

BAB II

TINJAUAN PUSTAKA

2.1. Mikrokontroler

2.1.1. Pengertian Mikrokontroler

Saat ini perkembangan teknologi semakin pesat berkat adanya teknologi mikrokontroler, sehingga rangkaian kendali atau rangkaian kontrol semakin banyak dibutuhkan untuk mengendalikan berbagai peralatan yang digunakan manusia dalam kehidupan sehari-hari. Dari rangkaian kendali inilah akan terciptanya suatu alat yang dapat mengendalikan sesuatu. Rangkaian kendali atau rangkaian kontrol adalah rangkaian yang dirancang sedemikian rupa sehingga dapat melakukan fungsi–fungsi kontrol tertentu sesuai dengan kebutuhan.

Bermula dari dibuatnya *Integrated Circuit* (IC). Selain IC, alat yang dapat berfungsi sebagai kendali adalah *chip* sama halnya dengan IC. *Chip* merupakan perkembangan dari IC, dimana *chip* berisikan rangkaian elektronika yang dibuat dari artikel *silicon* yang mampu melakukan proses logika. *Chip* berfungsi sebagai media penyimpan program dan data, karena pada sebuah *chip* tersedia RAM dimana data dan program ini digunakan oleh logic *chip* dalam menjalankan prosesnya.

Chip lebih di identikkan dengan dengan kata mikroprosesor. Mikroprosesor adalah bagian dari *Central Processing Unit* (CPU) yang terdapat pada computer tanpa adanya memory, I/O yang dibutuhkan oleh sebuah system yang lengkap. Selain mikroprosesor ada sebuah *chip* lagi yang dikenal dengan nama mikrokomputer. Berbeda dengan mikroprosesor, pada mikrokomputer ini telah tersedia I/O dan memory.

Dengan kemajuan teknologi dan dengan perkembangan *chip* yang pesat sehingga saat ini didalam sekeping *chip* terdapat CPU memory dan control I/O. *Chip* jenis ini sering disebut *microcontroller*.

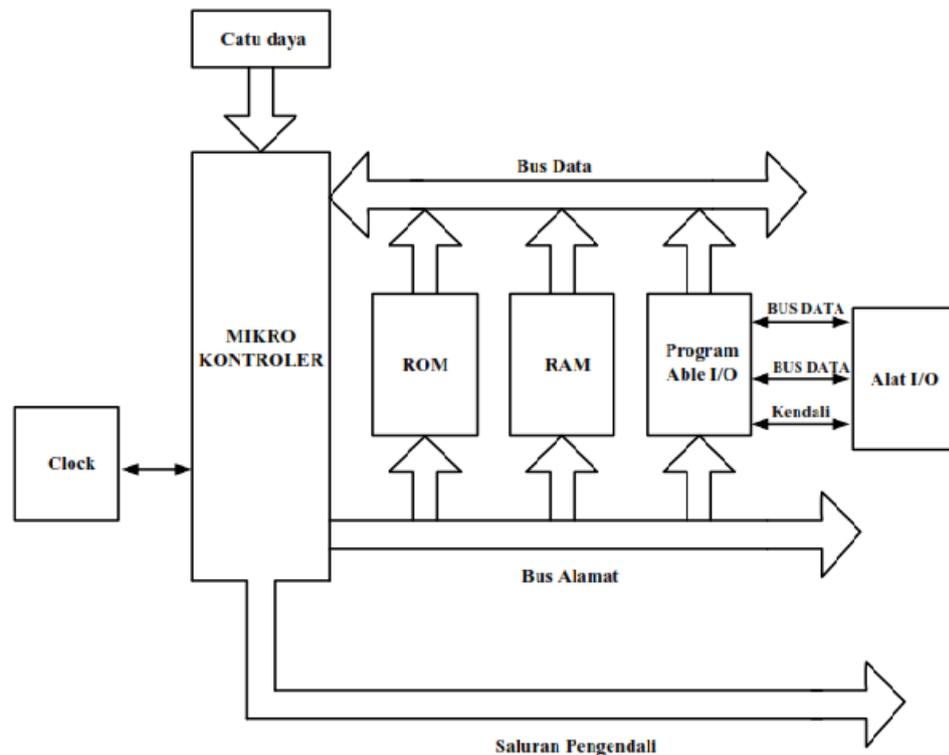
Mikrokontroler merupakan sebuah sistem komputer di mana seluruh atau sebagian besar elemennya dikemas dalam satu chip IC (*Integrated Circuit*), sehingga sering disebut *single chip microcomputer*. Mikrokontroler ini juga

merupakan sebuah sistem komputer yang memiliki satu atau beberapa tugas yang spesifik, berbeda dengan PC yang memiliki beragam fungsi. Perbedaan yang lain adalah perbandingan RAM dan ROM yang sangat besar antara mikrokontroller dengan komputer. Dalam mikrokontroller ROM jauh lebih besar dibanding RAM, sedangkan dalam komputer atau PC RAM jauh lebih besar dibanding ROM.

Mikrokontroller memiliki kemampuan untuk mengolah serta memproses data sekaligus juga dapat digunakan sebagai unit kendali, maka dengan sekeping *chip* yaitu mikrokontroller kita dapat mengendalikan suatu alat. Mikrokontroller mempunyai perbedaan dengan mikroprosesor dan mikrokomputer. Suatu mikroprosesor merupakan bagian dari CPU tanpa memori dan I/O pendukung dari sebuah komputer, sedangkan mikrokontroller umumnya terdiri atas CPU, memory, I/O tertentu dan unit – unit pendukung lainnya.

Pada dasarnya terdapat perbedaan sangat mencolok antara mikrokontroller dan mikroprosesor serta mikrokomputer yaitu pada aplikasinya, karena mikrokontroller hanya dapat digunakan pada aplikasi tertentu saja. Kelebihan lainnya yaitu terletak pada perbandingan *Random Access Memory* (RAM) dan *Read Only Memory* (ROM). Sehingga ukuran *board* mikrokontroller menjadi sangat ringkas atau kecil, dari kelebihan yang ada terdapat keuntungan pemakaian mikrokontroller dengan mikroprosesor yaitu pada mikrokontroller sudah terdapat RAM dan peralatan I/O pendukung sehingga tidak perlu menambahnya lagi. Pada dasarnya struktur dari mikroprosesor memiliki kemiripan dengan mikrokontroller.

