

BAB II

LANDASAN TEORI

2.1. Tinjauan Jurnal

Di Indonesia, peran teknologi dan informasi tidak kalah pentingnya dibandingkan dengan negara lain. Indonesia terus berusaha meningkatkan kemajuan di bidang teknologi dan informasi, karena itu masyarakat berusaha untuk menggunakan manfaat positif dari teknologi dan informasi agar dapat membantu dalam segala bidang pekerjaan. Seperti halnya sistem parkir, dalam membantu kemudahan dan ketelitian dalam pekerjaan sistem parkir juga dapat digunakan bantuan komputer.

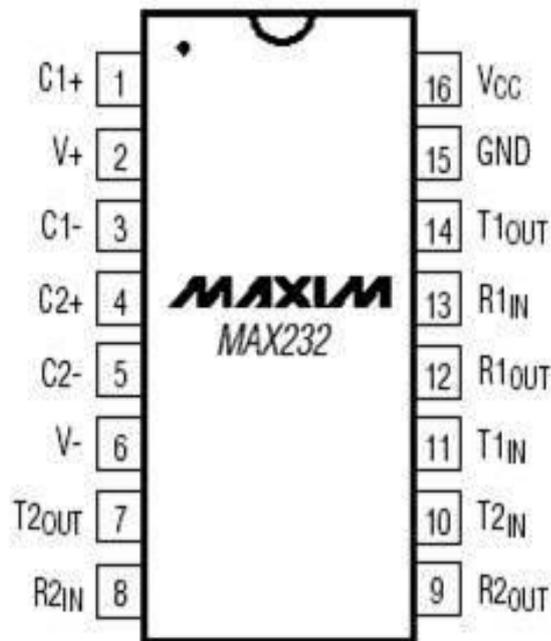
Menurut Rahmaniah, dkk (2008:401) mengatakan bahwa:
Sistem perparkiran yang konvensional khususnya di pusat-pusat perbelanjaan seringkali memerlukan penanganan ekstra di lapangan. Petugas harus selalu berkomunikasi untuk menentukan keadaan lokasi parkir yang sudah terisi dan masih kosong. Belum adanya sistem informasi parkir yang dapat memberikan informasi tentang keadaan lokasi parkir yang sudah terisi dan yang masih kosong itulah yang mendorong penulis untuk melakukan penelitian ini.

Kemudian menurut Pranata, dkk (2015:132) menjelaskan bahwa:
Prototipe sistem otomatisasi dapat dirancang menggunakan mikrokontroler, dimana mikrokontroler berfungsi sebagai pusat pengendalian seluruh sistem parkir, baik dari otomatisasi pintu parkir hingga pemberian informasi alokasi. Mikrokontroler adalah sebuah chip yang berfungsi sebagai pengontrol rangkaian elektronik dan dapat menyimpan instruksi program didalamnya.

2.2. Konsep Dasar Alat

A. Rangkaian Dasar Alat

1. IC MAX232



Sumber : <http://dc263.4shared.com/doc/NozBqLFj/preview.html>

Gambar II.1

Konfigurasi Pin IC MAX232

MAX232 merupakan salah satu jenis IC rangkaian antar muka *dual RS-232 transmitter / receiver* yang memenuhi semua spesifikasi standar EIA-232-E. Menurut Jatmika (2011:75) menjelaskan bahwa “IC merupakan komponen elektronika yang terbuat dari kumpulan puluhan, ratusan, hingga ribuan transistor, resistor, dioda, dan komponen elektronika lainnya”. IC MAX232 hanya membutuhkan *power supply* 5V (*single power supply*) sebagai catu daya. IC MAX232 di sini berfungsi untuk merubah level tegangan pada COM1 menjadi level tegangan TTL / CMOS. IC MAX232 terdiri atas tiga bagian yaitu *dual charge-pump voltage converter*, *driver RS232*, dan *receiver RS232*.

1. *Dual Charge-Pump Voltage Converter.*

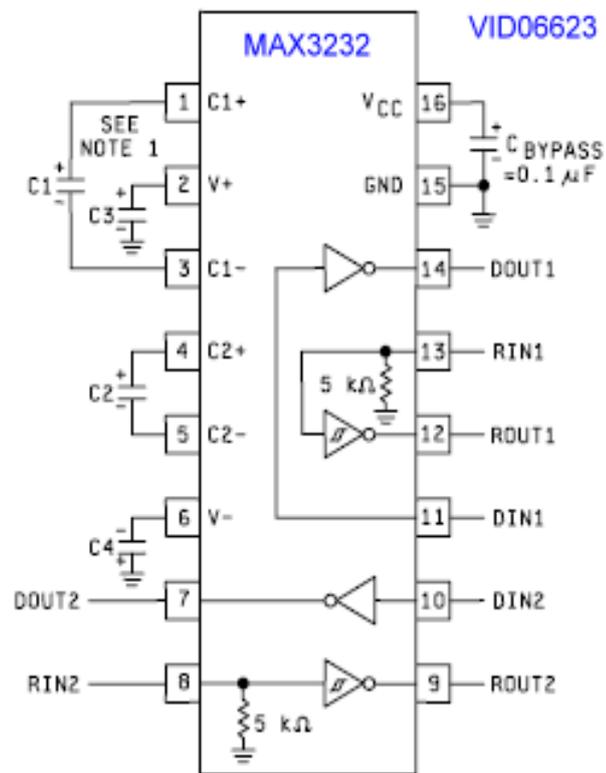
IC MAX232 memiliki dua *charge-pump internal* yang berfungsi untuk mengkonversi tegangan +5V menjadi $\pm 10V$ (tanpa beban) untuk operasi driver RS232. Konverter pertama menggunakan kapasitor C1 untuk menggandakan tegangan input +5V menjadi +10V saat C3 berada pada *output V+*. Konverter kedua menggunakan kapasitor C2 untuk merubah +10V menjadi -10V saat C4 berada pada *output V-*.

2. Driver RS232

Output ayunan tegangan (*voltage swing driver typical*) adalah $\pm 8V$. Nilai ini terjadi saat driver dibebani dengan beban nominal receiver RS232 sebesar $5k\Omega$ atau $V_{cc} = 5V$. *Input* pada *driver* yang tidak digunakan bisa dibiarkan tidak terhubung kemana – mana. Hal ini dapat terjadi karena dalam kaki *input driver* IC MAX232 terdapat resistor *pull-up* sebesar $400k\Omega$ yang terhubung ke V_{cc} . Resistor *pull-up* mengakibatkan *output driver* yang tidak terpakai menjadi *low* karena semua *output driver* diinversikan.

3. Receiver RS232

EIA mendefinisikan level tegangan lebih dari 3V sebagai logic 0, berdasarkan hal tersebut semua *receiver* diinversikan. *Input receiver* dapat menahan tegangan input sampai dengan $\pm 25V$ dan menyiapkan resistor terminasi input dengan nilai nominal 5k. Nilai *input receiver hysteresis typical* adalah 0,5V dengan nilai minimum 0,2V dan nilai *delay propogasi typicalnya* adalah 600ns.



Sumber : http://www.interfacebus.com/Design_Connector_RS232.html

Gambar II.2

Typical Operasi Rangkaian MAX232

2. IC Regulator 7805



Sumber : <http://www.popscreen.com/tagged/ic-7805/images>

Gambar II.3

IC Regulator 7805

IC Regulator ini merupakan Regulator yang dapat menghasilkan tegangan *output* stabil sesuai jenisnya dengan syarat tegangan *input* minimal yang diberikan harus lebih besar dari tegangan *output*nya. Sedangkan batas maksimal tegangan *input* juga harus disesuaikan karena jika tidak maka tegangan *output* yang dihasilkan tidak akan stabil. Menurut Tim Pustena ITB (2011:67) menerangkan bahwa” Regulator tegangan berfungsi untuk mengatur tegangan keluaran adaptor supaya sesuai dengan kebutuhan tegangan pada rangkaian elektronika yang akan digunakan”. Regulator 7805 akan menghasilkan tegangan *output* stabil 5V, sedangkan Regulator 7809 akan menghasilkan tegangan *output* stabil 9V. Berikut ini merupakan keunggulan yang dimiliki IC Regulator seri 78xx.

- a. Untuk regulasi tegangan DC, tidak memerlukan komponen elektronik tambahan.
- b. Aplikasi mudah dan hemat ruang.
- c. Memiliki proteksi terhadap *overload* (beban lebih), *overheat* (panas lebih), dan hubungsingkat.
- d. Dalam keadaan tertentu, kemampuan pembatasan arus peranti 78XX tidak hanya melindunginya sendiri, tetapi juga melindungi rangkaian yang ditopangnya.

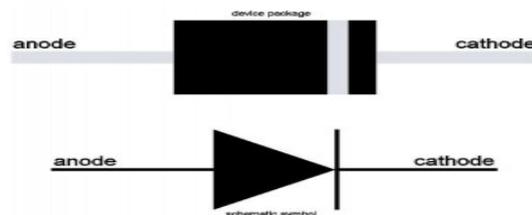
3. Dioda

Dioda adalah komponen elektronika semikonduktor yang memiliki 1 buah *junction*, sering disebut sebagai komponen 2 lapis (lapis N dan P). Menurut Jatmika (2011:71) menjelaskan bahwa “Fungsi paling umum dari dioda adalah untuk memperbolehkan arus listrik mengalir dalam satu arah (disebut kondisi panjar maju) dan untuk menahan arus dari arah sebaliknya (disebut kondisi panjar

mundur)”. Bahan pokok untuk pembuatan dioda adalah Germanium (Ge) dan Silikon/Silsilum (Si).

a. Dioda Silikon

Dioda silikon adalah dioda yang mempunyai fungsi sebagai pengendali. Dioda silikon masih termasuk semikonduktor dengan karakteristik yang serupa dengan tabung *thiratron*. Dioda silikon ini terbuat dari bahan silikon, dan banyak digunakan pada peralatan catu daya sebagai penyearah arus, pengamanan tegangan kejut, dan sebagainya. Contoh dioda silikon yaitu, 1N4001, 1N4007, dan 1N5404.



Sumber : <http://elektronika-dasar.web.id/teori-elektronika/konsep-dasar-diode/>

Gambar II.4

Simbol dan Gambar Dioda

4. Transistor

Transistor adalah alat semikonduktor yang dipakai sebagai penguat, sebagai sirkuit pemutus dan penyambung (*switching*), stabilisasi tegangan, modulasi sinyal atau sebagai fungsi lainnya. Transistor dapat berfungsi semacam keran listrik, dimana berdasarkan arus inputnya (BJT) atau tegangan inputnya (FET), memungkinkan pengaliran listrik yang sangat akurat dari sirkuit sumber listriknya.



