) atau *FALSE* (salah) dan hanya menggunakan 1 bit dari *RAM*.

5. Array

Adalah tipe terstruktur yang terdiri dari sejumlah komponen-komponen yang mempunyai tipe yang sama. Banyaknya komponen dalam array ditunjuk oleh suatu indeks untuk membedakan antar variabel.

Format: TipeData NamaVariabel *[ukuran]*

a. *Type* data, untuk menyatakan *type* data yang digunakan.

b. Nama variabel, untuk menyatakan nama variabel yang digunakan.

c. Ukuran, untuk menyatakan jumlah maksimum elemen array.

Contoh: *Float* Jumlah[8]

6. Operator Bahasa

Merupakan simbol atau kata yang digunakan dalam program untuk melakukan suatu operasi atau manipulasi, seperti menjumlahkan dua buah nilai, memberikan nilai suatu variabel, membandingkan dua buah nilai, dan lain-lain. Berikut ini operator-operator yang biasa digunakan dalam bahasa C, yaitu:

a. Operator Aritmatika

Operator ini digunakan untuk memanipulasi angka yang bekerja seperti matematika sederhana. Operator Aritmatika digolongkan dalam Operator *binary* dan operator *unary*.

Tabel II.4.
Tabel Operator *Binary*

Operator	Keterangan	Contoh
+	Penjumlahan	4 + 5
-	Pengurangan	3 – 1
*	Perkalian	4 * 2
/	Pembagian	8 / 4
%	Sisa Pembagian (mod)	5 % 2

Sumber : www.duniailkom.com

Tabel II.5.
Tabel Operator *Unary*

Operator	Keterangan	Contoh
+	Tanda Plus	+5
-	Tanda Minus	-5

Sumber : www.duniailkom.com

b. Operator Relasi / Perbandingan

Operator ini digunakan untuk membandingkan dua buah nilai dengan hasil berupa nilai numerik 1 (*True*) atau 0 (*False*).

Tabel II.6.
Tabel Operator Relasi

Operator	Keterangan
==	Sama dengan (bukan pemberi nilai)
!=	Tidak Sama dengan
>	Lebih dari
<	Kurang dari
>=	Lebih dari atau Sama dengan
<=	Kurang dari atau Sama dengan

Sumber : www.duniailkom.com

c. Operator Logika / Boolean

Digunakan untuk mengekspresikan satu atau lebih data atau ekspresi logika (boolean) dengan menghasilkan data logika (boolean) baru.

Tabel II.7.
Operator Logika

Operator	Keterangan
&&	Operator Logika AND
||	Operator Logika OR
!	Operator Logika NOT

Sumber : www.duniailkom.com

Operator Logika AND digunakan untuk menghubungkan dua atau lebih ekspresi relasi, akan bernilai benar jika semua ekspresi relasi yang dihubungkan bernilai benar. Operator Logika OR digunakan untuk menghubungkan dua atau lebih ekspresi relasi, akan bernilai benar jika salah satu ekspresi relasi yang dihubungkan bernilai benar dan akan bernilai salah jika semua ekspresi relasi yang dihubungkan bernilai salah. Operator Logika NOT akan memberikan nilai kebalikan dari ekspresi yang dihubungkan, jika ekspresi bernilai benar maka akan menghasilkan nilai salah, begitu dengan sebaliknya.

7. Konstanta

Adalah nilai yang telah ditentukan yang digunakan untuk membuat program lebih mudah dibaca. Konstanta dikelompokkan menjadi sebagai berikut:

a. *TRUE/FALSE*

Adalah konstanta boolean yang mendefinisikan level logika.

FALSE didefinisikan sebagai 0 (nol), sedangkan *TRUE* didefinisikan sebagai 1 (satu).