Mikrokontroller biasanya dikelompokkan dalam satu keluarga, masing-masing mikrokontroller memiliki spesifikasi tersendiri namun cocok dalam pemrogramannya misalnya keluarga MCS-51 yang diproduksi ATMEL seperti AT89C51, AT89S52 dan lainnya sedangkan keluarga AVR seperti Atmega 8535 dan lain sebagainya.



Gambar 2.1. Blok Diagram Mikrokontroler Secara Umum

(Sumber : Aprianah, 2013)

1. *Central Processing Unit (CPU)*

CPU adalah suatu unit pengolah pusat yang terdiri atas dua bagian, yaitu unit pengendali (*control unit*) dan unit logika (*arithmetic and logic unit*). Disamping itu juga CPU mempunyai beberapa simpanan yang berukuran kecil yang disebut dengan register. Adapun fungsi utama dari unit pengendali ini adalah mengatur dan mengendalikan semua peralatan yang ada pada sistem komputer dan juga dapat mengatur kapan alat input menerima data dan kapan data diolah serta ditampilkan pada alat output.

Sedangkan unit logika berfungsi untuk melakukan semua perhitungan aritmatika yang terjadi sesuai dengan instruksi program dan dapat juga melakukan keputusan dari operasi logika atau pengambilan keputusan sesuai dengan instruksi yang diberikan padanya.

2. Bus Alamat

Bus alamat berfungsi sebagai sejumlah lintasan saluran pengalamatan alamat dengan sebuah computer. Pengalamatan ini harus ditentukan terlebih dahulu untuk menghindari terjadinya kesalahan pengiriman sebuah instruksi dan terjadinya bentrok antar dua buah alamat yang bekerja secara bersamaan.

3. Bus Data

Bus data merupakan sejumlah lintasan saluran keluar masuknya data dalam sebuah mikrokontroller. Pada umumnya saluran data yang masuk sama dengan saluran data yang keluar.

4. Bus Kontrol

Bus kontrol atau bus kendali berfungsi untuk menyamakan operasi mikrokontroler dengan operasi rangkaian luar.

5. Memori

Didalam sebuah mikrokontroler terdapat sebuah memori yang berfungsi untuk menyimpan data atau program. Ada beberapa jenis memori, diantaranya adalah RAM dan ROM serta ada tingkat memori, diantaranya adalah register internal, memori utama dan memori massal. Registrasi internal adalah memori yang terdapat didalam ALU. Memori utama adalah memori yang ada pada suatu system, waktu aksesnya lebih lambat dibandingkan register internal. Sedangkan memori massal dipakai untuk penyimpanan berkapasitas tinggi, yang biasanya berbentuk disket, pita magnetic atau kaset.

6. RAM (*Random Access Momory*)

RAM adalah memori yang dapat dibaca atau ditulis. Data dalam RAM bersifat volatile dimana isinya akan hilang begitu IC kehilangan catu daya, karena sifat yang demikian RAM hanya digunakan untuk menyimpan data pada saat program bekerja.

7. ROM (*Read Only Memory*)

ROM merupakan memory yang hanya dapat dibaca, dimana isinya tidak dapat berubah apabila IC telah kehilangan catu daya. ROM dipakai untuk menyimpan program, pada saat di reset maka mikrokontroler akan langsung bekerja dengan program yang terdapat didalam ROM tersebut. Ada beberapa jenis ROM antara lain ROM murni, PROM (*Programable Read Only Memory*), EPROM (*Erasable Programmable Only Memory*), yang paling banyak digunakan diantara tipe-tipe diatas adalah EPROM yang dapat diprogram ulang dan dapat juga dihapus dengan sinar ultraviolet.

8. *Input / Output*

Setiap system computer memerlukan sistem *input* dan *output* yang merupakan media keluar masuk data dari dan ke komputer. Contoh peralatan I/O yang umum yang terhubung dengan sebuah komputer seperti *keyboard*, *mouse*, *monitor*, sensor, *printer*, LED, dan lain-lain.

9. *Clock*

Clock atau pewaktuan berfungsi memberikan referensi waktu dan sinkronisasi antar elemen.

2.1.2. Sistem Mikrokontroler

Mikroprosesor dan mikrokontroler berasal dari ide dasar yang sama. Mikroprosesor adalah istilah yang merujuk pada *central processing unit* (CPU) computer digital untuk tujuan umum. Untuk membuat sistem computer, CPU harus ditambahkan memori, umumnya *read only memory* (ROM) dan *random access memory* (RAM), dekoder memori, osilator dan sejumlah *input/output device* seperti port data parallel dan serial. Gambar 2.1 menunjukkan sebuah diagram blok sistem mikroprosesor tujuan umum yang terdiri atas *central processing unit* (CPU), RAM, ROM, *I/O port*, *timer*, dan *port serial* COM. Tambahan lain, *special-purpose device*, seperti *interrupt handler* dan *counter*. Penambahan seperti *mass storage*, *hard drive*, *I/O peripheral* seperti *keyboard*

dan *display* (CRT/LCD) menghasilkan sebuah computer yang dapat digunakan untuk aplikasi-aplikasi *general-purpose software*. (Utari, 2013)

Mikrokontroller umumnya dikelompokkan dalam satu keluarga besar, contoh-contoh keluarga mikrokontroller :

1. Keluarga MCS-51
2. Keluarga MC68HC05
3. Keluarga MC68HC11
4. Keluarga AVR
5. Keluarga PIC8

2.2. Mikrokontroller ATMega 16

2.2.1. Pengertian Mikrokontroller ATMega 16

Mikrokontroller sering juga disebut sebagai mikrokomputer atau *embedded system*. Mikrokontroller dapat dipandang sebagai suatu sistem yang terdiri atas *input*, program dan *output*. Perancang dapat mengatur perilaku mikrokontroller melalui program. Proses memasukkan program ke dalam mikrokontroller disebut proses *download* dan alat yang digunakan disebut *downloader*. Seperti sistem komputer, nilai tambah sistem mikrokontroller dapat dilipatgandakan melalui program. (Sulistiyanto, 2008: 1)

Mikrokontroller AVR (*Alf and Vegard's Risc processor*) merupakan pengontrolan utama standar industri dan riset saat ini. Hal ini dikarenakan berbagai kelebihan yang dimilikinya dibandingkan mikroprosesor antara lain lebih murah, dukungan software dan dokumentasi yang memadai dan memerlukan komponen pendukung yang sangat sedikit. Salah satu tipe mikrokontroller AVR untuk aplikasi standar yang memiliki fitur memuaskan ialah ATmega16. (Budiharto, 2008:1)

