BAB II

LANDASAN TEORI

2.1 Logika *Fuzzy*

Logika *Fuzzy* pertama kali diperkenalakan oleh Prof. Lutfi Zadeh pada tahun 1965. Logika *Fuzzy* adalah "Suatu sistem yang digunakan untuk menangani konsep kebenaran parsial yaitu kebenaran yang berada diantara sepenuhnya benar dan sepenuhnya salah" (Yan, Jun. 1994;14). Logika *Fuzzy* berbeda dengan logika digital biasa, dimana logika digital biasanya hanya mengenal dua keadaan yaitu "ya"-"tidak" atau "on"-"off" atau "*high"*-"*low*" atau "1"-"0". Logika *Fuzzy* menggantikan logika *Boolean* dalam tingkat kebenaran.

Logika *Fuzzy* telah banyak digunakan pada bidang-bidang seperti statistik, teori pengendalian, dan lain sebagainya. Logika *Fuzzy* merupakan suatu metoda untuk memindahkan atau meniru pengetahuan manusia ke dalam mesin. Logika *Fuzzy* yang memiliki nilai kekaburan atau kesamaran *(Fuzzyness)* antara benar atau salah. Dalam teori logika *Fuzzy* suatu nilai bisa bernilai benar atau salah secara bersamaan. Nilai kebenaran dan kesalahan tergantung pada bobot keanggotaan atau derajat keanggotaan *(Membership Function)* dalam rentang 0 hingga 1. Berbeda dengan logika digital yang hanya memiliki dua nilai 1 atau 0. Logika *Fuzzy* digunakan untuk menterjemahkan suatu besaran yang diekspresikan menggunakan bahasa *(linguistic),* misalkan besaran kecepatan laju kendaraan yang diekspresikan dengan pelan, agak cepat, cepat, dan sangat cepat. Logika *Fuzzy* juga menunjukan sejauh mana suatu nilai itu benar dan sejauh mana suatu

nilai itu salah. Derajat keanggotaan 0 (nol) artinya nilai bukan merupakan anggota himpunan dan 1 (satu) berarti nilai tersebut adalah anggota himpunan.

Logika *Fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang *input* ke dalam suatu ruang *output,* mempunyai nilai kontinyu. *Fuzzy* dinyatakan dalam derajat dari suatu keanggotaan dan derajat dari kebenaran. Oleh sebab itu sesuatu dapat dikatakan sebagian benar dan sebagian salah pada waktu yang sama (Kusumadewi. 2004)

Logika *Fuzzy* memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk linguistik, konsep tidak pasti seperti "sedikit", "lumayan" dan "sangat" (Zadeh 1965).

2.1.1 Himpunan *Fuzzy*

Himpunan adalah suatu kumpulan atau koleksi objek-objek yang mempunyai kesamaan sifat tertentu (Frans Susilo, 2006). Himpunan *Fuzzy* merupakan suatu pengembangan lebih lanjut tentang konsep himpunan dalam matematika.

Himpunan *Fuzzy* adalah rentang nilai-nilai, masing-masing nilai mempunyai derajat keanggotaan antara 0 sampai dengan 1. Himpunan *Fuzzy* memiliki 2 atribut, yaitu:

a. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti muda, parobaya, tua.
b. Numerik, suatu nilai (angka) yang menunjukkan ukuran dari suaru variabel seperti 10,20,30.

Beberapa hal yang perlu diketahui dalam memahami *Fuzzy*, yaitu:

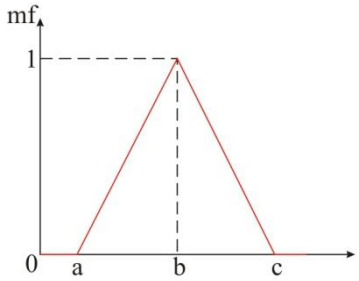
a. Variable *Fuzzy*, merupakan variabel yang hendak dibahas dalam suatu sistem *Fuzzy*. Contoh: umur, *temperature*, permintaan, dan lain sebagainya.
b. Semesta pembicaraan, merupakan keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *Fuzzy*. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif.
c. Domain himpunan *Fuzzy*, merupakan keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *Fuzzy*. Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai domain dapat berupa bilangan positif dan bilangan negatif.

2.1.2 Fungsi Keanggotaan *(Membership Function)*

Fungsi keanggotaan adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data kedalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Apabila U menyatakan himpunan universal dan A adalah himpunan fungsi *Fuzzy* dalam U, maka A dapat dinyatakan sebagai pasangan terurut (Wang, 1997 dari Wulandari, F., 2005). Ada 2 fungsi yang digunakan dalam tugas akhir ini, yaitu:

a. Representasi Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis linear.