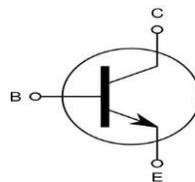
Sumber : <http://www.kootation.com/high-power-transistor-lifier-circuits-eleccircuit-com.html>

Gambar II.5
Transistor

Menurut Winarno dan Arifianto (2011:26) mengemukakan bahwa “Transistor merupakan komponen elektronika pertama yang mengantarkan dunia elektronika kuno menjadi elektronika modern”. Pada umumnya, transistor memiliki 3 terminal, yaitu Basis (B), Emitor (E) dan Kolektor (C). Tegangan yang di satu terminalnya misalnya emitor dapat dipakai untuk mengatur arus dan tegangan yang lebih besar daripada arus input basis, yaitu pada keluaran tegangan dan arus *output* kolektor.

Transistor secara umum dibagi menjadi 2 macam yaitu PNP dan NPN.

a. Transistor NPN

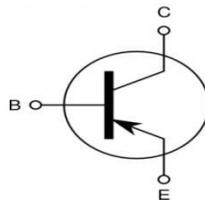


Sumber : www.easyvectors.com

Gambar II.6
Transistor NPN

Prinsip kerja dari transistor NPN adalah arus akan mengalir dari kolektor ke emitor jika basisnya dihubungkan ke ground (negatif). Arus yang mengalir dari basis harus lebih kecil daripada arus yang mengalir dari kolektor ke emitor, oleh sebab itu maka ada baiknya jika pada *pin* basis dipasang sebuah resistor.

b. Transistor PNP



Sumber : www.easyvectors.com

Gambar II.7 Transistor PNP

Prinsip kerja dari transistor PNP adalah arus akan mengalir dari emitter menuju ke kolektor jika pada *pin* basis dihubungkan ke sumber tegangan (diberi logika 1). Arus yang mengalir ke basis harus lebih kecil daripada arus yang mengalir dari emitor ke kolektor, oleh sebab itu maka ada baiknya jika pada *pin* basis dipasang sebuah resistor.

5. Resistor

Resistor adalah komponen pasif elektronika yang berfungsi untuk membatasi arus listrik yang mengalir. Menurut Jatmika (2011:51) menjelaskan bahwa “ Resistor, sesuai namanya yang berarti penghambat, berfungsi untuk menghambat arus listrik yang mengalir pada sebuah rangkaian”. Berdasarkan kelasnya resistor dibagi menjadi 2 yaitu : *Fixed Resistor* dan *Variable Resistor*. Pada umumnya *resistor* terbuat dari *carbon* film atau metal film, tetapi tidak menutup kemungkinan untuk dibuat dari material yang lain. Pada dasarnya semua

bahan memiliki sifat *resistif*, namun beberapa bahan tembaga, perak, emas dan bahan metal umumnya memiliki *resistansi* yang sangat kecil. Bahan–bahan tersebut menghantar arus listrik dengan baik, sehingga dinamakan konduktor. Kebalikan dari bahan yang konduktif, bahan material seperti karet, gelas, karbon memiliki *resistansi* yang lebih besar menahan aliran *elektron* dan disebut sebagai *insulator*.

1. *Fixed Resistor*

Fixed resistor yaitu *resistor* yang nilai hambatannya relatif tetap, biasanya terbuat dari karbon, kawat atau paduan logam. Nilainya hambatannya ditentukan oleh tebalnya dan panjangnya lintasan karbon. Adapun fungsi *fixed resistor* adalah sebagai berikut :

- a. Menahan sebagian arus listrik agar sesuai dengan kebutuhan suatu rangkaian elektronika.
- b. Menurunkan tegangan sesuai dengan yang dibutuhkan oleh rangkaian elektronika
- c. Membagi tegangan.
- d. Membangkitkan frekuensi tinggi dan frekuensi rendah dengan bantuan transistor dan kondensator (kapasitor).



Sumber : http://www.engineeringtoolbox.com/electrical-resistor-colour-code-value-tolerance-d_1655.htm

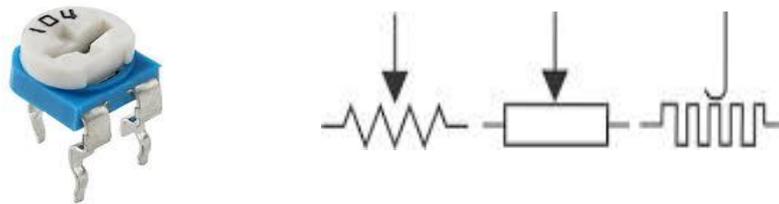
Gambar II.8

Simbol dan gambar *fixed resistor*

2. Variable Resistor

Variable resistor adalah *resistor* yang memiliki hambatan dapat diubah-ubah dengan memutar atau menggesernya. *Variable resistor* ini memiliki fungsi dan kegunaan antara lain:

- a. Pengatur *volume* (keras/lemah suara)
- b. Pengatur nada (*bass/treble*)
- c. Pengatur cerah redup gambar pada pesawat TV
- d. Pengatur kontras gambar
- e. Pengatur frekuensi



Sumber : http://www.biriecig.co.uk/index.php?main_page=products_new

Gambar II.9

Simbol dan Gambar *Variable Resistor* (trimpot)

6. Kapasitor

Menurut Jatmika (2011:58) mengemukakan bahwa “Kapasitor adalah komponen elektronika yang mempunyai kemampuan menyimpan elektron-elektron dalam bentuk muatan listrik selama waktu yang tidak tertentu”. Kapasitor berbeda dengan akumulator dalam menyimpan muatan listrik terutama tidak terjadi perubahan kimia pada bahan kapasitor, besarnya kapasitansi dari sebuah kapasitor dinyatakan dalam farad.

Pengertian lain Kapasitor adalah komponen elektronika yang dapat menyimpan dan melepaskan muatan listrik. Struktur sebuah kapasitor terbuat dari 2 buah plat metal yang dipisahkan oleh suatu bahan dielektrik. Bahan-bahan dielektrik yang umum dikenal misalnya udara vakum, keramik, gelas, elektrolit dan lain-lain. Jika kedua ujung plat metal diberi tegangan listrik, maka muatan-muatan positif akan mengumpul pada salah satu kaki (elektroda) metalnya dan pada saat yang sama muatan-muatan negatif terkumpul pada ujung metal yang satu lagi. Muatan positif tidak dapat mengalir menuju ujung kutub negatif dan sebaliknya muatan negatif tidak bisa menuju ke ujung kutub positif, karena terpisah oleh bahan dielektrik yang non-konduktif. Muatan elektrik ini “tersimpan” selama tidak ada konduksi pada ujung-ujung kakinya. Kemampuan untuk menyimpan muatan listrik pada kapasitor disebut dengan kapasitansi atau kapasitas.

Berikut adalah tabel contoh konstanta (k) dari beberapa bahan dielektrik yang disederhanakan

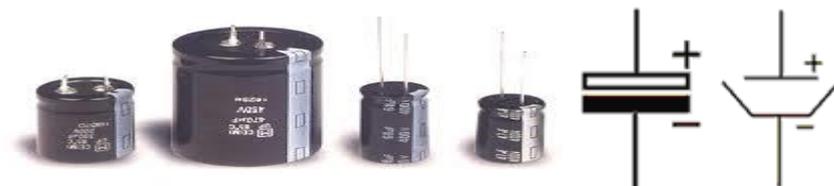
Tabel II.1
Konstanta Bahan Dielektrik

Udara vakum	$k = 1$
Aluminium oksida	$k = 8$
Keramik	$k = 100 - 1000$
Gelas	$k = 8$
Polyethylene	$k = 3$

Kapasitor tetap adalah kapasitor yang memiliki kapasitansi tetap dan tidak dapat diubah-ubah. Pada kategori kapasitor tetap, terdapat 2 jenis kapasitor yang dapat dibedakan berdasarkan polaritas elektrodanya.

a. Kapasitor Polar (*Electrolytic Capacitor*)

Kelompok kapasitor *electrolytic* terdiri dari kapasitor-kapasitor yang bahan dielektriknya adalah lapisan metal-oksida. Umumnya kapasitor yang termasuk kelompok ini adalah kapasitor polar dengan tanda + dan – di badannya. Mengapa kapasitor ini dapat memiliki polaritas, adalah karena proses pembuatannya menggunakan elektrolisa sehingga terbentuk kutup positif anoda dan kutup negatif katoda. Telah lama diketahui beberapa metal seperti tantalum, aluminium, magnesium, titanium, niobium, zirconium dan seng (*zinc*) permukaannya dapat dioksidasi sehingga membentuk lapisan metal-oksida (*oxide film*). Lapisan oksidasi ini terbentuk melalui proses elektrolisa, seperti pada proses penyepuhan emas. Elektroda metal yang dicelup kedalam larutan elektrolit (*sodium borate*) lalu diberi tegangan positif (anoda) dan larutan elektrolit diberi tegangan negatif (katoda). Oksigen pada larutan elektrolit terlepas dan mengoksidasi permukaan plat metal. Contohnya, jika digunakan Aluminium, maka akan terbentuk lapisan Aluminium-oksida (Al_2O_3) pada permukaannya.



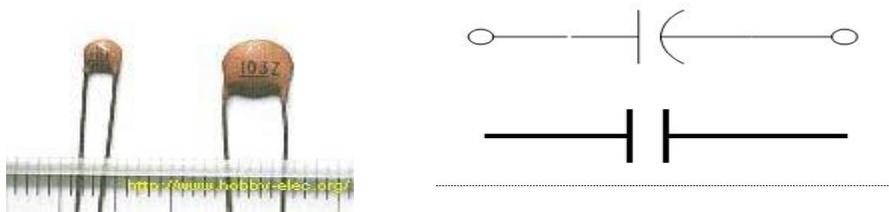
Sumber : <http://elektronika-dasar.web.id/teori-elektronika/jenis-jenis-kapasitor/>

Gambar II.10

Simbol dan Gambar *Electrolytic Capacitor* (ELCO)

b. Kapasitor Non Polar (Kapasitor Keramik)

Kapasitor non polar adalah kelompok kapasitor yang dibuat dengan bahan dielektrik dari keramik, film dan mika. Keramik dan mika adalah bahan yang populer serta murah untuk membuat kapasitor yang kapasitansinya kecil. Tersedia dari besaran pF sampai beberapa uF, yang biasanya untuk aplikasi rangkaian yang berkenaan dengan frekuensi tinggi. Termasuk kelompok bahan dielektrik film adalah bahan-bahan material seperti *polyester* (*polyethylene terephthalate* atau dikenal dengan sebutan mylar), *polystyrene*, *polypropylene*, *polycarbonate*, *metalized paper* dan lainnya.



