

**PENGATURAN ARAH DAN KECEPATAN PUTAR DISK INERSIA  
UNTUK MENJAGA KESEIMBANGAN MINIATUR SEPEDA  
MOTOR MENGGUNAKAN KONTROL LOGIKA FUZZY**

**PROPOSAL SKRIPSI**

*Diajukan Untuk Memenuhi Persyaratan*

*Memperoleh Gelar Sarjana Teknik*



**Disusun Oleh:**

**ADHITYA YOGA GURITNA**

**0810633001**

**KEMENTERIAN PENDIDIKAN NASIONAL**

**UNIVERSITAS BRAWIJAYA**

**FAKULTAS TEKNIK**

**MALANG**

**2012**

## 1. Judul

PENGATURAN ARAH DAN KECEPATAN PUTAR *DISK INERSIA* UNTUK MENJAGA KESEIMBANGAN MINIATUR SEPEDA MOTOR MENGGUNAKAN KONTROL LOGIKA FUZZY.

## 2. Latar Belakang

Seiring dengan perkembangan teknologi dan ilmu pengetahuan terutama kemajuan dalam bidang robotika, mesin maupun robot dirancang agar mempunyai kemampuan yang sama bahkan melebihi kapasitas manusia, salah satu kemampuan manusia yang tidak dimiliki robot adalah kemampuan menjaga keseimbangan secara mandiri, dimana manusia memiliki sensor dan aktuator penyeimbang yang sangat kompleks.

Dalam kehidupan sehari-hari kendaraan roda dua menjadi kendaraan yang paling banyak dipilih masyarakat sebagai kendaraan pribadi. Namun kendaraan roda dua mempunyai kelemahan dimana sepeda roda dua tidak dapat menjaga keseimbangannya, sepeda akan mudah jatuh ke kiri atau kekanan apabila mendapat gangguan. Dari hal itu munculah ide untuk membuat sistem keseimbangan otomatis pada miniatur sepeda roda dua, dimana sistem ini adalah sebuah sistem sensorik dan mekanis yang akan menjaga miniatur sepeda roda dua dalam keadaan seimbang dengan menggunakan sensor keseimbangan “*gyro*” dan mekanis penyeimbang “*disk inertia*” dan kecerdasan buatan yang ditanam ke dalam mikrokontroler Atmega8.

Kondisi miniatur sepeda roda dua tidaklah setimbang walaupun beberapa detik saja, agar tetap seimbang seluruh sistem harus bekerja secara sempurna. Yakni kecepatan putar dan arah mekanis penyeimbang harus sesuai dengan data yang diperoleh dari sensor keseimbangan, sehingga tiga elemen (sensor, mekanis, dan kontroler) dituntut bekerja secara sempurna.

Karena pada sistem keseimbangan ini dibutuhkan respon dengan akurasi tinggi maka pembacaan sensor *gyro* menggunakan filter digital agar didapat data keseimbangan yang akurat, sedangkan mekanis penyeimbang harus dihitung

persamaan matematisnya agar dapat dikalkulasi perkiraan sinyal kontrol yang dikeluarkan oleh kontroler fuzzy yang ditanam di mikrokontroler Atmega8.

### **3. Rumusan Masalah**

Dari latar belakang masalah diatas, dibuat rumusan masalah sebagai berikut :

1. Bagaimana merancang hardware sebuah miniatur sepeda motor roda dua, beserta piringan inersia dan kontroler?
2. Bagaimana merancang software sebagai pembaca keseimbangan menggunakan sensor *gyro*, dan kontroler fuzzy yang akan digunakan untuk mengendalikan arah dan kecepatan putar "*disk inertia*"?

### **4. Batasan Masalah**

Dalam perancangan, permasalahan dibatasi oleh hal-hal sebagai berikut :

1. Menggunakan mikrokontroler Atmega8 sebagai pusat pengontrol dengan kendali kontroler fuzzy.
2. Sensor yang digunakan adalah potensiometer *Gyroscope* produk robotis GS 12 dengan range  $\pm 300^\circ/\text{s}$ .
3. Motor yang digunakan untuk memutar piringan inersia adalah jenis Mini Metal Gear 12VDC kecepatan maksimum 200rpm dengan torsi 1,2kg.cm.
4. Gangguan alat yaitu hanya berupa dorongan ujung jari tangan yang akan mengakibatkan kemiringan robot berubah sudut maksimal  $(\pm) 1^\circ/\text{s}$ .
5. Pembahasan dietekankan pada pengendalian keseimbangan miniatur sepeda motor pada saat diberi gangguan, kinerja driver dan elektronika tidak dibahas mendalam.