Gambar 2.1 Representasi Kurva Segitiga

Representasi kurva segitiga adalah sebagai berikut:

$$\mu[x,a,b,c] = \begin{cases} 0 & ;a \leq x \; atau \; x \geq c \\ (x-a)/(b-a); & a \leq x \leq b \\ (c-x)/(c-b); & b \leq x \leq c \end{cases} \quad \text{…….....……..…(2.1)}$$

Atau

$$u[x,a,b,c] = \max\left(min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \quad \text{…….....…….....……(2.2)}$$

Dimana:

a = nilai domain terkecil yang mempunyai derajat keanggotaan nol

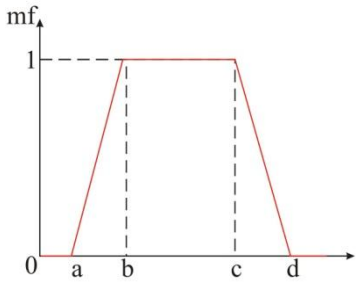
b = nilai domain yang mempunyai derajat keanggotaan satu

c = nilai domain terbesar yang mempunyai derajat keanggotaan nol

x = nilai *input* yang akan diubah ke dalam bilangan *Fuzzy*

b. Representasi Kurva Trapesium

Kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1.



Gambar 2.2 Representasi Kurva Trapesium

Representasi kurva segitiga adalah sebagai berikut:

$$\mu[x,a,b,c,d] = \begin{cases} 0 & ;x \le a \; atau \; x \ge d \\ (x-a)/(b-a); & a \le x \le b \\ 1 & ;b \le x \le c \\ (d-x)/(d-c); & c \le x \le d \\ 0 & ;x \ge d \end{cases} \quad \text{……..……..…..(2.3)}$$

Atau

$$u[x,a,b,c,d] = \max\left(min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \quad \text{…....……..…(2.4)}$$

Dimana:

a = nilai domain terkecil yang mempunyai derajat keanggotaan nol

b = nilai domain terkecil yang mempunyai derajat keanggotaan satu

c = nilai domain terbesar yang mempunyai derajat keanggotaan satu

d = nilai domain terbesar yang mempunyai derajat keanggotaan nol

x = nilai *input* yang akan diubah ke dalam bilangan *Fuzzy*

2.1.3 Operasi Logika *Fuzzy*

Operasi logika adalah operasi yang mengkombinasikan dan memodifikasi 2 atau lebih himpunan *Fuzzy*. Nilai keanggotaan baru hasil operasi dua himpunan disebut *firing strength* atau α predikat, ada 3 operasi dasar yang diciptakan oleh Zadeh:

a. Operator AND, berhubungan dengan operasi intersection pada himpunan, α predikat diperoleh dengan mengambil nilai minimum antar kedua himpunan.

$$\mu A \cap B = \min(\mu A[x], \mu B[y]) \quad \text{.....(2.5)}$$

b. Operator OR, berhubungan dengan operasi union pada himpunan, α predikat diperoleh dengan mengambil nilai maximum antar kedua himpunan.

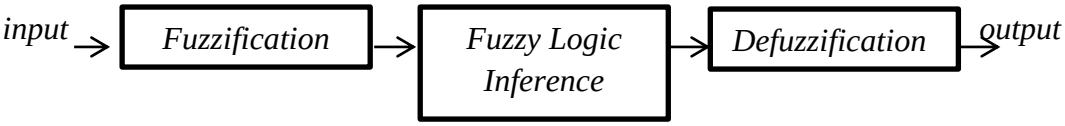
$$\mu A \cup B = \max(\mu A[x], \mu B[y]) \quad \text{.....(2.6)}$$

c. Operasi NOT, berhubungan dengan operasi komplemen pada himpunan, α predikat diperoleh dengan mengurangkan nilai keanggotaan elemen pada himpunan dari 1.

$$\mu A' = 1 - \mu A[x] \quad \text{.....(2.7)}$$

2.1.4 Tahapan Membangun Sistem *Fuzzy*

Tahapan membangun sistem *Fuzzy* tergantung metoda yang digunakan, karena banyak metoda untuk membangun sistem *Fuzzy*. Namun secara garis besar dapat disimpulkan sebagai berikut :



Gambar 2.3 Proses Sistem Logika *Fuzzy*

a. *Fuzzification*

Proses *Fuzzification* yaitu mengubah nilai suatu masukan menjadi suatu fungsi keanggotaan *Fuzzy*. Pada proses ini *Membership Function* ditentukan.

b. *Fuzzy Logic Inference*

Ada beberapa proses pada *Fuzzy Logic Inference*, yaitu :

- Mengaplikasikan aturan *(Fuzzy Rule)* pada masukan *Fuzzy* yang dihasilkan dalam proses *Fuzzification.*
- Mengevaluasi tiap aturan dengan masukan yang dihasilkan dari proses *Fuzzyfikasi* dengan mengevaluasi hubungan atau derajat keanggotaannya.

c. *Defuzzification*

Proses *Defuzzification* merupakan pengubahan kembali data-data *Fuzzy* kembali kebentuk *numeric* yang dapat dikirimkan ke peralatan pengendalian. Proses *Defuzzification* dapat dilakukan dengan beberapa cara, diantaranya:

- *Centroid Of Area,*

$$z_{COA} \frac{\int z\, \mu A(z) z\, dz}{\int z\, \mu A(z)\, dz} \quad \text{.....(2.8)}$$