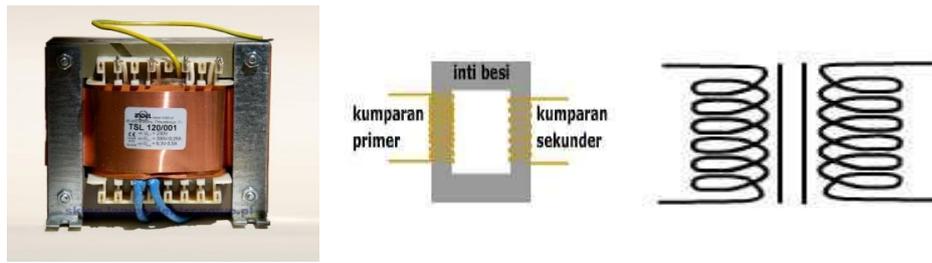
Sumber : <http://elektronikadasar.info/capasitor.htm>

Gambar II.11

Simbol dan Gambar Keramik Kapasitor

7. Transformator (Trafo)

Menurut Tim Pustena ITB (2011:61) menjelaskan bahwa “Trafo adalah komponen elektronika yang berfungsi menaikkan atau menurunkan tegangan AC”. Transformator terdiri dari 3 komponen pokok yaitu: kumparan pertama (primer) yang bertindak sebagai *input*, kumparan kedua (sekunder) yang bertindak sebagai *output*, dan inti besi yang berfungsi untuk memperkuat medan magnet yang dihasilkan.



Sumber : <http://sklep.lampyelektronowe.pl/?114,tsl-120-001-transormator-sieciowy-indel>

Gambar II.12

Simbol dan Gambar Transformator

Prinsip kerja dari sebuah transformator adalah ketika kumparan primer dihubungkan dengan sumber tegangan bolak-balik, perubahan arus listrik pada kumparan primer menimbulkan medan magnet yang berubah. Medan magnet yang berubah diperkuat oleh adanya inti besi dan dihantarkan inti besi ke kumparan sekunder, sehingga pada ujung-ujung kumparan sekunder akan timbul GGL induksi. Efek ini dinamakan induktansi timbal-balik (*mutual inductance*). Ketika arus listrik dari sumber tegangan yang mengalir pada kumparan primer berbalik arah (berubah polaritasnya) medan magnet yang dihasilkan akan berubah arah sehingga arus listrik yang dihasilkan pada kumparan sekunder akan berubah polaritasnya.

Hubungan antara tegangan primer, jumlah lilitan primer, tegangan sekunder, dan jumlah lilitan sekunder, dapat dinyatakan dalam persamaan:

$$\frac{V_p}{V_s} = \frac{N_p}{N_s}$$

V_p = tegangan primer (volt)

V_s = tegangan sekunder (volt)

N_p = jumlah lilitan primer

N_s = jumlah lilitan sekunder

Berdasarkan perbandingan antara jumlah lilitan primer dan jumlah lilitan sekunder transformator ada dua jenis yaitu:

- a. Transformator *step up* yaitu transformator yang mengubah tegangan bolak-balik rendah menjadi tinggi, transformator ini mempunyai jumlah lilitan kumparan sekunder lebih banyak daripada jumlah lilitan primer ($N_s > N_p$).
- b. Transformator *step down* yaitu transformator yang mengubah tegangan bolak-balik tinggi menjadi rendah, transformator ini mempunyai jumlah lilitan kumparan primer lebih banyak daripada jumlah lilitan sekunder ($N_p > N_s$).

8. *Push Button*

Menurut Nalwan (2012:37) menerangkan bahwa “*Switch Push Button* adalah saklar tekan yang berfungsi untuk menghubungkan atau memisahkan bagian – bagian dari suatu instalasi listrik satu sama lain”. Suatu sistem saklar tekan *push button* terdiri dari saklar tekan *start*, *stop reset* dan saklar tekan untuk *emergency*. *Push button* memiliki kontak NC (*normally close*) dan NO (*normally open*).



Sumber : <http://www.skpang.co.uk/catalog/mini-push-button-switch-p-540.html>

Gambar II.13
Simbol dan Gambar *Push Button*

Prinsip kerja *Push Button* adalah apabila dalam keadaan normal tidak ditekan maka kontak tidak berubah, apabila ditekan maka kontak NC akan berfungsi sebagai *stop* (memberhentikan) dan kontak NO akan berfungsi sebagai *start*.

9. XTAL

XTAL merupakan komponen yang berfungsi untuk membangkitkan frekuensi osilasi dengan stabilitas yang sangat tinggi. Frekuensi osilasi didapat dari efek *piezoelektrik*. *Piezoelektrisitas* adalah kemampuan dari suatu benda (pada umumnya kristal dan keramik) untuk menghasilkan potensial listrik sebagai respon terhadap tekanan mekanik yang diberikan. Bahan yang biasa digunakan untuk memperoleh efek *piezoelektrik* diantaranya kwarsa, garam *Rochelle* dan *tourmaline*. Bahan yang banyak digunakan adalah kristal kwarsa. XTAL memiliki beberapa ukuran frekuensi yang dapat digunakan sesuai dengan kebutuhan, diantaranya 1Mhz, 4Mhz, 8Mhz, 11.0592Mhz, 12Mhz dan lain-lain.



Sumber : http://quarndon.co.uk/index.php?main_page=index&cPath=56_81

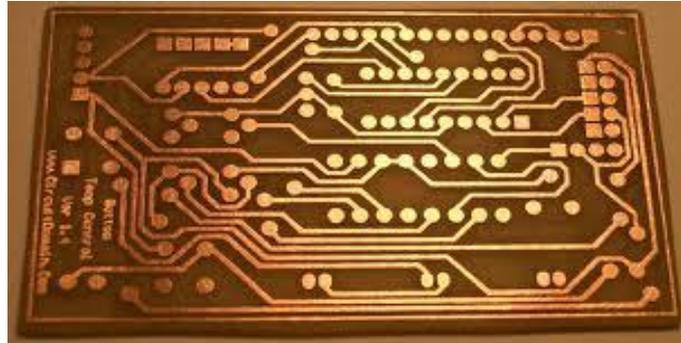
Gambar II.14

Simbol dan gambar Xtal

10. Printed Circuit Board (PCB)

Menurut Winarno dan Arifianto (2011:34) menjelaskan bahwa “PCB adalah papan rangkaian yang terbuat dari bahan isolator dengan permukaan yang dilapisi tembaga”. Papan PCB juga memiliki jalur-jalur konduktor yang terbuat

dari tembaga dan berfungsi untuk menghubungkan antara satu komponen dengan komponen lainnya.



Sumber : <http://www.circuitdomain.com/PCB%20Prototyping/PCB.htm>

Gambar II.15

Printed Circuit Board

Bahan yang digunakan untuk membuat PCB adalah sejenis *fiber* sebagai media isolasinya yang dilapisi cat berwarna hijau, sedangkan untuk jalur konduktor menggunakan tembaga. Ada beberapa macam jenis PCB menurut kegunaannya, yaitu PCB 1 side (biasa digunakan pada rangkaian elektronika seperti radio, TV, dll.) dan PCB *double side* (maksudnya kedua sisi PCB digunakan untuk menghubungkan komponen) dan PCB *multi side* (bagian PCB luar maupun dalam digunakan sebagai media penghantar, misalnya pada rangkaian-rangkaian PC).

11. RS232

RS232 adalah *standard* komunikasi *serial* yang digunakan untuk koneksi *peripheral* ke *peripheral*. Biasa juga disebut dengan jalur I/O (*input /output*). Menurut Andriyanto (2013:86) mengemukakan bahwa “RS232 merupakan standar *interface* untuk komunikasi asinkron yang menghubungkan *Data*

Terminal Equipment (DTE) dengan *Data Communication Equipment* (DCE) atau dapat juga menghubungkan antara DTE dengan DTE”. Contoh yang paling sering kita temui adalah koneksi antara komputer dengan *modem*, atau komputer dengan *mouse* bahkan bisa juga antara komputer dengan sebuah alat elektronik, semua biasanya dihubungkan lewat jalur *port serial RS232*.

Serial port RS232 pada konektor DB9 memiliki pin 9 buah. Fungsi dari masing-masing pin ditunjukkan pada tabel di bawah ini:

Tabel II.2
Fungsi pin pada RS232

Pin DB9	Singkatan	Keterangan
Pin 3	TD	Transmit Data
Pin 2	RD	Receive Data
Pin 7	RTS	Request To Send
Pin 8	CTS	Clear To Send
Pin 6	DSR	Data Set Ready
Pin 5	SG	Signal Ground
Pin 1	CD	Carrier Detect
Pin 4	DTR	Data Terminal Ready
Pin 9	RI	Ring Indikator

Keterangan mengenai fungsi saluran RS232 pada konektor DB-9 adalah sebagai berikut:

- a. *Received Line Signal Detect*, dengan saluran ini DCE memberitahukan ke DTE bahwa pada terminal masukan ada data masuk.
- b. *Receive Data*, digunakan DTE menerima data dari DCE.
- c. *Transmit Data*, digunakan DTE mengirimkan data ke DCE.
- d. *Data Terminal Ready*, pada saluran ini DTE memberitahukan kesiapan terminalnya.
- e. *Signal Ground*, saluran *ground*.

- f. *DCE ready*, sinyal aktif pada saluran ini menunjukkan bahwa DCE sudah siap.
- g. *Request to Send*, dengan saluran ini DCE diminta mengirim data oleh DTE.
- h. *Clear to Send*, dengan saluran ini DCE memberitahukan bahwa DTE boleh mulai mengirim data.
- i. *Ring Indicator*, pada saluran ini DCE memberitahukan ke DTE bahwa sebuah stasiun menghendaki hubungan dengannya.



Sumber : <http://www.usb-port.com/converters.html>

Gambar II.16
Kabel *USB to serial*

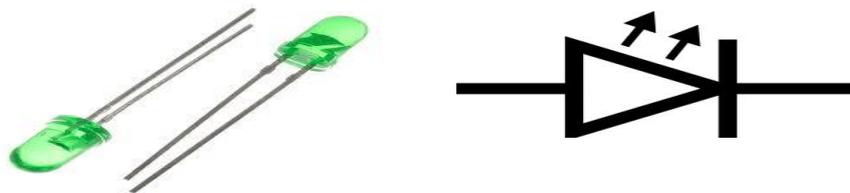
DTE (*Data Terminal Equipment*) adalah instrumen akhir yang mengubah informasi pengguna menjadi sinyal atau *reconverts* sinyal yang diterima. Sedangkan DCE (*Data Circuit-Terminating Equipment*) adalah perangkat yang berada di antaraperalatan terminal data (DTE) dan rangkaian transmisi data.

