BAB II

TEORI DASAR

2.1 Mikrokontroler AVR ATMEGA16

AVR merupakan seri mikrokontroler *Complementary Metal Oxide Semiconductor* (CMOS) 8-*bit* buatan Atmel berbasis arsitektur RISC *(Reduced Instruction Set Computer*). Hampir semua instruksi pada program dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 register *general-purpose*, *timer/counter* fleksibel dengan mode *compare*, interupsi *internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, *power saving mode*, ADC dan PWM. AVR pun mempunyai In-*System Programmable* (ISP) *Flash on-chip* yang mengijinkan memori program untuk diprogram ulang (*read/write*) dengan koneksi secara serial yang disebut *Serial Peripheral Inteface* (SPI).

AVR memilki keunggulan dibandingkan dengan mikrokontroler lain, keunggulan mikrokontroler AVR yaitu memiliki kecepatan dalam mengeksekusi program yang lebih cepat, karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock* (lebih cepat dibandingkan mikrokontroler keluarga MCS 51 yang memiliki arsitektur *Complex Intrukstion Set Compute*).

ATMEGA16 mempunyai *throughput* mendekati 1 *Millions Instruction Per Second* (MIPS) per MHz, sehingga membuat konsumsi daya menjadi rendah terhadap kecepatan proses eksekusi perintah.

Beberapa keistimewaan dari AVR ATMEGA16 antara lain:

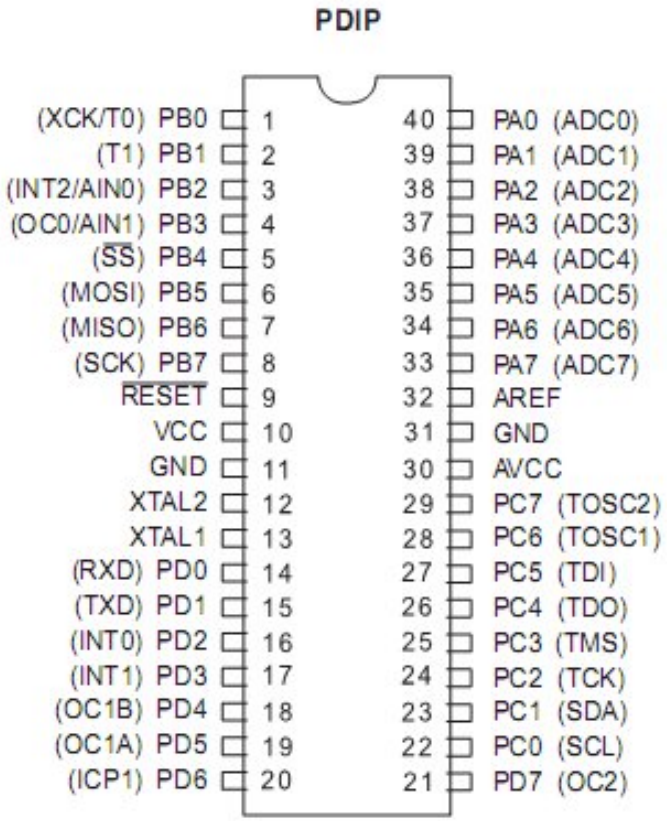
1. Mikrokontroler AVR 8 *bit* yang memilliki kemampuan tinggi dengan konsumsi daya rendah

2. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16MHz
3. Memiliki kapasitas *Flash* memori 16 Kbyte, EEPROM 512 Byte dan SRAM 1 Kbyte
4. Saluran I/O sebanyak 32 buah, yaitu *Port* A, *Port* B, *Port* C dan *Port* D
5. CPU yang terdiri dari 32 buah *register*
6. Unit interupsi dan eksternal
7. *Port* USART untuk komunikasi serial
8. Fitur *peripheral*
    - Tiga buah *Timer/Counter* dengan kemampuan perbandingan (*compare*)
        - Dua buah *Timer/Counter* 8 *bit* dengan *Prescaler* terpisah dan *Mode Compare*
        - Satu buah *Timer/Counter* 16 *bit* dengan *Prescaler* terpisah, *Mode Compare* dan *Mode Capture*
    - *Real Time Counter* dengan *Oscillator* tersendiri
    - Empat kanal PWM
    - 8 kanal ADC
        - 8 *Single-ended Channel* dengan keluaran hasil konversi 8 dan 10 resolusi (register ADCH dan ADCL)
        - 7 *Diferrential Channel* hanya pada kemasan *Thin Quad Flat Pack* (TQFP)
        - 2 *Differential Channel* dengan *Programmable Gain*
    - Antarmuka *Serial Peripheral Interface* (SPI) *Bus*
    - *Watchdog Timer* dengan *Oscillator Internal*

- *On-chip Analog Comparator*

9. *Non-volatile program memory*

2.1.1 Konfigurasi *Pin* AVR ATMEGA16



Gambar 2.1 Konfigurasi Kaki (*pin*) ATMEGA16

Konfigurasi *pin* ATMEGA16 dengan kemasan 40 *pin Dual In-line Package* (DIP) dapat dilihat pada Gambar 2.13. dari gambar diatas dapat dijelaskan fungsi dari masing-masing *pin* ATMEGA16 sebagai berikut.

1. VCC merupakan *pin* yang brfungsi sebagai masukan catu daya
2. GND merupakan *pin Ground*
3. *Port* A (PA0 – PA7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* masukan ADC

4. *Port* B (PB0 – PB7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus, seperti dapat dilihat pada tabel dibawah ini.

Tabel 2.1 Fungsi Khusus *Port* B

Pin	Fungsi Khusus
PB0	XCK (USART *External Clock Input/Output*) T0 (*Timer/Counter0 External Counter Input*)
PB1	T1 (*Timer/Counter1 External Counter Input*)
PB2	INT2 (*External Interupt 2 Input*) AIN0 (*Analaog Comparator Negative Input*)
PB3	OC0 (*Timer/Counter0 Output Compare Macth Output*) AIN1 (*Analaog Comparator Negative Input*)
PB4	(SPI *Slave Select Input*)
PB5	MOSI (SPI *Bus Master Output /Slave Input*)
PB6	MISO (SPI *Bus Master Input/Slave Output*)
PB7	SCK (SPI Bus Serial Clock)

5. *Port* A (PC0 – PC7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus, seperti dapat dilihat pada tabel dibawah ini.

Tabel 2.2 Fungsi Khusus *Port* C

Pin	Fungsi Khusus
PC0	SCL (*Two-wire Serial Bus Clock Line*)
PC1	SDA (*Two-wire Serial BusData Input/Output Line*)
PC2	TCK (*Joint Test Action Group Test Clock*)
PC3	TMS (*JTAG Test Mode Select*)
PC4	TDO (*JTAG Data Out*)
PC5	TDI (*JTAG Test Data In*)
PC6	TOSC1 (*Timer Oscillator pin 1*)
PC7	TOSC2 (*Timer Oscillator pin 2*)

6. *Port* D (PD0 – PD7) merupakan *pin input/output* dua arah (*full duplex*) dan selain itu merupakan *pin* khusus, seperti dapat dilihat pada tabel dibawah ini.

Tabel 2.3 Fungsi Khusus *Port* D

Pin	Fungsi Khusus
PD0	RXD (USART *Input Pin*)
PD1	TXD (USART *Output Pin*)
PD2	INT0 (*External Interupt 0 Input*)
PD3	INT1 (*External Interupt 1 Input*)
PD4	OC1B (*Timer/Counter1 Output Compare B Macth Output*)
PD5	OC1A (*Timer/Counter1 Output Compare A Macth Output*)
PD6	ICP (*Timer/Counter1 Input Capture Pin*)
PD7	OC2 (*Timer/Counter2 Output Compare Macth Output*)

7. RESET merupakan *pin* yang digunakan untuk me-*reset* mikrokontroler

8. XTAL1 dan XTAL2, merupakan *pin masukan external clock*
9. AVCC merupakan *pin* masukan tegangan untuk ADC
10. AREF merupakan *pin* masukan tegangan referensi untuk ADC.

2.1.2 *Timer/Counter* Mikrokontroler ATMEGA16

Mikrokontroler AVR ATMEGA16 memiliki tiga buah *Timer/Counter*, yaitu: *Timer* 0 (8 *bit*), *Timer* 1 (16 *bit*) dan *Timer* 2 (8 *bit*). Namun, pada sub bab ini hanya akan membahas mengenai *Timer/Counter* 1 saja. *Timer/Counter* 1 mempunyai keunggulan dibanding *Timer/Counter* 0 atau 2, namun cara mengatur *Timer 0, 1, 2* sama saja, yaitu pada masing-masing registernya. *Timer/Counter 1* dapat menghitung sampai dengan 65536 *Timer/Counter* 0 atau 2 hanya sampai dengan 256. Selain itu, *Timer* 1 ini memiliki mode operasi sebanyak 16 mode (Tabel 2.8). *Register* pada *Timer* ini dibagi menjadi beberapa *register* dengan fungsi khusus, yaitu: *control register* A, *control register* B dan *interrupt mask*.