- *Bisector Of Area,*

$$\int_{\alpha}^{z\,BOA} \mu A(z) dz = \int_{z\,BOA}^{\beta} \mu A(z) dz \quad \text{.....(2.9)}$$

- *Mean Of Maximum,*

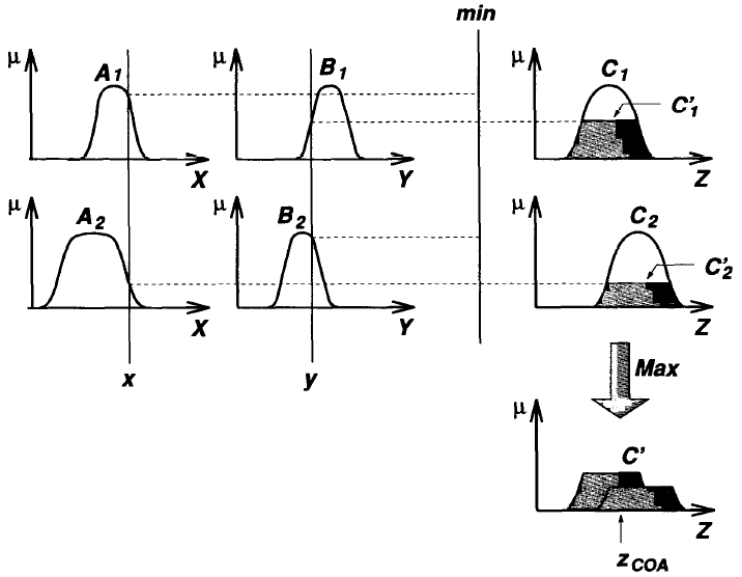
$$z_{MOM} = \frac{\int z_{\prime}\, z\, dz}{\int z'\, dz} \quad \text{.....(2.10)}$$

2.1.5 Fuzzy Inference System

Sistem Inferensi *Fuzzy (Fuzzy Inference System/FIS)* disebut juga *Fuzzy Inference Engine* adalah sistem yang dapat melakukan penalaran dengan prinsip serupa seperti manusia melakukan penalaran dengan nalurinya. Terdapat beberapa jenis FIS yang dikenal yaitu Mamdani, Sugeno dan lain sebagainya.

a. FIS Mamdani

FIS yang paling mudah dimengerti, karena paling sesuai dengan naluri manusia adalah FIS Mamdani. FIS tersebut bekerja berdasarkan kaidah-kaidah linguistik dan memiliki algoritma *Fuzzy* yang menyediakan sebuah aproksimasi untuk dimasuki analisa matematik.

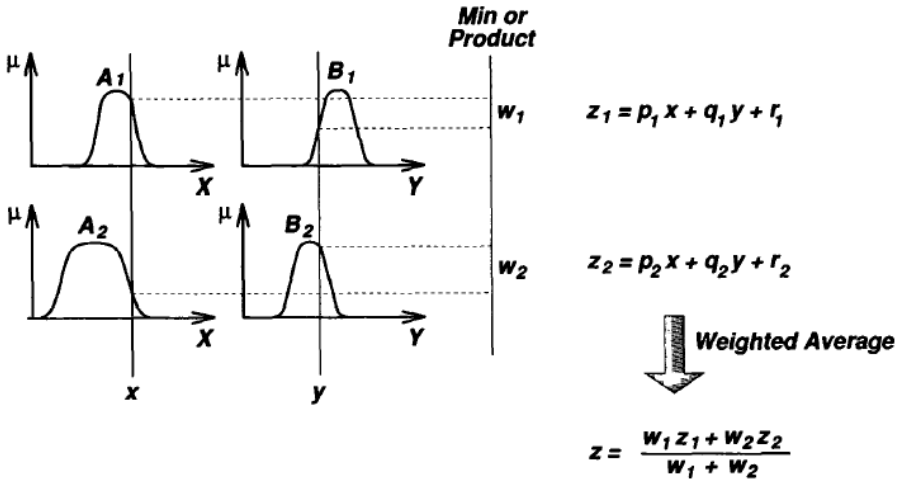


Gambar 2.4 FIS Mamdani

b. FIS Sugeno

Penalaran dengan metode Sugeno hampir sama dengan penalaran Mamdani, hanya saja *output* sistem tidak berupa himpunan *Fuzzy* melainkan berupa konstanta atau persamaan linier. Metoda ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985. Sistem *Fuzzy* Sugeno memperbaiki kelemahan yang dimiliki oleh sistem *Fuzzy* murni untuk menambah suatu perhitungan matematika sederhana sebagai bagian THEN. Pada perubahan ini, sistem *Fuzzy* memiliki suatu nilai rata-rata *tertimbang (Weighted Average Values)* di dalam bagian aturan *Fuzzy* IF-THEN. Sistem *Fuzzy* Sugeno juga memiliki kelemahan terutama pada

bagian THEN, yaitu dengan adanya perhitungan matematika sehingga tidak dapat menyediakan kerangka alami untuk erepresentasikan pengetahuan manusia dengan sebenarnya.