If (b == TRUE);

{ pernyataan; }

b. *HIGH/LOW*

Konstanta ini menentukan level tinggi atau rendah dan digunakan saat membaca atau menulis ke pin digital. *HIGH* didefinisikan sebagai logika 1, *on*, atau 5 *volt*. Sedangkan *LOW* didefinisikan sebagai logika 0, *off*, atau 0 *volt*.

digital*write(13, HIGH);*

c. *INPUT/OUTPUT*

Konstanta ini digunakan dengan fungsi pinMode() untuk menentukan mode pin digital sebagau *input* atau *output*.

pinMode*(13, OUTPUT);*

8. Struktur Pengaturan

Sebuah program mempunyai suatu struktur dan pengaturan untuk menjalankan perintah programnya, yang biasa disebut juga dengan seleksi kondisi. Beberapa seleksi kondisi yang biasa digunakan dalam bahasa C sebagai berikut:

a. Perintah *If*

Digunakan untuk memilih satu dari dua atau lebih pernyataan.

Format: *if (kondisi)*

{ pernyataan }

b. Perintah *If...Else*

Digunakan untuk memilih satu dari dua atau lebih pernyataan dengan menjalankan pernyataan *if* terlebih dahulu, jika kondisinya *TRUE*

maka akan diteruskan perintah programnya, jika kondisinya *FALSE* maka pernyataan *else* yang akan dijalankan.

Format: *if (kondisi) { Pernyataan }*

Else { pernyataan }

c. Perintah *If* dalam *If*

Perintah *if* dalam *if* sering disebut *nasted-if* , dengan menjalankan kondisi1 if terlebih dahulu kemudian meneruskannya ke kondisi2 *if* yang jika pernyataan1 benar/*TRUE* maka diteruskan perintah programnya, jika kondisinya *FALSE* maka pernyataan2 yang akan dijalankan.

Format: *if (kondisi1)*

{ if (kondisi2) { pernyataan1 }

Else { pernyataan2 } }

d. Perintah *For*

Digunakan untuk melakukan perulangan pada suatu kondisi menggunakan perhitungan (*counter*) yang pasti.

Format: *for (inisialisasi; syarat perulangan; pengubah nilai pencacah)*

{ pernyataan; }

Inisialisasi merupakan bagian untuk memberikan nilai awal untuk variabel-variabel tertentu. Syarat Perulangan yaitu memegang kontrol terhadap perulangan, karena bagian ini yang akan menentukan suatu perulangan diteruskan atau dihentikan. Pengubah Nilai Pencacah; mengatur kenaikan atau penurunan nilai pencacah.

9. Digital *I/O*

BINA SARANA INFORMATIKA UNIVERSITAS

I/O digital pada Arduino mempunyai fungsi sendiri untuk mengaksesnya. Fungsi ini menyederhanakan perintah yang berhubungan dengan pin *I/O* Arduino.

a. pinMode (pin, mode)

Berfungsi untuk mengkonfigurasi pin tertentu dan digunakan sebagai *input* atau *output*. Pin adalah nomor pin yang akan digunakan dari 0-19 (pin analog 0-5 adalah 14-19).

b. digital*Write* (pin, *value*)

Berfungsi untuk memberi nilai *HIGH* atau *LOW* ke pin digital. Jika pin dikonfigurasi sebagai *OUTPUT* dengan pinMode(), maka tegangan akan diatur ke nilai yang sesuai, yaitu 5V atau 3.3V untuk *HIGH* dan 0V (*ground*) untuk *LOW*. Jika pin dikonfigurasi sebagai *INPUT* dengan pinMode(), maka digital*Write*() akan mengaktifkan (*HIGH*) atau menonaktifkan (*LOW*) dengan pullup internal pada pin *input*.

c. digital*Read*(pin)

Berfungsi untuk membaca nilai dari pin digital tertentu, baik itu *HIGH* atau *LOW*.