### **5. Tujuan**

Tujuan yang ingin dicapai pada skripsi ini adalah membuat sebuah sistem pengaturan arah dan kecepatan *disk inertia* yang dapat mempertahankan kesetimbangan posisi miniatur sepeda roda dua dengan maupun ada gangguan.

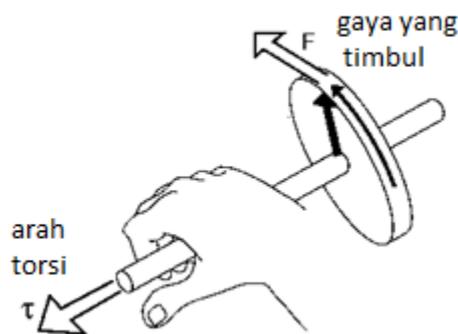
## 6. Manfaat

Desain dan implementasi sistem ini diharapkan dapat bermanfaat sebagai referensi untuk pengendalian sistem keseimbangan otomatis dan bermanfaat dalam pengembangan penelitian mengenai *balancing robot* dimasa mendatang.

## 7. Tinjauan Pustaka

### 7.1 Momen Inersia

Momen inersia (Satuan SI :  $\text{kg m}^2$ ) adalah ukuran kelembaman suatu benda untuk berotasi terhadap porosnya. Besaran ini adalah analog rotasi daripada massa. Momen inersia berperan dalam dinamika rotasi seperti massa dalam dinamika dasar, dan menentukan hubungan antara momentum sudut dan kecepatan sudut, momen gaya dan percepatan sudut, dan beberapa besaran lain. Meskipun pembahasan skalar terhadap momen inersia, pembahasan menggunakan pendekatan tensor memungkinkan analisis sistem yang lebih rumit seperti gerakan giroskopik.

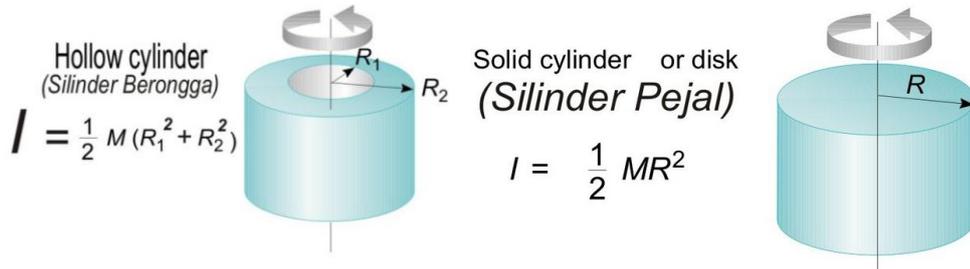


Gambar 1 momen inersia

Momen inersia dari beberapa partikel diperoleh dengan menjumlahkan secara aljabar biasa tiap-tiap momen inersia.

$$I = \sum_i m_i r_i^2$$

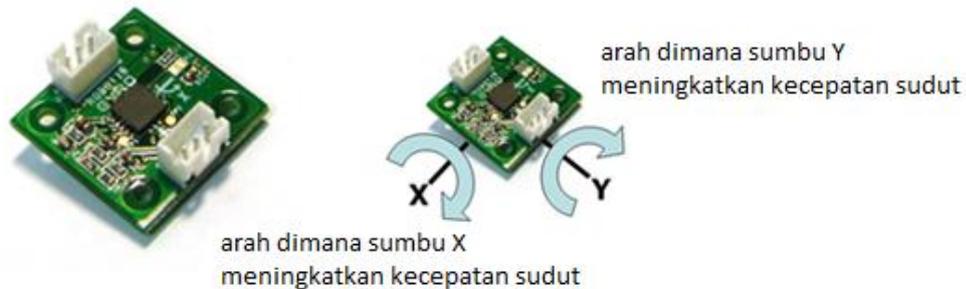
Momen Inersia Berbagai Benda Tegar :