Gambar 2.5 FIS Sugeno

2.2 Sensor Jarak Ultrasonik (SR-04)

Sensor ultrasonik SR-04 adalah sebuah sensor jarak yang terdiri dari rangkaian pemancar ultrasonik atau *transmitter* dan rangkaian penerima ultrasonik atau *receiver*. Gelombang ultrasonik adalah gelombang dengan besar frekuensi diatas frekuensi gelombang suara yaitu 40khz.



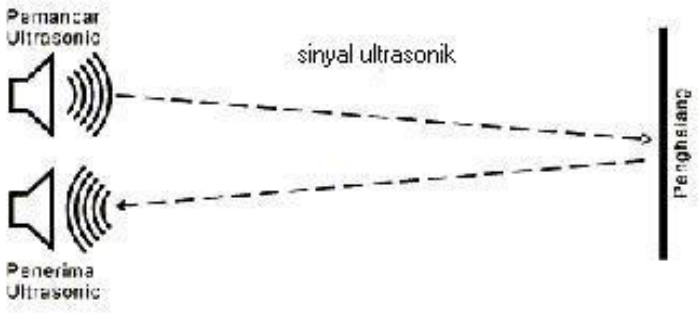
Gambar 2.6 Sensor Jarak Ultrasonik (SR-04)

Prinsip kerja dari sensor SR-04 adalah sebagai berikut :

- Gelombang ultrasonik akan dibangkitkan dan dipancarkan melalui *transmitter* secara menyebar.
- Gelombang ultrasonik yang dipancarkan tersebut kemudian akan merambat sebagai sinyal / gelombang bunyi dengan kecepatan bunyi yang berkisar 340 m/s. Sinyal tersebut kemudian akan dipantulkan dan akan diterima kembali oleh bagian *receiver* ultrasonik.
- Setelah sinyal tersebut diterima kembali, kemudian sinyal tersebut akan diproses untuk menghitung jaraknya. Jarak dihitung berdasarkan rumus :

  $$S = 340.t/2 \quad \text{.....(2.11)}$$

  dimana S adalah jarak antara sensor ultrasonik dengan bidang pantul, dan t adalah selisih waktu antara pemancaran gelombang ultrasonik sampai diterima kembali oleh bagian penerima ultrasonik.



Gambar 2.7 Cara Kerja Sensor Ultrasonik

Jadi secara ringkas cara kerja dari sensor ultrasonik ini yaitu memancarkan gelombang ultrasonik oleh *transmitter* yang bersifat menyebar dan akan memantul jika terkena halangan atau benda yang berada di depannya, pantulan tersebut akan diterima oleh *receiver* sehingga akan didapat jarak berdasarkan persamaan 2.11.

2.3 Optocoupler

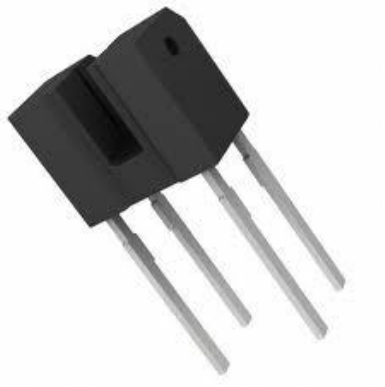
Optocoupler adalah suatu piranti elektronika yang terdiri dari 2 bagian yaitu *transmitter* (pengirim) dan *receiver* (penerima), yaitu antara bagian yang menghasilkan cahaya dengan bagian deteksi sumber cahaya tersebut terpisah.

a. *Transmitter*

Merupakan bagian yang terhubung dengan rangkaian *input* atau rangkaian kontrol. Pada bagian ini terdapat sebuah LED infra merah (IR LED) yang berfungsi untuk mengirimkan sinyal kepada *receiver*. Pada *transmitter* dibangun dari sebuah LED infra merah. Jika dibandingkan dengan menggunakan LED biasa, LED infra merah memiliki ketahanan yang lebih baik terhadap sinyal tampak. Cahaya yang dipancarkan oleh LED infra merah tidak terlihat oleh mata telanjang.