Mikrokontroller AVR standar memiliki arsitektur 8 bit, dimana semua instruksi dikemas dalam kode 16 bit dan sebagian besar instruksi dieksekusi dalam satu siklus *clock*. Berbeda dengan instruksi MCS51 yang membutuhkan 12 siklus *clock*, karena kedua jenis mikrokontroller tersebut memiliki arsitektur yang berbeda. (Budiharto & Jefri, 2007:52)

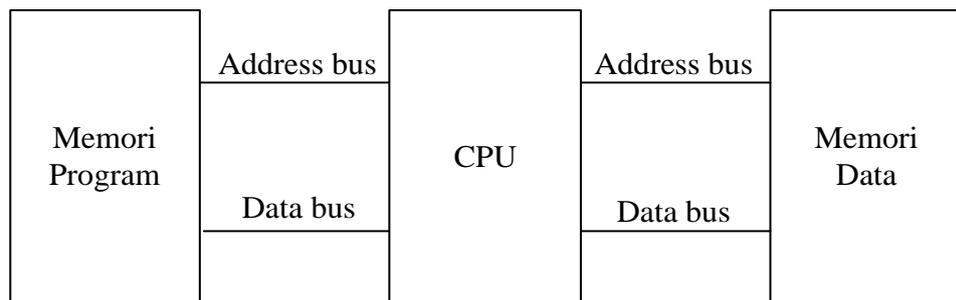


Gambar 2.2. ATMega 16

(Sumber : Andrianto, 2013)

2.2.2. Arsitektur ATMega 16

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



Gambar 2.3. Arsitektur Harvard

(Sumber : Andrianto, 2013)

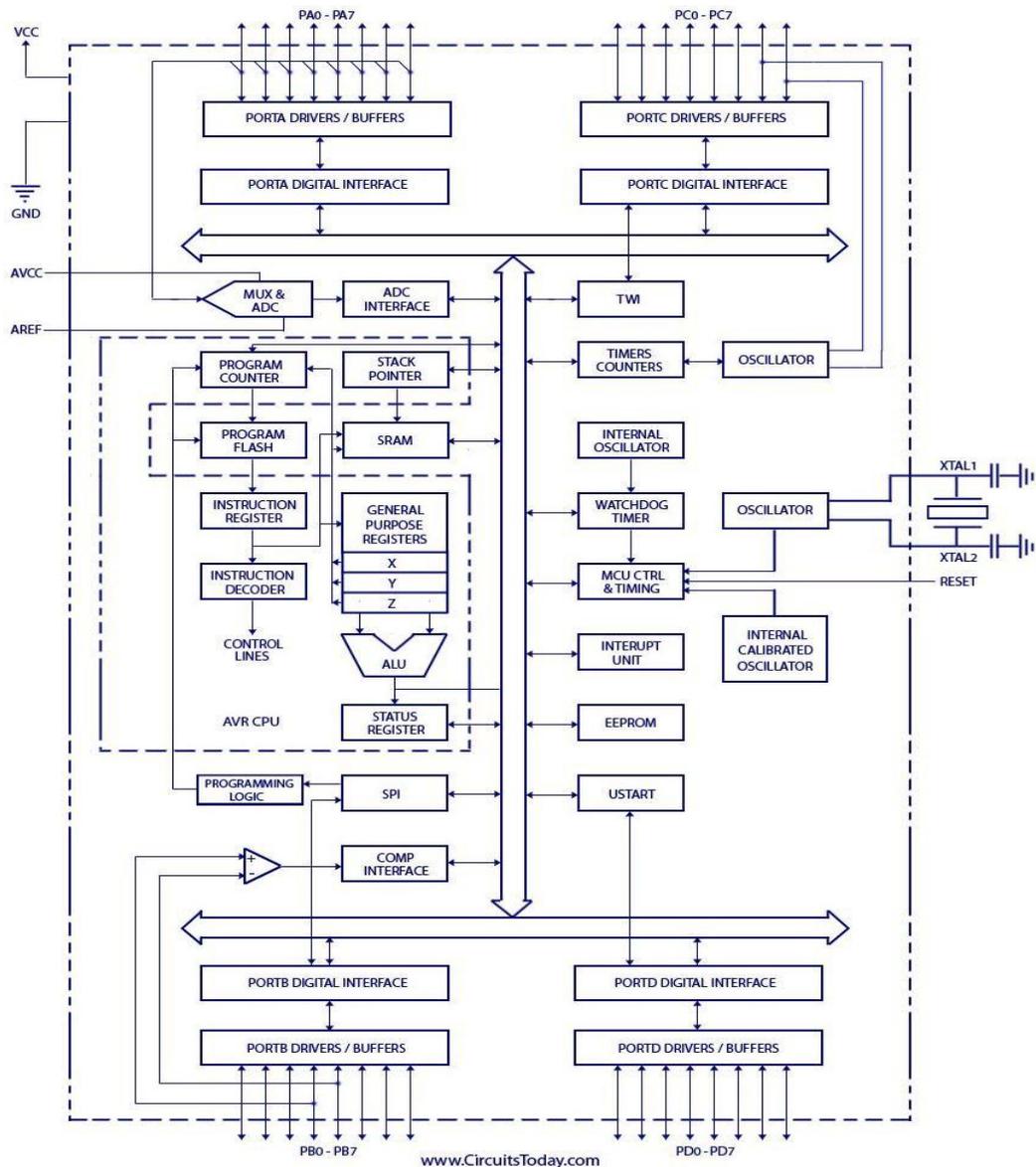
Dari gambar diatas, AVR menggunakan arsitektur Harvard dengan memisahkan antara memori dan *bus* untuk program dan data untuk memaksimalkan kemampuan dan kecepatan. Instruksi dalam memori program dieksekusi dengan *pipelining singel level*. Dimana ketika satu instruksi dieksekusi, instruksi berikutnya diambil dari memori program. Konsep ini mengakibatkan instruksi

dieksekusi setiap *clock cycle*. CPU terdiri dari 32x8 – bit general purpose register yang dapat diakses dengan cepat dalam satu *clock cycle*, yang mengakibatkan operasi *Arithmetic Logic Unit (ALU)* dapat di lakukan dalam satu *cycle*. Pada operasi ALU, dua *operand* berasal dari *register*, kemudian operasi dieksekusi dan hasilnya disimpan kembali ke *register* dalam satu *clock cycle*. Operasi aritmatika logic pada ALU akan mengubah bit – bit yang terdapat pada Status *Register (SREG)*. Proses pengambilan instruksi dan pengekseskuan instruksi berjalan secara *parallel*. (Andrianto, 2013:14)