Register – register pada *Timer/Counter* 1 yang berfungsi untuk mengatur timer dan mode operasinya. Register tersebut mempunyai fungsi masing-masing sebagai berikut.

a. *Timer/Counter* 1 *Control Register* A (TCCR1A)

Tabel 2.4 *Register* TCCR1A

Bit	7	6	5	4	3	2	1	0
TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10

Keterangan:

Bit 7 dan 6 : Compare Output untuk kanal A

Bit 5 dan 4 : Compare Output untuk kanal B

Bit COM1 ini mempunyai *Compare Output Mode* pada setiap mode operasinya. Mode tersebut mempengaruhi *pin* I/O OC1 A dan B.

Tabel 2.5 *Compare Output Mode*, *Non*-PWM

COM1A1/COM1B1	**COM1A1/COM1B1**	**Deskripsi**
0	0	*Normal Port Operation*, OC1A/OC1B *disconnected*
0	1	*Toggle* OC1A/OC1B *on compare match*
1	0	*Clear* OC1A/OC1B *on compare match* (*low level*)
1	1	*Set* OC1A/OC1B *on compare match* (*high level*)

Tabel 2.6 *Compare Output Mode*, *Fast* PWM

COM1A1/COM1B1	**COM1A1/COM1B1**	**Deskripsi**
0	0	*Normal Port Operation*, OC1A/OC1B *disconnected*
0	1	*Toggle* OC1A *on compare match,* OC1B *disconnected*
1	0	*Clear* OC1A/OC1B *on compare match, set* OC1A/OC1B *at* BOTTOM (*non-inverting mode*)
1	1	*Set* OC1A/OC1B *on compare match, clear* OC1A/OC1B *at* BOTTOM (*inverting mode*)

Tabel 2.7 *Compare Output Mode*

Phase Correct dan *Phase Correct & Frequency* PWM

COM1A1/COM1B1	**COM1A1/COM1B1**	**Deskripsi**
0	0	*Normal Port Operation*, OC1A/OC1B *disconnected*
0	1	*Toggle* OC1A *on compare match,* OC1B *disconnected*
1	0	*Clear* OC1A/OC1B *on compare match when up-counting, set*
		OC1A/OC1B *on compare match when down-counting*
1	1	*Clear* OC1A/OC1B *on compare match when up-counting, set* OC1A/OC1B *on compare match when down-counting*

Bit 3 : *Force Output* untuk kanal A

Bit 2 : *Force Output* untuk kanal B

Bit 1 dan 0 : *Waveform Generation Mode*

Mode operasi sebanyak 16 mode, diatur dalam *bit* WGM ini. Mode operasi tersebut ditunjukkan oleh Tabel 2.8 di bawah ini.

Tabel 2.8 Deskripsi *Bit* WGM

Mode	WGM13	WGM12 (CTC1)	WGM11 (PWM11)	WGM10 (PWM10)	Mode Operasi	TOP	Update of OCRn	TOVn Flag Set-on
0	0	0	0	0	Normal	0xFFFF	*Immediate*	MAX
1	0	0	0	1	PWM, *Phase Correct* 8-*Bit*	0x00FF	*TOP*	*BOTTOM*
2	0	0	1	0	PWM, *Phase Correct* 9-*Bit*	0x01FF	*TOP*	*BOTTOM*
3	0	0	1	1	PWM, *Phase Correct* 10-*Bit*	0x03FF	*TOP*	*BOTTOM*
4	0	1	0	0	CTC	OCR1A	*Immediate*	MAX
5	0	1	0	1	*Fast* PWM, 8-*Bit*	0x00FF	*BOTTOM*	*TOP*
6	0	1	1	0	*Fast* PWM, 9-*Bit*	0x01FF	*BOTTOM*	*TOP*
7	0	1	1	1	*Fast* PWM, 10-*Bit*	0x03FF	*BOTTOM*	*TOP*
8	1	0	0	0	PWM, *Phase and Frequency Correct*	ICR1	*BOTTOM*	*BOTTOM*
9	1	0	0	1	PWM, *Phase and Frequency Correct*	OCR1A	*BOTTOM*	*BOTTOM*
10	1	0	1	0	PWM, *Phase Correct*	ICR1	*TOP*	*BOTTOM*
11	1	0	1	1	PWM, *Phase Correct*	OCR1A	*TOP*	*BOTTOM*
12	1	1	0	0	CTC	ICR1	*Immediate*	MAX
13	1	1	0	1	*Reserved*	-	-	-
14	1	1	1	0	*Fast* PWM	ICR1	*BOTTOM*	*TOP*
15	1	1	1	1	*Fast* PWM	OCR1A	*BOTTOM*	*TOP*

b. *Timer/Counter Control Register* 1 B (TCCR1B)

Tabel 2.9 TCCR1B

Bit	7	6	5	4	3	2	1	0
TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10

Keterangan:

Bit 7 : *Input Capture Noise Canceler*, ketika *bit* ini diset 1(*high*) maka *Noise Canceler* aktif dan masukkan dari *Input Capture Pin* (ICP1) terfilter.

Bit 6 : *Input Capture Edge Select*, *bit* ini digunakan untuk *trigger* yang disebabkan oleh *edge* ICP1. Jika *bit* ini diset 1 maka sebuah *rising edge* (positif) akan men-*trigger capture,* Jika *bit* ini diset 0 maka sebuah *falling edge* (negatif) akan men-*trigger capture.*

Bit 5 : *Reserved*, *bit* ini akan digunakan pada tahap pengembangan selanjutnya.

Bit 4 dan 3 : lihat deskripsi *register* TCCR1A.

Bit 2, 1 dan 0 : *Clock Select*, *bit* ini digunakan untuk memilih jenis sumber *clock* untuk digunakan pada suatu *timer/counter.*

Tabel 2.10 Deskripsi *Clock Select Bit*

CS12	**CS11**	**CS10**	Deskripsi
0	0	0	Tidak ada *clock* (Timer/Counter terhenti)
0	0	1	CLK_I/O/1 (tanpa *Prescaling*)
0	1	0	CLK_I/O/8 (dari *Prescaling*)
0	1	1	CLK_I/O/64 (dari *Prescaling*)
1	0	0	CLK_I/O/256 (dari *Prescaling*)
1	0	1	CLK_I/O/1024 (dari *Prescaling*)
1	1	0	Sumber *clock* (eksternal)berasal dari *pin* T1, *clock* pada *falling edge*
1	1	1	Sumber *clock* (eksternal)berasal dari *pin* T1, *clock* pada *rising edge*

c. TCNT1, digunakan untuk menyimpan nilai *timer* yang diinginkan. TCNT1 dibagi menjadi 2 *register* 8 *bit*, yaitu TCNT1H dan TCNT1L.

Tabel 2.11 *Register* TCNT1

Bit	7	6	5	4	3	2	1	0
TCNT1H	**TCNT1[15:8]**							
TCNT1L	**TCNT1[7:0]**							

d. TIMSK dan TIFR, *Timer Interrupt Mask Register* (TIMSK) dan *Timer Interrupt Flag* (TIFR) digunakan untuk mengendalikan *interrupt* mana yang diaktifkan, dengan cara melakukan *setting* pada TIMSK dan untuk mengetahui *interrupt* mana yang sedang terjadi.

Tabel 2.12 *Register* TIMSK

Bit	7	6	5	4	3	2	1	0
TIMSK	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0

Keterangan:

Bit 7 : *Timer/Counter*2 *Output Compare Match Interrupt Enable*

Bit 6 : *Timer/Counter*2 *Overflow Interrupt Enable*

Bit 5 : *Timer*1 *Input Capture Interrupt Enable*

Bit 4 : *Timer/Counter*A *Output Compare Match Interrupt Enable*

Bit 3 : *Timer/Counter*B *Output Compare Match Interrupt Enable*

Bit 2 : *Timer/Counter*1 *Overflow Interrupt Enable*

Bit 1 : *Timer/Counter*0 *Output Compare Match Interrupt Enable*

Bit 0 : *Timer/Counter*0 *Overflow Interrupt Enable*

Tabel 2.13 *Register* TIFR

Bit	7	6	5	4	3	2	1	0
TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0

Keterangan:

Bit 7 : *Output Compare Flag2*

Bit 6 : *Timer/Counter*2 *Overflow Flag*

Bit 5 : *Timer*1 *Input Capture Interrupt Flag*

Bit 4 : *Output Compare Flag*1A

Bit 3 : *Output Compare Flag*1B

Bit 2 : *Timer/Counter*1 *Overflow Flag*

Bit 1 : *Output Compare Flag*0

Bit 0 : *Timer/Counter*0 *Overflow Flag*

e. OCR1n, *Output Compare Register Timer* 1 n (n = A, B) merupakan register yang digunakan untuk membangkitkan interupsi eksternal dengan melakukan perbandingan (*Output Compare*) atau juga dapat digunakan untuk membangkitkan bentuk gelombang (PWM). Fungsi tersebut di atas dikeluarkan oleh *pin* OC1n (n = A, B).