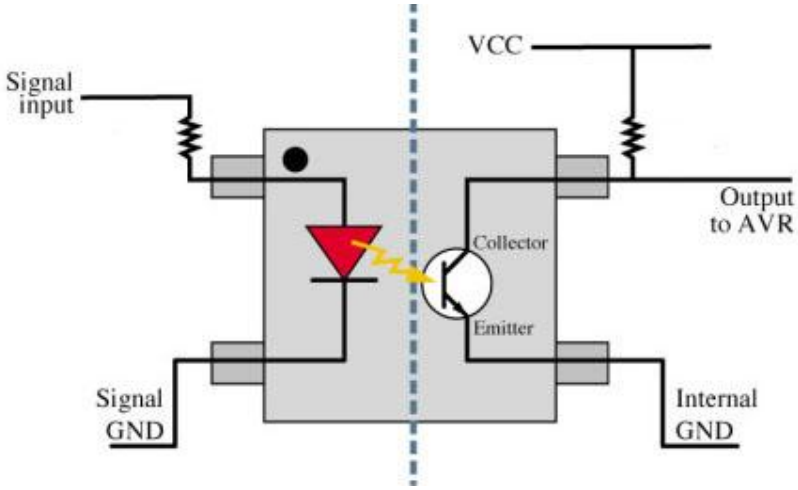
b. *Receiver*

Merupakan bagian yg terhubung dengan rangkaian *output* atau rangkaian beban, dan berisi komponen penerima cahaya yang dipancarkan oleh *transmitter*. Komponen penerima cahaya ini dapat berupa photodioda atapun phototransistor. Pada bagian *receiver* dibangun dengan dasar komponen phototransistor. Phototransistor merupakan suatu transistor yang peka terhadap tenaga cahaya. Suatu sumber cahaya menghasilkan energi panas, begitu pula dengan spektrum infra merah. Karena spekrum infra mempunyai efek panas yang lebih besar dari cahaya tampak, maka phototransistor lebih peka untuk menangkap radiasi dari sinar infra merah.



Gambar 2.8 *Optocoupler*

Optocoupler merupakan gabungan dari LED infra merah dengan *photo-transistor* yang terbungkus menjadi satu *chips*. LED infra merah sebagai *transmitter* akan menghasilakan cahaya infra merah termasuk dalam gelombang elektromagnetik yang tidak tampak oleh mata telanjang. Sinar ini tidak tampak oleh mata karena mempunyai panjang gelombang berkas cahaya yang terlalu panjang bagi tanggapan mata manusia. Sinar infra merah mempunyai daerah frekuensi 1 x 1012 Hz sampai dengan 1 x 1014 GHz atau daerah frekuensi dengan panjang gelombang 1µm – 1mm. Cahaya infra merah ini akan diterima oleh phototransistor. *Photo-transistor* merupakan komponen elektronika yang berfungsi sebagai detektor cahaya infra merah. Detektor cahaya ini mengubah efek cahaya menjadi sinyal listrik, oleh sebab itu *photo-transistor* termasuk dalam golongan detektor optik.



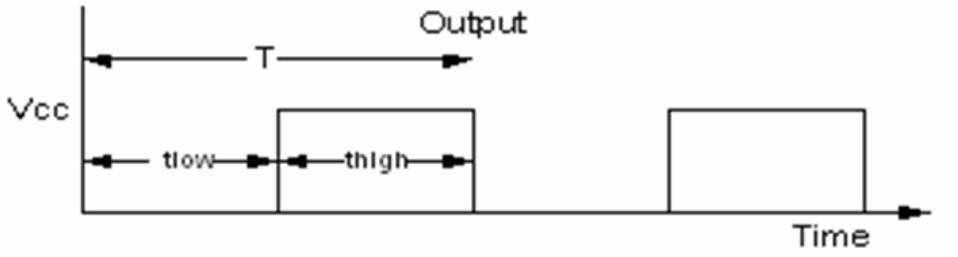
Gambar 2.9 Cara Kerja *Optocoupler*

Prinsip kerja dari optocoupler adalah sebagai berikut:

- Jika antara *photo-transistor* dan LED terhalang maka *photo-transistor* tersebut akan *off* sehingga *output* dari kolektor akan berlogika *high.*
- Sebaliknya jika antara *photo-transistor* dan LED tidak terhalang maka phototransistor dan LED tidak terhalang maka *photo-transistor* tersebut akan *on* sehingga *output*-nya akan berlogika *low*.

2.4 *Pulse Width Modulation (PWM)*

PWM merupakan suatu metoda untuk membangkitkan sinyal keluaran berupa pulsa dengan frekuensi dan amplitudo tetap dengan lebar pulsa yang dapat diubah-ubah. Lebar pulsa berupa berupa kondisi *high* dan *low* dapat diatur durasinya sesuai dengan keinginan. PWM bisa dibangkitkan secara *software* maupun *hardware.* Sehingga dapat dibentuk gelombang dengan *duty cycle* yang dapat diatur sesuai dengan program.



Gambar 2.10 Sinyal PWM

Duty cycle merupakan perbandingan atau rasio lamanya suatu sistem bernilai logika *high* dan *low*. Variasi *duty cycle* ini memberikan harga tegangan rata–rata yang berbeda–beda. Sinyal PWM dengan *duty cycle* yang besar memiliki nilai rata-rata tegangan yang lebih besar dibandingkan dengan *duty cycle* kecil. Nilai tegangan yang diberikan sebanding dengan nilai duty cycle yang diberikan. Adapun rumus *duty cycle*, yaitu:

$$Duty\ Cycle = \frac{t_{high}}{(t_{high}+t_{low})} \times 100\% \quad \text{.....(2.12)}$$

Sedangkan untuk menghitung nilai tegangan rata-rata *output* dapat dihitung dengan rumus sebagai berikut:

$$V_{out} = Duty\ Cycle\ \times V_{total} \quad \text{.....(2.13)}$$

Dimana Vtotal merupakan tegangan yang diberikan untuk mengaktifkan motor.