Menurut (Andrianto, 2013:8) Fitur-fitur yang dimiliki ATmega 16 sebagai berikut:

1. Mikrokontroler AVR 8 Bit yang memiliki kemampuan tinggi, dengan daya rendah.
2. Memiliki kapasitas Flash memori 16 KByte, EEPROM 512 Byte dan SRAM 1 Kbyte.
3. Saluran I/O sebanyak 32 buah, yaitu Port A, Port B, Port C dan Port D.
4. CPU terdiri atas 32 buah *register*.
5. Unit Interupsi *internal* dan *eksternal*.
6. ADC internal dengan fidelitas 10 bit 8 channel.
7. Sistem mikroprosesor 8 bit berbasis RISC dengan kecepatan maksimal 16 MHz.
8. Port USART untuk komunikasi serial.

Dengan fitur-fitur seperti diatas, pembuatan alat menggunakan ATmega16 menjadi lebih sederhana dan tidak memerlukan IC pendukung yang banyak. Agar lebih jelasnya dapat dilihat pada gambar di bawah.



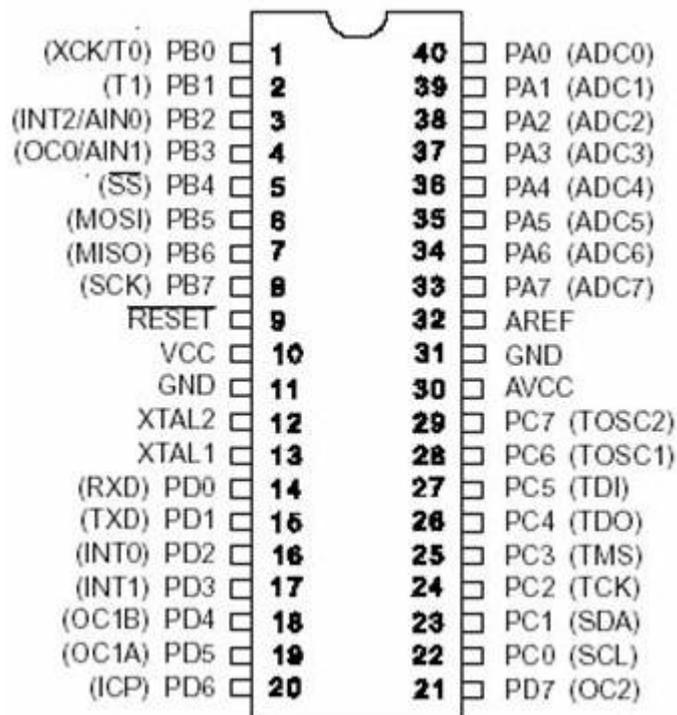
Gambar 2.4. Blok Diagram ATMEGA16

(Sumber : Setiawan, 2010)

2.2.3. Konfigurasi Pin ATMEGA 16

Konfigurasi pin mikrokontroler Atmega16 dengan kemasan 40 pin dapat dilihat pada gambar di bawah. ATmega16 memiliki 32 pin yang digunakan untuk input/output, pin-pin tersebut terdiri dari 8 pin sebagai port A, 8 pin sebagai port B, 8 pin sebagai port C, 8 pin sebagai port D. Dalam komunikasi serial, maka hanya port D yang dapat digunakan karena fungsi khusus yang dimilikinya. Untuk lebih

kelas akan ditunjukkan pada tabel-tabel fungsi khusus port. Susunan pin Mikrokontroler ATmega16 diperlihatkan pada gambar dibawah ini.



Gambar 2.5. Susunan kaki Mikrokontroler ATmega16

(Sumber : Andrianto, 2013)

Berikut ini adalah penjelasan umum susunan kaki dari ATmega16:

1. VCC merupakan pin masukan positif catu daya. Setiap peranti elektronika digital membutuhkan sumber daya yang umumnya sebesar 5V. Oleh karena itu, biasanya di PCB kit mikrokontroler selau ada IC regulator 7805,
2. GND sebagai pin Ground,
3. Port A (PA0...PA7),

Merupakan pin I/O dua arah dan dapat diprogram sebagai pin masukan ADC. 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. Pin-pin 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 pin PA0 ke PA7 digunakan sebagai input dan secara eksternal ditarik rendah, pin-pin akan memungkinkan arus sumber jika resistor *internal pull-up* diaktifkan. Pin port A adalah tri-stated manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

4. Port B (PB0...PB7),

Merupakan pin I/O dua arah dan pin fungsi khusus, yaitu Timer/Counter, komparator analog dan SPI. Port B adalah suatu port 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, pin port B yang secara eksternal ditarik rendah akan arus sumber jika resistor *pull-up* diaktifkan. Pin port B adalah tri-stated manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

Tabel 2.1. Fungsi Khusus Port B Atmega 16

Port Pin	Alternate Functions
PB 7	SCK: SPI serial waktu
PB 6	MISO: SPI master input / slave output
PB 5	MOSI: SPI master output / slave input
PB 4	SS: SPI slave select input
PB 3	AIN1: pembanding analog, input negative OC0: Timer / counter 0 output (pembanding output)
PB 2	AIN0: Pembanding analog, input positive INT2: External interrupt 2 input
PB 1	T1: Timer / counter1 external counter input
PB 0	T0: Timer/Counter 0 external counter input XCK: USART waktu eksternal input / output)

(Sumber : Andrianto, 2013)

5. Port C (PC0...PC7),

Merupakan pin I/O dua arah dan pin fungsi khusus, yaitu komparator TWI, komparator analog dan timer osilator. Port C adalah suatu port I/O 8-bit dua

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

Tabel 2.2. Fungsi Khusus Port C Atmega 16

Port Pin	Alternate Functions
PC 7	TOSC2: Waktu oscillator Pin2
PC 6	TOSC1: Waktu oscillator Pin1
PC 5	TDI: JTAG test data input
PC 4	TDO: JTAG test data output
PC 3	TMS: JTAG test mode select
PC 2	TCK: JTAG test clock
PC 1	SDA: Dua penghubung serial data input / output
PC 0	SCL: Dua penghubung serial waktu

