

## **BAB II**

### **TINJAUAN PUSTAKA**

Pada pembahasan di bab 2 (dua) ini akan membahas komponen-komponen berupa *input*, proses dan *ouput*. Dimana *inputnya* adalah baterai aki (*accu*) 12V dan baterai, prosesnya adalah *joystick*, mikrokontroler dan *driver relay* serta *output*-nya adalah motor. Berikut adalah penjelasan dari beberapa komponen dibawah ini :

#### **2.1 Power Supply (DC)**

*Power Supply (DC/arus searah)* adalah arus listrik yang mengalir pada suatu hantaran yang tegangannya berpotensi tetap, tidak berubah-ubah. Listrik DC adalah listrik yang asli, artinya listrik yang dapat dihasilkan dari sumber-sumber susunan material alam. Muatan-muatan listrik yang terjadi akibat adanya gesekan pada dua jenis material adalah muatan listrik yang berbentuk DC. ([www.sandielektronik.com/pengertian-dc.html](http://www.sandielektronik.com/pengertian-dc.html),2013)

##### **2.1.1 Sumber Power Supply (DC)**

DC adalah listrik alami, dimana sumber kelistrikan DC yang telah dibuat :

1. Aki dan segala jenis baterai,
2. *Spoel* generator pada kendaraan mobil,
3. Solar *cell*,
4. Generator serta adaptor

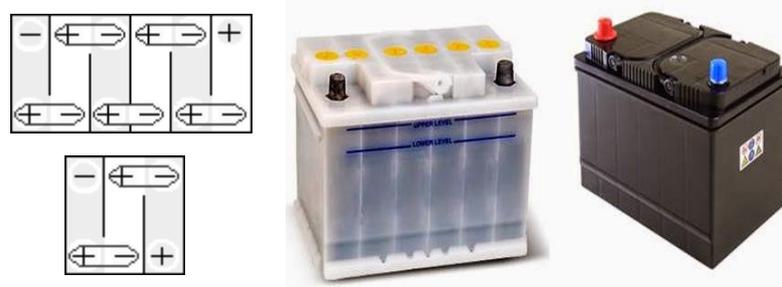
#### **2.2 Aki (Accu)**

Aki atau *accu* adalah suatu proses kimia listrik, dimana pada saat pengisian, cas, *charge* energi listrik diubah menjadi kimia dan saat pengeluaran (*discharge*) energi kimia diubah menjadi energi listrik. (UNIKOM Bandung,2012,4-5)

Aki terdiri dari sel-sel dimana tiap sel memiliki tegangan sebesar 2V, artinya aki mobil dan aki motor yang memiliki tegangan 12V terdiri dari 6 sel



yang dipasang secara seri ( $12V = 6 \times 2V$ ) sedangkan aki yang memiliki tegangan 6V memiliki 3 sel yang dipasang secara seri ( $6V = 3 \times 2V$ ).



**Gambar 2.1** Simulasi dalam Aki dan Contoh Aki

(a) Aki 12Volt dan Aki 6Volt, (b) Aki Basah dan Kering

(<http://www.wartasaranamedia.com/2014/06/bagian-komponen-utama-aki-kering-dan-aki-basah-197.html>)

Antara satu sel dengan sel lainnya dipisahkan oleh dinding penyekat yang terdapat dalam bak aki, artinya tiap ruang pada sel tidak berhubungan karena itu cairan elektrolit pada tiap sel juga tidak berhubungan (dinding pemisah antar sel tidak boleh ada yang bocor atau merembes).

Sebuah aki kering 12V dan 18Ah mencantumkan nilai spesifikasi sebagai berikut :

- a.  $20\text{hr} \times 0,9A = 18A$
- b.  $5\text{hr} \times 3,06A = 15,3A$
- c.  $1\text{hr} \times 10,8A = 10,8A$
- d.  $1/2\text{hr} \times 18A = 9A$

Jika dilihat dari spesifikasi maka aki ini memiliki kapasitas efektif sebesar 18Ah namun suplai dari aki sebenarnya hanya bisa dilakukan selama :

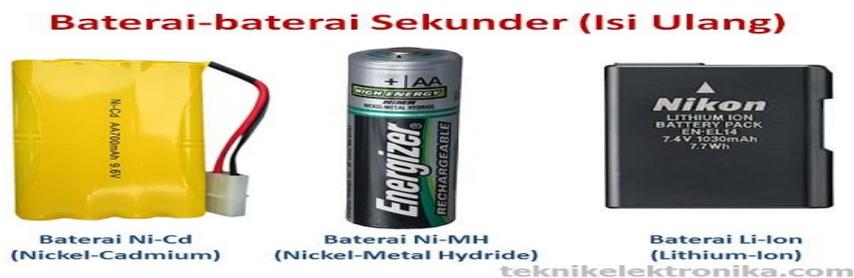
- a. 20 jam jika kuat arus yang dipakai hanya sebesar 0,9A untuk tiap jam artinya hanya memakai daya sebesar 10,8Watt/jam ( $12V \times 0,9A$ ). Kapasitas = 18Ah ( $0,9A \times 20\text{hour}$ ).



- b. 5 jam jika kuat arus yang dipakai 3,06A atau berdaya 36,72Watt/jam ( $12V \times 3,06A$ ). Kapasitas = 15,3Ah ( $3,06A \times 5\text{hour}$ ).
- c. 1 jam jika kuat arus yang dipakai 10,8A atau berdaya 129,6Watt/jam ( $12V \times 10,8A$ ). Kapasitas = 10,8Ah ( $10,8A \times 1\text{hour}$ )
- d. 1/2 jam jika kuat arus yang dipakai sama dengan kapasitas efektifnya yang 18 Ah atau berdaya 216 Watt/jam ( $12V \times 18A$ ). Kapasitas = 9 Ah ( $18A \times 0,5\text{hour}$ ).

## 2.2 Baterai

Baterai merupakan sekumpulan sel–sel kimia yang masing–masing berisi dua elektron logam yang dicelupkan dalam larutan penghantar yang disebut elektrolit. Akibat reaksi–reaksi kimia antara konduktor–konduktor dan elektrolit satu elektroda anoda bermuatan positif dan lainnya katoda bermuatan negatif. Secara harfiah baterai berfungsi sebagai media penyimpan dan penyedia energi listrik.



**Gambar 2.2** Baterai Sekunder

(<http://teknikelektronika.com/pengertian-baterai-jenis-jenis-baterai/>)

## 2.3 Sistem Komunikasi antara Joystick dengan Mobile Robot

Sebagai bentuk cara komunikasi atau interaksi manusia dengan robot, maka perlu adanya data-data yang *diinputkan* kedalam suatu sistem robot untuk dibaca oleh robot tersebut. Terkait hal tersebut, teknologi *wireless* dapat digunakan pada media transmisi data dari media *input* kedalam sistem *mobile robot* yaitu dengan memanfaatkan gelombang radio dengan pemancar Radio Frekuensi (RF).



*Mobile robot* merupakan sebuah robot yang dapat bergerak dengan leluasa karena memiliki alat gerak untuk berpindah posisi. Dan mempunyai aktuator berupa roda untuk menggerakkan keseluruhan badan robot tersebut, sehingga robot tersebut dapat melakukan perpindahan posisi dari satu titik ke titik lain. (Zaenurrohman-Utis Sutisna, 2014. Perancangan Sistem Kontrol *Wireless* pada *Mobile Robot*, Purwokerto)

### 2.3.1 Pemancar Radio

Pemancar radio adalah teknologi yang digunakan untuk pengiriman sinyal dengan cara modulasi dan gelombang elektromagnetik. Gelombang ini melintas dan merambat lewat udara dan bisa juga merambat lewat ruang angkasa yang hampa udara, karena gelombang ini tidak memerlukan medium pengangkut. Gelombang radio adalah satu bentuk dari radiasi elektromagnetik dan terbentuk ketika obyek bermuatan listrik dimodulasi pada frekuensi yang terdapat dalam frekuensi gelombang radio (*RF*) dalam suatu spektrum elektromagnetik.