2.4.1 PWM Sebagai Pengatur Kecepatan Motor DC

Motor DC adalah piranti elektronik yang mengubah energi listrik menjadi energi mekanik berupa gerak rotasi. Kecepatan motor DC berbanding lurus dengan tegangan listrik yang diberikan. Sebagai contoh, jika tegangan motor dikurangi dari 6 volt menjadi 3 volt, maka kecepatan motor akan turun menjadi setengahnya. Salah satu cara untuk menurunkan kecepatan motor yaitu dengan menurunkan rata-rata tegangan yang diberikan. Yaitu dengan cata memutus dan

menghubungkan kembali *supply* tegangan listrik dengan sangat cepat. Cara kerja inilah yang diadopsi oleh PWM. Saat tegangan listrik terputus putaran motor akan melambat, dan dipercepat kembali ketika tegangan listrik tersambung.



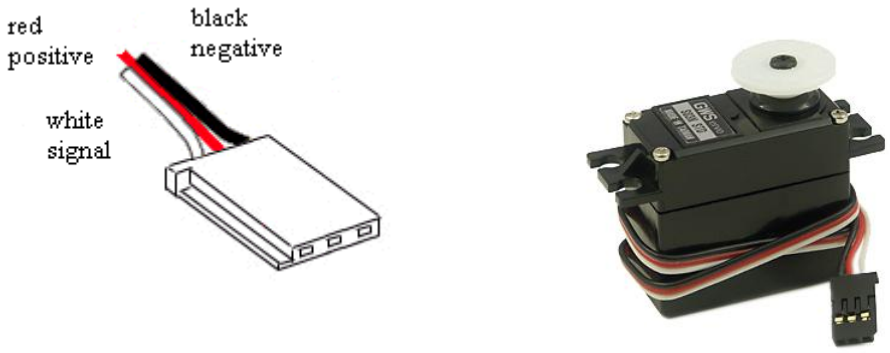
Gambar 2.11 Motor DC

Cepat lambatnya motor DC dapat diatur berdasarkan *duty cycle* yang diberikan, jika *duty cycle* pada sinyal PWM besar, maka akan menghasilakan rata-rata tegangan yang besar pula sehingga motor DC akan berputar cepat, begitu juga sebaliknya jika *duty cycle* yang diberikan kecil akan mengahasilkan tegangan rata-rata yang kecil dan motor DC akan berputar lebih lambat. Pengaturan nilai tegangan rata-rata tersebut dapat diatur dengan menggunakan persamaan (2.13).

2.4.2 PWM Sebagai Pengatur Sudut Motor Servo

Motor servo adalah sebuah motor DC yang dilengkapi rangkaian kendali dengan sistem *closed feedback* yang terintegrasi dalam motor tersebut. Motor servo disusun dari sebuah motor DC, gearbox, variabel resistor (VR) atau potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas maksimum putaran sumbu (axis) motor servo. Sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang pada pin kontrol motor servo. Pemberian pulsa sebagai pengontrolan motor servo dapat dilakukan dengan metoda PWM.

Berbeda dengan motor DC, lebar pulsa yang diberikan pada motor servo ini diberikan untuk mengatur posisi sudut motor. Motor servo akan bekerja secara baik jika pada bagian pin kontrolnya diberikan sinyal PWM dengan frekuensi 50 Hz. Servo motor mempunyai 3 buah pin yang terdiri dari VCC, GND dan pin SIGNAL. Berikut adalah contoh konfigurasi pin motor servo :



Gambar 2.12 Motor Servo

Motor servo ini terbagi menjadi 2 jenis, yaitu :

a. Motor servo standard

   Motor servo jenis ini hanya mampu bergerak dua arah (CW dan CCW) dengan defleksi masing-masing sudut mencapai 90° sehingga total defleksi sudut dari kanan – tengah – kiri adalah 180°.

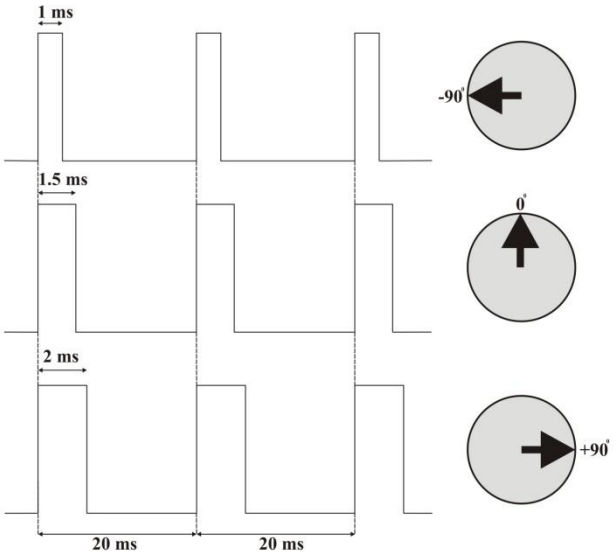
b. Motor servo continius

   Motor servo jenis ini mampu bergerak dua arah (CW dan CCW) tanpa batasan defleksi sudut putar (dapat berputar secara kontinyu).