(Sumber : Andrianto, 2013)

6. Port D (PD0...PD7),

Merupakan pin I/O dua arah dan pin fungsi khusus, yaitu komparator analog, interupsi eksternal dan komunikasi serial. 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, pin port D yang secara eksternal ditarik rendah akan arus sumber jika resistor *pull-up* diaktifkan. Pin port D adalah tri-stated manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

Tabel 2.3. Fungsi Khusus Port D Atmega 16

Port Pin	Alternate Functions
PD 7	OC2: Timer / counter 2 output (hasil output)
PD 6	ICP1: Timer / counter 1 input Pin
PD 5	OC1A: Timer / counter 1 hasil output A (hasil output A)
PD 4	OC1B: Timer / counter 1 hasil output B (hasil Output B)
PD 3	INT1: External interrupt 1 input
PD 2	INT0: External interrupt 0 input
PD 1	TXD: USART output Pin
PD 0	RXD: USART input Pin

(Sumber : Andrianto, 2013)

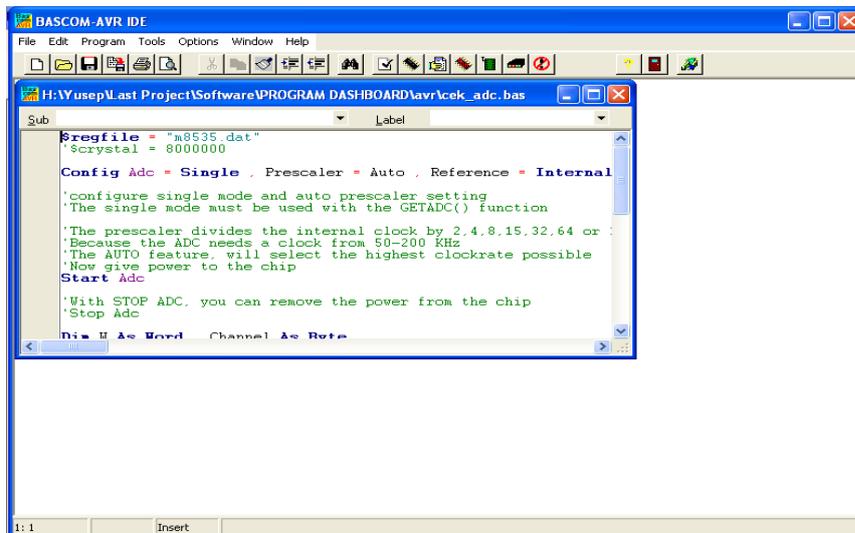
7. Reset merupakan pin yang digunakan untuk me-reset mikrokontroler,
8. XTAL 1 dan XTAL 2 sebagai pin masukan clock eksternal. Suatu mikrokontroler membutuhkan sumber detak (clock) agar dapat mengeksekusi instruksi yang ada di memori. Semakin tinggi nilai kristalnya maka semakin cepat mikrokontroler tersebut,
9. AVCC sebagai pin masukan tegangan untuk ADC,
10. AREF sebagai pin masukan tegangan referensi.

2.3. Software (BASCOS AVR)

2.3.1. Pengertian BASCOM AVR

Bascom AVR atau yang biasa disebut *basic compiler* adalah suatu piranti lunak yang termasuk bahasa tingkat tinggi yang sangat mudah untuk dipelajari. Sebagai *compiler*, yaitu perubah instruksi dari bahasa basic ke file yang berbentuk hexa dengan tujuan dimengerti oleh mesin atau mikrokontroler, sehingga mikrokontroler mampu menerjemahkan instruksi-instruksi yang kita buat dengan benar dan tepat.

Interface dari BASCOM AVR dapat dilihat pada gambar dibawah ini :



Gambar 2.6. Interface BASCOM AVR

(Sumber : Aprianah, 2013)

Tabel 2.4. Keterangan lengkap ikon-ikon dari program BASCOM

Icon	Nama	Fungsi	Shortcut
	<i>File New</i>	Membuat file baru	Ctrl+N
	<i>Open File</i>	Untuk Membuka File	Ctrl+N
	<i>File Close</i>	Untuk Menutup proram yang dibuka	Ctrl+O
	<i>File Save</i>	Untuk menyimpan file	Ctrl+S
	<i>Save as</i>	Menyimpan dengan nama yang lain	-
	<i>Print preview</i>	Untuk melihat tampilan sebelum dicetak	-
	<i>Print</i>	Untuk mencetak dokumen	Ctrl+P
	<i>Exit</i>	Untuk Keluar dari program	-
	<i>Program compile</i>	Untuk mengkompile program yang dibuat, Outputnya bisa berupa *.hex, *.bin dll	F7
	<i>Syntax check</i>	Untuk memeriksa kesalahan bahasa	Ctrl+F7
	<i>Show result</i>	Untuk menampilkan hasil kompilasi program	Ctrl+W

(Sumber : Aprianah, 2013)

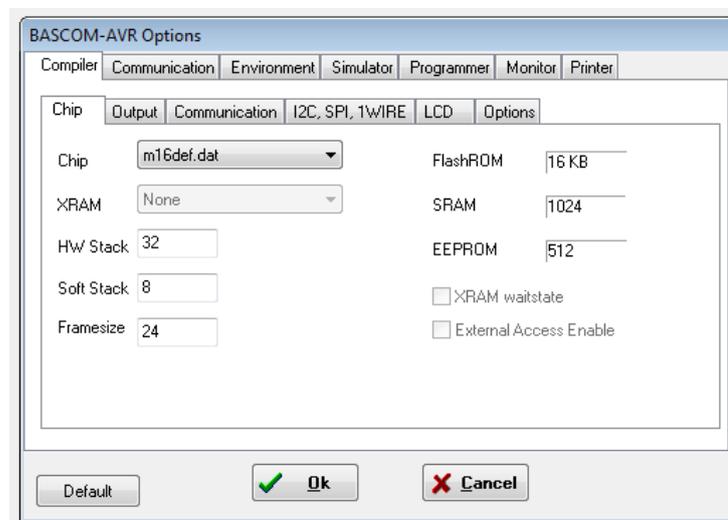
Tabel 2.5. Keterangan menu *show result*