Gambar 2 momen inersia benda tegar

### 7.2 Sensor Kecepatan Sudut (Gyro GS10)

Untuk mendeteksi keseimbangan robot menggunakan sensor keseimbangan yaitu *gyroscope*, pada Gambar 3 dibawah ini adalah sensor *gyroscope* dari [www.robotis.com](http://www.robotis.com).



Gambar 3 Gyroscope sensor GS10

(Sumber : [www.robotis.com](http://www.robotis.com))

Tabel 1 Tabel Keluaran (output) sensor berupa tegangan analog

Output Value	455	-	250	-	45
AngularVellocity	+300°/s	-	0°/s	-	-300°/s
Voltage Value	2.23 V	-	1.23 V	-	0.23 V

(Standard VCC Voltage Value sebesar 5V)

Cara kerja *gyroscope* adalah mengukur kecepatan sudut (kecepatan putar), menghasilkan output “noI” apabila dalam keadaan diam dan mengeluarkan nilai keluaran positif atau negatif jika sensor berputar. Seperti pada penjelasan Gambar 4 berikut ini.

giroskop:

- Tingkat ukuran sudut (kecepatan rotasi).
- Membaca "nol" saat stasioner.
- Membaca positif atau negatif ketika berputar.



Gambar 4 Gyroscope sensor GS10 principle

(Sumber : [www.robotis.com](http://www.robotis.com))

### 7.3 Filter Digital

Filter digital adalah sebuah metode pemrosesan data secara digital / program dari data sebuah sensor atau lebih untuk mendapatkan nilai perkiraan data sebaik dan seakurat mungkin dari gangguan, ketidak linieran, dan berbagai masalah akuisisi data yang lainnya.

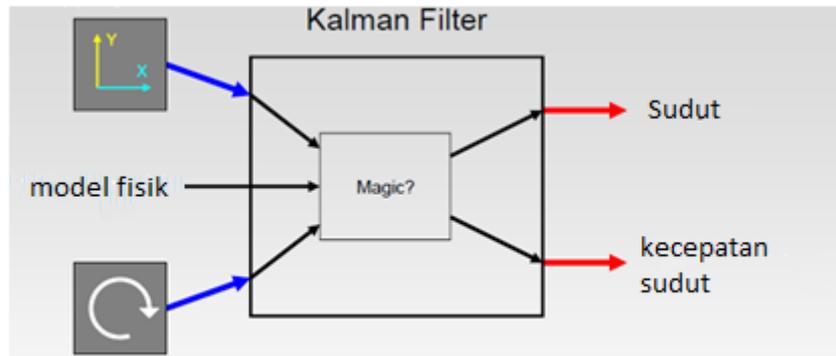
Terdapat banyak sekali jenis-jenis filter digital diantaranya adalah :

#### 1. Kalman Filter

Kalman Filter adalah rangkaian teknik perhitungan matematika (algoritma) yang memberikan perhitungan efisien dalam mengestimasi *state* proses, yaitu dengan cara meminimalkan rata-rata kuadrat galat (*Mean Squared Error/MSE*). Kalman filter diskrit digunakan pada suatu sistem dengan waktu diskrit, artinya jarak antar waktu adalah sama (konstan).

Filter sangat berguna dalam beberapa aspek yaitu dapat menaksir *state* masa lalu, masa kini, maupun masa depan. Pada Kalman Filter, Filter diasumsikan sebagai suatu alat yang bagus untuk memisahkan sinyal dari sinyal lain yang tidak dikehendaki. Pada faktanya, hasil pengukuran tidak akurat atau mengandung sinyal yang tidak dikehendaki (*noise*), sehingga dengan menggunakan filter terhadapnya, maka hasil pengukuran akan mendekati hasil sebelumnya.

Pada kasus keseimbangan, metode kalman filter digunakan untuk mengkolaborasikan dua buah sensor mendapatkan nilai sudut dan kecepatan sudut, seperti gambar dibawah ini.