10. Analog *I/O*

Berikut ini adalah fungsi analog dari Arduino:

a. Analog*Read*(pin)

Berfungsi untuk membaca nilai analog melalui pin analog. Pada Arduino Uno sendiri memiliki 6 *channel* analog. Ketika pin analog ditetapkan sebagai *INPUT*, maka keluaran voltase dapat dibaca, yang keluarannya antara angka 0 (0V) dan 1024 (5V).

b. Analog*Write*(pin, *value*)

BINUS SABANA INFORMATIKA UNIVERSITAS

Pin pada Arduino ada yang mendukung *PWM* (*pulse width modulation*) yaitu pin 3, 5, 6, 9, 10, 11, yang dapat merubah pin hidup (*on*) atau mati (*off*) dengan cepat sehingga dapat membuatnya berfungsi seperti keluaran analog. *Value* atau nilai pada format kode tersebut adalah antara angka 0.

(*0% duty cycle – 0V*) dan 255 (*100% duty cycle – 5V*).

11. *Delay*(*time*)

Fungsi ini untuk menghentikan sementara program selama waktu yang ditentukan dalam milidetik. 1000 sama dengan 1 detik

Delay(1000); //tunggu 1 detik

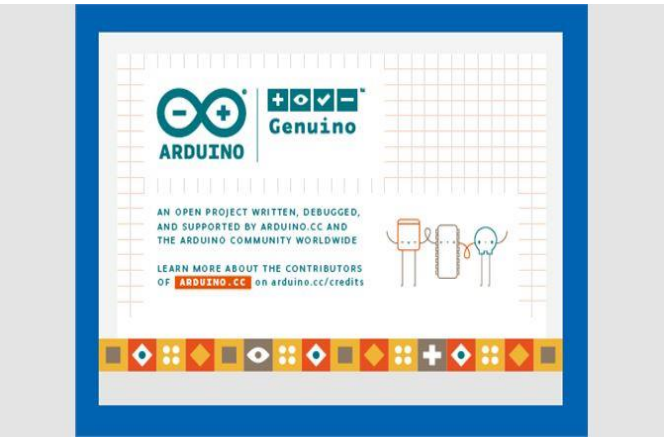
BINA SARANA INFORMATIKA UNIVERSITAS

2.2.2. *Software* Editor

1. Arduino *IDE*

Menurut (Ananda, 2018) mengatakan bahwa "Arduino *IDE* (*Integrated Development Program*) merupakan *software* yang dipakai untuk membuat *sketch* yang akan di *upload* ke *board* arduino sendiri".

Arduino *IDE* juga tergolong bentuk *software* pengembang program yang terintegrasi sehingga berbagai *tools* atau tampilan disediakan dan dinyatakan dalam bentuk antarmuka berbasis menu. Dengan menggunakan Arduino *IDE*, kesalahan penulisan *sketch* atau kebenaran penulisan *sketch* langsung dapat dibuktikan.



Sumber : www.microsoft.com

Gambar II.25

Arduino *IDE*

2. Bagian-bagian Arduino *IDE*

Berikut adalah fungsi *toolbar* yang terdapat pada Arduino *IDE* :

Verify : Berfungsi untuk melakukan *checking* kode yang dibuat pada kesalahan kompilasi.

Upload : Berfungsi untuk melakukan kompilasi program dan meng*upload*nya ke *Board* arduino yang sudah dikonfigurasi.

New : Berfungsi untuk membuat *sketch* baru.

Open : Berfungsi untuk membuka *sketch* yang sudah pernah disimpan.

Save : Berfungsi untuk menyimpan *sketch* yang telah dibuat.

Serial Monitor : Berfungsi untuk membuka *serial monitor*. *Serial* monitor merupakan jendela yang menampilkan data apa saja yang dikirimkan atau dipertukarkan antara arduino dengan *sketch* pada port

BINA SARANA INFORMATIKA UNIVERSITAS

*serial*nya. *Serial monitor* ini dapat digunakan untuk menampilkan nilai proses,nilai pembacaan, dan pesan *error*. Perintah tambahan dapat ditemukan dalam lima menu: ***File, Edit, Sketch, Tool, Help***.