B. Sensor

1. LED (*Light Emitting Diode*) dan *Infrared*

Menurut Jatmika (2011:74) menjelaskan bahwa “LED (*Light Emitting Diode*) merupakan jenis dioda yang jika diberikan tegangan *forward-bias* akan menimbulkan cahaya dengan warna-warna tertentu seperti merah, hijau dan kuning”. Pada dasarnya LED itu merupakan komponen elektronika yang terbuat

dari bahan semi konduktor jenis dioda yang mampu memancarkan cahaya. LED merupakan produk temuan lain setelah dioda. Strukturnya juga sama dengan dioda, tetapi belakangan ditemukan bahwa elektron yang menerjang sambungan P-N. Untuk mendapatkan emisi cahaya pada semikonduktor, *doping* yang dipakai adalah galium, *arsenic* dan *phosporus*. Jenis *doping* yang berbeda menghasilkan warna cahaya yang berbeda pula.



Sumber : <http://www.tandyonline.co.uk/components/optoelectronics/leds.html>

Gambar II.17
Simbol dan Gambar LED

Karena LED adalah salah satu jenis dioda maka LED memiliki 2 kutub yaitu anoda dan katoda. Dalam hal ini LED akan menyala bila ada arus listrik mengalir dari anoda menuju katoda. Pemasangan kutub LED tidak boleh terbalik karena apabila terbalik kutubnya maka LED tersebut tidak akan menyala. LED memiliki karakteristik berbeda-beda menurut warna yang dihasilkan. Semakin tinggi arus yang mengalir pada led maka semakin terang pula cahaya yang dihasilkan, namun perlu diperhatikan bahwa besarnya arus yang diperbolehkan 10mA-20mA dan pada tegangan 1,6V–3,5V menurut karakter warna yang dihasilkan. Apabila arus yang mengalir lebih dari 20mA maka led akan terbakar. Untuk menjaga agar LED tidak terbakar perlu kita gunakan *resistor* sebagai penghambat arus.

Keunggulan LED antara lain konsumsi listrik rendah, tersedia dalam berbagai warna, murah dan umur panjang. Keunggulannya ini membuat LED digunakan secara luas sebagai lampu indikator pada peralatan elektronik dan sensor infra merah pada LED *infrared*.

2. *Photodiode*

Menurut Jatmika (2011:74) menerangkan bahwa “*Photodiode* adalah dioda yang bekerja berdasarkan intensitas cahaya, di mana jika *photodiode* terkena cahaya, maka *photodiode* bekerja seperti dioda pada umumnya”. *Photodiode* merupakan sensor cahaya semikonduktor yang dapat mengubah besaran cahaya menjadi besaran listrik. *Photodiode* merupakan sebuah dioda dengan sambungan pn yang dipengaruhi cahaya dalam kerjanya. Cahaya yang dapat dideteksi oleh *photodiode* ini mulai dari cahaya infra merah, cahaya tampak, ultra ungu sampai dengan sinar-X.



Sumber: <http://www.digikey.com/product-detail/en/SFH213/475-1077-ND/607286>

Gambar II.18
Simbol dan Gambar *Photodiode*

Prinsip kerja dari *photodiode* jika sebuah sambungan pn dibias maju dan diberikan cahaya padanya maka pertambahan arus sangat kecil sedangkan jika sambungan pn dibias mundur arus akan bertambah cukup besar. Cahaya yang dikenakan pada *photodiode* akan mengakibatkan terjadinya pergeseran foton yang akan menghasilkan pasangan *electron-hole* di kedua sisi dari sambungan. Ketika

elektron-elektron yang dihasilkan itu masuk ke pita konduksi maka elektron-elektron itu akan mengalir ke arah positif sumber tegangan sedangkan hole yang dihasilkan mengalir ke arah negatif sumber tegangan sehingga arus akan mengalir di dalam rangkaian. Besarnya pasangan elektron ataupun *hole* yang dihasilkan tergantung dari besarnya intensitas cahaya yang dikenakan pada *photodiode*.

C. *Buzzer* dan LCD

1. *Buzzer*

Menurut Supriatna (2013:16) menerangkan bahwa “*Buzzer* merupakan alat yang dapat menghasilkan bunyi atau suara karena mempunyai membran yang terhubung dengan magnet dan koil”. Pada dasarnya prinsip kerja *buzzer* hampir sama dengan *loud speaker*, jadi *buzzer* juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara. *Buzzer* biasa digunakan sebagai indikator bahwa proses telah selesai atau terjadi suatu kesalahan pada sebuah alat (*alarm*).



Sumber : <http://www.oddwires.com/piezo-electronic-tone-buzzer-alarm-3-24v-12vdc-with-mounting-holes/>

Gambar II.19
Simbol dan Gambar *Buzzer*

2. LCD (*Liquid Crystal Display*) 16x2

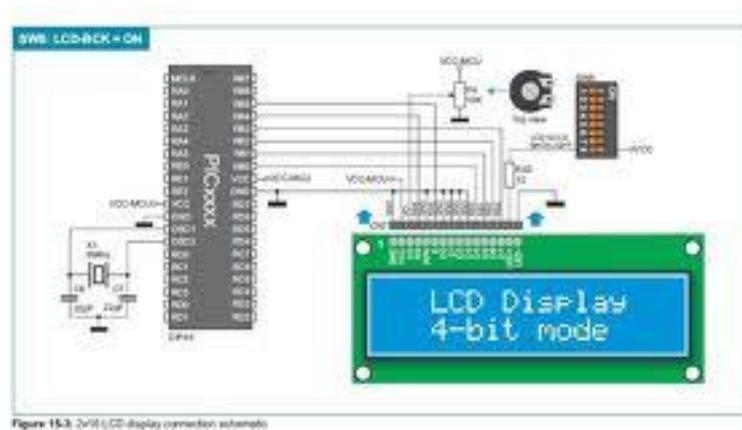
Menurut Nurcahyo (2012:42) menjelaskan bahwa “LCD merupakan jenis *device* penampil yang menggunakan teknologi *crystal* cair. *Crystal* cair disusun dalam kaca kemudian dilengkapi rangkaian elektronik sehingga dapat di konfigurasi untuk menampilkan titik, garis, huruf, angka ataupun gambar”. Kegunaan LCD banyak sekali dalam perancangan suatu sistem dengan menggunakan menggunakan mikrokontroler, LCD (*Liquid Crystals Display*) dapat berfungsi untuk menampilkan suatu nilai hasil sensor, menampilkan teks, atau menampilkan menu pada aplikasi mikrokontroler. M1632 merupakan modul LCD *matrix* dengan konfigurasi 16 karakter dan 2 baris dengan setiap karakternya dibentuk oleh baris *pixel* dan 5 kolom *pixel* (1 baris *pixel* terakhir adalah kursor).

Didalam modul M1632 sudah tersedia HD44780 yang dikeluarkan oleh Hitachi, Hyundai dan modul-modul M1632 lainnya. HD44780 sebetulnya merupakan mikrokontroler dirancang khusus untuk mengendalikan LCD dan mempunyai kemampuan untuk mengatur proses *scanning* pada layar LCD yang terbentuk oleh 16 COM dan 40 SEG sehingga mikrokontroler/perangkat yang mengakses modul LCD ini tidak perlu lagi mengatur *scanning* pada layar LCD. Mikrokontroler atau perangkat tersebut hanya mengirim data-data yang merupakan karakter yang akan ditampilkan pada LCD atau perintah yang mengatur proses tampilan pada LCD saja.

Adapun konfigurasi dan deskripsi dari pin-pin LCD M1632 antara lain:

- a. Pin 1 dihubungkan ke Gnd.
- b. Pin 2 dihubungkan ke Vcc +5V.

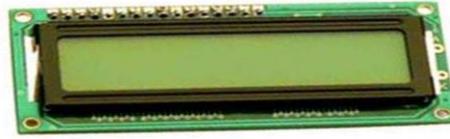
- c. Pin 3 dihubungkan ke bagian tengah potensiometer 10K ohm sebagai pengatur kontras.
- d. Pin 4 untuk memberitahukan LCD bahwa sinyal yang dikirim adalah data, jika pin ini di set ke logika 1 (*high*, +5V), atau memberitahukan bahwa sinyal yang dikirim adalah perintah jika pin ini di set ke logika 0 (*low*, 0V).
- e. Pin 5 digunakan untuk mengatur fungsi LCD. Jika di set ke logika 1 (*high*, +5V) maka LCD berfungsi untuk menerima data (membaca data). Dan fungsi untuk mengeluarkan data, jika pin ini di set ke logika 0 (*low*, 0V). Namun kebanyakan aplikasi hanya digunakan untuk menerima data, sehingga pin 5 ini biasanya dihubungkan ke Gnd.
- f. Pin 6 adalah terminal *enable*. Berlogika 1 setiap kali pengiriman atau pembaca data.
- g. Pin 7 – Pin 14 adalah data 8 bit data bus (Aplikasi ini menggunakan 4 bit MSB saja, sehingga pin data yang digunakan hanya pin 11 – pin 14).
- h. Pin 15 dan Pin 16 adalah tegangan untuk menyalakan lampu LCD.



Sumber : <http://www.mikroe.com/forum/viewtopic.php?f=12&t=44881>

Gambar II.20
Skematik LCD 16x2

Adapun gambar dari LCD 16x2 adalah sebagai berikut:



Sumber : http://www.skpang.co.uk/catalog/lcd-displays-16x2-lcd-33v-c-91_207.html

Gambar II.21

LCD 16x2

Modul LCD M1632 memiliki beberapa jenis memori yang digunakan untuk menyimpan atau memproses data-data yang ditampilkan pada layar LCD. Setiap memori mempunyai fungsi-fungsi tersendiri.

1. DDRAM

DDRAM merupakan memori tempat karakter yang ditampilkan berada. Contohnya karakter 'A' atau 41h yang ditulis pada alamat 00 akan tampil pada baris pertama dan kolom pertama dari LCD. Apabila karakter tersebut di alamat 40h, karakter tersebut akan tampil pada baris kedua kolom pertama dari LCD.

2. CGRAM

CGRAM merupakan memori untuk menggambarkan pola seluruh karakter dan bentuk karakter dapat diubah-ubah sesuai keinginan. Akan tetapi isi memori akan hilang saat *power supply* tidak aktif sehingga pola karakter akan hilang.