Pemancar radio *FM* digunakan sebagai perangkat yang dapat mengirim sinyal modulasi yang ditransmisikan melalui media udara. Sinyal modulasi yang dipancarkan *Radio Frequency FM* di bagian *transmitter* ke udara kemudian diterima oleh *Radio Frequency FM* di bagian *receiver*. Kemudian sinyal modulasi yang sudah diterima *Radio Frequency FM* di bagian penerima disalurkan ke *input* demodulator untuk melalui proses selanjutnya sampai sinyal termodulasi tersebut menjadi sinyal informasi.

### 2.3.2 Gelombang Modulasi Frekuensi

Modulasi frekuensi didefinisikan sebagai deviasi frekuensi sesaat sinyal pembawa (dari frekuensi tak termodulasinya) sesuai dengan amplitudo sesaat sinyal pemodulasi. Sinyal pembawa dapat berupa gelombang sinus, sedangkan sinyal pemodulasi (informasi) dapat berupa gelombang apa saja (sinusoidal, kotak, segitiga, atau sinyal lain misalnya sinyal audio). Secara matematis, sinyal termodulasi *FM* dapat dinyatakan dengan.

$$e_{FM} = V_c \sin(\omega_c t + m_f \sin \omega_m t) \dots \dots \dots (1)$$



dimana

$e_{FM}$  : sinyal termodulasi *FM*

$V_c$  : amplitudo maksimum sinyal pembawa

$m_f$  : indeks modulasi *FM*

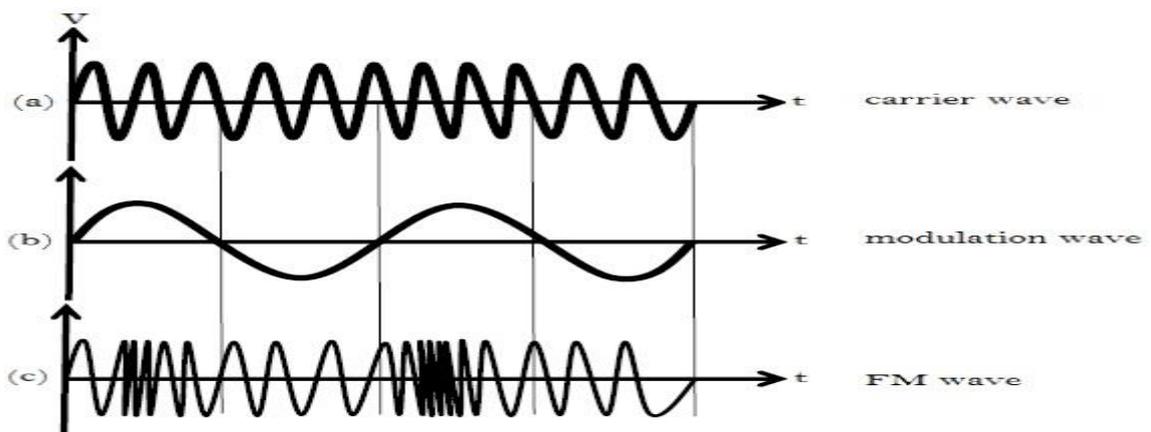
$\omega_c$  : frekuensi sudut sinyal pembawa (radian/detik)

$\omega_m$  : frekuensi sudut sinyal pemodulasi (radian/detik)

Pada modulasi frekuensi, amplitudo sinyal pembawa selalu tetap (tidak berubah-ubah), sedangkan frekuensinya berubah-ubah tergantung pada amplitudo sinyal modulasi. Perubahan naik turunnya amplitudo pemodulasi akan berpengaruh pada simpangan frekuensi sinyal pembawa yang disebut dengan frekuensi deviasi. Di dalam teknik *FM* terdapat tiga jenis frekuensi yaitu.

- Frekuensi *carrier* (pembawa) : Pada *FM* berkisar dari 87.5 MHz – 108 MHz
- Frekuensi simpangan : Perubahan frekuensi *carrier* dinamakan frekuensi simpangan yang mewakili kekuatan amplitudo dari sinyal informasi
- Frekuensi informasi : Kecepatan perubahan frekuensi simpangan dalam satu detik dinamakan frekuensi informasi.

Pemakaian saluran *FM* memberikan respon yang cukup untuk frekuensi audio dan menyediakan hubungan radio dengan *noise* rendah. Gambar 2.3 mengilustrasikan modulasi frekuensi sinyal pembawa sinusoidal dengan menggunakan sinyal pemodulasi yang juga berbentuk sinyal sinusoidal.



**Gambar 2.3 (a) Sinyal Pembawa, (b) Sinyal Pemodulasi, (c) Sinyal Termodulasi *FM***

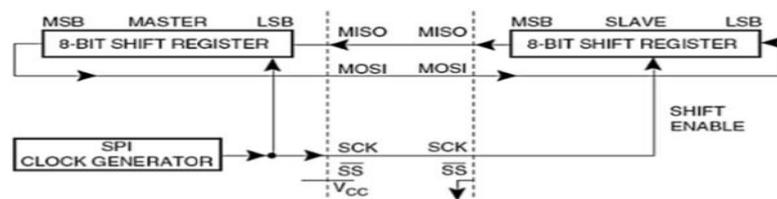


### 2.3.3 Transmitter dan Receiver Wireless

Pada *wireless transmitter* terdapat beberapa tombol sebagai masukan data. Jika *wireless transmitter* menerima suatu masukan data, maka kemudian masukan data tersebut dikirim ke bagian *receiver*. Selanjutnya dari *receiver* dikirim ke mikrokontroler melalui komunikasi SPI.

### 2.3.4 Serial Peripheral Interface (SPI)

*Serial Peripheral Interface* (SPI) SPI merupakan hubungan data serial yang standar untuk mikroprosesor, mikrokontroler dan *peripheral* yang dikeluarkan oleh perusahaan Motorola. Hubungan dalam SPI merupakan hubungan data serial yang *full-duplex, synchronous*. SPI dipakai untuk menyediakan komunikasi antara kontroler dengan piranti *peripheral*. Komunikasi antara mikroprosesor dan *peripheral* atau *inter-processor* dapat dilakukan dengan SPI. Piranti *peripheral* SPI tersedia dari *shift register* sederhana untuk ADC, DAC dan chip memori. Kontroler yang terintegrasi dengan port SPI menyediakan hubungan ke piranti *peripheral* dengan *port* SPI. Sistem SPI cukup fleksibel sebagai antarmuka secara langsung dengan banyak *peripheral* yang tersedia. *Port* SPI memiliki sinyal sebagai berikut. Ketika dikonfigurasi sebagai *slave*, antarmuka SPI akan menjadi status *sleep* dengan jalur MISO ke kondisi *tri state (high impedance)* selama pin SS dibawa ke logika tinggi. Pada kondisi ini, perangkat lunak mungkin memperbarui isi dari SPI *Data register* (SPDR), tetapi data tidak akan digeser keluar oleh pulsa clock yang diterima pada pin SCK sampai pin SS dibuat rendah. Jika bit SPI *Interrupt Enable* (SPIE) pada register diset, sebuah interupsi diminta. *Slave* mungkin melanjutkan untuk data baru dikirim ke SPDR sebelum membaca data yang diterima. *Byte* terakhir yang diterima akan dijaga dalam *Buffer Register* untuk digunakan selanjutnya. (Zaenurrohman, 2014)



**Gambar 2.4** Interkoneksi SPI (*Serial Peripheral Interface*)



## 2.4 Joystick Wireless PS2

*Joystick Wireless PS2* terdiri dari dua modul, yaitu modul *transmitter* dan modul *receiver*. Modul *transmitter* berfungsi sebagai data *input* dan mengirim data *input* tersebut ke modul *receiver*. Sedangkan modul *receiver* berfungsi sebagai penerima data yang dikirim dari modul *transmitter*. Pada setiap Stik PS (*joystick Playstation*) terdapat kontroler yang bertugas untuk berkomunikasi dengan *console playstation*. Komunikasi yang digunakan adalah serial sinkron, yaitu data dikirim satu per satu melalui jalur data. Untuk mengkoordinasikan antara pengirim dan penerima terdapat satu jalur *clock*.