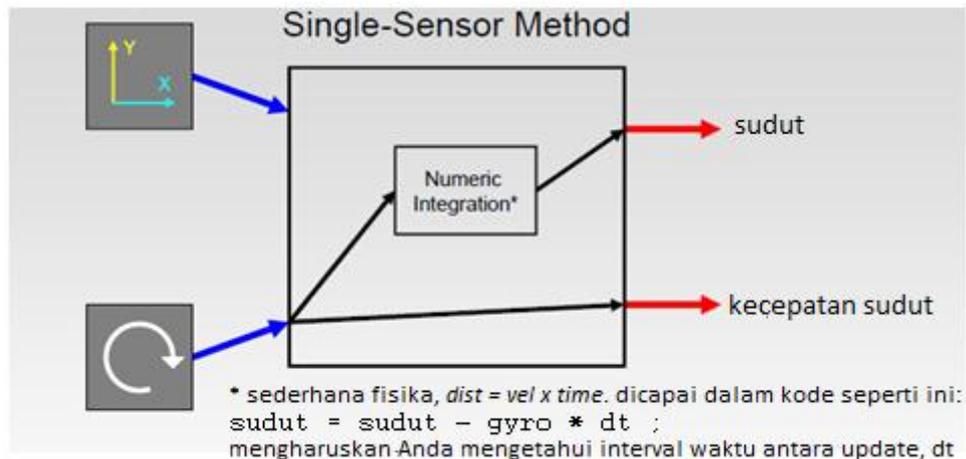


Gambar 5 Filter Digital

Sumber : [scolton@mit.edu](mailto:scolton@mit.edu)

## 2. Singular-Sensor

Penerapan dalam bahasa program yaitu dengan cara membaca masing masing nilai dari sensor dan di proses secara terpisah, kemudian dijumlahkan hingga didapatkan nilai perkiraan sudut.

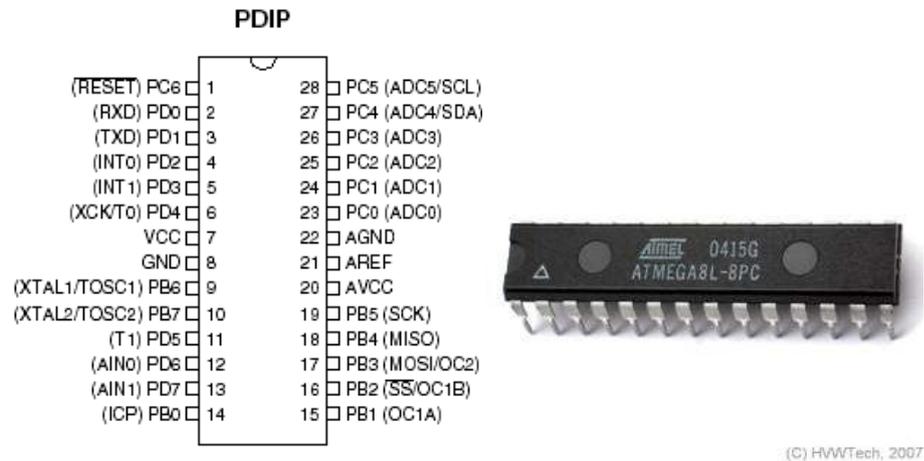


Gambar 6 Single-Sensor Method

## 7.4 Mikro Kontroler

Mikro kontroler adalah sebuah IC yang memiliki input dan output yang dapat diprogram sesuai kehendak yang diinginkan, fasilitas yang dimiliki oleh mikro kontroler jenis Atmega8 antara lain adalah menerima input digital dan analog (terdapat konverter ADC internal), output berupa sinyal digital dan analog

(berupa sinyal PWM), mempunyai pin khusus antar muka komunikasi serial (RX/TX), kelebihanannya adalah berbentuk kecil sehingga cocok untuk aplikasi yang sederhana, detail port i/o seperti pada gambar dibawah ini.