Tabel 2.14 *Register* OCR1n

Bit	7	6	5	4	3	2	1	0
OCR1nH	OCR1n[15:8]							
OCR1nL	OCR1n[7:0]							

Setiap mode *timer* seperti CTC, *Fast* PWM, *Phase Correct* PWM dan *Phase and Frequency Correct* PWM, mempunyai persamaan untuk menghitung frekuensi (*clock*) yang akan dihasilkannya.

a. *Clear Timer On Compare Match* (CTC)

$$f_{OCnA} = \frac{}{\cdot \cdot (\quad)} \quad \text{.....(2.1)}$$

Dimana:

f_{OCnA} = frekuensi mode CTC pada OCRnA (n: H atau L)

$f_{clk_I/O}$ = frekuensi kristal yang digunakan

N = variable *prescaler factor* (1, 8, 64, 256 atau 1024)

OCRnA = nilai register OCRnA (n: 1 atau 2)

b. *Fast* PWM

$$R_{FPWM} = \frac{(\quad)}{(\quad)} \quad \text{.....(2.2)}$$

Dimana:

R_{FPWM} = Resolusi mode *Fast* PWM

TOP = nilai maksimal pada register TCCRn (n: A atau B)

$$f_{OCnxPWM} = \frac{\quad}{\cdot(\quad)} \quad \text{.....(2.3)}$$

Dimana:

$f_{OCnxPWM}$ = frekuensi mode *Fast* PWM pada OCRnx (n: H atau L; x: A/B)

$f_{clk_I/O}$ = frekuensi kristal yang digunakan

N = variable *prescaler factor* (1, 8, 64, 256 atau 1024)

TOP = nilai maksimal pada register TCCRn (n: A atau B)

c. *Phase Correct* PWM

$R_{PCPWM} = R_{FPWM}$ .....(2.4)

Dimana: R_{PCPWM}= resolusi mode *Phase Correct* PWM

$$f_{OCnxPCPWM} = \frac{\quad}{\cdot\ \cdot} \quad \text{.....(2.5)}$$

Dimana:

$f_{OCnxPCxPWM}$ = frekuensi mode *Phase Correct* PWM pada OCRnx (n: H atau L; x: A atau B)

$f_{clk_I/O}$ = frekuensi kristal yang digunakan

N = variable *prescaler factor* (1, 8, 64, 256 atau 1024)

TOP = nilai maksimal pada register TCCRn (n: A atau B)

d. *Phase and Frequency Correct* PWM

$R_{PFCPWM} = R_{FPWM}$ ..... (2.6)

Dimana: R_{PFCPWM} = resolusi mode *Phase and Frequency Correct* PWM

$f_{OCnxPFCPWM} = f_{OCnxPCPWM}$ ..... (2.7)

Dimana: R_{PCPWM} = frekuensi mode *Phase and Frequency Correct* PWM pada OCRnx (n: H atau L; x: A atau B)

2.1.3 Interupsi Eksternal

Interupsi eksternal merupakan fitur tambahan dari mikrokontroler AVR ATMEGA 16 yang khusus difungsikan untuk interupsi. Interupsi eksternal adalah jenis interupsi asinkron yang pengaktifannya bukan dipicu dari fitur: timer/counter, ADC, komparator analog ataupun dari komunikasi antarmuka, tetapi dipicu secara logika dari luar mikrokontroler (eksternal). mikrokontroler AVR ATMEGA 16 mempunyai 3 buah pemicu interupsi eksternal, yaitu pada *pin* INT0 (PORTD2), *pin* INT1 (PORTD3), *pin* INT2 (PORTB2). Interupsi eksternal tersebut dapat diaktifkan dengan cara mengatur register - register sebagai berikut.

Tabel 2.15 *Register* MCUCR

Bit	7	6	5	4	3	2	1	0
MCUCR	SM2	SE2	SM1	SM0	ISC11	ISC10	ISC01	ISC00

Keterangan:

a. ISC11 dan ISC10: *Interrupt Sense Control* 1 Bit 1 and Bit 0

Tabel 2.16 *Interrupt Sense Control* 1

ISC11	ISC10	**Deskripsi**
0	0	Interupsi terjadi jika pada pin INT1 berlogika “0” (*low*)
0	1	Interupsi terjadi setiap perubahan logika pada pin INT1
1	0	Interupsi terjadi setiap *falling edge* pada pin INT1
1	1	Interupsi terjadi setiap *rising edge* pada pin INT1

b. ISC01 dan ISC00: *Interrupt Sense Control* 0 Bit 1 and Bit 0

Tabel 2.17 *Interrupt Sense Control* 0

ISC01	ISC00	**Deskripsi**
0	0	Interupsi terjadi jika pada pin INT0 berlogika “0” (*low*)
0	1	Interupsi terjadi setiap perubahan logika pada pin INT0
1	0	Interupsi terjadi setiap *falling edge* pada pin INT0
1	1	Interupsi terjadi setiap *rising edge* pada pin INT0

Tabel 2.18 *Register* MCUCSR

Bit	7	6	5	4	3	2	1	0
MCUCR	-	ISC2	-	-	WDRF	BORF	EXTRF	PORF

Keterangan:

ISC02: *Interrupt Sense Control* 2

Ketiga Interupsi eksternal ini akan aktif apabila *bit*-I pada *register* SREG dan GICR diberi logika “1” (*high*). Lebar pulsa minimum pada interupsi eksternal asinkron ini sebesar 50nS.

Tabel 2.19 *Register* GICR

Bit	7	6	5	4	3	2	1	0
MCUCR	INT1	INT0	INT2	-	-	-	IVSEL	IVCE

Keterangan:

INT1, INT0, INT2: *Eksternal Interrupt Request* 1, 0 atau 2 *Enable*

2.1.4 *Prescaler*

Pada dasarnya *Timer* hanya menghitung pulsa *clock*. Frekuensi pulsa *clock* yang dihitung tersebut bias sama dengan frekuensi Kristal yang diginakan atau dapat diperlambat menggunakan *prescaler* dengan faktor 1, 8, 64, 256 atau 1024. Untuk memahami *prescaler* ini, berikut contoh penggunaan *prescaler* dalam menentukan waktu suatu timer.

Contoh:

Sebuah AVR menggunakan kristal dengan frekuensi 8 MHz dengan *timer* yang digunakan adalah *timer* 16 *bit*, maka maksimum waktu *timer* tersebut adalah sebesar:

TMAX = ⁄ × (h) 1 ..... (2.8)

= 0,125 × 65536 = 0,008192

Untuk menghasilkan waktu *timer* yang lebih lama, dapat digunakan *prescaler*, misalnya 1024. Maka maksimum waktu *timer* tersebut adalah

TMAX = ⁄ × (h) 1 ..... (2.9)

= 0,125 × 65536 × 1024 = 8,388608

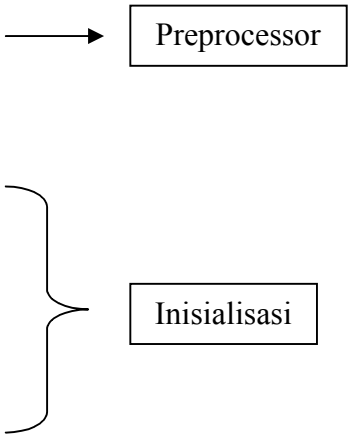
2.1.5 Pemrograman Mikrokontroler ATMEGA16

Pengembangan sebuah system menggunakan mikrokontroler AVR buatan ATMEL menggunakan *software* AVR STUDIO dan CodeVision AVR. AVR STUDIO merupakan *software* khusus untuk bahasa *assembly* yang mempunyai fungsi sangat lengkap, yaitu digunakan untuk menulis program, kompilasi, simulasi dan *download* program ke IC mikrokontroler AVR dapat dilakukan pada CodeVision. CodeVision AVR memilki fasilitas terminal, yaitu untuk melakukan komunikasi serial dengan mikrokontroler yang sudah deprogram. Proeses *download* program ke IC mikrokontroler AVR dapat menggunakan *system download* secara In-System Programming (ISP). ISP *Flash On-chip* mengijinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial SPI.

2.1.5.1 Bahasa *Assembly* AVR

Bahasa *assembly* AVR merupakan bahasa asli/mesin yang berupa instruksi – instruksi tertentu sesuai dengan ketentuan pabrikan ATMEL. Di bawah ini merupakan contoh sebuah penulisan program dalam bahasa *assembly* untuk mikrokontroler AVR ATMEGA16.

```
.include “C:\Appmotes\m16def.inc”
.org 0x0000
Rjmp main
Main:
    ldi r16,low(RAMEND)
    out       SPI,r16
    ldir16,high(RAMEND)
    out       SPH,r16
    ldir16,0xff
```



```
out     ddrc,r16
out     PORTC,r16
out     ddrd,r16
out     PORTD,r16
henti:
cbi     PORTD,5
cbi,    PORTC,0
cbi     PORTC,1
nop
rjmp    henti
```

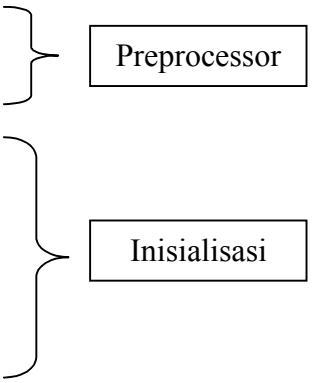
Setiap program terdiri dari inisialisasi program dan program utama. Inisialisasi program berisi definisi *chip* yang digunakan, mendefinisikan nama variable, konstanta, alamat awal program, *stack pointer*.