### 2.4.1 Pin Konfigurasi dan Communication

Konfigurasi pin *wiring Conector Joystick Wireless PS2*, berikut gambarnya :



**Gambar 2.5 Connector PS2 Controller**

(Sumber : <http://data-uploadku.com/2013/04/pin-konfigurasi-communication.html?m=1>)

*Joystick Wireless PS2* menggunakan komunikasi SPI (*Serial Peripheral Interface*) atau biasa orang sebut sebagai 3 *wire interface*. Adapun penjelasan setiap pin dari *connector PS2 controller* sebagai berikut :

1. Data : Dihubungkan pada pin PA.7 pada mikrokontroler
2. *Command* : Dihubungkan pada pin PA.6 pada mikrokontroler
3. *Motor Vibrate* : Vcc
4. Gnd : *Ground*
5. Vcc : +5v
6. *Attention* : Dihubungkan pada pin PA.5
7. *Clock* : Dihubungkan pada pin PA.4 (SCK)



8. NC : Tidak dihubungkan

9. ACK : Tidak dihubungkan

Pada penjelasan pin-pin dari *joystick PS2* diketahui bahwa untuk melakukan hubungan anatar *joystick PS2* dan mikrokontroler dibutuhkan 3 jalur utama yaitu pin 1 Data dihubungkan ke mikrokontroler pin PA.7 (MISO), pin 2 *Command* dihubungkan ke mikrokontroler pin PA.6 (MOSI) dan pin 7 *Clock* dihubungkan ke mikrokontroler pin PA.4 (SCK). Dari ketiga pin tersebut penjelasannya sebagai berikut :

1. MISO : *Master Output Slave Input* artinya jika dikonfigurasi sebagai *master* maka pin MOSI sebagai *output* tetapi jika dikonfigurasi sebagai *slave* maka pin MOSI sebagai *input*.

2. MOSI : *Master Input Slave Output* artinya jika dikonfigurasi sebagai *master* maka pin MOSI sebagai *input* tetapi jika dikonfigurasi sebagai *slave* maka pin MOSI sebagai *output*.

3. CLK : *Clock* jika dikonfigurasi sebagai *master* maka pin CLK berlaku sebagai *output* tetapi jika dikonfigurasi sebagai *slave* maka pin CLK berlaku sebagai *input*.

#### 2.4.2 Data Protokol

Untuk berkomunikasi dengan kontroler pada *joystick wireless PS2* diperlukan beberapa proses pengiriman ID. Mikrokontroler mengirim data &H01 (*start up*), setelah itu mikrokontroler mengirim data &H42 (*read data*). Kemudian disaat yang sama mikrokontroler akan menerima data tipe *joystick* yang digunakan. &H41 = Konsul Digital dan &H73 = Konsul Analog. Setelah itu mikrokontroler akan menerima data &H5. Data *byte* pertama akan diterima kemudian *byte* kedua, setelah itu data analog 1 dan analog 2.



### 2.4.3 Frame Data

**Tabel 2.1** Frame Data PS2 Controller

Byte	PsxCMD	PsxData	Keterangan							
01	&H01									
02	&H42	&H41/&H73								
03	-	&H5A	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
04	-	Digital 1	←	↓	→	↑	start	Joy R	Joy L	Select
05	-	Digital 2	Ⓜ	x	Ⓜ	Δ	R1	L1	R2	L2
06	-	Analog 1 X	Joystick Analog kanan sumbu x 128 center							
07	-	Analog 1 Y	Joystick Analog kanan sumbu y 128 center							
08	-	Analog 2 X	Joystick Analog kiri sumbu x 128 center							
09	-	Analog 2 Y	Joystick Analog kiri sumbu Y 128 center							

### 2.4.4 Komponen Joystick

Joystick yang digunakan pada umumnya memiliki komponen-komponen sebagai berikut :

1. *Switch* : Menggunakan saklar-saklar dua keadaan sebagai pemberi *input* ke mikrokontroler.
2. *Wiring* : Menggunakan kabel dan konektor DB9 sebagai penghubung dari joystick menuju mikrokontroler.
3. *Power Supply* : Membutuhkan tegangan DC untuk dapat mengaktifkan joystick.
4. Sebuah joystick terdiri dari 4 buah potensio yang yang disusun menjadi pada sebuah rangkaian yang berguna menentukan arah gerak joystick.

Berikut merupakan contoh gambar dari joystick yang ditunjukkan pada gambar 2.4 :



**Gambar 2.6** Joystick Wireless PS2

(Sumber : <http://data-uploadku.com/2013/04/konfigurasi-joystick-di-emulator-psx.html?m=1>)

## 2.5 Mikrokontroler

Mikrokontroler adalah sebuah *system microprocessor* dimana didalamnya sudah terdapat CPU, ROM, RAM, I/O, *Clock* dan peralatan *internal* lainnya yang



sudah saling terhubung dan terorganisasi (teralamati) dengan baik oleh pabrik pembuatnya dan dikemas dalam satu *chip* yang siap pakai. Sehingga kita tinggal memprogram isi ROM sesuai aturan penggunaan oleh pabrik yang membuatnya menurut. (Winoto, 2010)

Teknologi yang digunakan pada mikrokontroler AVR berbeda dengan mikrokontroler seri MCS-51. AVR berteknologi RISC (*Reduced Instruction Set Computer*), sedangkan seri MCS-51 berteknologi CISC (*Complex Instruction Set Computer*). Mikrokontroler AVR dapat dikelompokkan menjadi empat kelas, yaitu keluarga ATtiny, keluarga AT90Sxx, keluarga ATMega, dan keluarga AT89RFxx. Pada dasarnya yang membedakan masing-masing kelas adalah memori, kelengkapan peripheral dan fungsi-fungsi tambahan yang dimiliki.

Berikut ini penjelasan lebih lengkap mengenai Mikrokontroler ATMega8535.

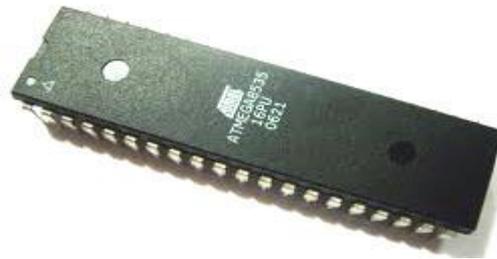
### **2.5.1 Mikrokontroler ATMega8535**

ATMega8535 adalah mikrokontroler CMOS 8 *bit* daya rendah berbasis arsitektur RISC. Instruksi dikerjakan pada satu siklus *clock*, ATMega8535 mempunyai *throughput* mendekati 1 MIPS per MHz, hal ini membuat ATMega8535 dapat bekerja dengan kecepatan tinggi walaupun dengan penggunaan daya rendah. Mikrokontroler ATMega8535 memiliki beberapa fitur atau spesifikasi yang menjadikannya sebuah solusi pengendali yang efektif untuk berbagai keperluan.

Mikrokontroler AVR ATMega8535 memiliki fitur yang cukup lengkap. Mikrokontroler AVR ATMega8535 telah dilengkapi dengan ADC *internal*, EEPROM *internal*, *Timer/Counter*, PWM, analog *comparator* dan lain-lain (Ary Heryansto.M, 2008).