Dari kedua jenis motor servo tersebut, metoda PWM dapat digunakan untuk menentukan posisi sudut motor, hanya saja terdapat perbedaan pada besarnya arah puataran motor.

Pada motor servo standard hanya dapat bergerak sebesar 180° yaitu dari 0°-180° atau -90°-90°. Secara umum untuk mengakses motor servo tipe standard

adalah dengan cara memberikan pulsa *high* selama 1,5 ms dan mengulangnya setiap 20 ms, maka posisi servo akan berada ditengah atau netral (0°). Untuk pulsa 1 ms maka akan bergerak berkebalikan arah jarum jam dengan sudut -90°. Dan pulsa *high* selama 2 ms akan bergerak searah jarum jam sebesar 90° seperti terlihat pada gambar berikut:



Gambar 2.13 Sinyal Kontrol Motor Servo

Pengaturan pergerakan motor servo continius tidak jauh berbeda dengan pengaturan pengendalian motor servo standard, secara umum untuk berputar (rotasi) searah jarum jam harus diberi pulsa high selama 1,3 ms. Sedangkan untuk berputar berlawanan arah jarum jam harus diberi logika high selama 1,7 ms. Jika motor servo continous diberi pulsa high selama 1,5 ms maka akan berhenti. Pin signal pada motor servo dapat dikoneksi ke pin mikrokontroler sebagai pengendali dari pergerakan motor servo ini.

2.5 Mikrokontroler

Mikrokontroler adalah sebuah *chip* yang berfungsi sebagai pengontrol rangkaian elektronik dan umunya dapat menyimpan program didalamnya. Mikrokontroler umumnya terdiri dari CPU *(Central Processing Unit),* memori, I/O tertentu dan unit pendukung seperti *Analog-to-Digital Converter* (ADC) yang sudah terintegrasi di dalamnya.



Gambar 2.14 ATmega 32

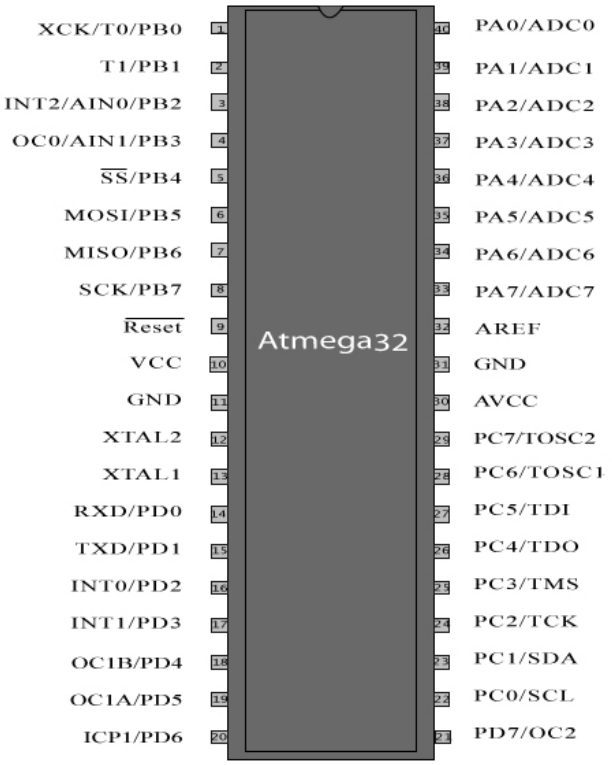
Kelebihan utama dari mikrokontroler ialah tersedianya RAM dan peralatan I/O pendukung sehingga ukuran board mikrokontroler menjadi sangat ringkas. AVR Atmega32 merupakan sebuah mikrokontroler low power CMOS 8 bit berdasarkan arsitektur AVR RISC. Karakteristik dari mikrokontroler ini adalah:

- Menggunakan arsitektur AVR RISC
    - 131 perintah dengan satu clock cycle
    - 32 x 8 register umum
- Data dan program memori
    - 32 Kb In-System Programmable Flash
    - 2 Kb SRAM
    - 1 Kb In- System EEPROM

- 8 Channel 10-bit ADC
- Two Wire Interface
- USART Serial Communication
- Master/Slave SPI Serial Interface
- On-Chip Oscillator
- Watch-dog Timer
- 32 Bi-directional I/O
- Tegangan operasi 2,7 – 5,5 V

Arsitektur AVR ini menggabungkan perintah secara efektif dengan 32 register umum. Semua register tersebut langsung terhubung dengan *Arithmetic Logic Unit* (ALU) yang memungkinkan 2 register terpisah diproses dengan satu perintah tunggal dalam satu *clock cycle*. Hal ini menghasilkan kode yang efektif dan kecepatan prosesnya 10 kali lebih cepat dari pada mikrokontroler CISC biasa.

AVR ATMega32 terdiri dari 40 pin yang terbagi menjadi 4 port yaitu PORTA sebanyak 8 pin, PORTB sebanyak 8 pin, PORTC sebanyak 8 pin, dan PORTC sebanyak 8 pin. Masing-masing pin dapat digunakan sebagai *input-output* dari sistem.