2.1.5.2 Bahasa C Pada AVR ATMEGA16

Mikrokontroler AVR dapat pula menggunakan bahasa C dalam penulisan programnya, sehingga dapat memudahkan dan mempersingkat instruksi – intruksi yang digunakan dalam bahasa *assembly.* Dalam pembuatan program yang menggunakan fungsi atau aritmatika, Bahasa C menawarkan kemudahan dengan menyediakan fungsi – fungsi khusus, seperti: pembuatan konstanta, operator aritmatika, operatot logika, operator *bitwise* dan operator *Assigment.* Selain itu, bahasa C menyediakan Program kontrol seperti: Percabangan (*if* dan *if…else*), Percabangan switch, *Looping* (*for, while* dan *do…while*), *Array,* serta fungsi – fungsi lainnya. Di bawah ini merupakan contoh penulisan program dalam bahasa C untuk mikrokontroler AVR ATMEGA16.

```
//Preprocessor
#include      <mega16.h>
#include      <delay.h>
# define      tachometer    PINA.0
# define      motor_AC      PORTB.0
//variable global
Unsigned int i,j;
void main(void)

{
//inisialisasi port, timer, dsb.
Char data_rx;
DDRA=0x00;
PORTA=0xFF;
DDRB=0xFF;
PORTB=0x00;
…
…
…
While(1)
{
        for (i=0; i<=255; i ++)
                {
                if (tachometer==0)
                {motor_AC = 1;}
                }
};
}
```



Preprocessor digunakan untuk memasukkan (*include*) *text* dari *file* lain, mendefinisikan *macro* yang dapat mengurangi beban kerja pemrograman dan

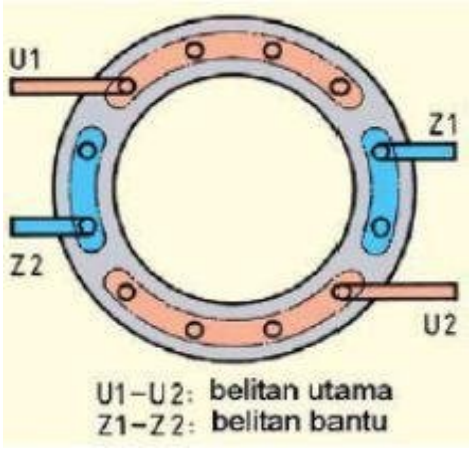
meningkatkan *legability source code* (mudah dibaca). Inisialisasi merupakan pengaturan awal yang akan dibutuhkan dalam membuat suatu program.

2.2 Motor Listrik AC Satu Fasa

Motor listrik AC Satu Fasa termasuk kedalam kategori mesin listrik dinamis dan merupakan sebuah perangkat elektromagnetik yang mengubah energi listrik menjadi energi mekanik. Energi mekanik ini banyak digunakan pada pemutar *impeller* pompa, *blower*, penggerakan kompresor, pengangkat beban, mixer, bor listrik, kipas angin, dll. Motor listrik dibagi menjadi dua jenis berdasarkan arus listriknya, yaitu: Motor arus bolak balik (AC) dan motor arus searah (DC). Berdasarkan karakteristik dari arus listrik, motor AC terdiri dari 2 jenis, yaitu: Motor listrik AC/arus bolak-balik 1 fasa dan Motor listrik AC/arus bolak-balik 3 fasa. Pembahasan teori dasar motor listrik AC dalam tugas akhir ini menitik beratkan pada motor listrik AC 1 fasa berjenis Motor Kapasitor.

2.2.1 Prinsip kerja Motor AC Satu Fasa

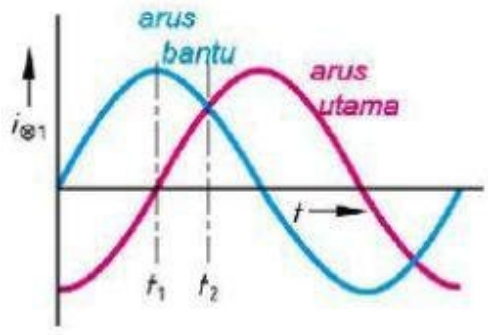
Cara kerja motor AC satu fasa berbeda dengan motor AC tiga fasa, dimana pada belitan stator motor AC tiga fasa terdapat tiga buah belitan yang menghasilkan medan putar, serta pada rotor sangkar terjadi induksi dan interaksi torsi yang menghasilkan putaran, sedangkan pada motor satu fasa memiliki dua belitan stator, yaitu belitan fasa utama (belitan U1-U2) dan belitan fasa bantu (belitan Z1-Z2), yang dapat dilihat pada Gambar 2.2.



Gambar 2.2 Belitan Stator Fasa Utama dan Belitan Stator Fasa Bantu

Belitan utama menggunakan penampang kawat tembaga lebih besar daripada penampang kawat belitan bantu, sehingga memiliki impedansi lebih kecil. Sedangkan belitan bantu terbuat dari tembaga berpenampang kecil dan jumlah belitannya lebih banyak, sehingga impedansinya lebih besar dibanding impedansi belitan utama.

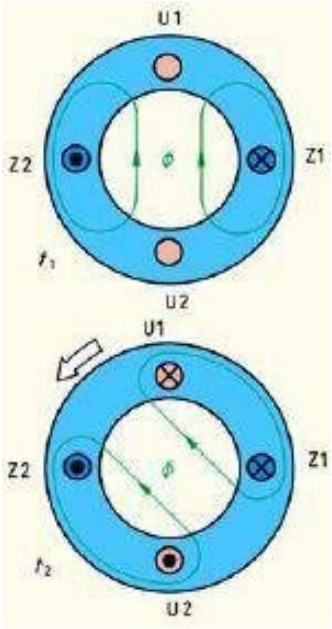
Grafik arus belitan bantu dan arus belitan utama berbeda fasa sebesar φ, hal ini disebabkan karena perbedaan besarnya impedansi kedua belitan tersebut. Perbedaan arus beda fasa ini menyebabkan arus total, dimana arus total merupakan penjumlahan vektor arus utama dan vektor arus bantu. Medan magnet utama yang dihasilkan belitan utama juga berbeda fasa sebesar dengan medan magnet bantu.



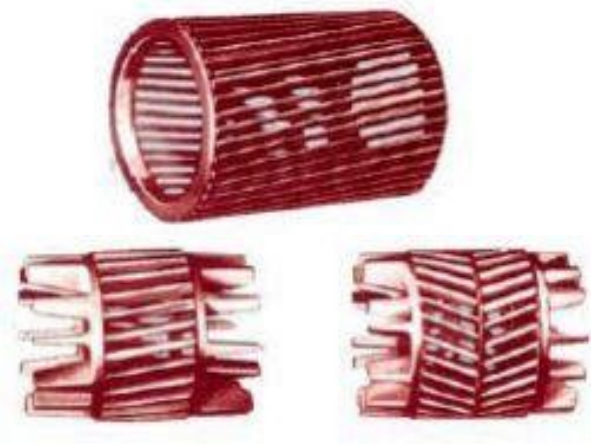
Gambar 2.3 Grafik Gelombang Arus Medan Bantu Dan Arus Medan Utama

Belitan bantu Z1-Z2 pertama-tama dialiri arus bantu menghasilkan fluks magnet tegak lurus, beberapa saat kemudian belitan utama U1-U2 dialiri arus utama yang bernilai positip. Hasilnya adalah medan magnet yang bergeser sebesar 45° dengan arah berlawanan jarum jam. Kejadian ini berulang setiap satu siklus sinusoida, sehingga menghasilkan medan magnet yang berputar pada belitan statornya.

Rotor motor satu fasa sama dengan rotor motor tiga fasa yaitu berbentuk batang - batang kawat yang pada bagian ujungnya dihubung singkatkan dan menyerupai bentuk sangkar tupai, maka sering disebut rotor sangkar tupai.



Gambar 2.4 Medan Magnet Pada Stator Motor Satu Fasa



Gambar 2.5 Rotor Sangkar

Belitan rotor yang dipotong oleh medan putar stator menghasilkan tegangan induksi, interaksi antara medan putar stator dan medan magnet rotor akan menghasilkan torsi putar pada rotor.

2.2.2 Motor Kapasitor

Motor kapasitor satu fasa banyak digunakan dalam peralatan rumah tangga seperti motor pompa air, motor mesin cuci, motor lemari es, motor *Air Conditioning*. Konstruksinya sederhana dengan daya kecil dan bekerja dengan tegangan suplai PLN 220VAC, oleh karena itu menjadikan motor kapasitor ini banyak digunakan pada peralatan rumah tangga.