Berikut ini gambar Mikrokontroler ATmega8535 :



**Gambar 2.7** Mikrokontroler ATmega8535

### 2.5.2 Arsitektur ATmega8535

- Saluran IO sebanyak 32 buah, yaitu Port A, Port B, Port C dan Port D
- ADC 10 bit sebanyak 8 Channel
- Tiga buah *timer / counter*
- 32 register
- *Watchdog Timer* dengan *oscilator* internal
- SRAM sebanyak 512 *byte*
- Memori *Flash* sebesar 8 kb
- Sumber *Interrupt* internal dan eksternal
- Port SPI (*Serial Pheriperal Interface*)
- EEPROM *on board* sebanyak 512 *byte*
- Komparator analog
- Port USART (*Universal Shynchronous Ashynchronous Receiver Transmitter*)

### 2.5.3 Fitur ATmega8535

- Sistem processor 8 bit berbasis RISC dengan kecepatan maksimal 16 MHz.
- Ukuran *memory flash* 8KB, SRAM sebesar 512 *byte*, EEPROM sebesar 512 *byte*.
- ADC internal dengan resolusi 10 bit sebanyak 8 *channel*.
- Port komunikasi serial USART dengan kecepatan maksimal 2.5 Mbps.



- Mode *Sleep* untuk penghematan penggunaan daya listrik.

Penjelasan :

- *Flash* adalah suatu jenis *Read Only Memory* yang biasanya diisi dengan program hasil buatan manusia yang harus dijalankan oleh mikrokontroler.
- RAM (*Random Acces Memory*) merupakan memori yang membantu CPU untuk penyimpanan data sementara dan pengolahan data ketika program sedang *running*.
- EEPROM (*Electrically Erasable Programmable Read Only Memory*) adalah memori untuk penyimpanan data secara permanen oleh program yang sedang *running*.
- Port I/O adalah kaki untuk jalur keluar atau masuk sinyal sebagai hasil keluaran ataupun masukan bagi program *Timer* adalah modul dalam *hardware* yang bekerja untuk menghitung waktu/pulsa.
- UART (*Universal Asynchronous Receive Transmit*) adalah jalur komunikasi data khusus secara serial *asynchronous*.
- PWM (*Pulse Width Modulation*) adalah fasilitas untuk membuat modulasi pulsa.
- ADC (*Analog to Digital Converter*) adalah fasilitas untuk dapat menerima sinyal analog dalam range tertentu untuk kemudian dikonversi menjadi suatu nilai digital dalam *range* tertentu.
- SPI (*Serial Peripheral Interface*) adalah jalur komunikasi data khusus secara serial secara serial *synchronous*.
- ISP (*In System Programming*) adalah kemampuan khusus mikrokontroler untuk dapat diprogram langsung dalam sistem rangkaiannya dengan membutuhkan jumlah pin yang minimal.

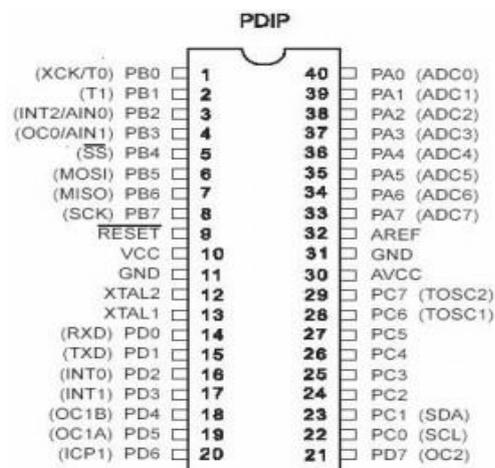
Secara garis besar, arsitektur mikrokontroler ATMEGA8535 terdiri dari :

- 32 saluran I/O (Port A, Port B, Port C, dan Port D)
- 10 bit 8 Channel ADC (*Analog to Digital Converter*)
- 4 channel PWM



- 6 *Sleep Modes* : *Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby*
- 3 buah *timer/counter*
- *Analog comparator*
- *Watchdog timer* dengan osilator internal
- 512 *byte SRAM*
- 512 *byte EEPROM*
- 8 kb *Flash memory* dengan kemampuan *Read While Write*
- Unit interupsi (internal & eksternal)
- Port antarmuka SPI “*memory map*”
- Port USART untuk komunikasi serial dengan kecepatan maksimal 2,5Mbps
- 4.5 sampai 5.5V *operation*, 0 sampai 16MHz

#### 2.5.4 Konfigurasi Pin ATmega8535



**Gambar 2.8** Konfigurasi Pin ATmega8535

Berikut ini merupakan penjelasan pada konfigurasi Pin ATmega8535 gambar 2.6 :

- VCC merupakan Pin yang berfungsi sebagai pin masukan catudaya
- GND merupakan Pin *Ground*
- Port A (PA0...PA7) merupakan pin I/O dan pin masukan ADC



- Port B (PB0...PB7) merupakan pin I/O dan pin yang mempunyai fungsi khusus yaitu *Timer/Counter*, komparator Analog dan SPI
- Port C (PC0...PC7) merupakan port I/O dan pin yang mempunyai fungsi khusus, yaitu komparator analog dan *Timer Oscillator*
- Port D (PD0...PD1) merupakan port I/O dan pin fungsi khusus yaitu komparator analog dan interrupt eksternal serta komunikasi serial
- RESET merupakan pin yang digunakan untuk mereset mikrokontroler
- XTAL1 dan XTAL2 merupakan pin masukan *clock* eksternal
- AVCC merupakan pin masukan untuk tegangan ADC
- AREF merupakan pin masukan tegangan referensi untuk ADC

### 2.5.5 Keterangan PIN ATmega8535

- Port A

Merupakan 8-bit *directional port I/O*. Setiap pinnya dapat menyediakan internal *pull-up* resistor (dapat diatur per bit). *Output buffer Port A* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. Data *Direction Register port A* (DDRA) harus disetting terlebih dahulu sebelum *Port A* digunakan. Bit-bit DDRA diisi 0 jika ingin memfungsikan pin-pin port A yang bersesuaian sebagai *input*, atau diisi 1 jika sebagai *output*. Selain itu, kedelapan pin port A juga digunakan untuk masukan sinyal analog bagi *A/D converter*.

- Port B

Merupakan 8-bit *directional port I/O*. Setiap pinnya dapat menyediakan internal *pull-up* resistor (dapat diatur per bit). *Output buffer Port B* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. Data *Direction Register port B* (DDRB) harus disetting terlebih dahulu sebelum *Port B* digunakan. Bit-bit DDRB diisi 0 jika ingin memfungsikan pin-pin *port B* yang bersesuaian sebagai *input*, atau diisi 1 jika sebagai *output*.

Pin-pin *port B* juga memiliki untuk fungsi-fungsi alternatif khusus seperti yang dapat dilihat dalam tabel :



Tabel 2.2 Fungsi Khusus Port B

Port Pin	Fungsi Khusus
PB0	T0 = <i>timer/counter 0 external counter input</i>
PB1	T1 = <i>timer/counter 0 external counter input</i>
PB2	AIN0 = <i>analog comparator positive input</i>
PB3	AIN1 = <i>analog comparator negative input</i>
PB4	SS = <i>SPI slave select input</i>
PB5	MOSI = <i>SPI bus master output / slave input</i>
PB6	MISO = <i>SPI bus master input / slave output</i>
PB7	SCK = <i>SPI bus serial clock</i>

- Port C

Merupakan 8-bit *directional port I/O*. Setiap pinnya dapat menyediakan internal *pull-up* resistor (dapat diatur per bit). *Output buffer Port C* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. Data *Direction Register port C (DDRC)* harus di-*setting* terlebih dahulu sebelum *Port C* digunakan. Bit-bit *DDRC* diisi 0 jika ingin memfungsikan pin-pin *port C* yang bersesuaian sebagai *input*, atau diisi 1 jika sebagai *output*. Selain itu, dua pin *port C* (PC6 dan PC7) juga memiliki fungsi alternatif sebagai *oscillator* untuk *timer/counter 2*.

- Port D

Merupakan 8-bit *directional port I/O*. Setiap pinnya dapat menyediakan internal *pull-up* resistor (dapat diatur per bit). *Output buffer Port D* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. Data *Direction Register port D (DDRD)* harus di-*setting* terlebih dahulu sebelum *Port*



D digunakan. Bit-bit DDRD diisi 0 jika ingin memfungsikan pin-pin *port D* yang bersesuaian sebagai *input*, atau diisi 1 jika sebagai *output*.