3. Menulis *Sketch*

Jalankan program arduino dari file master arduino yang kita simpan ataupun dari ikon di komputer desktop. Kemudian akan muncul program arduino *IDE* dengan tampilan *sketch* sesuai dengan tanggal dan bulan pada saat program dijalankan. *Sketch* atau kode pemrograman Arduino ditulis dibagian editor *text* dan disimpan dalam file dengan ekstensi .ino.

4. Verifikasi *Sketch*

Jika *sketch* telah disimpan selanjutnya adalah verifikasi memastikan bahwa kode program telah benar. Klik tombol verifikasi untuk melakukannya, dan jendela pesan akan muncul "*Done Compilling*" jika verifikasi sukses.

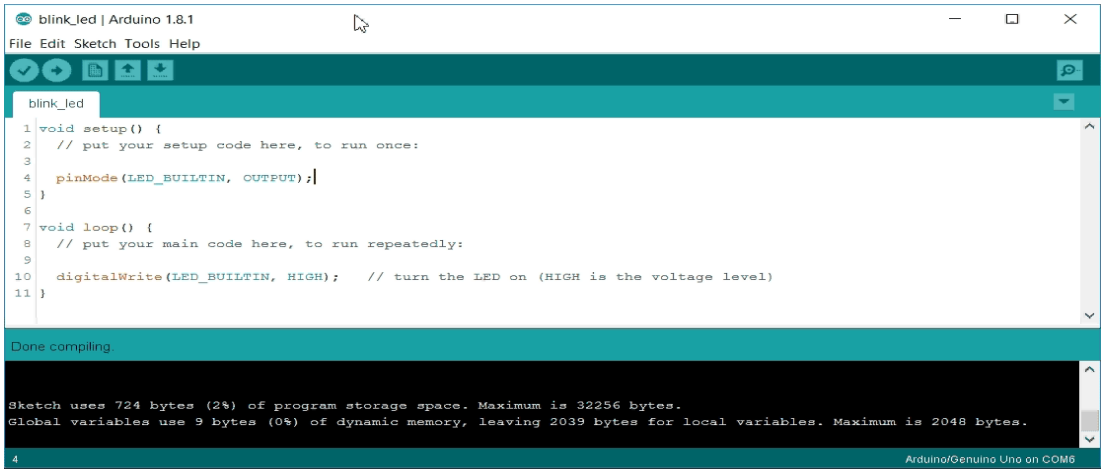
a. Memilih Jenis dan Port *Board* Arduino

Sebelum meng*upload sketch* ke *board* arduino, diharuskan memilih jenis dan port *board* Arduino yang digunakan. Menu ***tools* > *Board*** untuk memilih *board* yang sesuai, menu ***tools* > Port** untuk memilih port yang digunakan. Pada Windows kemungkinan **COM1** dan **COM2** untuk port *serial* atau COM4, COM5, COM7, dan lebih tinggi untuk port *USB*. Untuk mengetahuinya, dapat dilihat perangkat *USB serial* pada seksi port dari *Windows Device Manager*.

b. Mengunggah *Sketch*

Klik tombol *Upload*, *LED* RX dan TX di *board* akan berkedip dan jendela Arduino *IDE* akan memberi status bar yang menunjukkan

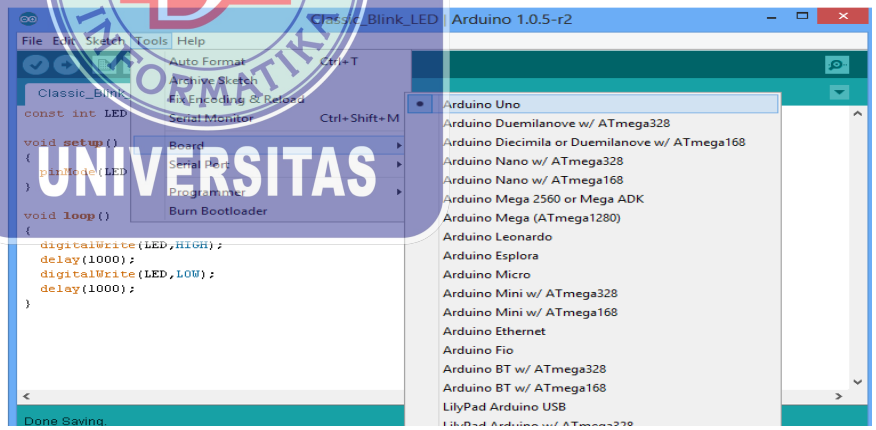
berapa banyak kemajuan yang telah diunggah. Jika unggahan berhasil, pesan "*Done uploading*", akan muncul di jendela pesan.