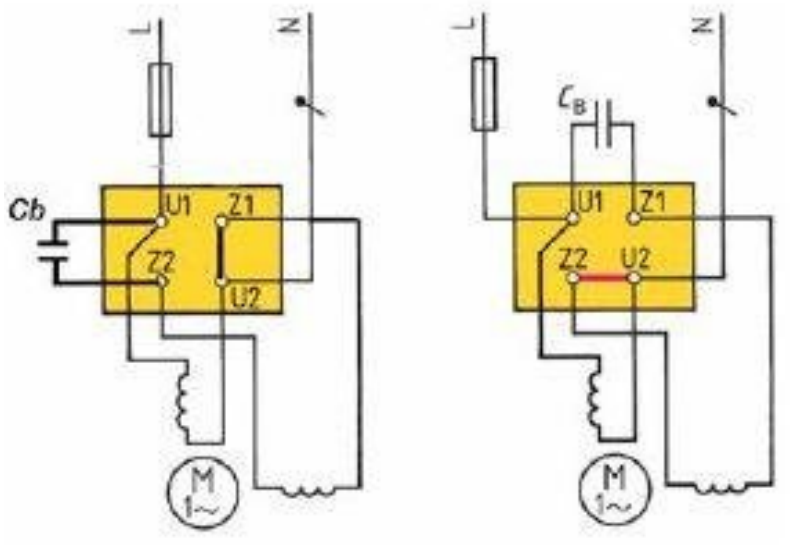
Gambar 2.6 Motor Kapasitor

Belitan stator terdiri atas belitan utama dengan notasi terminal U1-U2, dan belitan bantu dengan notasi terminal Z1-Z2. Jala-jala L1 terhubung dengan

terminal U1 dan kawat netral N terhubung dengan terminal U2. kapasitor kerja (Cb) berfungsi agar perbedaan sudut phasa belitan utama dengan belitan bantu mendekati 90°.

Pengaturan arah putaran motor kapasitor dapat dilakukan dengan:

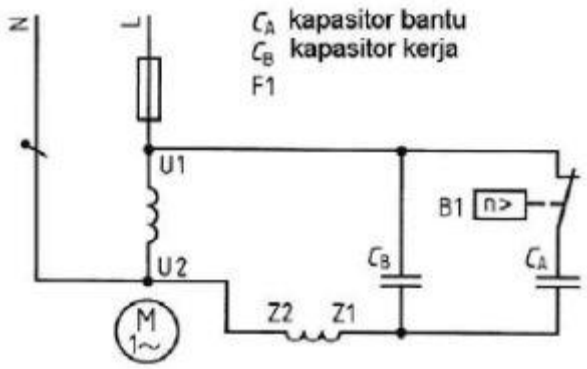
- Untuk menghasilkan putaran ke kiri (berlawanan jarum jam) kapasitor kerja CB disambungkan ke terminal U1 dan Z2 dan terminal Z1 dikopel dengan terminal U2.
- Untuk menghasilkan putaran ke kanan (searah jarum jam) kapasitor kerja disambung kan ke terminal U1 dan Z1 dan terminal Z2 dikopel dengan terminal U2.



Gambar 2.7 Pengawatan Motor Kapasitor Dengan Pembalik Putaran

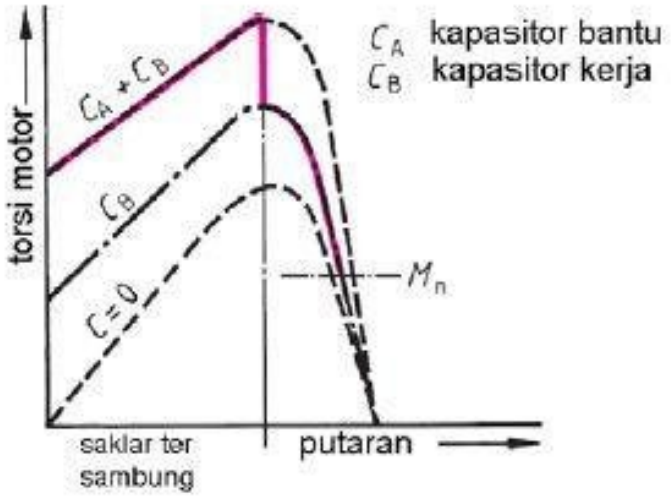
Motor kapasitor dengan daya diatas 1 KW di lengkapi dengan dua buah kondensator dan satu buah saklar sentrifugal. Belitan utama U1-U2 dihubungkan langsung dengan jala-jala L1 dan Netral N. Belitan bantu Z1-Z2 dihubungkan secara seri dengan kapasitor kerja Cb dan kontak *Normally Close* (NC) dari saklar sentrifugal terhubung seri dengan sebuah kapasitor *starting* Ca (Gambar 2.7).

Pada saat pertama kali motor mendapatkan tegangan, buah kapasitor Cb dan Ca membentuk loop tertutup sehingga rotor mulai berputar, ketika putaran mendekati 70% putaran normalnya saklar sentrifugal akan membuka, sehingga memutuskan kapasitor bantu Ca.



Gambar 2.8 Pengawatan Dengan Dua Kapasitor

Fungsi dari dua kondensator yang disambungkan secara paralel adalah untuk meningkatkan nilai torsi awal pada saat mengangkat beban. Setelah putaran motor mencapai 70% putaran, saklar sentrifugal terputus sehingga hanya kondensator kerja CB saja yang tetap bekerja. Jika kedua kondensator rusak maka torsi motor akan menurun drastis (Gambar 2.11).



Gambar 2.9 Karakteristik Torsi Motor Kapasitor

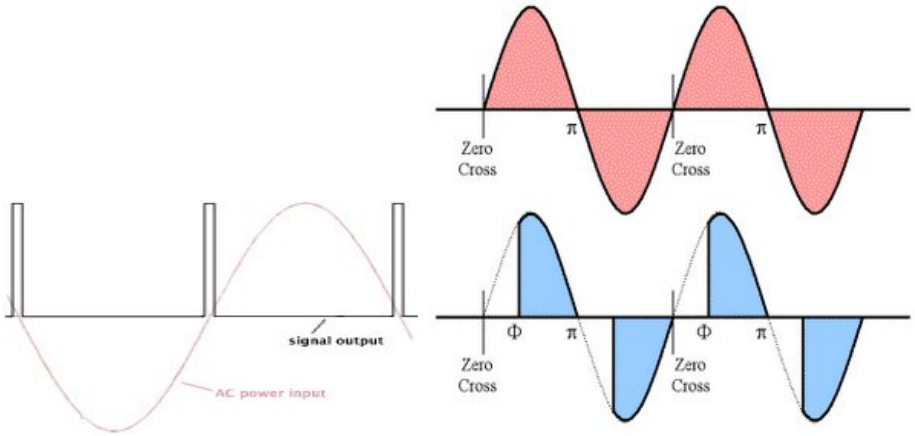
2.3 *Driver* Motor AC

Motor AC bekerja pada saat siklus tegangan positif dan siklus tegangan negatif (gelombang sinus). Siklus – siklus tersebut mempunyai periode/frekuensi tertentu atau dengan kata lain mempunyai lebar pulsa pada tiap siklus/polaritas tegangannya. Lebar pulsa pada tiap siklus tersebut dapat diatur dan dimanfaatkan untuk pengaturan kecepatan motor AC. Cara mengatur lebar pulsa dari suatu frekuensi dikenal dengan metode *Pulse Width Modulation* (PWM). Dengan menggunakan prinsip PWM ini, lebar pulsa dari frekuensi tegangan AC tersebut dapat diatur dengan syarat panjang periode (frekuensi) dan titik nol (*zero point*) dari gelombang AC tersebut dideteksi terlebih dahulu. Sinyal PWM akan memicu/mengatur lebar pulsa setiap siklus tegangan AC, dimulai dari perpotongan titik nol pada tegangan AC tersebut.

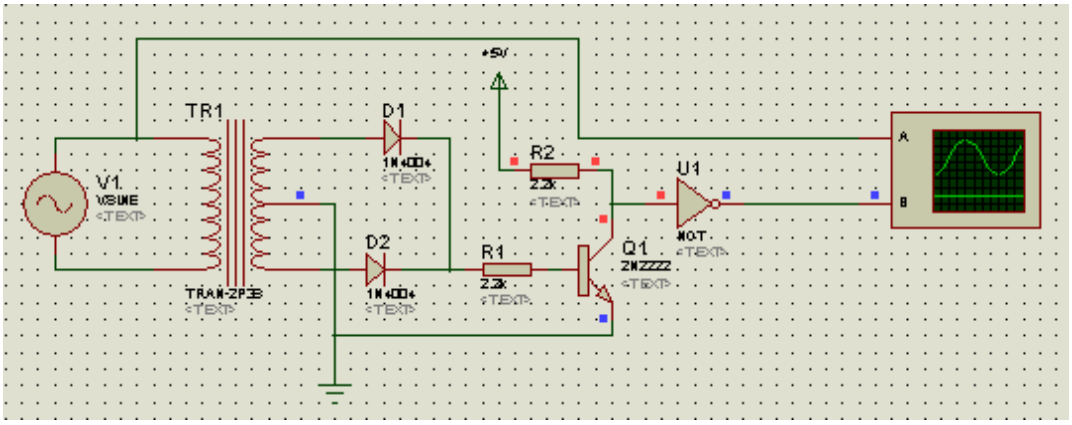
2.3.1 Metode *Zero Crossing Detection*

Metode *zero crossing detection* adalah metode paling umum untuk mengetahui frekuensi/periode suatu gelombang. Metode ini berfungsi untuk menentukan frekuensi suatu gelombang dengan cara mendeteksi banyaknya *zero point* pada suatu rentang waktu. *Zero crossing detector* adalah rangkaian yang berfungsi untuk mendeteksi perpotongan gelombang sinus pada tegangan AC dengan *zero point* tegangan AC tersebut, sehingga dapat memberikan sinyal acuan saat dimulainya pemicuan sinyal PWM. Dengan menggunakan rangkaian *zero crossing detector* ini, kita dapat mendeteksi *zero point* sekaligus mengubah suatu sinyal sinusoidal (*sine wave*) menjadi sinyal kotak (*square wave*). Perpotongan titik nol yang dideteksi adalah pada saat peralihan dari siklus positif menuju siklus negatif dan peralihan dari siklus negatif menuju siklus positif.