Info	Keterangan
Compiler	Versi dari compiler yang digunakan
Processor	Menampilkan target prosesor yang dipilih
Date and time	Tanggal dan waktu kompilasi
Baud rate dan xtal	Baudrate yang dipilih dan kristal yang digunakan uP.
Error	Error nilai Baud yang di set dengan nilai baud sebenarnya
Flash Used	Persentase flash ROM yang terisi program
Stack Start	Lokasi awal stack pointer memori
RAM Start	Lokasi awal eksternal RAM.
LCD Mode	Mode LCD yang digunakan, 4 bit atau 8 bit

(Sumber : Aprianah, 2013)

2.3.2. *Compiler*

BASCOM-AVR menyediakan pilihan untuk memodifikasi pilihan-pilihan pada kompilasi. Dengan memilih menu *Compiler* maka jendela berikut akan ditampilkan :



Gambar 2.7. Jendela *Option*

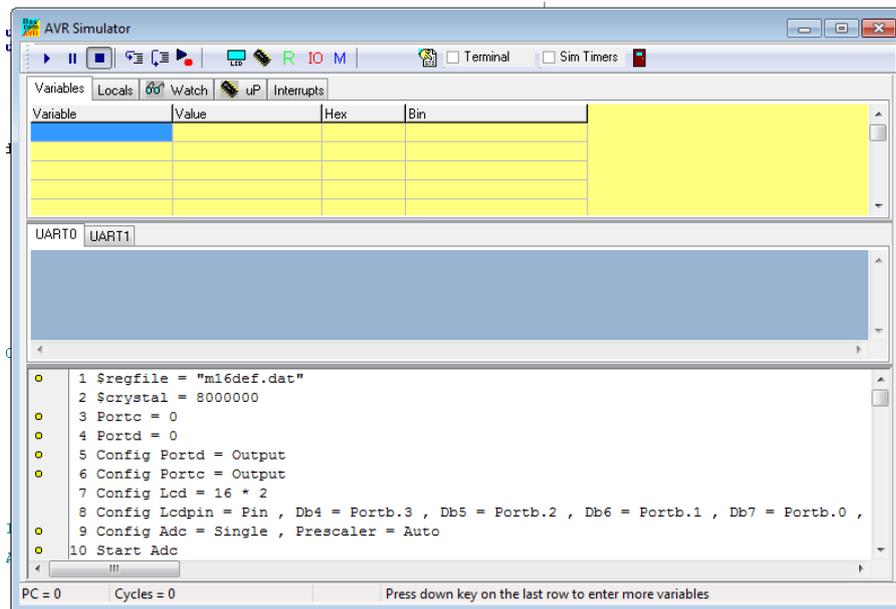
(Sumber : *Print Screen Program*)

Tabel 2.6. Keterangan dari Tab Menu Chip

TAB Menu	OPTION	Keterangan
Chip	Chip	Mikrokontroler yang digunakan, sebagai contoh m8535.dat untuk ATMEGA8535
	XRAM	Jika menggunakan ekstrenal RAM nilai ini bisa ditampilkan
	HW Stack	Stack memory hardware, setiap Gosub membutuhkan 2 byte. Jika menggunakan interupsi, naikan nilainya
	Soft stack	Stack software, nilai defaultnya 8
	FlashROM	Nilai flashROM Chip yang dipilih
	SRAM	Nilai RAM internal Chip yang dipilih
	EEPROM	Nilai EEPROM chip yang terpilih
Output		File Output yang akan dihasilkan dalam proses kompilasi
Communication	Baudrate 0	Nilai Baudrate yang digunakan dalam komunikasi serial
	Frekuensi	Nilai osilator yang digunakan
	Error	Error antara baudrate yang dipilih dengan nilai sebenarnya, hal ini tergantung pada osilator yang dipilih
I2C, SPI, 1 wire	SDA	Pin yang berfungsi untuk data serial dalam komunikasi I2C
	SCL	Pin yang berfungsi untuk data clock dalam komunikasi I2C
	1Wire	Pin yang digunakan untuk komunikasi 1 wire
	SPI	Pin yang digunakan untuk komunikasi serial sinkron
LCD		Pemilihan Port yang digunakan untuk tampilan LCD, jenis LCD

(Sumber : Aprianah, 2013)

BASCOM-AVR menyediakan pilihan yang dapat mensimulasikan program. Agar dapat menjalankan simulator ini, file DBG dan OBJ harus dipilih pada menu *Options Compiler Output*. Tampilan program simulasi adalah sebagai berikut:



Gambar 2.8. Interface Simulator BASCOM AVR

(Sumber : *Print Screen Program*)

Tekan tombol  untuk memulai simulasi. Dan untuk memberhentikan simulasi atau menahan proses simulasi gunakan tombol disebelahnya. Layar biru ditengah merupakan simulasi layar monitor ketika menggunakan komunikasi serial. Untuk dapat mengamati perubahan-perubahan nilai register atau variabel selama program berjalan, simulator ini menyediakan beberapa jendela, antara lain :

1. Variabel

Jendela ini berisi tabel yang berfungsi untuk mengamati nilai variabel-variabel yang digunakan dalam program yang sedang disimulasikan. Untuk menambahkan variabel klik ganda pada kolom variabel maka daftar variabel akan ditampilkan, klik variabel yang ingin diamati.

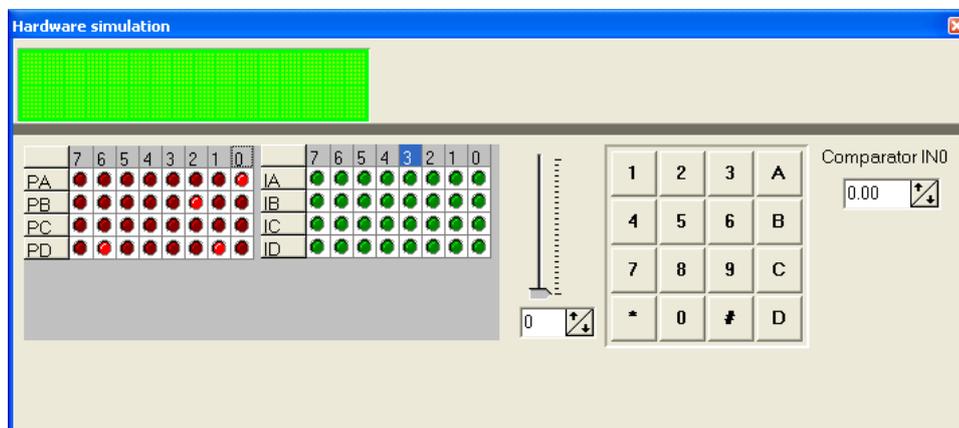
Variable	Value	Hex	Bin
B	0	0	00000000
idx	0	0	00000000