Gambar 7 ATmega8

Fitur :

- 8K byte self-programmable Flash
- 512byte EEPROM
- 1024byte internal SRAM
- Two 8bit timer
- One 16bit timer
- 3 PWM Channel
- 8-channel 10-bit ADC
- USART
- SPI
- WDT
- 23-programmable I/O lines

2.7-5.5V for ATmega8L (max 8MHz)

### 7.5 Program CodeVision AVR

CodeVisionAVR merupakan sebuah cross-compiler C, Integrated Development Environment (IDE), dan Automatic Program Generator yang didesain untuk mikrokontroler buatan Atmel seri AVR. CodeVisionAVR dapat dijalankan pada sistem operasi Windows 95, 98, Me, NT4, 2000, dan XP.

Cross-compiler C mampu menerjemahkan hampir semua perintah dari bahasa ANSI C, sejauh yang diijinkan oleh arsitektur dari AVR, dengan tambahan beberapa fitur untuk mengambil kelebihan khusus dari arsitektur AVR dan kebutuhan pada sistem *embedded*.

*File object* COFF hasil kompilasi dapat digunakan untuk keperluan *debugging* padatingkatan C, dengan pengamatan variabel, menggunakan *debugger* Atmel AVR Studio.

IDE mempunyai fasilitas internal berupa software AVR Chip In-System Programmer yang memungkinkan pengguna untuk melakukan transfer program kedalam chip mikrokontroler setelah sukses melakukan kompilasi/assembly secara otomatis. *Software* In-System Programmer didesain untuk bekerja dengan Atmel STK500/AVRISP/AVRProg, Kanda Systems STK200+/300, Dontronics DT006, Vogel Elektronik VTEC-ISP, Futurlec JRAVR dan MicroTronics ATCPU/Mega2000 *programmers/development boards*.

Untuk keperluan *debugging* sistem *embedded*, yang menggunakan komunikasi serial, IDE mempunyai fasilitas internal berupa sebuah Terminal.

Selain *library* standar C, CodeVisionAVR juga mempunyai *library* tertentu untuk:

- a) Modul LCD alphanumeric
- b) Bus I2C dari Philips
- c) Sensor Suhu LM75 dari National Semiconductor
- d) *Real-Time Clock*: PCF8563, PCF8583 dari Philips, DS1302 dan DS1307 dari Maxim/Dallas Semiconductor
- e) Protokol 1-Wire dari Maxim/Dallas Semiconductor
- f) Sensor Suhu DS1820, DS18S20, dan DS18B20 dari Maxim/Dallas Semiconductor
- g) Termometer/Termostat DS1621 dari Maxim/Dallas Semiconductor
- h) EEPROM DS2430 dan DS2433 dari Maxim/Dallas Semiconductor
- i) SPI
- j) *Power Management*
- k) Delay
- l) Konversi ke Kode Gray

CodeVisionAVR juga mempunyai Automatic Program Generator bernama CodeWizardAVR, yang mengijinkan pengguna untuk menulis, dalam hitungan menit, semua instruksi yang diperlukan untuk membuat fungsi-fungsi berikut:

- a) *Set-up* akses memori eksternal
- b) Identifikasi sumber *reset* untuk *chip*
- c) Inisialisasi port *input/output*
- d) Inisialisasi interupsi eksternal
- e) Inisialisasi *Timer/Counter*
- f) Inisialisasi *Watchdog-Timer*
- g) Inisialisasi UART (USART) dan komunikasi *serial* berbasis *buffer* yang digerakkan oleh interupsi
- h) Inisialisasi Pembanding Analog
- i) Inisialisasi ADC
- j) Inisialisasi Antarmuka SPI
- k) Inisialisasi Antarmuka Two-Wire
- l) Inisialisasi Antarmuka CAN
- m) Inisialisasi Bus I2C, Sensor Suhu LM75, *Thermometer/Thermostat* DS1621 dan *Real-Time Clock* PCF8563, PCF8583, DS1302, dan DS1307
- n) Inisialisasi *Bus 1-Wire* dan Sensor Suhu DS1820, DS18S20
- o) Inisialisasi modul LCD

## 7.6 Kontroler

Sistem pengendalian dirancang untuk melakukan dan menyelesaikan tugas tertentu. Syarat utama sistem pengendalian adalah harus stabil. Di samping kestabilan mutlak, maka sistem harus memiliki kestabilan secara relatif, yakni tolok ukur kualitas kestabilan sistem dengan menganalisis sampai sejauh mana batas-batas kestabilan sistem tersebut jika dikenai gangguan (Ogata.K.,1997). Selain itu analisis juga dilakukan untuk mengetahui bagaimana kecepatan sistem dalam merespons *input*, dan bagaimana peredaman terhadap adanya lonjakan (*over shoot*).