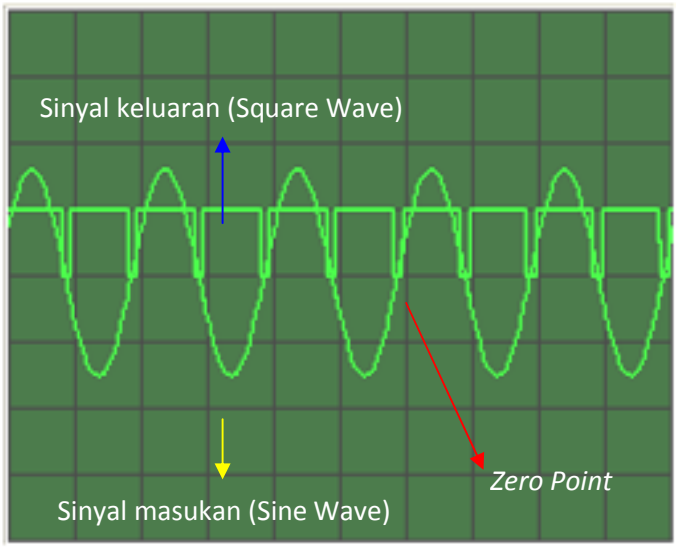
Dalam rangkaian *driver* motor ini, sinyal acuan (*zero point*) akan digunakan sebagai interupsi eksternal mikrokontroler dan selanjutnya mikrokontroler akan mengatur dan membangkitkan sinyal PWM untuk memicu *gate* TRIAC *Opto Osillator*.



Gambar 2.10 Prinsip kerja Metode *Zero Cross Detection*



Gambar 2.11 Contoh Rangkaian *Zero Cross Detector*



Gambar 2.12 Hasil Deteksi *Zero Point* Oleh Rangkaian *Zero Cross Detector*

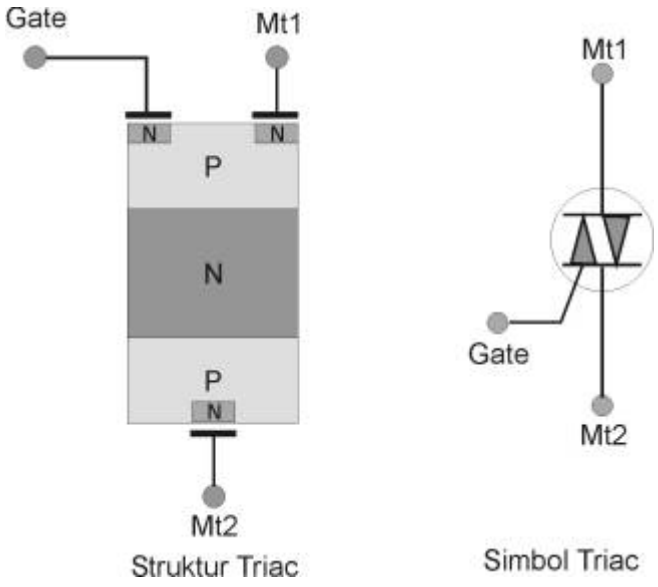
2.3.2 TRIAC

Pada sub bab ini membahas tentang pengertian TRIAC, karakteristik TRIAC, penggunaan TRIAC dan TRIAC jenis *Optoisolators*.

2.3.2.1 Pengertian TRIAC

TRIAC merupakan singkatan dari *TRIode Alternating Current*, yang artinya adalah saklar *triode* untuk arus bolak-balik. TRIAC adalah pengembangan

dari pendahulunya yaitu *Diode Alternating Current* (DIAC) dan *Silicon Control Rectifier* (SCR). Ketiganya merupakan sub-jenis dari Thyristor, piranti berbahan silikon yang umum digunakan sebagai saklar elektronik, disam*pin*g transistor dan *Field Effect Transistor* (FET). Perbedaan diantara ketiganya adalah dalam penggabungan unsur-unsur penyusunnya, serta dalam segi arah penghantaran arus listrik yang melaluinya. TRIAC sebenarnya adalah gabungan dua buah SCR atau *Thyristor* yang dirancang anti paralel dengan satu buah elektroda gerbang (*gate electrode*) yang menyatu. SCR merupakan piranti zat padat (*solid state*) yang berfungsi sebagai sakelar daya berkecepatan tinggi.

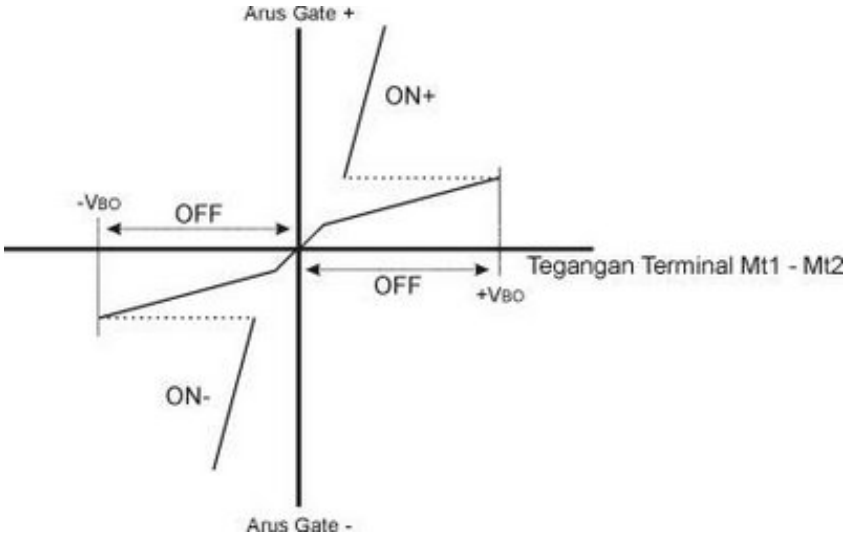


Gambar 2.13 Sruktur dan Simbol TRIAC

2.3.2.2 Karakteristik TRIAC

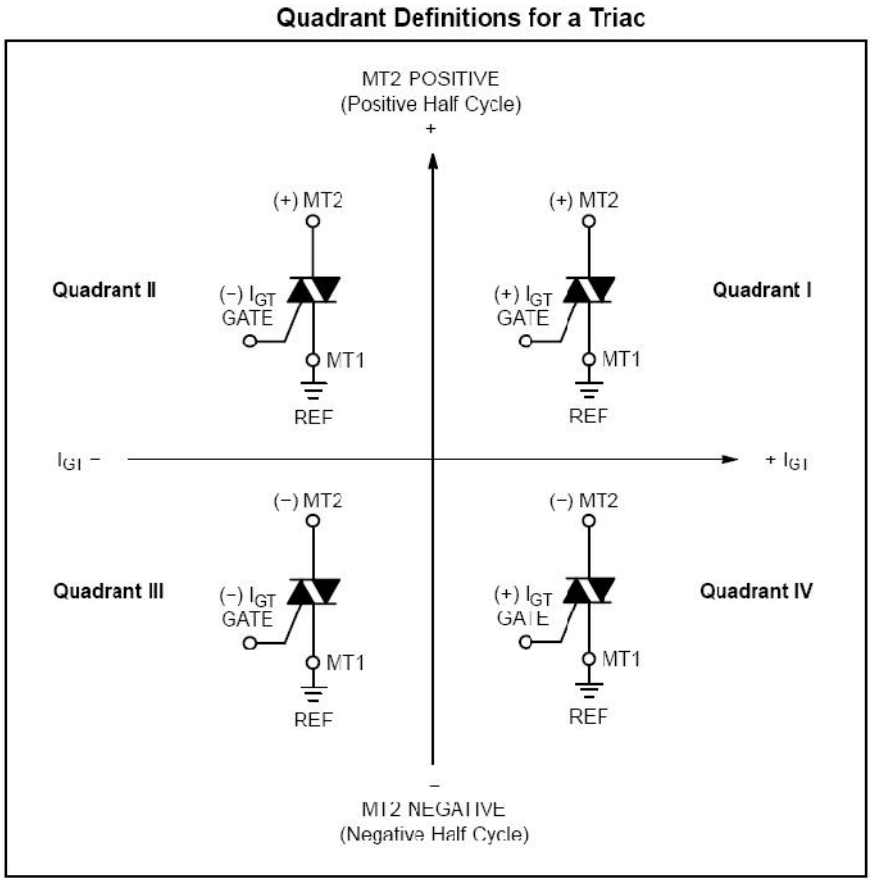
TRIAC memiliki karakteristik *swicthing* seperti pada SCR, kecuali bahwa TRIAC dapat berkonduksi dalam berbagai arah. TRIAC dapat digunakan untuk mengontrol aliran arus dalam rangkaian AC. Elemen seperti penyearah dalam dua arah menunjukkan kemungkinan dua aliran arus antara terminal utama

M1 dan M2. Pengaturan dilakukan dengan memberi sinyal antara *gate* (gerbang) dan M1.