Sumber : http://www.circuitstoday.com/blink-led-with-arduino

Gambar II.26

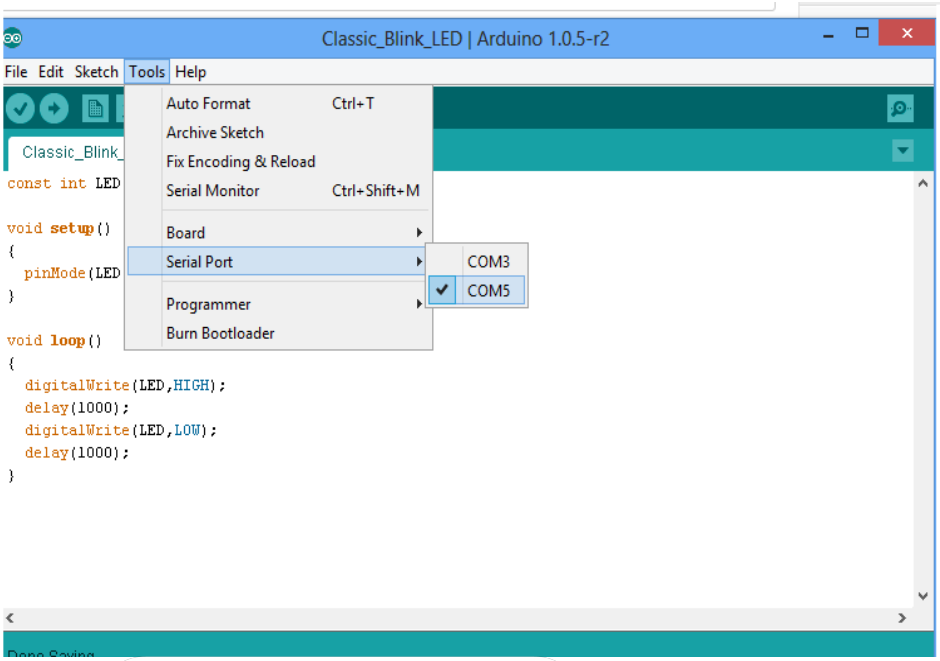
Verifikasi *Sketch*



Sumber : http://www.circuitstoday.com/blink-led-with-arduino

Gambar II.27

Memilih *Board*



Sumber : http://www.circuitstoday.com/blink-led-with-arduino

Gambar II.28

Memilih Port

Sumber : http://www.circuitstoday.com/blink-led-with-arduino

Gambar II.29

Sukses Mengunggah *Sketch*

BINA SARANA INFORMATIKA UNIVERSITAS

2.2.3 MIT App Inventor 2

Menurut (Kadir , 2018) App Inventor untuk Android adalah sebuah aplikasi web open-source asli yang disediakan oleh Google , dan sekarang dikelola oleh Massachusetts Institute of Technology (MIT). MIT App Inventor adalah sebuah inovasi pengantar bagi pemula untuk membuat program dan menciptakan aplikasi yang mengubah bahasa yang rumit dari coding berbasis text ke dalam bentuk visual, dengan men-drag-and-drop bangunan berupa blok-blok. Merupakan antar muka dengan grafik sederhana yang bahkan memberikan kemampuan bagi pemula untuk membuat suatu aplikasi yang berfungsi penuh dalam waktu satu jam atau kurang dari satu jam. berikut adalah gambar aplikasi MIT App Inventor.

Sumber: http//:wariornux.com

Gambar II.30.

MIT App Inventor 2