**Tabel 2.3** Fungsi Khusus Port D

Port Pin	Fungsi Khusus
PD0	RDX (UART <i>input line</i> )
PD1	TDX (UART <i>output line</i> )
PD2	INT0 ( external <i>interrupt 0 input</i> )
PD3	INT1 ( external <i>interrupt 1 input</i> )
PD4	OC1B ( <i>Timer/Counter1 output compareB match output</i> )
PD5	OC1A ( <i>Timer/Counter1 output compareA match output</i> )
PD6	ICP ( <i>Timer/Counter1 input capture pin</i> )
PD7	OC2 ( <i>Timer/Counter2 output compare match output</i> )

## 2.6 Program *Downloader*

Diperlukan sebuah program yang dapat mengirim data dari hasil kompil oleh program BASCOM AVR, ada banyak program *downloader* diinternet baik itu yang gratis maupun yang berbayar, didalam kesempatan ini penulis akan menjelaskan salah satu program *downloader* yaitu *Universal ISP Programmer*, dimana untuk menggunakan *software* ini membutuhkan perangkat hardware yang mendukung, berikut rangkaian elektronika pendukung dari *software Universal ISP Programmer*.

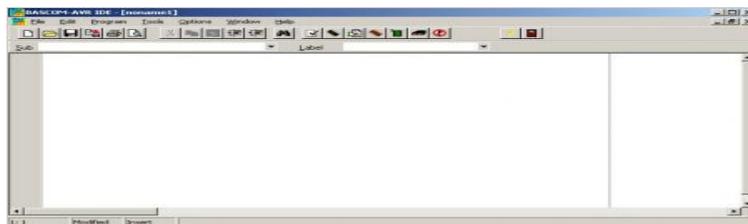
Berikut ini adalah penjelasan dari program *downloader* saat memprogram sebuah data pada suatu rangkaian yang akan diisi program :



## 2.7 Program Compiler

*Basic compiler* (BASCOM) AVR merupakan *editor list* program yang berbasis bahasa *basic*, *software* BASCOM AVR sangat mudah digunakan.

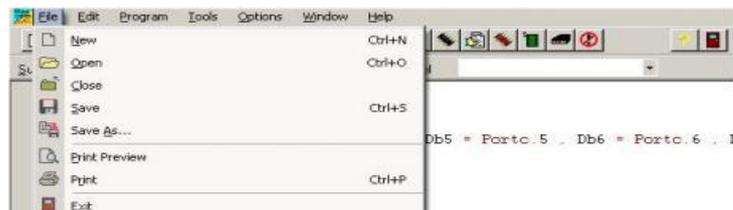
*Compiler Bascom AVR* sangat populer bagi para penghobi dunia mikrokontroler, dikarenakan struktur program didalam *bascom avr* dibuat untuk mudah difahami bagi para pemula didalam belajar mikrokontroler.



**Gambar 2.9** Halaman Editor *Bascom AVR*

Untuk menjalankan atau menggunakan *software editor* ini terlebih dahulu kita harus mengetahui setiap fungsi-fungsi dari *toolbar* yang ada pada *software* tersebut, berikut merupakan fungsi dari *toolbar* pada *Bascom AVR*.

### 2.7.1 Toolbar pada File



**Gambar 2.10** *Toolbar File*

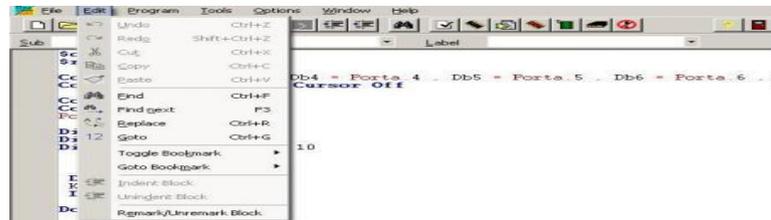
Berikut merupakan penjelasan *toolbar file* :

- a. *New*, digunakan untuk membuat *project* baru atau membuat *file* program baru.
- b. *Open*, digunakan untuk membuka *project* atau *file* program
- c. *Save*, digunakan untuk menyimpan *project* atau menyimpan *file* program.
- d. *Save As*, digunakan digunakan untuk menyimpan *project* atau menyimpan *file* dengan nama yang berbeda dari sebelumnya.
- e. *Print Preview*, digunakan untuk melihat hasil cetakan *print out* dari *sintak*
- f. *Print*, digunakan untuk mencetak *file* program.



g. *Exit*, digunakan untuk keluar dari BASCOM AVR

### 2.7.2 Toolbar pada Edit

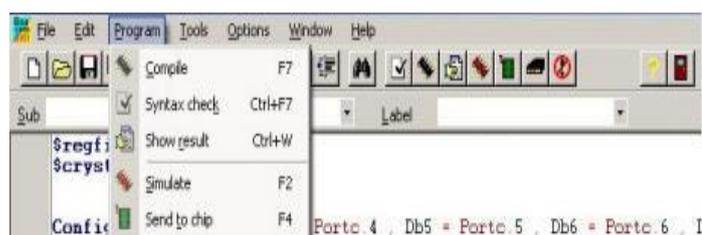


Gambar 2.11 Toolbar Edit

Berikut merupakan penjelasan *toolbar edit* :

- Undo*, digunakan untuk kembali ke langkah sebelumnya.
- Redo*, kebalikan dari *undo*.
- Cut*, digunakan untuk mengcopy dan menghapus *teks* sekaligus
- Copy*, digunakan untuk mengcopy *teks*.
- Paste*, digunakan untuk menyalin bagian yang telah dikopi.
- Find*, digunakan untuk mencari *teks* yang diinginkan.
- Find next*, sama halnya dengan *find* hanya saja berikutnya.

### 2.7.3 Toolbar pada Program

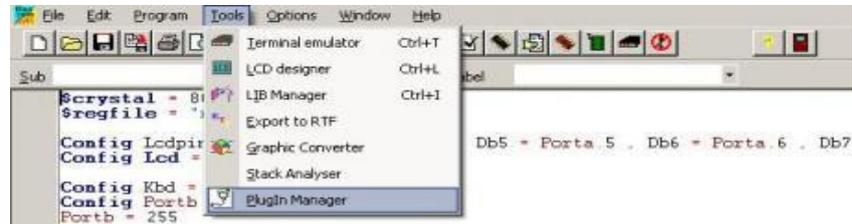


Gambar 2.12 Toolbar Program

- compile*, digunakan untuk mengkompilasi program. Proses ini akan menghasilkan *file* berekstensi \*.hex
- syntax check*, digunakan untuk memeriksa apakah terjadi kesalahan pada penulisan program atau tidak.
- Simulate*, digunakan untuk mensimulasikan program.
- Send to chip*, digunakan untuk mengirim file \*.hex ke dalam *chip* mikrokontroler (mendownload program mikrokontroler).



### 2.7.4 Toolbar pada Tools



Gambar 2.13 Toolbar Tools

- Terminal emulator**, digunakan untuk simulasi komunikasi serial dengan komputer (RS232) hampir sama dengan *Hypert Terminal*
- Lcd designer**, digunakan untuk mendesain karakter LCD yang diinginkan.
- Libray Manager**, digunakan untuk *library* yang terdapat pada BASCOM AVR
- Export to RTF**, digunakan untuk mengkonversi penulisan program pada RTF
- Graphic Converter**, digunakan untuk menkonversi gambar ke LCD yang menjang RGB (*high kualitas LCD*).
- Stack Analyser**, digunakan untuk menganalisa *stack* program.
- PlugIn Manager**, digunakan untuk mengatur *plugin* yang ada.

### 2.7.5 Toolbar pada Options



Gambar 2.14 Toolbar Options

- Compiler**, digunakan untuk mensetting *chip*, *output*, *communication*, I2C, LCD
- Communication**, digunakan untuk mensetting komunikasi mikrokontroler.
- Simulator**, digunakan untuk mensetting simulasi pada BASCOM AVR.
- Programmer**, digunakan untuk mensetting *downloader programmer* yang akan digunakan.
- Monitor**, untuk *mensetting* tampilan.