Suatu sistem dikatakan stabil jika diberi gangguan maka sistem tersebut akan kembali ke keadaan *steady state* di mana *output* berada dalam keadaan tetap seperti tidak ada gangguan. Sistem dikatakan tidak stabil jika *output*nya berosilasi terus menerus ketika dikenai suatu gangguan. Karena suatu sistem pengendalian biasanya melibatkan penyimpanan energi maka *output* sistem ketika diberi suatu *input*, tidak dapat mengikuti *input* secara serentak, tapi menunjukkan respons transien berupa suatu osilasi teredam sebelum mencapai *steady state*.

Dalam sistem pengendalian terdapat 2 macam *loop*:

1. Pengendalian dengan *loop* terbuka

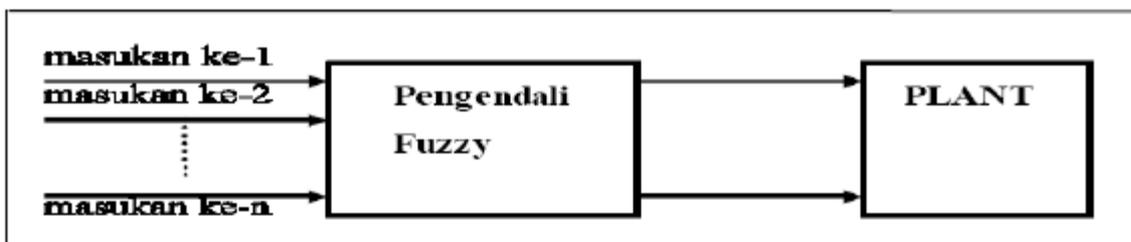
Sistem kontrol *loop* terbuka adalah sistem kontrol yang keluarannya tidak berpengaruh pada aksi pengontrolan. Jadi pada sistem kontrol *loop* terbuka, keluaran tidak diukur atau diumpun balikan untuk dibandingkan dengan masukan.

2. Pengendalian dengan *loop* tertutup

Sistem kontrol *loop* tertutup adalah sistem kontrol yang keluarannya mempunyai pengaruh langsung pada aksi pengontrolan. Disebut juga sistem kontrol yang menggunakan umpan balik untuk memperkecil kesalahan sistem.

### 7.6.1 Fuzzy Kontroler

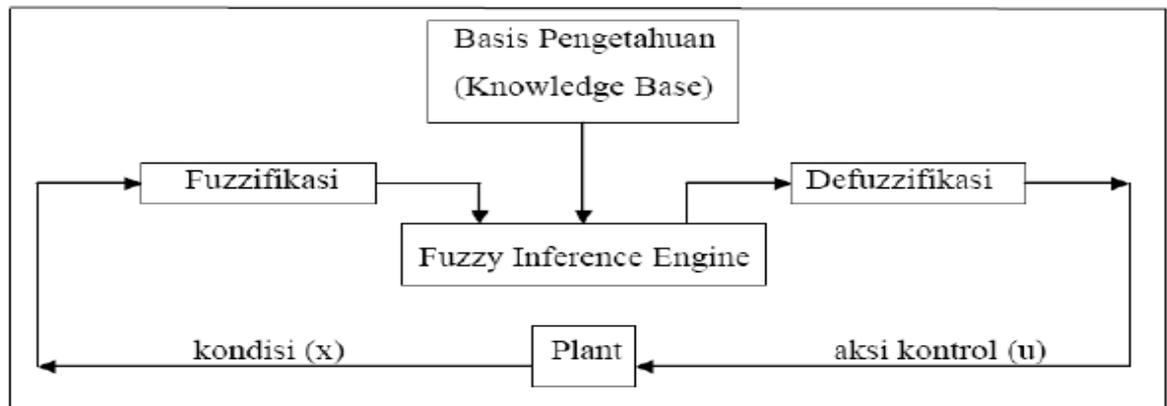
Dalam sistem pengendalian dengan logika *fuzzy* dilibatkan suatu blok pengendali yang menerima satu atau lebih masukan dan mengumpankan satu atau lebih keluaran ke *plant* atau blok lain sebagaimana terlihat dalam Gambar 8