Gambar 2.15 Pin-Pin ATMega32

Secara fungsional konfigurasi pin ATMega32 adalah sebagai berikut:

a. VCC, tegangan sumber

b. GND (Ground)

c. Port A (PA7 – PA0)

   Port A adalah 8-bit port I/O yang bersifat bi-directional dan setiap pin memilki internal *pull-up* resistor. *Output* buffer port A dapat mengalirkan arus sebesar 20 mA. Ketika port A digunakan sebagai *input* dan di *pull-up* secara langsung, maka port A akan mengeluarkan arus jika internal *pull-up* resistor diaktifkan. Pin-pin dari port A memiliki fungsi khusus yaitu dapat berfungsi sebagai channel ADC *(Analog to Digital Converter)* sebesar 10 bit. Fungsi-fungsi khusus pin-pin port A dapat ditabelkan seperti yang tertera pada tabel.

Tabel 2.1 Fungsi Khusus Port A

Port	Fungsi
PA7	ADC7 (ADC *input channel* 7)
PA6	ADC6 (ADC *input channel* 6)
PA5	ADC5 (ADC *input channel* 5)
PA4	ADC4 (ADC *input channel* 4)
PA3	ADC3 (ADC *input channel* 3)
PA2	ADC2 (ADC *input channel* 2)
PA1	ADC1 (ADC *input channel* 1)
PA0	ADC0 (ADC *input channel* 0)

d. Port B (PB7 – PB0)

Port B adalah 8-bit port I/O yang bersifat *bi-directional* dan setiap pin mengandung internal *pull-up* resistor. *Output buffer* port B dapat mengalirkan arus sebesar 20 mA. Ketika port B digunakan sebagai *input* dan di *pull-down* secara *external*, port B akan mengalirkan arus jika *internal pull-up* resistor diaktifkan. Fungsi-fungsi khusus pin-pin port B dapat ditabelkan seperti pada tabel.

Tabel 2.2 Fungsi Khusus Port B

Port	Fungsi
PB7	SCK *(SPI Bus Serial Clock)*
PB6	MISO *(SPI Bus Master Input/Slave Output)*
PB5	MOSI *(SPI Bus Master Output/Slave Input)*
PB4	SS *(SPI Slave Select Input)*
PB3	AIN1 *(Analog Comparator Negative Input)* OCO *(Timer/Counter0 Output Compare Match Output)*
PB2	AIN0 *(Analog Comparator Positive Input)* INT2 *(External Interrupt 2 Input)*
PB1	T1 *(Timer/Counter1 External Counter Input)*
PB0	T0 *(Timer/Counter External Counter Input)* XCK *(USART External Clock Input/Output)*

e. Port C (PC7 – PC0)

Port C adalah 8-bit port I/O yang berfungsi *bi-directional* dan setiap pin memiliki *internal pull-up* resistor. *Output buffer* port C dapat mengalirkan arus sebesar 20 mA. Ketika port C digunakan sebagai *input* dan di *pull-down* secara langsung, maka port C akan mengeluarkan arus jika *internal pull-up* resistor diaktifkan. Fungsi-fungsi khusus pin-pin port C dapat ditabelkan seperti yang tertera pada tabel dibawah ini.

Tabel 2.3 Fungsi Khusus Port C

Port	Fungsi
PC7	TOSC2 *(Timer Oscillator* Pin 2)
PC6	TOSC1 (*Timer Oscillator* Pin 1)
PC5	TD1 (JTAG *Test Data In)*
PC4	TD0 (JTAG *Test Data Out)*
PC3	TMS (JTAG *Test Mode Select)*
PC2	TCK (JTAG *Test Clock)*
PC1	SDA *(Two-wire Serial Bus Data Input/Output Line)*
PC0	SCL *(Two-wire Serial Bus Clock Line)*

f. Port D (PD7 – PD0)

Port D adalah 8-bit port I/O yang berfungsi *bi-directional* dan setiap pin memiliki *internal pull-up* resistor. *Output buffer* port D dapat mengalirkan arus sebesar 20 mA. Ketika port D digunakan sebagai *input* dan di *pull-down* secara langsung, maka port D akan mengeluarkan arus jika *internal pull-up* resistor diaktifkan. Fungsi-fungsi khusus pin-pin port D dapat ditabelkan seperti yang tertera pada tabel dibawah ini.

Tabel 2.4 Fungsi Khusus Port D

Port	Fungsi
PD7	OC2 *(Timer / Counter2 Output Compare Match Output)*
PD6	ICP1 *(Timer/Counter1 Input Capture Pin)*
PD5	OCIB *(Timer/Counter1 Output Compare B Match)*
PD4	TD0 (JTAG *Test Data Out)*
PD3	INT1 *(External Interrupt 1 Input)*
PD2	INT0 *(External Interrupt 0 Input)*
PD1	TXD (USART *Output Pin)*
PD0	RXD (USART *Input Pin)*