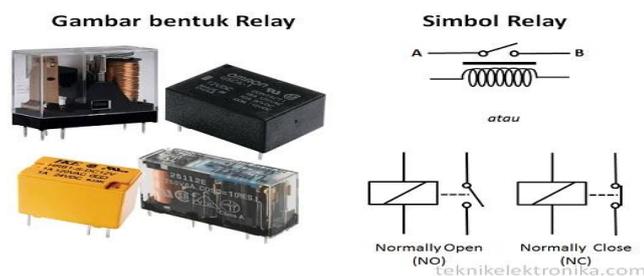
## 2.8 Relay

*Relay* merupakan bentuk hambatan terdiri atas titik-titik kontak bawah dengan gulungan *spool*-nya tidak bergerak dan titik kontak bagian atas yang bergerak. Prinsip kerja hambatan adalah menghubungkan titik-titik kontak bagian bawah dengan titik bagian atas yaitu terletak gulungan *spool* dialiri arus listrik yang timbul elektromagnet. (Handy Wicaksono,1996,1-12).

Secara sederhana *relay* elektromekanis ini didefinisikan sebagai berikut :

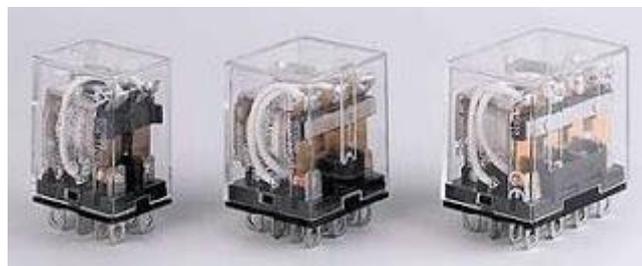
- Alat yang menggunakan gaya elektromagnetik untuk menutup (atau membuka) kontak saklar.
- Saklar yang digerakkan (secara mekanis) oleh daya/energi listrik.

Dibawah ini adalah gambar fisik, bentuk dan Simbol *Relay* yang sering ditemukan di Rangkaian Elektronika.



**Gambar 2.15** Gambar dan Simbol *Relay*

(<http://teknikelektronika.com/pengertian-relay-fungsi-relay/>)



**Gambar 2.16** *Relay*

(Sumber : Kilian, Christopher T, *Modern Control Technology*, (West Published Co : 1996)



Bagian titik kontak dibagi menjadi 2 bagian yaitu bagian kontak utama dan kontak bantu yaitu : Bagian kontak utama gunanya untuk menghubungkan dan memutuskan arus listrik bagian yang menuju beban/pemakai. Bagian kontak bantu gunanya untuk menghubungkan dan memutuskan arus listrik ke bagian yang menuju bagian pengendali. Kontak Bantu mempunyai 2 kontak yaitu kontak hubung (NC) dan kontak putus (NO) menandakan masing-masing kontak dan gulungan *spool*. Secara umum, *relay* digunakan untuk memenuhi fungsi-fungsi berikut :

- *Remote control* : dapat menyalakan atau mematikan alat dari jarak jauh.
- Penguatan daya : menguatkan arus atau tegangan.
- Pengatur logika kontrol suatu sistem.

Susunan kontak pada *relay* adalah:

- *Normally Open* : *Relay* akan menutup bila dialiri arus listrik.
- *Normally Close* : *Relay* akan membuka bila dialiri arus listrik.
- *Changeover* : *Relay* ini memiliki kontak tengah yang akan melepaskan diri dan membuat kontak lainnya berhubungan.

### 2.8.1 Prinsip Kerja *Relay*

Pada dasarnya, *Relay* terdiri dari 4 komponen dasar yaitu :

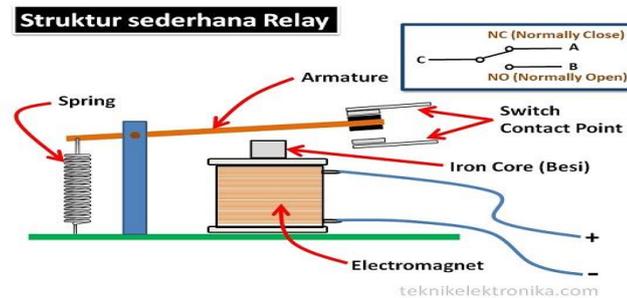
1. *Electromagnet (Coil)*
2. *Armature*
3. *Switch Contact Point (Saklar)*
4. *Spring*

Seperti saklar, *relay* juga dibedakan berdasar *pole* dan *throw* yang dimilikinya.

- *Pole* : banyaknya *contact* yang dimiliki oleh *relay*
- *Throw* : banyaknya kondisi (*state*) yang mungkin dimiliki *contact*.



Berikut ini merupakan gambar dari bagian-bagian *Relay* :



**Gambar 2.17** Struktur Sederhana *Relay*

(<http://teknikelektronika.com/pengertian-relay-fungsi-relay/>)

Kontak *normally open* akan membuka ketika tidak ada arus mengalir pada kumparan, tetapi tertutup secepatnya setelah kumparan menghantarkan arus atau diberi tenaga. Kontak *normally close* akan tertutup apabila kumparan tidak diberi tenaga dan membuka ketika kumparan diberi daya. Masing-masing kontak biasanya digambarkan sebagai kontak yang tampak dengan kumparan tidak diberi tenaga atau daya.

*Relay* terdiri dari 2 terminal *trigger*, 1 terminal *input* dan 1 terminal *output*.

1. Terminal *trigger* : yaitu terminal yang akan mengaktifkan *relay*, seperti alat elektronik lainnya *relay* akan aktif apabila di aliri arus + dan arus -. Pada contoh *relay* yang kita gunakan terminal *trigger* ini adalah 85 dan 86.
2. Terminal *input* : yaitu terminal tempat kita memberikan masukan, pada contoh adalah terminal 30.
3. Terminal *output* : yaitu tempat keluarnya *output* pada contoh adalah terminal 87.

### 2.8.2 Jenis-jenis *Relay*

Berikut ini penggolongan *relay* berdasar jumlah *pole* dan *throw* :

- *DPST (Double Pole Single Throw)*, *relay* golongan ini memiliki 6 terminal, diantaranya 4 terminal yang terdiri dari 2 pasang terminal saklar sedangkan

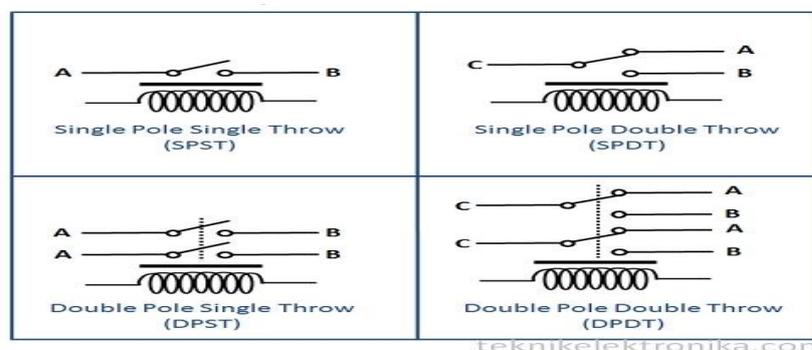


2 terminal lainnya untuk *coil*. Relay DPST dapat dijadikan 2 saklar yang dikendalikan oleh 1 *coil*.

- *SPST (Single Pole Single Throw)*, relay golongan ini memiliki 4 terminal, 2 terminal untuk saklar dan 2 terminalnya lagi untuk *coil*.
- *SPDT (Single Pole Double Throw)*, relay golongan ini memiliki 5 terminal, 3 terminal untuk saklar dan 2 terminalnya lagi untuk *coil*.
- *DPDT (Double Pole Double Throw)*, relay golongan ini memiliki terminal sebanyak 8 terminal, diantaranya terminal yang merupakan 2 pasang relay SPDT yang dikendalikan oleh 1 (*single*) *coil*. Sedangkan 2 terminal lainnya untuk *coil*.