Gambar 8 Pengendali Fuzzi

Sumber : Donald Coughanowr, 1991

Komponen utama penyusun *Fuzzy Logic Controller* adalah unit *fuzzifikasi*, *fuzzy inference*, basis pengetahuan dan unit *defuzzifikasi*. Struktur dasar *Fuzzy LogicController* dapat dilihat dalam Gambar 9.



Gambar 9 Struktur Dasar Logika Fuzzy

Sumber : Jun Yan, 1994.

Basis pengetahuan terdiri dari dua jenis :

- **Basis data**

Mendefinisikan parameter *fuzzy* sebagai bagian dari himpunan *fuzzy* dengan menentukan batas-batas fungsi keanggotaan pada semesta pembicaraan untuk tiap-tiap variabel

- **Basis aturan**

Memetakan nilai masukan *fuzzy* menjadi nilai keluaran *fuzzy*.

### 7.6.2 Fuzzifikasi

Proses fuzzifikasi merupakan proses untuk mengubah variabel non-fuzzy (variable numerik) menjadi variabel fuzzy (variabel linguistik). Nilai masukan-masukan yang masih dalam bentuk variabel numerik yang telah dikuantisasi sebelum diolah oleh pengendali logika fuzzy harus diubah terlebih dahulu ke dalam variable fuzzy. Melalui fungsi keanggotaan yang telah disusun, maka dari nilai-nilai masukan tersebut menjadi informasi fuzzy yang berguna nantinya untuk proses pengolahan secara fuzzy pula. Proses ini disebut fuzzifikasi. ( Jun Yan, 1994 : 49).

Proses fuzzifikasi diekspresikan sebagai berikut:

$x = \text{fuzzifier}(x_0)$

dengan :

$x_0$  = nilai craps variabel masukan  
 $x$  = himpunan fuzzy variabel yang terdefinisi  
fuzzifier = operator fuzzifikasi yang memetakan himpunan craps ke himpunan fuzzy

Pedoman memilih fungsi keanggotaan untuk proses fuzzifikasi, menurut Jun Yan, menggunakan :

1. Himpunan fuzzy dengan distribusi simetris.
2. Gunakan himpunan fuzzy dengan jumlah ganjil, berkaitan erat dengan jumlah kaidah (rules).
3. Mengatur himpunan fuzzy agar saling menumpuk.
4. Menggunakan fungsi keanggotaan bentuk segitiga atau trapezoidal.

### 7.6.3 Aturan Fuzzy (Fuzzy Rule)

*Fuzzy rule* adalah bagian yang menggambarkan dinamika suatu sistem terhadap masukan yang dikarakteristikan oleh sekumpulan variabel-variabel linguistik dan berbasis pengetahuan seorang operator ahli. Pernyataan tersebut umumnya dinyatakan oleh suatu pernyataan bersyarat. Dalam pengendali berbasis *fuzzy* aturan pengendalian *fuzzy* berbentuk aturan “ IF – THEN”. Untuk sebuah sistem MISO (*Multi Input Single Output*) basis aturan pengendalian *fuzzy* berbentuk seperti berikut ini.

Rule 1 IF  $X_1$  is  $A_{11}$  AND ... AND  $x_m$  is  $A_{1m}$  THEN  $Y$  is  $B_1$

Rule 2 IF  $X_1$  is  $A_{21}$  AND ... AND  $x_m$  is  $A_{2m}$  THEN  $Y$  is  $B_2$

..

..

..