3. CGROM

Adalah memori untuk menggambarkan pola sebuah karakter dan pola tersebut ditentukan secara permanen dari HD44780 sehingga pengguna tidak dapat mengubah lagi. Oleh karena ROM bersifat permanen, pola karakter tersebut akan hilang walaupun *power supply* tidak aktif.

D. Mikrokontroler ATmega16

Menurut Andrianto (2015:1) menjelaskan bahwa “Mikrokontroler atau pengendali mikro adalah sebuah komputer kecil (“*special purpose computers*”) di dalam sebuah IC/*chip*. Mikrokontroler lebih dari sekedar sebuah mikroprosesor karena sudah terdapat atau berisikan ROM (*Read-Only Memory*), RAM (*Read-Write Memory*), beberapa bandar masukan maupun keluaran, dan beberapa *peripheral* seperti pencacah/pewaktu, ADC (*Analog to Digital converter*), DAC (*Digital to Analog converter*) dan serial komunikasi. Salah satu mikrokontroler yang banyak digunakan saat ini yaitu mikrokontroler AVR.

Secara umum mikrokontroler AVR dapat dikelompokkan menjadi 3 kelompok, yaitu keluarga AT90Sxx, ATmega dan ATtiny. Pada dasarnya yang membedakan masing-masing kelas adalah memori, *peripheral*, dan fiturnya. Seperti mikroprosesor pada umumnya, secara internal mikrokontroler ATmega16 terdiri atas unit-unit fungsionalnya *Arithmetic and Logical Unit* (ALU), himpunan register kerja, register dan dekoder instruksi, dan pewaktu beserta komponen kendali lainnya. Berbeda dengan mikroprosesor, mikrokontroler menyediakan memori dalam serpih yang sama dengan prosesornya (*in chip*).

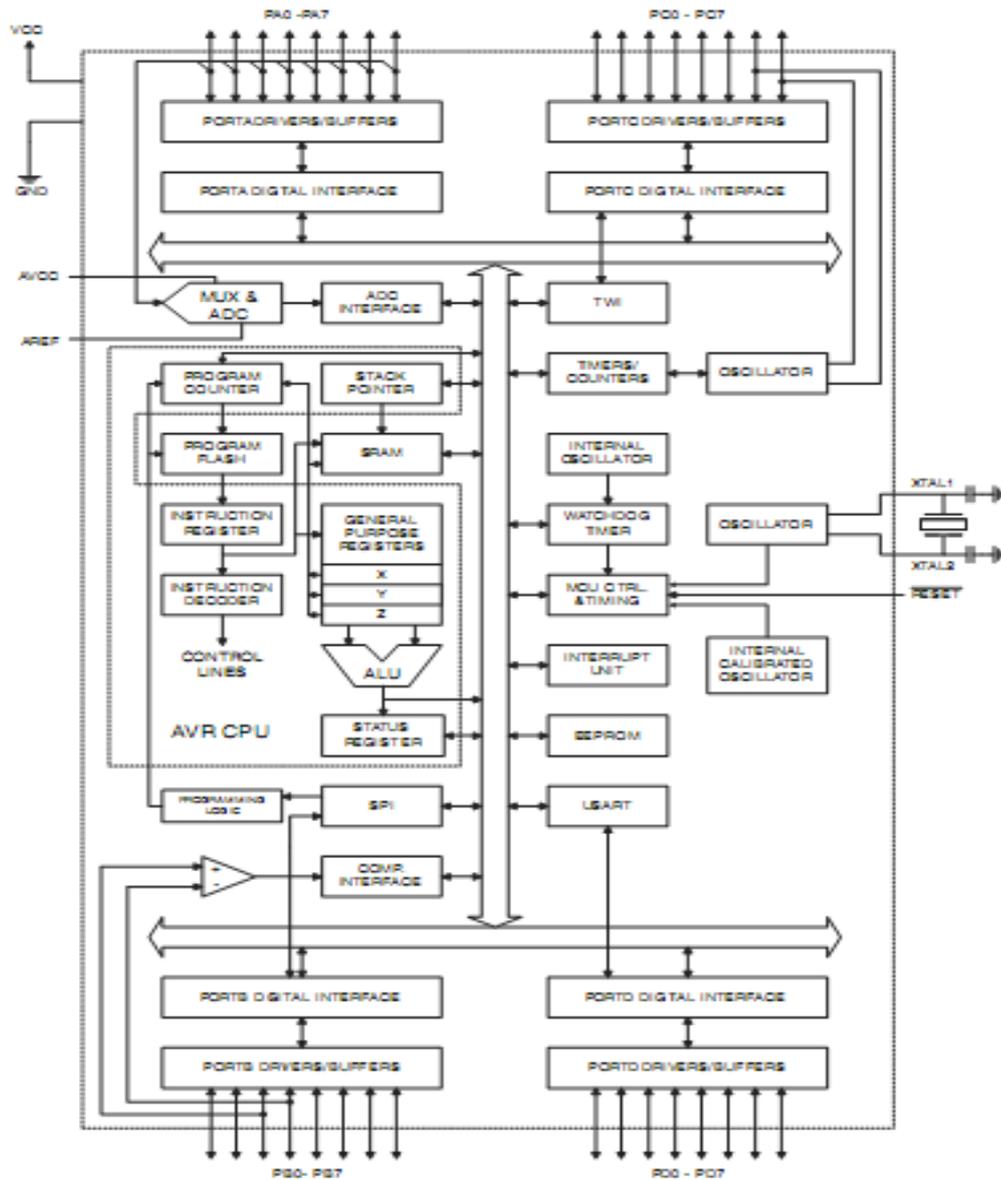
1. Arsitektur ATmega16

Mikrokontroler ini menggunakan arsitektur Harvard yang memisahkan memori program dari memori data, baik bus alamat maupun bus data, sehingga pengaksesan program dan data dapat dilakukan secara bersamaan (*concurrent*).

Secara garis besar mikrokontroler ATMEGA16 terdiri dari:

- a. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16Mhz.

- b. Memiliki kapasitas *Flash* memori 16Kbyte, EEPROM 512 Byte, danSRAM 1Kbyte
- c. Saluran I/O 32 buah, yaitu *Port A*, *Port B*, *Port C*, dan *Port D*.
- d. CPU yang terdiri dari 32 buah register.
- e. User interupsi internal dan eksternal.
- f. Bandar antarmuka SPI dan Bandar USART sebagai komunikasi serial
- g. Fitur Peripheral
 - 1) Dua buah 8-bit *timer/counter* dengan prescaler terpisah dan mode *Compare*
 - 2) Satu buah 16-bit timer/counter dengan prescaler terpisah, mode *compare*, dan mode *capture*
 - 3) *Real time counter* dengan osilator tersendiri
 - 4) Empat kanal PWM dan Antarmuka komparator analog
 - 5) 8 kanal, 10 bit ADC
 - 6) *Byte-oriented Two-wire Serial Interface*
 - 7) *Watchdog timer* dengan osilator internal.



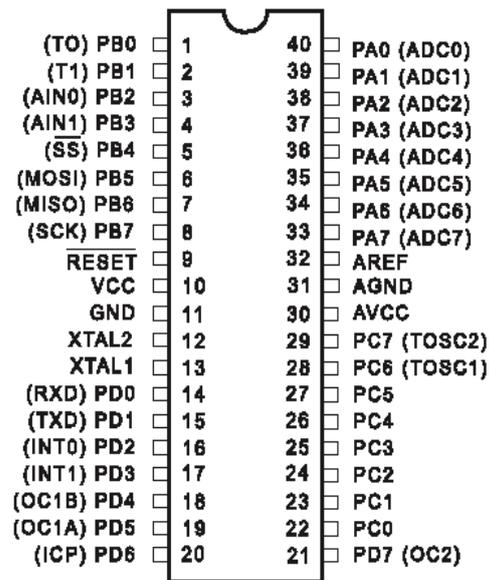
sumber: www.circuitstoday.com

Gambar II.22

Blok Diagram ATMEGA16

2. Konfigurasi Pin ATMEGA16

Konfigurasi pena (*pin*) mikrokontroler ATMEGA16 dengan kemasan 40-pena dapat dilihat pada Gambar II.22. Dari gambar tersebut dapat terlihat ATMEGA16 memiliki 8 pena untuk masing-masing *Port A*, *Port B*, *Port C*, dan *Port D*.



Sumber: <http://www.intellecta.net>

Gambar II.23

Pena-Pena AMEGA16

3. Deskripsi Mikrokontroler ATMEGA16

Berikut ini deskripsi mikrokontroler yang ada pada ATmega16.

- a. VCC (*Power Supply*) dan GND(*Ground*)
- b. Port A (PA7..PA0)

Port A berfungsi sebagai *input* analog pada konverter A/D. Port A juga sebagai suatu *port* I/O 8-bit dua arah, jika A/D konverter tidak digunakan.

Pena-pena *port* dapat menyediakan resistor *internal pull-up* (yang dipilih untuk masing-masing bit). Port A *output* buffer mempunyai karakteristik gerakan simetris dengan keduanya sink tinggi dan kemampuan sumber. Ketika pena PA0 ke PA7 digunakan sebagai input dan secara eksternal ditarik rendah, pena-pena akan memungkinkan arus sumber jika resistor *internal pull-up* diaktifkan. Pena PortA adalah *tri-stated* manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

c. *Port B (PB7..PB0)*

Port B adalah suatu bandar I/O 8-bit dua arah dengan resistor *internal pull-up* (yang dipilih untuk beberapa bit). *Port B output buffer* mempunyai karakteristik gerakan simetris dengan keduanya sink tinggi dan kemampuan sumber. Sebagai input, pena *Port B* yang secara eksternal ditarik rendah akan arus sumber jika resistor *pull-up* diaktifkan. Pena *Port B* adalah *tri-stated* manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

d. *Port C (PC7..PC0)*

Port C adalah suatu *Port I/O* 8-bit dua arah dengan resistor *internal pull-up* (yang dipilih untuk beberapa bit). *Port C output buffer* mempunyai karakteristik gerakan simetris dengan keduanya sink tinggi dan kemampuan sumber. Sebagai input, pena *Port C* yang secara eksternal ditarik rendah akan arus sumber jika resistor *pull-up* diaktifkan. Pena *Port C* adalah *tri-stated* manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

e. *Port D (PD7..PD0)*

Port D adalah suatu *Port I/O* 8-bit dua arah dengan resistor *internal pull-up* (yang dipilih untuk beberapa bit). *Port D output buffer* mempunyai karakteristik gerakan simetris dengan keduanya sink tinggi dan kemampuan sumber. Sebagai input, pena *Port D* yang secara eksternal ditarik rendah akan arus sumber jika resistor *pull-up* diaktifkan. Pena *Port D* adalah *tri-stated* manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

f. RESET (*Reset input*).

g. XTAL1 (*Input Oscillator*).

h. XTAL2 (*Output Oscillator*).