Selain golongan *relay* diatas, terdapat juga *relay-relay* yang *Pole* dan *Throw*-nya melebihi dari dua. Misal-nya 3PDT (*Triple Pole Double Throw*) ataupun 4PDT (*Four Pole Double Throw*) dan lain sebagainya.

Berikut ini merupakan gambar dari jenis *Relay* berdasarkan *Pole* dan *Throw*-nya :



**Gambar 2.18** Jenis Relay berdasarkan *Pole* dan *Throw*

(<http://teknikelektronika.com/pengertian-relay-fungsi-relay/>)

### 2.8.3 Fungsi-fungsi *Relay*

Beberapa fungsi *relay* yang telah umum diaplikasikan kedalam peralatan elektronika diantara-nya adalah :

1. *Relay* digunakan untuk menjalankan fungsi logika (*logic function*).
2. *Relay* digunakan untuk memberikan fungsi penundaan waktu (*time delay function*).



3. *Relay* digunakan untuk mengendalikan sirkuit tegangan tinggi dengan bantuan dari signal tegangan rendah.
4. Ada juga *relay* yang berfungsi untuk melindungi motor ataupun komponen lainnya dari kelebihan tegangan ataupun hubung singkat (*short*).

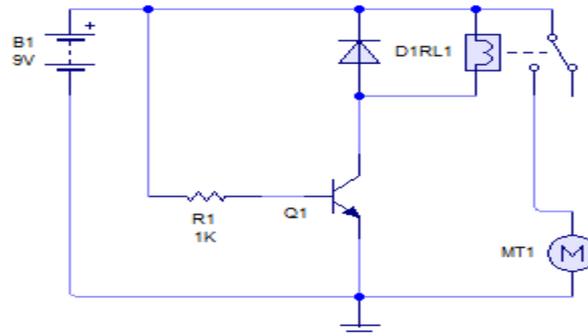
## 2.9 *Driver Relay*

Rangkaian *driver relay* berfungsi untuk mengendalikan motor arus searah (dc) yang dihasilkan dari *port* paralel I/O. Sinyal dari keluaran port biasanya berupa sinyal-sinyal yang kecil, sehingga tidak mampu untuk menggerakkan sistem daya berupa motor arus searah. Untuk dapat dimanfaatkan sinyal keluaran port, diperlukan suatu rangkaian *driver relay* agar sinyal yang kecil dapat dipergunakan untuk penggerak objek yang akan dikendalikan dari jarak jauh. Rangkaian *driver relay* ini dibangun oleh suatu komponen utama yaitu transistor dan *relay*. Transistor di rangkain *driver relay* difungsikan sebagai penguat sinyal dan *switching*, serta *relay* sebagai penggerak motor dc. *Driver relay* ini selain sebagai sebagai penguat dan *switching*, sekaligus difungsikan untuk mengendalikan motor dc dalam sistem pembalik putaran. Jadi, *driver relay* ini dapat mengatur arah putaran motor *forward* dan *reverse*. Semua *driver relay* pada sistem ini memiliki rangkaian dan karakteristik yang sama. Saat *relay* 1 bekerja maka posisi positif motor akan mendapat sumber tegangan positif dan posisi negatif motor terhubung dengan kutub negatif sumber tegangan. Sehingga, motor akan berputar dengan arah putaran searah jarum jam (*clockwise*). Dengan cara yang sama untuk menggerakkan kontak *relay* 2, maka terjadi kondisi yang berkebalikan yaitu motor akan berputar dengan arah putaran yang berlawanan arah jarum jam (*counter clockwise*).

Penggunaan *driver relay* ini menjadi pilihan karena *driver relay* mudah dikontrol, dapat diberi beban yang besar baik beban AC maupun DC serta sebagai isolator yang baik antara rangkaian beban dengan rangkaian kendali. Rangkaian *driver relay* dapat dibangun menggunakan konsep transistor sebagai saklar. Teknik antara *relay* dengan rangkaian digital atau mikrokontroler adalah rangkaian *driver relay* dengan menggunakan transistor sebagai penguat.



Berikut merupakan contoh dari gambar rangkaian *Driver Relay* :



**Gambar 2.19** Rangkaian *Driver Relay*

Pada rangkaian menyerupai sirkuit diatas, dapat dilihat untuk mengoperasikan transistor sebagai saklar transistor dalam keadaan sepenuhnya "OFF" (*cut-off*) atau dalam keadaan "ON" (saturasi).. Namun, ketika dinyalakan dalam kondisi ON (saturasi) , maka aliran arus maksimum. Dalam prakteknya ketika transistor diaktifkan "OFF", arus kebocoran akan kecil ketika mengalir melalui transistor dan ketika diaktifkan "ON" maka rangkaian tersebut akan memiliki tegangan saturasi kecil ( $V_{CE}$ ) Meskipun transistor tidak dalam saklar yang sempurna, baik di *cut-off* dan daerah saturasi. Agar arus Basis mengalir, terminal input Basis harus dibuat lebih positif daripada Emitter dengan meningkatkan itu di atas 0,7 volt yang dibutuhkan untuk perangkat silikon. Dengan memvariasikan *Base-Emitter* ini tegangan  $V_{BE}$  arus basis juga mengontrol jumlah arus kolektor yang mengalir melalui transistor.

Ketika arus kolektor maksimum mengalir maka transistor dikatakan saturasi. Nilai dari resistor Basis menentukan berapa banyak masukan tegangan yang diperlukan dan sesuai saat Basis untuk beralih transistor sepenuhnya "ON".

Transistor BC108 adalah transistor umum NPN *bipolar junction* (BJT) digunakan untuk memperkuat daya rendah atau aplikasi *switching*. Hal ini dirancang untuk arus yang rendah sampai medium, daya yang rendah , tegangan medium, dan dapat beroperasi pada kecepatan yang cukup tinggi.



Semua variasi memiliki beta atau *gain* arus ( $h_{FE}$ ) minimal 100 dalam kondisi yang optimal. Hal ini digunakan dalam berbagai amplifikasi dan beralih aplikasi analog.

### 2.9.1 Transistor Sebagai Saklar (*Switching Transistor*)

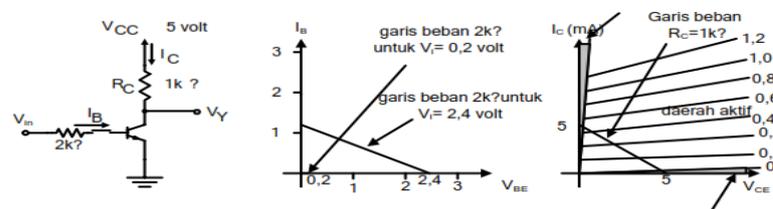
Salah satu cara termudah untuk memahami cara kerja transistor adalah dengan menganggapnya sebagai sebuah saklar. Transistor dapat di analogikan sebagai saklar *push button*. Agar saklar *push button* dapat difungsikan diperlukan gaya yang bergantung dengan konstanta pegas yang terdapat di dalam saklar tersebut, sedangkan pada transistor diperlukan arus tertentu pada basis agar dapat menghidupkan saklar transistor. (Sastra Wijaya Kusuma:2015)

Untuk menghasilkan kondisi *on/off* seperti pada saklar, transistor dioperasikan pada salah satu titik kerjanya, titik saturasi dan *cut off*. Transistor akan aktif apabila diberikan arus pada basis transistor sebesar :

$$I_B = I_{B(\text{saturasi})}$$

Saat kondisi saturasi, transistor seperti sebuah saklar yg tertutup (*on*) sehingga arus dapat mengalir dari kolektor menuju emitor. Sedangkan saat kondisi *cutoff*, transistor seperti sebuah saklar yg terbuka (*off*) sehingga tidak ada arus yg mengalir dari kolektor ke emitor. Dalam prakteknya, kontrol arus  $I_B$  biasanya dihasilkan dari sumber tegangan seperti ditunjukkan pada Gambar 2.20.