Gambar 1.14 Karakteristik TRIAC

Karena dapat bersifat konduktif dalam dua arah, biasanya TRIAC digunakan untuk mengendalikan fasa arus AC. Selain itu, karena TRIAC merupakan *bidirectional device*, terminalnya tidak dapat ditentukan sebagai anode atau katode. Jika terminal MT2 positif terhadap terminal MT1, TRIAC dapat dimatikan dengan memberikan sinyal gerbang positif antara gerbang *Gate* dan MT1, sebaliknya jika terminal MT2 negatif terhadap MT1 maka TRIAC akan dapat dihidupkan dengan memberikan sinyal pulsa negatif antara gerbang G dan terminal MT1.



Gambar 2.15 Kuadran Operasi TRIAC

Dalam kenyataannya, sensitifitas bervariasi antara satu kuadran dengan kuadran lain dan TRIAC biasanya beroperasi di kuadran I+ (tegangan dan arus gerbang positif) atau kuadran III- (tegangan dan arus gerbang negatif). Arus pada terminal M1 dan M2 akan mengalir sesuai dengan besar arus yang di berikan pada terminal *gate*. Semakin besar tegangan pada terminal *gate*, semakin besar pula arus tegangan yang mengalir pada M2 ke M1, dengan syarat tegangan yang diberikan pada terminal *gate* tersebut tidak lebih kecil atau melebihi tegangan yang diberikan pada terminal M1 dan M2.

2.3.2.3 Penggunaan TRIAC

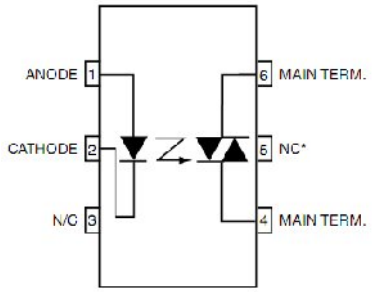
Piranti TRIAC dipakai secara luas untuk menggantikan ke-dudukan relai dan saklar mekanik konvensional. TRIAC dapat digunakan sebagai penyearah, tergantung dari cara pemakaian gerbangnya. TRIAC juga banyak dipakai untuk mengatur siklus piksel LCD, dengan menyambung/memutus arus yang mengalir ke setiap piksel (*picture element*) dalam satuan milidetik. Pengembangan karakteristik unsur penyusun TRIAC dapat menghasilkan waktu *on-off* yang lebih singkat.

TRIAC kebanyakan digunakan dalam rangkaian kontrol gelombang penuh AC, karena TRIAC memberikan dua kelebihan dibandingkan dengan dua *thyristor*. Kelebihan TRIAC tersebut adalah rancangan ke*pin*g pendingin yang lebih sederhana dan rangkaian pemicu yang relatif lebih ekonomis.

2.3.2.4 TRIAC *Optoisolators*

TRIode Alternating Current (TRIAC) *Optoisolators* merupakan jenis TRIAC yang mempunyai prinsip kerja seperti saklar elektronik yang diaktifkan oleh cahaya (LED). TRIAC ini tertanam bersama sebuah LED dalam sebuah rangkaian terintegrasi (*Integrated Circuit*). Perbedaan TRIAC *Optoisolators* dengan TRIAC biasa yaitu terletak dari cara pengaktifannya. TRIAC pada umumnya diaktifkan dengan cara memberi arus listrik secara langsung pada terminal *gate* TRIAC tersebut, sehingga mengakibatkan arus pada terminal M1 dan terminal M2 terhubung. Pada TRIAC *Optoisolators*, terminal *gate* tidak diberi arus listrik secara langsung, akan tetapi terminal *gate* yang berupa optik terisolasi diaktifkan oleh cahaya dari sebuah LED. Salah satu contoh dari IC TRIAC

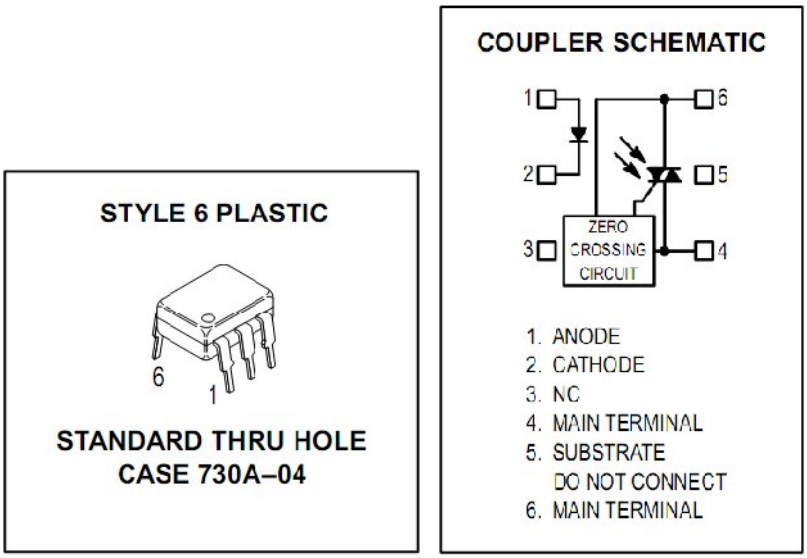
Optoisolators adalah IC tipe MOC3011 yang mempunyai konfigurasi seperti gambar dibawah ini.



Gambar 2.16 Konfigurasi IC MOC 3011

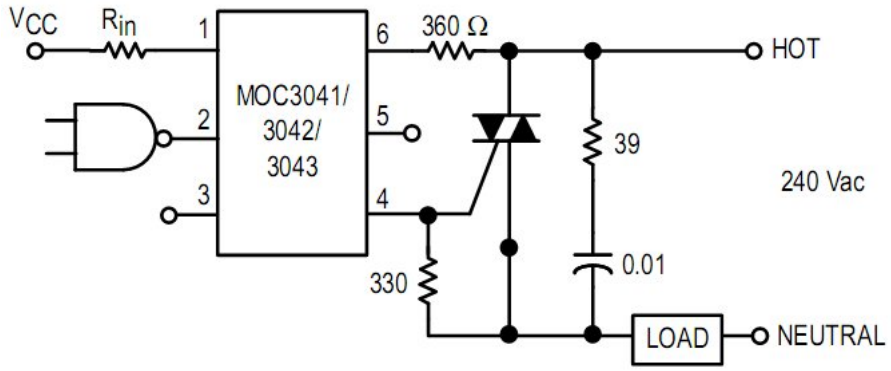
TRIAC akan mengalirkan arus pada M1 dan M2 (*pin* 4 dan 6) apabila tidak ada arus yang mengalir pada *pin* 1 dan 2 (LED padam). Apabila pada *pin* 1 dan 2 diberi arus (LED menyala), maka TRIAC tidak akan mengalirkan arus pada M1 dan M2 (*pin* 4 dan 6).

Berdasarkan tegangan kerjanya, TRIAC *Optoisolators* ini mempunyai daerah tegangan kerja yang berbeda-beda, contohnya TRIAC tipe MOC3011 di atas, mempunyai daerah tegangan kerja maksimal sebesar 250V$_{AC}$. Berbeda halnya dengan TRIAC tipe MOC3041, TRIAC ini memiliki fitur lain. Selain bekerja dapat bekerja pada level tegangan 400V$_{AC}$, MOC3041 memiliki rangkaian *zero crossing.* Rangkaian *zero crossing* ini berfungsi untuk mendeteksi perpotongan gelombang sinus pada tegangan AC dengan titik nol pada tegangan tersebut (*zero point*), sehingga dapat memberikan acuan untuk memulai waktu pen-*trigger*-an. Konfigurasi IC MOC3041 dan rangkaian TRIAC *Opto Osillator* ini dapat dilihat pada gambar di bawah ini.



Gambar 2.17 Bentuk dan Konfigurasi IC MOC3041

TRIAC *Optoisolators* banyak diaplikasikan pada *Selenoid/Valve Controls, Lighting Controls, Statics Power Switches, AC Motor Drivers, Temperature Controls, AC Motor starters, Solid State Relays.*

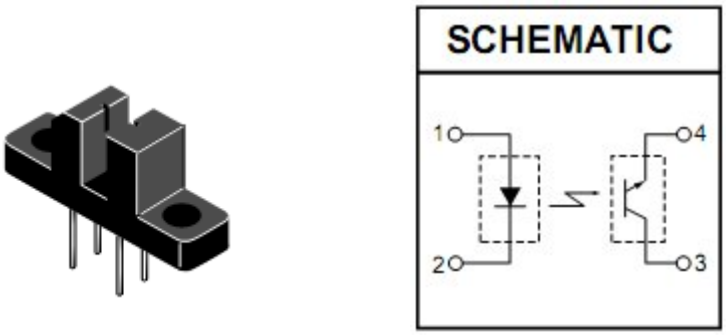


Gambar 2.18 Rangkaian aplikasi TRIAC *Optoisolators* untuk kontrol

Untuk mengontrol tegangan AC dapat dilakukan dengan cara memberikan sinyal PWM pada *Pin* 2 pada IC MOC3041.