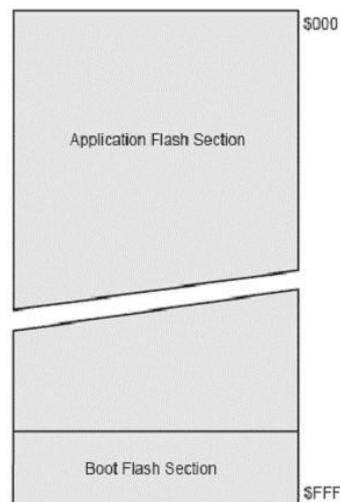
- i. AVCC adalah pena penyedia tegangan untuk bandar A dan Konverter A/D.
- j. AREF adalah pena referensi analog untuk konverter A/D.

4. Peta Memori ATmega16

Berikut ini peta memori program pada ATmega16.

a. Memori Program

Arsitektur ATMEGA16 mempunyai dua memori utama, yaitu memori data dan memori program. Selain itu, ATMEGA16 memiliki memori EEPROM untuk menyimpan data. ATMEGA16 memiliki 16K byte *On-chip In-System Reprogrammable Flash Memory* untuk menyimpan program. Instruksi ATMEGA16 semuanya memiliki format 16 atau 32 bit, maka memori *flash* diatur dalam 8K x 16 bit. Memori *flash* dibagi kedalam dua bagian, yaitu bagian program *boot* dan aplikasi seperti terlihat pada Gambar 2.23. *Bootloader* adalah program kecil yang bekerja pada saat sistem dimulai yang dapat memasukkan seluruh program aplikasi ke dalam memori prosesor.



Sumber: <http://www.dc312.4shared.com>

Gambar 2.24

Peta Memori ATMEGA16

b. Memori Data (SRAM)

Memori data AVR ATMEGA16 terbagi menjadi 3 bagian, yaitu 32 register umum, 64 buah register I/O dan 1 Kbyte SRAM internal. *General purpose register* menempati alamat data terbawah, yaitu \$00 sampai \$1F. Sedangkan memori I/O menempati 64 alamat berikutnya mulai dari \$20 hingga \$5F. Memori I/O merupakan register yang khusus digunakan untuk mengatur fungsi terhadap berbagai fitur mikrokontroler seperti kontrol register, *timer/counter*, fungsi-fungsi I/O, dan sebagainya. 1024 alamat berikutnya mulai dari \$60 hingga \$45F digunakan untuk SRAM internal.

c. Memori Data EEPROM

ATMEGA16 terdiri dari 512 *byte* memori data EEPROM 8 bit, data dapat ditulis/dibaca dari memori ini, ketika catu daya dimatikan, data terakhir yang ditulis pada memori EEPROM masih tersimpan pada memori ini, atau dengan kata lain memori EEPROM bersifat *nonvolatile*. Alamat EEPROM mulai dari \$000 sampai \$1FF.

5. Analog To Digital Converter

AVR ATMega16 merupakan tipe AVR yang telah dilengkapi dengan 8 saluran ADC internal dengan resolusi 10 bit. Dalam mode operasinya, ADC dapat dikonfigurasi, baik *single ended* input maupun *differential* input. Selain itu, ADC ATMega16 memiliki konfigurasi pewaktuan, tegangan referensi, mode operasi, dan kemampuan filter derau (*noise*) yang amat fleksibel sehingga dapat dengan mudah disesuaikan dengan kebutuhan dari ADC itu sendiri. ADC pada ATMEGA16 memiliki fitur-fitur antara lain:

a. Resolusi mencapai 10-bit

- b. Akurasi mencapai ± 2 LSB
- c. Waktu konversi 13-260 μ s
- d. 8 saluran ADC dapat digunakan secara bergantian
- e. Jangkauan tegangan input ADC bernilai dari 0 hingga VCC
- f. Disediakan 2,56V tegangan referensi internal ADC
- g. Mode konversi kontinyu atau mode konversi tunggal
- h. Interupsi ADC complete
- i. *Sleep Mode Noise canceler*

Proses inisialisasi ADC meliputi proses penentuan *clock*, tegangan referensi, format data keluaran, dan modus pembacaan. Register-register yang perlu diatur adalah sebagai berikut:

- a. *ADC Control and Status Register A – ADCSRA*

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Sumber: <http://www.maxembedded.com>

Gambar II.25

ADC Control and Status Register A – ADCSRA

- b. ADEN : 1 = *adc enable*, 0 = *adc disable*
- c. ADSC : 1 = mulai konversi, 0 = konversi belum terjadi
- d. ADATE : 1 = *auto trigger* diaktifkan, *trigger* berasal dari sinyal yang dipilih (set pada *trigger* SFIOR bit ADTS). ADC akan *start* konversi pada *edge* positif sinyal *trigger*.

- e. ADIF : diset ke 1, jika konversi ADC selesai dan data register ter-update.
Namun *ADC Conversion Complete Interrupt* dieksekusi jika bit ADIE dan bit-I dalam register SREG diset.
- f. ADIE : diset 1, jika bit-I dalam register SREG di-set.
- g. ADPS[0..2] : Bit pengatur clock ADC, faktor pembagi 0 ... 7 = 2, 4, 8, 16, 32, 64, 128.

Tabel II.3
Konfigurasi Clock ADC

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

h. *ADC Multiplexer-ADMUX*

ADLAR = 0		Bit	15	14	13	12	11	10	9	8	ADMUX
009 (011)									ADSC	ADIF	ADIF
004 (004)			ADSC7	ADSC6	ADSC5	ADSC4	ADSC3	ADSC2	ADSC1	ADSC0	ADIF
Read/Write			R	R	R	R	R	R	R	R	R
Initial Value			1	0	0	0	0	0	0	0	0
			1	0	0	0	0	0	0	0	0

ADLAR = 1		Bit	15	14	13	12	11	10	9	8	ADMUX
005 (005)			ADSC8	ADSC7	ADSC6	ADSC5	ADSC4	ADSC3	ADSC2		ADIF
004 (004)			ADSC1	ADSC0							ADIF
Read/Write			R	R	R	R	R	R	R	R	R
Initial Value			1	0	0	0	0	0	0	0	0
			1	0	0	0	0	0	0	0	0

Sumber:electronics.stackexchange.com

Gambar II.26.
ADC Multiplexer

REFS 0, 1 : Pemilihan tegangan referensi ADC

00 : Vref = Aref

01 : vref = AVCC dengan eksternal kapasitor pada AREF

10 : vref = internal 2.56 volt dengan eksternal kapasitor pada AREF

ADLAR : Untuk setting format data hasil konversi ADC, default = 0

i. Special Function IO Register-SFIOR

SFIOR merupakan register 8 bit pengatur sumber picu konversi ADC, apakah dari picu eksternal atau dari picu internal, susunannya seperti yang terlihat pada Gambar 2.26 berikut :

Bit	7	6	5	4	3	2	1	0	
	ADTS2	ADTS1	ADTS0	ADHSM	ACME	PUD	PSR2	PSR10	SFIOR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Sumber: <http://msoe.us/taylor/tutorial/ce2800/2800sfior.png&imgrefurl>

Gambar II.27.

Register SFIOR

ADTS[0...2] : Pemilihan trigger (pengatur picu) untuk konversi ADC, bit-bit ini akan berfungsi jika bit ADATE pada register ADCSRA bernilai 1.

Konfigurasi bit ADTS[0...2] dapat dilihat pada Tabel II.3

Tabel II.4

Pemilihan sumber picu ADC

ADTS2	ADTS1	ADTS0	Trigger Source
0	0	0	Free Running mode
0	0	1	Analog Comparator
0	1	0	External Interrupt Request 0
0	1	1	Timer/Counter0 Compare Match
1	0	0	Timer/Counter0 Overflow
1	0	1	Timer/Counter Compare Match B
1	1	0	Timer/Counter1 Overflow
1	1	1	Timer/Counter1 Capture Event

ADHSM : 1. ADC *high speed mode enabled*. Untuk operasi ADC, bit ACME, PUD, PSR2 dan PSR10 tidak diaktif.

2.3. Konsep Dasar Program

A. *Flowchart*

Flowchart merupakan gambar atau bagan yang memperlihatkan urutan dan hubungan antar proses beserta instruksinya. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung.

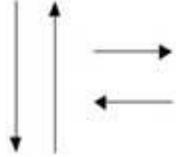
Flowchart ini merupakan langkah awal pembuatan program. Dengan adanya *flowchart* urutan poses kegiatan menjadi lebih jelas. Jika ada penambahan proses maka dapat dilakukan lebih mudah. Setelah *flowchart* selesai disusun, selanjutnya pemrogram (programmer) menerjemahkannya ke bentuk program dengan bahasa pemrograman.

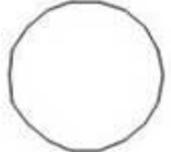
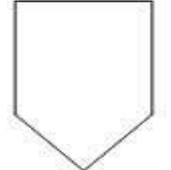
1. Simbol-Simbol *Flowchart*

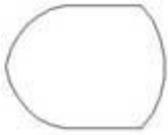
Flowchart disusun dengan simbol-simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses di dalam program. Simbol-simbol yang dipakai antara lain :

Tabel II.5

Simbol-Simbol *Flowchart*

	<p><i>Flow Direction symbol</i></p> <p>Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga <i>connecting line</i>.</p>
---	--

	<p><i>Terminator Symbol</i></p> <p>Yaitu simbol untuk permulaan (<i>start</i>) atau akhir (<i>stop</i>) dari suatu kegiatan.</p>
	<p><i>Connector Symbol</i></p> <p>Yaitu simbol untuk keluar – masuk atau penyambungan proses dalam lembar / halaman yang sama.</p>
	<p><i>Connector Symbol</i></p> <p>Yaitu simbol untuk keluar – masuk atau penyambungan proses pada lembar / halaman yang berbeda.</p>
	<p><i>Processing Symbol</i></p> <p>Simbol yang menunjukkan pengolahan yang dilakukan oleh komputer.</p>
	<p><i>Simbol Manual Operation</i></p> <p>Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh komputer</p>
	<p><i>Simbol Decision</i></p> <p>Simbol pemilihan proses berdasarkan kondisi yang ada.</p>
	<p><i>Simbol Input-Output</i></p> <p>Simbol yang menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya</p>
	<p><i>Simbol Manual Input</i></p> <p>Simbol untuk pemasukan data secara manual on-line keyboard</p>
	<p><i>Simbol Preparation</i></p> <p>Simbol untuk mempersiapkan penyimpanan yang akan digunakan sebagai tempat pengolahan di dalam <i>storage</i>.</p>
	<p><i>Simbol Predefine Proses</i></p> <p>Simbol untuk pelaksanaan suatu bagian (sub-program) atau prosedur.</p>

	<p>Simbol <i>Display</i></p> <p>Simbol yang menyatakan peralatan output yang digunakan yaitu layar, plotter, printer dan sebagainya.</p>
	<p>Simbol <i>disk and On-line Storage</i></p> <p>Simbol yang menyatakan input yang berasal dari <i>disk</i> atau disimpan ke <i>disk</i>.</p>

2. Kaidah-kaidah pembuatan *Flowchart*

Dalam pembuatan *flowchart* tidak ada rumus atau patokan yang bersifat mutlak. Karena *flowchart* merupakan gambaran hasil pemikiran dalam menganalisa suatu masalah dengan komputer. Sehingga *flowchart* yang dihasilkan dapat bervariasi antara satu pemrogram dengan pemrogram lainnya.