Gambar 2.9. Jendela Variabel

(Sumber : Aprianah, 2013)

2. Simulasi Hardware

Selain itu Untuk dapat melihat perubahan data pada setiap port atau ketika kita ingin memberikan input pada pin-pin tertentu dari mikrokontroller, maka gunakan tombol  untuk menampilkan jendela sebagai berikut:



Gambar 2.10. Jendela Simulasi *Hardware*

(Sumber : *Print Screen Program*)

2.3.3. Dasar Pemrograman BASCOM

2.3.3.1. Tipe Data

Setiap variabel dalam BASCOM memiliki tipe data yang menunjukkan daya tampung variabel tersebut, hal ini berhubungan dengan penggunaan memori dari mikrokontroler. Berikut ini adalah tipe data pada BASCOM berikut keterangannya.

Tabel 2.7. Tipe Data BASCOM

Tipe Data	Kapasitas (byte)	Jangkauan Nilai
Bit	1/8	0 dan 1
Byte	1	0-255
Integer	2	-32.768 – +32.767
Word	2	0 – 65.535
Long	4	-2.147.483.648 – +2.147.483.647
Single	4	$1,5 \times 10^{-45} - 3,4 \times 10^{38}$
Double	8	$5 \times 10^{-324} - 1,7 \times 10^{308}$
String	254	-

(Sumber : Setiawan, 2010)

2.3.3.2. Variabel

Variabel dalam sebuah pemrograman berfungsi sebagai tempat penyimpanan data atau penampung data sementara, misalnya menampung hasil perhitungan, menampung data hasil pembacaan register dan lain lain. Variabel merupakan pointer yang menunjuk pada alamat memori fisik di mikrokontroler.

Dalam BASCOM ada beberapa aturan dalam penamaan sebuah variabel:

1. Nama variabel maksimum terdiri atas 32 karakter
2. Karakter bias berupa angka atau huruf
3. Nama variabel harus dimulai dengan huruf
4. variabel tidak boleh menggunakan kata-kata yang digunakan oleh BASCOM sebagai perintah, pernyataan, internal register dan nama operator (AND,OR, DIM dan lain-lain).

Sebelum variabel digunakan maka variabel tersebut harus dideklarasikan terlebih dahulu, dalam BASCOM ada beberapa cara untuk mendeklarasikan sebuah variabel. Yang pertama dengan menggunakan pernyataan “**DIM**” diikuti nama dan tipe datanya, contoh pendeklarasian menggunakan DIM sebagai berikut:

```
Dim nama as byte
Dim tombol1 as integer
Dim tombol2 as word
Dim tombol3 as word
```

2.3.3.3. Alias

Dengan menggunakan **ALIAS** sebuah variabel yang sama dapat diberikan nama yang lain, tujuannya untuk mempermudah proses pemrograman. Biasanya ALIAS digunakan untuk mengganti nama variabel yang telah baku seperti port mikrokontroler.

```
LEDBAR alias portA
Tombol1 alias portA.1
```

Dengan deklarasi seperti di atas maka perubahan pada tombol1 akan merubah kondisi dari P0.1. Selain mengganti nama port **ALIAS** juga dapat digunakan untuk mengakses bit tertentu dari sebuah variabel yang telah dideklarasikan :

```
Dim LedBar as byte
Led1 as LedBar.0
Led2 as LedBar.1
```

2.3.4. Kontrol Program

Pada BASCOM, Kontrol program dikelompokkan kedalam tiga kategori yaitu berdasarkan perulangan (Do..Loop, For.. Next), kondisi/percabangan (IF.. Then .., Select- Case) dan Lompatan (Gosub, Goto). Berikut akan dijelaskan satu persatu :

2.3.4.1.DO...LOOP

Perintah **Do..loop** digunakan untuk mengulangi sebuah blok pernyataan terus menerus. Untuk membatasi perulangannya dapat ditambahkan sebuah syarat kondisi agar perulangan berhenti dan perintahnya menjadi **Do..loop until**. Sintaknya sebagai berikut:

```
Do
  <blok pernyataan>
```

Yang menggunakan perintah **Do..Loop Until**

```
Do
  <blok pernyataan>
loop Until <syarat kondisi>
```

2.3.4.2.FOR...NEXT

Perintah ini digunakan untuk mengeksekusi sebuah blok pernyataan secara berulang. Perintah ini hampir sama dengan perintah **Do..Loop** namun pada perintah **For..Next** ini nilai awal dan akhir perulangan serta tingkat kenaikan atau turunnya bisa ditentukan. Penggunaannya sebagai berikut:

```
FOR var = start TO/DOWNTO end [STEP value]
  <Blok pernyataan>
```

Untuk menaikkan nilai perulangan gunakan **To** dan untuk menurunkan gunakan **Downto**. Tingkat kenaikan merupakan pilihan, jadi bisa digunakan ataupun tidak. Jika nilai kenaikan tidak ditentukan maka secara otomatis BASCOM akan menentukan nilainya yaitu 1.

2.3.4.3. IF... THEN

Dengan pernyataan ini kita dapat mengetes sebuah kondisi tertentu dan kemudian menentukan tindakan yang sesuai dengan kondisi yang diinginkan. Sintak penulisannya adalah sebagai berikut:

IF <syarat Kondisi> **THEN** <Pernyataaan>

Sintak diatas digunakan jika hanya ada satu kondisi yang diuji dan hanya melakukan satu tindakan. Jika melakukan lebih dari satu tindakan maka sintaknya adalah:

```
IF <syarat Kondisi> THEN
    <Pernyataan ke-1>
    <Pernyataan ke-2>
    .
```

Pada sintak di atas BASCOM akan menguji kondisi 1, jika kondisi 1 tidak terpenuhi maka BASCOM akan menguji kondisi 2, begitu seterusnya hingga BASCOM menemukan satu kondisi yang memenuhi. Jika dari semua kondisi itu tidak ada yang memenuhi maka BASCOM akan mengeksekusi blok pernyataan ELSE.