2.4 Sensor Kecepatan Putaran

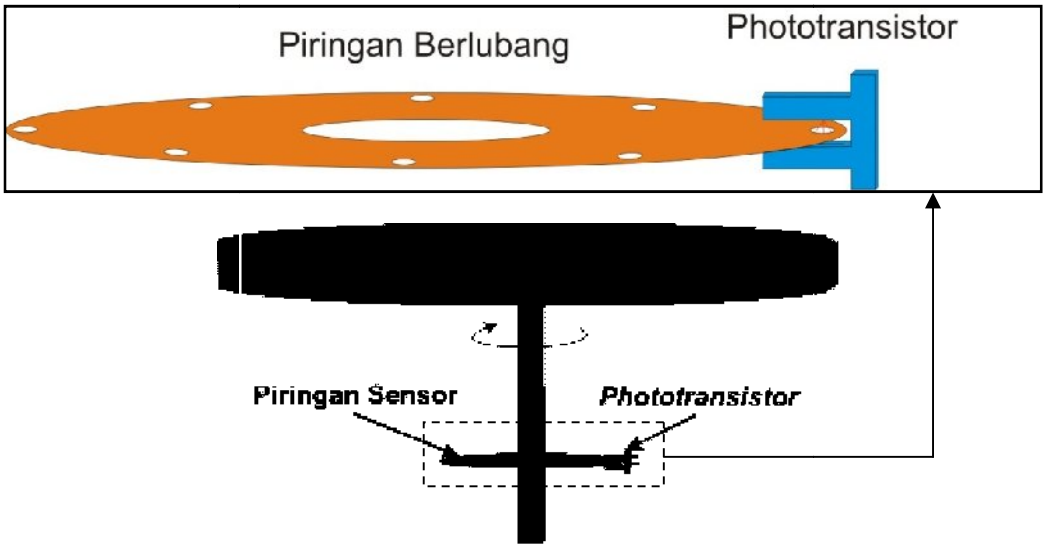
Sensor kecepatan putaran merupakan sensor yang berfungsi untuk mendeteksi atau mengukur kecepatan putaran suatu benda putar. Pada pokok bahasan kali ini akan membahas tentang sensor kecepatan putaran dengan menggunakan *phototransistor*. *Phototransistor* merupakan komponen elektronika yang mempunyai prinsip kerja seperti saklar, namun saklar tersebut diaktifkan oleh cahaya. Bentuk dan skematik dari *phototransistor* seperti gambar di bawah ini.



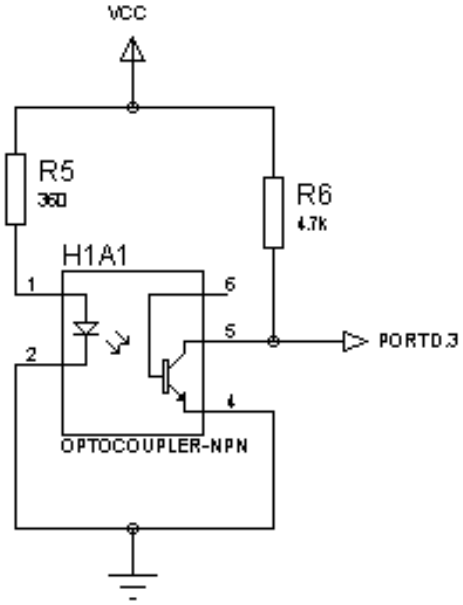
Gambar 2.19 Bentuk dan skematik *phototransistor*

Phototransistor akan aktif jika diberi tegangan sebesar ± 1,7V pada kaki 1 dan kaki 2. Pada saat keadaan aktif tersebut, jika celah antara dioda cahaya dan transistor tidak terhalangi suatu benda padat (tidak transparan), maka transistor cahaya akan meloloskan arus dari kaki kolektor ke kaki emitor. Besar maksimal tegangan yang diijinkan dari kaki kolektor dan emitor (*forward* V_{CE}) tersebut sebesar ±30V.

Untuk mendeteksi putaran suatu benda putar, pada umumnya dengan cara membuat piringan yang dilubangi pada tiap sisinya dengan ukuran tertentu, kemudian poros piringan sensor tersebut disatukan dengan poros benda putar.



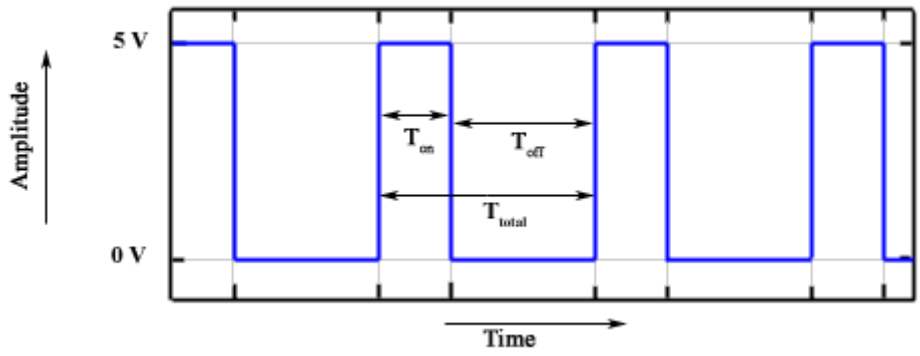
Gambar 2.20 Piringan Sensor



Gambar 2.21 Rangkaian Sensor Kecepatan Putaran

2.5 *Pulse Width Modulation* (PWM)

Pulse Width Modulation (PWM) adalah sebuah metode memanipulasi lebar sinyal atau tegangan yang dinyatakan dengan pulsa dalam suatu periode. Aplikasi PWM sangatlah luas, mulai dari *speed control* (kendali kecepatan), *power control* (kendali sistem tenaga), *measurement and communication* (pengukuran dan telekomunikasi). *regulator* tegangan, audio *effect*, penguatan, serta aplikasi-aplikasi lainnya. Pengaturan lebar pulsa modulasi merupakan salah satu teknik yang digunakan dalam sistem kendali (*control system*) saat ini. PWM dicapai/diperoleh dengan bantuan sebuah gelombang kotak yang mana siklus kerja (*Duty cycle*) gelombang dapat diubah-ubah untuk mendapatkan sebuah tegangan keluaran yang bervariasi yang merupakan nilai rata-rata dari gelombang tersebut. Penjelasan lebih lanjut ditunjukkan pada Gambar 2.22 dibawah ini,



Gambar 2.22 Sinyal PWM

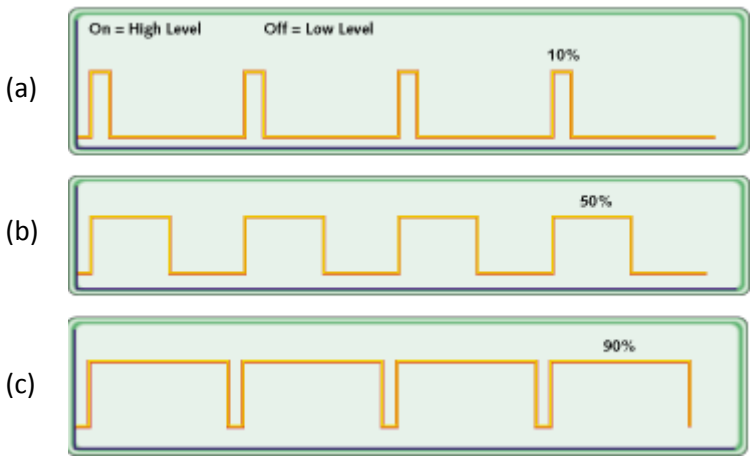
T_{on} adalah waktu dimana tegangan keluaran berada pada posisi tinggi (baca: *high* atau 1) dan T_{off} adalah waktu dimana tegangan keluaran berada pada posisi rendah (baca: *low* atau 0). T_{total} adalah waktu satu siklus atau penjumlahan T_{on} dan T_{off}, biasa dikenal dengan istilah "periode satu gelombang".

= + ... (2.10)

Siklus kerja (*Duty cycle*) sebuah gelombang dapat didefinisikan sebagai berikut.

=——— = —— ..... (2.11)

Gambar 2.23 dibawah ini menunjukan beberapa sinyal PWM dengan nilai *duty cycle* (satuan: %) yang berbeda-beda.



Gambar 2.23 Sinyal PWM Dengan Nilai *Duty Cycle* yang Berbeda-beda

Pada Gambar 2.23 (a) terlihat bahwa, lebar pulsa *high* per periodenya sangat kecil (10%). Pada Gambar 2.23 (b) terlihat bahwa, lebar pulsa *high* sama dengan lebar pulsa *low* (50%). Pada Gambar 2.23 (c) terlihat bahwa sinyal *high* lebih besar dari sinyal *low* (90%). Jika tinggi tegangan pada Gambar 2.23 di atas dimisalkan sebesar 5V dan sinyal PWM tersebut diaplikasikan pada pengontrolan kecepatan motor DC, maka kecepatan motor dengan besar *duty cycle* 90% akan lebih cepat dibandingkan dengan besar *duty cycle* 50% dan 10%. Kecepatan motor dengan besar *duty cycle* 50% akan lebih cepat dibandingkan dengan besar *duty cycle* 10% atau dengan kata lain kecepatan motor dengan besar *duty cycle* 10% akan lebih lambat dibandingkan dengan besar *duty cycle* 50% dan 90%.