Namun secara garis besar, setiap pengolahan selalu terdiri dari tiga bagian utama, yaitu;

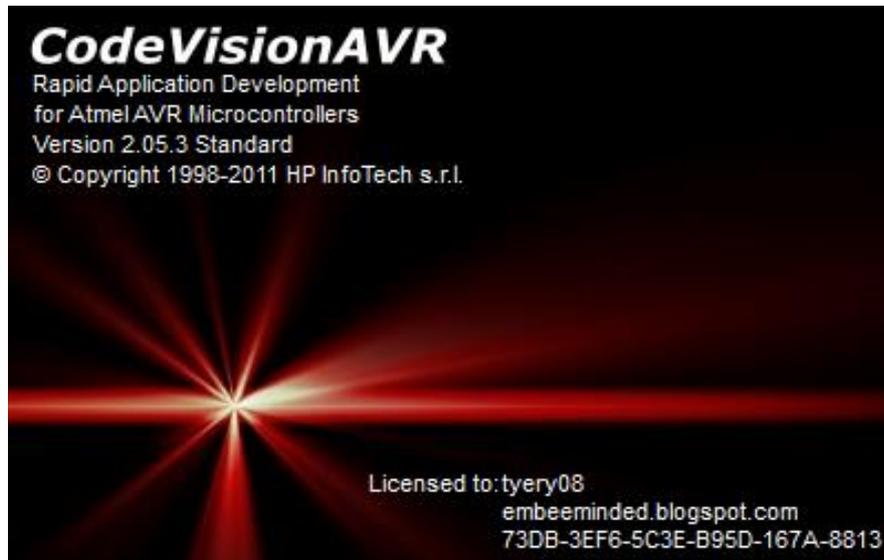
1. *Input*.
2. Proses.
3. *Output*.

Untuk pengolahan data dengan komputer, dapat dirangkum urutan dasar untuk pemecahan suatu masalah, yaitu:

- a. **START**: berisi instruksi untuk persiapan peralatan yang diperlukan sebelum menangani pemecahan masalah.
- b. **READ**: berisi instruksi untuk membaca data dari suatu peralatan input.
- c. **PROCESS**: berisi kegiatan yang berkaitan dengan pemecahan persoalan sesuai dengan data yang dibaca.

- d. **WRITE:** berisi instruksi untuk merekam hasil kegiatan ke peralatan output.
- e. **END:** mengakhiri kegiatan pengolahan.

B. CodeVisionAVR



Sumber : www.cursomicros.com

Gambar II.28

Codevision AVR

CodeVisionAVR adalah *cross-compiler* berbasis bahasa C, *Integrated Development Environment* (IDE) dan Program Generator otomatis yang khusus dirancang untuk mikrokontroler keluarga Atmel AVR Mikrokontroler. Program ini dirancang untuk dijalankan pada sistem operasi XP, Vista dan Windows 7 arsitektur 32 bit atau 64 bit. Selain library C standar, didalam IDE *codevision* telah dilengkapi dengan:

1. *Modul Alphanumerik* dan *Graphic LCD*.
2. Sensor suhu LM75 nasional semikonduktor.
3. *Philips PCF8563, PCF8583, Maxim/Dallas Semiconductor DS1307 DS1302* dan *Real Time clock*.

4. *Maxim/Dallas Semiconductor protokol one wire.*
5. *Maxim/Dallas Semiconductor DS1820, DS18S20 Sensor Suhu dan DS18B20.*
6. *Maxim/Dallas Semiconductor DS1621 Termometer/Thermostat.*
7. *Maxim/Dallas Semiconductor DS2433 DS2430 dan EEPROM.*
8. SPI.
9. *ADC (Analog To Digital Converter).*
10. Komparator.
11. TWI untuk XMEGA dan non-XMEGA *chip*.
12. *Power management.*
13. *delay/timer.*
14. MMC/SD/SD HC *FLASH*.

Selain itu *software* ini juga dilengkapi dengan *chip programmer* yang kompatibel dengan Atmel STK500, STK501, STK502, STK600, AVRISP (serial), AVRISP MkII (USB), AVR Dragon, JTAGICE MkII, Atmel AVR910 *application note*, Kanda Systems STK200+ and STK300 *development boards*, Vogel Elektronik VTEC-ISP, Dontronics DT006 sehingga kita tidak memerlukan *tool* tambahan untuk men-*download* hasil *compile* ke Mikrokontroler target.

Bahasa yang digunakan dalam *CodeVisionAVR* adalah bahasa C. Bahasa C mempunyai kemampuan lebih dari bahasa pemrograman lain. Bahasa C merupakan bahasa yang bersifat *portable* yaitu dengan sedikit perubahan suatu program yang ditulis dengan bahasa C pada suatu komputer dapat dijalankan di komputer lain.

Keunggulan bahasa C lainnya adalah kecepatan prosesnya, dengan tersedianya beberapa pustaka dari bahasa C memungkinkan seorang *programmer* mengembangkan programnya lebih luas tanpa harus menulis keseluruhan dari programnya.

Di dalam pemrograman bahasa C terdapat statement-statement kondisi yang digunakan antara lain:

1. Pernyataan *if*

Pernyataan *if* digunakan untuk melakukan pengambilan keputusan terhadap dua buah kemungkinan yaitu mengerjakan suatu blok pernyataan atau tidak, jika dan hanya jika persyaratannya terpenuhi.

```
if (kondisi)
{
    // pernyataan
};
```

Artinya adalah pernyataan akan dijalankan jika kondisi terpenuhi.

Perhatikan contoh :

```
if (a<0x50)
{
    PORTC=0x55;
};
```

Dalam contoh ini PORTC akan dikirim data 0x55 jika nilai a lebih kecil 0x50.

2. Pernyataan *if... else*

Pernyataan *if... else* digunakan untuk melakukan pengambilan keputusan terhadap dua buah kemungkinan, kedua kemungkinan tersebut adalah mengerjakan pernyataan satu atau mengerjakan pernyataan yang lain.

```
if (kondisi)
{
    // pernyataan a
}
else
{
    // pernyataan b
};
```

Artinya adalah pernyataan a akan dijalankan jika kondisi terpenuhi dan pernyataan b akan dijalankan jika kondisi tidak terpenuhi.

Perhatikan contoh dibawah ini:

```
if (a<0x50)
{
    PORTC=0x55;
}
else
{
    PORTC=0xAA;
};
```

PORTC akan dikirim data 0x55 jika a lebih kecil dari 0x50 dan PORTC akan dikirim data 0xAA jika $a \geq 0x50$.

3. Pernyataan *Switch*

Pernyataan *switch* digunakan untuk melakukan pengambilan keputusan terhadap banyak kemungkinan. Pernyataan *switch – case* digunakan jika terjadi banyak percabangan.

```

.....
switch (ekspresi)
{
case konstanta1: Pernyataan1; break;
case konstanta2: pernyataan2 break;
.....
case konstantaN: pernyataanN break;

```

Perhatikan contoh dibawah ini:

```

switch (a)
{
case 1 : PORTC=0x01;break;
case 2 : PORTC=0x02;break;
case 3 : PORTC=0x04;break;
}

```

PORTC akan dikirim data 0x01 jika nilai $a=1$, PORTC akan dikirim data 0x02 jika nilai $a=2$ dan PORTC akan dikirim data 0x04 jika nilai $a=3$.

4. Pernyataan *while*

Pernyataan *while* digunakan untuk pengulangan sebuah pernyataan atau blok pernyataan secara terus menerus selama kondisi tertentu masih terpenuhi.

Bentuk perulangan *while* adalah sebagai berikut:

```
while (kondisi)
{
    pernyataan-pernyataan;
}
```

Jika kondisi memenuhi (bernilai *true*) maka pernyataan-pernyataan dibawahnya akan dijalankan hingga selesai, kemudian akan menguji kembali kondisi diatas. Perhatikan contoh dibawah ini:

```
i=1;
a=1;
while (<50)
{
    a=a*2;
    PORTC=a;
    i++;
};
```

5. Pernyataan *do...while*

Pernyataan *do ... while* digunakan sama seperti penggunaan pernyataan dari *while*. Bentuk perulangan ini kebalikan dari *while-do*, yaitu pernyataan dilakukan terlebih dahulu kemudian diuji kondisinya.

```
do
{
```

```

pernyataan-pernyataan;
}
while (kondisi);

```

Perhatikan contoh berikut ini:

```

i=1;
a=1;
do
{
a=a*2;
PORTC=a;;
i++ ;
}
while (i<50);

```

6. Pernyataan *for*

Pernyataan *for* digunakan juga untuk melakukan pengulangan sebuah pernyataan atau blok pernyataan, tetapi beberapa kali jumlah pengulangannya dapat ditentukan secara lebih spesifik. Pernyataan *for* akan melakukan perulangan beberapa kali sesuai yang diinginkan. Berikut *sintaks* dari penulisan pernyataan *for*:

```

for (mulai ; kondisi ; penambahan/pengurangan)
{
Pernyataan-pernyataan;
};

```

Perhatikan contoh berikut ini:

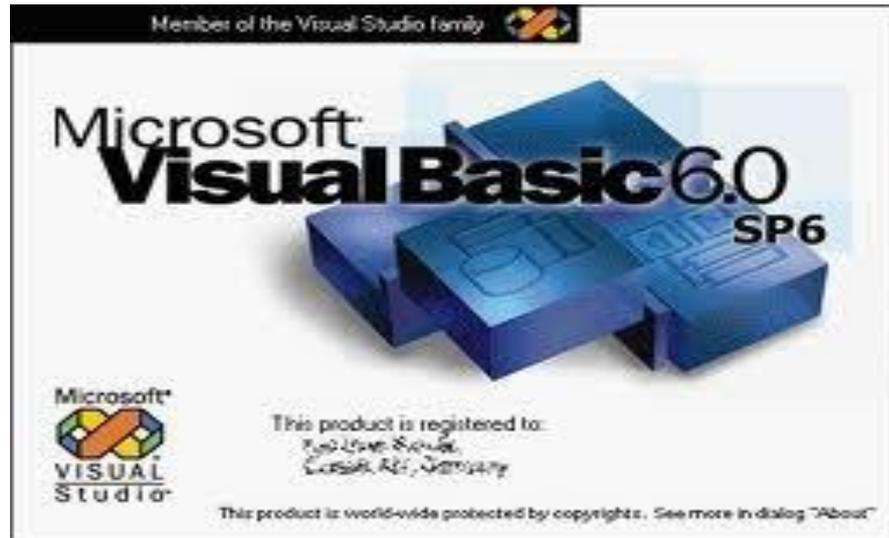
```
a=1;
for (i=1; i<50; i++)
{
    a=a*2;
    PORTC=a;
};
```

Contoh program tersebut akan melakukan perulangan 49 kali, yaitu dari 1 hingga 50 dengan menambahkan 1 (i++). Hasilnya PORTC akan dikirim data 1 kemudian data 2,4,8.... sesuai dengan persamaan $a=a*2$.

C. Visual Basic 6.0

Microsoft Visual Basic (VB) adalah sebuah turunan bahasa pemrograman *Visual Basic* yang dikembangkan oleh *Microsoft*. Kombinasi yang terintegrasi antara lingkungan pemrograman (*Visual Basic Editor*) dengan bahasa pemrograman (*Visual Basic*) yang memudahkan *user* untuk mendesain dan membangun program *Visual Basic* dalam aplikasi utama *Microsoft Office*, yang ditujukan untuk aplikasi-aplikasi tertentu.

Menurut Suhata, (2005:3), “*Visual Basic 6.0* merupakan salah satu bahasa pemrograman yang dapat digunakan untuk menyusun dan membuat program aplikasi pada lingkungan sistem operasi *windows*”. Program aplikasi dapat berupa program database, program grafis, dan lain sebagainya. Di dalam *visual basic 6.0* sudah terdapat komponen-komponen yang sangat membantu pembuatan program aplikasi.



Sumber : <http://www.thegeeksclub.com/16870-gambas-basic-ide-linux-review/>

Gambar II.29

Visual Basic 6.0

VB didesain untuk melakukan beberapa tugas, seperti halnya mengkustomisasi sebuah aplikasi layaknya *Microsoft Office* atau *Microsoft Visual Studio*. Kegunaan VB adalah mengotomatisasi pekerjaan. Pekerjaan yang dimaksud adalah pekerjaan yang dilakukan secara berulang-ulang dan pekerjaan yang *kompleks* diantaranya adalah mampu melakukan *transmit* dan *receive* data melalui komunikasi serial yang telah disediakan oleh *Microsoft Visual Basic*. *Microsoft Visual Basic* memberi banyak pemrograman dan fungsi tingkat lanjut hingga *Microsoft Visual Basic* dapat dihasilkan program yang lebih *kompleks* untuk sistem operasi *Microsoft Windows* maupun *Office*. *Microsoft* umumnya membuat 3 edisi *Visual Basic* yaitu:

- a. ***Standard Edition***, merupakan produk pasar.
- b. ***Professional Edition***, berisi tambahan *Microsoft Jet Data Access Engine* (*database*) dan pembuatan server *OLE automation*.
- c. ***Enterprise Edition***, merupakan edisi *client-server*.

Di dalam pemrograman *Visual Basic* terdapat statement-statement kondisi yang digunakan antara lain:

1. Percabangan

a. *If...Then*

Percabangan dengan menggunakan *If ... Then* digunakan untuk memilih dua kondisi atau lebih dengan memeriksa syarat pada tiap kondisi, jika syarat pada kondisi 1 benar (*True*) maka program akan menjalankan pernyataan 1 jika tidak (*False*) maka program akan memeriksa kondisi 2, jika syarat pada kondisi 2 benar (*True*) maka program akan menjalankan pernyataan 2 jika tidak (*False*) maka program akan memeriksa kondisi berikutnya dan begitu seterusnya hingga kondisi terakhir pada percabangan.

if Kondisi1 Then

Pernyataan1

Else if Kondisi2 Then

Pernyataan2

Else

Pernyataan3

End if

Contoh program yang menggunakan *if...then*

Dim Suhu Sebagai Double

...

if => Suhu Then

Debug. Print "Ini sangat panas"

Else if 30,0 <= Suhu Dan <Suhu = 39,0 Then

Debug. Print "Panas"

Else if 20,0 <= Suhu Dan <Suhu = 29,0 Then

Debug. Print "Ini hangat"

Else if 10,0 <= Suhu Dan <suhu = 19,0 Then

Debug. Print "It's cool"

Else if 0.0 <= Suhu Dan <Suhu = 9,0 Then

Debug. Print "Dingin"

Else

Debug. Print "Dingin"

End if

b. *Select Case*

Dibandingkan dengan percabangan *If ... Then*, *Select Case* lebih mudah dibaca dan di *debug* serta lebih mudah digunakan dalam alur program yang membutuhkan banyak percabangan.

Bentuk umum perabangan *select case* adalah sebagai berikut :

Select Case Pilihan

Case 1 Pernyataan1

Case 2 Pernyataan2

Case 3 Pernyataan3

End Select

Contoh program dengan menggunakan *Select Case* :

Pilih Kasus Suhu

Case Apakah > = 40 #

Debug. Print "Ini sangat panas"

Case 30 # Untuk 39 #

Debug. Print "Panas"

Case 20 # Untuk 29 #

Debug. Print "Ini hangat"

Case 10 # Untuk 19 #

Debug. Print "It's cool"

Case 0 # Untuk 9 #

Debug. Print "Dingin"

Case Else

Debug. Print "Dingin"

End Select

2 . Pengulangan

Dalam pemrograman ada kalanya kita memerlukan perulangan untuk melakukan suatu perintah yang sama untuk beberapa kali. Pada visual basic terdapat beberapa macam struktur kondisi perulangan, diantaranya adalah *Do ... Loop* dan *For ... Next*. Untuk lebih jelasnya berikut adalah macam – macam bentuk perulangan dalam *visual basic* :

a. Do While ... Loop

Kata *while* dalam pernyataan *Do While* memerintahkan program untuk melakukan perulangan apabila kondisi persyaratan masih bernilai *True*. Apabila kondisi persyaratan bernilai *False* maka program akan keluar dari perulangan dan

berpindah ke kode program berikutnya. Bentuk sintaks perulangan *Do While* adalah :

Do While Kondisi

Pernyataan

.....

Loop

Sintaks di atas adalah bentuk *Do While* dimana kondisi akhir perulangan terletak di awal sedangkan bentuk *Do While* yang letak kondisi akhirnya di akhir adalah sebagai berikut :

Do

Pernyataan

.....

Loop While Kondisi

b. *Do Until ... Loop*

Pada dasarnya, perulangan *DO Until* sama dengan perulangan *DO While*. Yang membedakannya adalah pada perulangan *Do While* perulangan akan terus berjalan selama kondisi bernilai *True* dan akan berhenti jika kondisi bernilai *False* sedangkan pada perulangan *Do Until* perulangan akan terus berjalan apabila kondisi bernilai salah dan akan berhenti jika kondisi bernilai *True*. Bentuk sintak dari perulangannya *Do Until* adalah sebagai berikut:

Do Until Kondisi

Pernyataan

.....

Loop

Perulangan *Do Until* juga memiliki bentuk *sintaks* yang lain dimana kondisi diletakkan di akhir pernyataan perulangan. Bentuk sintaksnya adalah sebagai berikut :

Do

Pernyataan

.....

Loop Until Kondisi

c. *While ... Wend*

Perulangan *While ... Wend* digunakan saat kita ingin pernyataan pada program dijalankan selama kondisi ekspresi yang di tetapkan masih bernilai *True* .

Sintaks perulangan *While Wend* adalah :

While Ekspresi

Pernyataan

.....

Wend

d. *For ... Next*

Perulangan *For ... Next* digunakan untuk melakukan perulangan dengan jumlah tertentu yang kita kehendaki. Kita harus mendeklarasikan sebuah variabel yang akan digunakan sebagai indeks untuk perulangan. *Sintaks* perulangan *For ...*

Next adalah :

For Indeks = NilaiAwal To NilaiAkhir

Pernyataan

.....

Next Indeks

Program *Microsoft Visual Basic* merupakan bahasa pemrograman tingkat tinggi (*High Level Language*). *Microsoft Visual Basic* juga merupakan bahasa pemrograman *Object Oriented Programming* (OOP), yaitu pemrograman berorientasi pada objek. *Microsoft Visual Basic* memiliki beberapa versi yaitu *Microsoft Visual Basic 3.0*, *Microsoft Visual Basic 5.0*, *Microsoft Visual Basic Versi 6.0*, VB. Net. Dan mungkin akan berkembang lagi dengan berbagai versi dan semakin sempurna dalam penggunaannya.

Adapun alasan penulis menggunakan pemrograman *Visual Basic Versi 6.0* yaitu antara lain:

- a. Bahasa pemrograman ini berbasis *Windows* sehingga seorang *programmer* dapat membuat penampilan semenarik mungkin.
- b. Program ini sangat *User Friendly*.
- c. Mudah dalam penanganan database serta mudah dalam pembuatan laporan.
- d. Cara penggunaan program ini cukup mudah bagi seorang *programmer* masih pemula.

Karena kemiripannya dengan pemrograman *basic*, bahasa pemrograman *Microsoft Visual Basic* ini menjadi lebih mudah untuk dipahami dan dipelajari. *Microsoft Visual Basic 6.0* ini mempunyai kemampuan yang sangat besar dalam membuat program-program yang lebih kompleks. *Microsoft Visual Basic* terdiri dari beberapa versi dan *Microsoft Visual Basic 6.0* merupakan penyempurnaan dari versi sebelumnya.