Berikut kurva V-I *input* dan kurva V-I *output*-nya :



**Gambar 2.20** Kurva Karakteristik Transistor

(Sumber :Wijaya, Sastra Kusuma. Analisa DC Transistor)



Agar transistor dapat bekerja sebagai saklar, ada beberapa hal yg harus diperhatikan diantaranya :

### 1. Menentukan $I_c$

$I_c$  adalah arus beban yg akan mengalir dari kaki kolektor ke emitor. Besarnya arus beban ini tidak boleh lebih besar dari  $I_c$  maksimum yang dpt dilewatkan oleh transistor. Arus beban ini dapat dicari dengan persamaan berikut :

$$I_{C(\text{beban})} < I_{C(\text{maks})} \leftarrow \text{syarat}$$

$$I_{C(\text{beban})} = \frac{V_{CC}}{R_L}$$

### 2. Menentukan $I_b$

Arus basis dc yang tepat cukup untuk menjenuhkan transistor diberikan rumus sebagai berikut :

$$I_B = \frac{V_B - V_{BE}}{R_B}$$

Rangkaian *Driver Relay* memiliki arti sebagai rangkaian elektronika yang biasanya digunakan untuk mengendalikan serta pengoperasian sesuatu dari jarak jauh atau semacam *remote*. Tentunya rangkaian ini bisa mempermudah dan juga memperlancar pekerjaan yang memang kadang membutuhkan rangkaian dari *relay* ini. Dengan menggunakan rangkaian *relay* tersebut, anda bisa melakukan kontrol dan juga mengoperasikan perangkat elektronik yang anda miliki dari jarak jauh dan tentu saja anda tidak perlu bergeser serta berpindah tempat duduk.

Rangkaian *relay* ini bisa juga dipasang atau diterapkan di dalam berbagai peralatan atau perangkat elektronik. Perangkat seperti televisi, radio *transmitter*, *sound system* dan juga perangkat lainnya.

Berikut dibawah ini merupakan dari contoh dari gambar rangkaian *driver relay* :

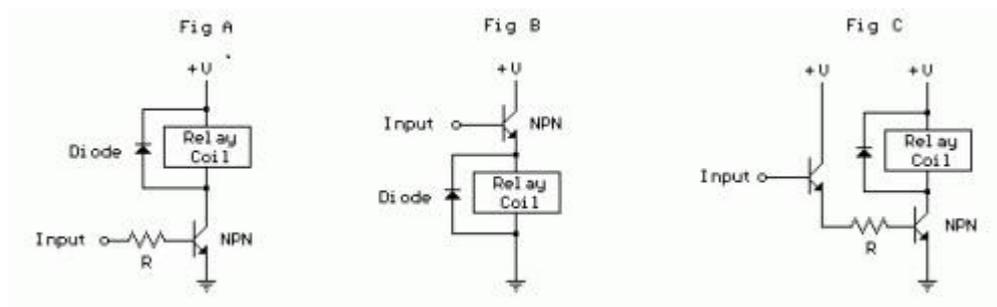
#### 2.9.2 Interface Driver Relay

Penggunaan *relay* sering menjadi pilihan karena *relay* mudah dikontrol, *relay* dapat diberi beban yang besar baik beban AC maupun DC, dan sebagai isolator yang baik antara rangkaian beban dengan rangkaian kendali. Rangkaian



*interface relay* dapat dibangun menggunakan konsep transistor sebagai saklar. Transistor yang digunakan untuk *driver relay* dapat dikonfigurasi dengan *common emitor*, *emitor follower* atau transistor *darlington*.

Teknik *interface* antara *relay* dengan rangkaian digital atau rangkaian *microcontroller* dapat dilihat pada gambar dibawah ini :



**Gambar 2.21** Rangkaian *Interface Driver Relay*

(<http://elektronika-dasar.web.id/teori-elektronika/interface-relay-ke-rangkaian-digital>)

Rangkaian *inteface* antar *relay* dengan rangkaian digital pada gambar diatas ada 3 jenis interface yang dapat digunakan. Bagian dan fungsi komponen dari rangkaian *interface relay* diatas sebagai berikut :

Rangkaian pada gambar A, rangkaian pada gambar A tersebut menggunakan mode *common emitor*, apabila basis mendapat sinyal *input* logika 1 (sumber tegangan positif) maka transistor pada gambar A akan mendapat bias maju, sehingga transistor ON dan memberikan sumber tegangan ke *relay* dan *relay* menjadi ON.

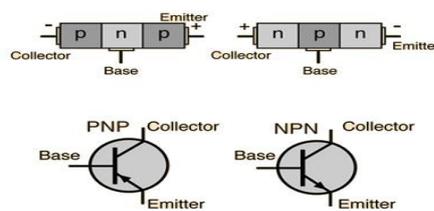
Rangkaian pada gambar B adalah *interface relay* yang menggunakan transistor teknik *emior folower* dimana *relay* diletakan pada kaki emitor trnasistor. Fungsi dioda yang dipasangkan pada rangkaian *interface* tersebut digunakan untuk menyerap tegangan induksi yang dihasilkan oleh *relay*.

Rangkaian pada gambar C merupakan teknik *inteface relay* ke rangkaian digital menggunakan transistor yang dirangkai secara *darlington*.



## 2.10 Transistor

Transistor adalah komponen elektronika yang tersusun dari bahan semikonduktor yang memiliki 3 kaki yaitu: basis (B), kolektor (C) dan emitor (E). Berdasarkan susunan semikonduktor yang membentuknya, transistor dibedakan menjadi dua tipe, yaitu PNP dan NPN. Untuk membedakan transistor PNP dan NPN dapat dari arah panah pada kaki emitornya. (Eko Purnomo,2010)



**Gambar 2.22** Gambar Transistor

(Sumber : <http://komponenelektronika/tipe-transistor/>)

### 2.10.1 Fungsi Transistor

1. Sebagai penguat amplifier
2. Sebagai pemutus dan penyambung (*switching*)
3. Sebagai pengatur stabilitas tegangan
4. Dapat menahan sebagian arus yang mengalir
5. Memperkuat arus dalam rangkaian
6. Sebagai pembangkit frekuensi rendah maupun tinggi

## 2.11 Motor DC

Motor DC adalah motor listrik yang memerlukan suplai tegangan arus searah pada kumparan medan untuk diubah menjadi energi gerak mekanik. Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Motor arus searah, sebagaimana namanya, menggunakan arus langsung yang tidak langsung/*direct-unidirectional*. (Rodwell International Corporation,1997)



Motor DC memiliki 3 bagian atau komponen utama untuk dapat berputar diantaranya adalah :

### 1. Kutub medan

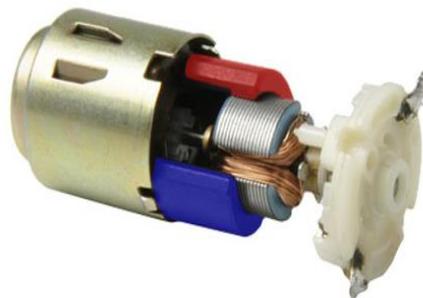
Motor DC sederhana memiliki dua kutub medan : kutub utara dan kutub selatan. Garis magnetik energi membesar melintasi ruang terbuka diantara kutub-kutub dari utara ke selatan. Untuk motor yang lebih besar atau lebih kompleks terdapat satu atau lebih elektromagnet.

### 2. *Current* Elektromagnet atau Dinamo

Dinamo yang berbentuk silinder, dihubungkan ke as penggerak untuk menggerakkan beban. Untuk kasus motor DC yang kecil, dinamo berputar dalam medan magnet yang dibentuk oleh kutub-kutub, sampai kutub utara dan selatan magnet berganti lokasi.

### 3. *Commutator*

Komponen ini terutama ditemukan dalam motor DC. Kegunaannya adalah untuk transmisi arus antara dinamo dan sumber daya.



**Gambar 2.23** Motor DC

([www.teknikelektronika.com](http://www.teknikelektronika.com))