2.4. LCD (*Liquid Crystal Display*)

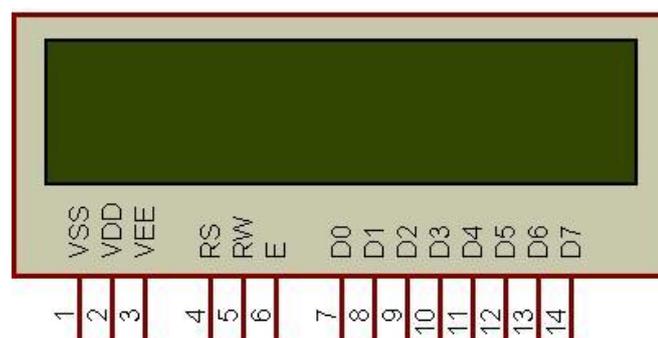
Menurut (Andrianto, 2013:77) *Liquid Crystal Display* (LCD) adalah suatu *display* dari bahan cairan kristal yang pengoperasiannya menggunakan sistem *dot* matriks. LCD banyak digunakan sebagai *display* dari alat-alat elektronika seperti kalkulator, *multitester digital*, jam *digital* dan sebagainya.

LCD dapat dengan mudah dihubungkan dengan mikrokontroler AVR ATmega 16. LCD yang digunakan dalam proyek akhir ini adalah LCD 16 x 2, lebar *display* 2 baris 16 kolom, yang memiliki 16 *pin* konektor.

Menurut (Setiawan, 2010:24) LCD merupakan salah satu perangkat penampil yang sekarang ini mulai banyak digunakan. Penampilan LCD mulai dirasakan menggantikan fungsi dari penampil CRT (*Cathode Ray Tube*), yang

sudah berpuluh – puluh tahun digunakan manusia sebagai penampil gambar / text baik monokrom (hitam dan putih), maupun yang berwarna. Teknologi LCD memberikan lebih keuntungan dibandingkan dengan teknologi CRT, karena pada dasarnya, CRT adalah tabung triode yang digunakan sebelum transistor ditemukan. Beberapa keuntungan LCD dibandingkan CRT adalah konsumsi daya relatif kecil, lebih ringan, tampilan yang lebih bagus, dan (menurut penulis) ketika berlama – lama didepan monitor CRT lebih cepat memberikan kejenuhan pada mata dibandingkan dengan LCD .

LCD berfungsi menampilkan suatu hasil sensor, menampilkan teks, atau menampilkan menu pada aplikasi mikrokontroler. LCD yang digunakan adalah jenis LCD M1632. LCD M1632 merupakan modul LCD dengan tampilan 16 x 2 baris dengan konsumsi daya rendah. Modul tersebut dilengkapi dengan mikrokontroler yang didesain khusus untuk mengendalikan LCD. (Tempuran, 2009).



Gambar 2.11. LCD (*Liquid Crystal Display*) 16 x 2

(Sumber : Setiawan, 2010)

Keunggulan LCD adalah hanya menarik arus yang kecil (beberapa mikro ampere), sehingga alat atau sistem menjadi portable karena dapat menggunakan catu daya yang kecil. Keunggulan lainnya adalah tampilan yang diperlihatkan dapat dibaca dengan mudah dibawah terang sinar matahari. (Setiawan, 2010:24)

2.5. Sensor PIR (*Passive Infra Red*)

Sensor PIR (*Passive Infra Red*) adalah sensor yang digunakan untuk mendeteksi adanya pancaran sinar infra merah. Sensor PIR bersifat pasif, artinya sensor ini tidak memancarkan sinar infra merah tetapi hanya menerima radiasi sinar infra merah dari luar.

Sensor ini biasanya digunakan dalam perancangan detektor gerakan berbasis PIR. Karena semua benda memancarkan energi radiasi, sebuah gerakan akan terdeteksi ketika sumber infra merah dengan suhu tertentu (misal: manusia) melewati sumber infra merah yang lain dengan suhu yang berbeda (misal: dinding), maka sensor akan membandingkan pancaran infra merah yang diterima setiap satuan waktu, sehingga jika ada pergerakan maka akan terjadi perubahan pembacaan pada sensor. Pada umumnya sensor PIR memiliki jangkauan pembacaan efektif hingga 5 meter, dan sensor ini sangat efektif digunakan sebagai *human detector*.

Ketika manusia berada di depan sensor PIR dengan kondisi diam, maka sensor PIR akan menghitung panjang gelombang yang dihasilkan oleh tubuh manusia tersebut. Panjang gelombang yang konstan ini menyebabkan energi panas yang dihasilkan dapat digambarkan hampir sama pada kondisi lingkungan disekitarnya. Ketika manusia itu melakukan gerakan, maka tubuh manusia itu akan menghasilkan pancaran sinar inframerah pasif dengan panjang gelombang yang bervariasi sehingga menghasilkan panas berbeda yang menyebabkan sensor merespon dengan cara menghasilkan arus pada material *Pyroelectricnya* dengan besaran yang berbeda beda. Karena besaran yang berbeda inilah *comparator* menghasilkan output.

Jadi sensor PIR tidak akan menghasilkan output apabila sensor ini dihadapkan dengan benda panas yang tidak memiliki panjang gelombang inframerah antar 8 sampai 14 mikrometer dan benda yang diam seperti sinar lampu yang sangat terang yang mampu menghasilkan panas, pantulan objek benda dari cermin dan suhu panas ketika musim panas.

2.6. Sensor LDR (*Light Dependent Resistor*)

LDR atau *Light Dependent Resistor* adalah salah satu jenis resistor yang nilai hambatannya dipengaruhi oleh cahaya yang diterima olehnya. Besarnya nilai hambatan pada LDR tergantung pada besar kecilnya cahaya yang diterima oleh LDR itu sendiri. Contoh penggunaannya adalah pada lampu taman dan lampu di jalan yang bisa menyala di malam hari dan padam di siang hari secara otomatis. Atau bisa juga kita gunakan di kamar kita sendiri. (Supatmi, 2011).

LDR adalah satu komponen elektronika yang memiliki hambatan yang bisa berubah sesuai dengan perubahan intensitas cahaya. Hambatan dari LDR akan berkurang seiring semakin besarnya intensitas cahaya yang mengenai permukaannya. Besarnya hambatan LDR itu berbeda dan tergantung sesuai dengan bentuk dan ukurannya. Semakin rapat pola garis di permukaan kumparan, biasanya perubahan hambatannya akan semakin besar dan bisa bersifat lebih sensitive terhadap cahaya. Dalam sensor cahaya LDR dipasang seri dengan Variabel Resistor atau (VR) yang berukuran 20k.