Rule n IF  $X_n$  is  $A_{n1}$  AND ... AND  $x_m$  is  $A_{nm}$  THEN  $Y$  is  $B_n$

Dengan  $X_j$  merupakan variabel masukan sistem ,  $A_{ij}$  merupakan *fuzzy* set untuk  $X_j$ ,  $Y$  merupakan variabel keluaran sistem,  $B_i$  merupakan *fuzzy* set untuk  $Y$  , AND adalah *operator fuzzy*.

#### 7.6.4 Metode Inferensi

Metode *inferensi* merupakan proses untuk mendapatkan keluaran dari suatu kondisi masukan dengan mengikuti aturan-aturan yang telah ditetapkan. Keputusan yang didapatkan pada proses ini masih dalam bentuk *fuzzy* yaitu derajat keanggotaan keluaran. Ada beberapa cara untuk mengidentifikasi aturan mana yang akan dipakai dengan menggunakan nilai keanggotaan masukan. Menurut Jun Yan, diantara bermacam metode inferensi *fuzzy* ada dua metode yang paling sering digunakan pada *fuzzy logic control* yaitu :

##### 1. Metode INFERENSI MAX – MIN

Pada metode Max – Min aturan operasi minimum Mamdani digunakan untuk implikasi *fuzzy*. Persamaan aturan minimum adalah

$$\bigcup_1^n \alpha_i \wedge \mu_{ci}$$

Dengan

$$\alpha_i = \mu_{A_i}(x_0) \wedge \mu_{B_i}(y_0)$$

Sebagai contoh , terdapat dua basis kaidah atur *fuzzy*, yaitu :

R1 : Jika x adalah A1 dan y adalah B1 maka z adalah C1

R2 : Jika x adalah A2 dan y adalah B2 maka z adalah C2

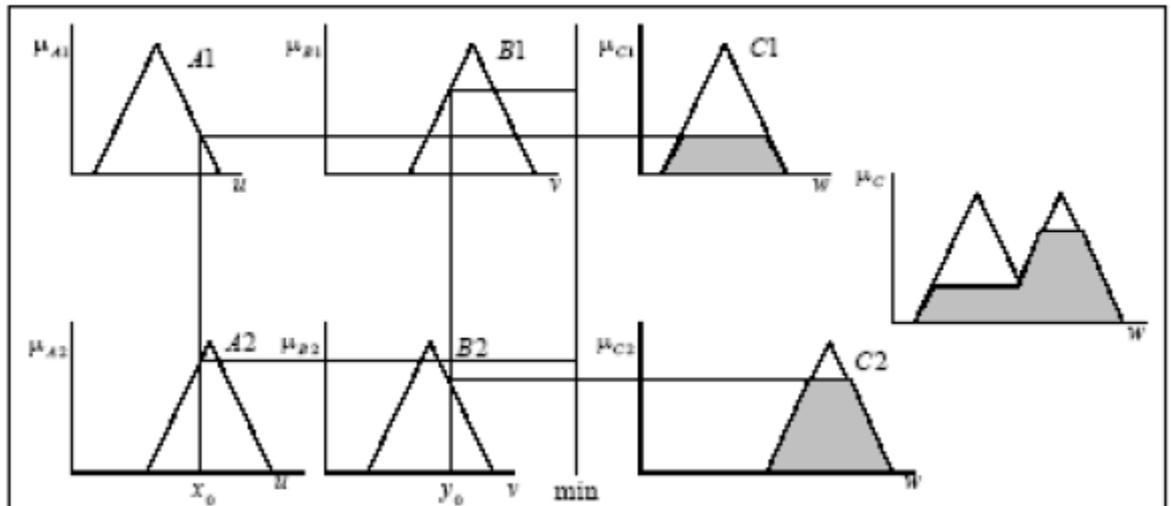
Pada metode penalaran MAX-MIN fungsi keanggotaan konsekuen dinyatakan dengan

Dimana

$$\alpha_1 = \mu_{A1}(x_0) \wedge \mu_{B1}(y_0)$$

$$\alpha_2 = \mu_{A2}(x_0) \wedge \mu_{B2}(y_0)$$

Lebih jelas metode ini dideskripsikan dalam Gambar 7.10



Gambar 10 Inferensi Fuzzy dengan metode MAX-MIN

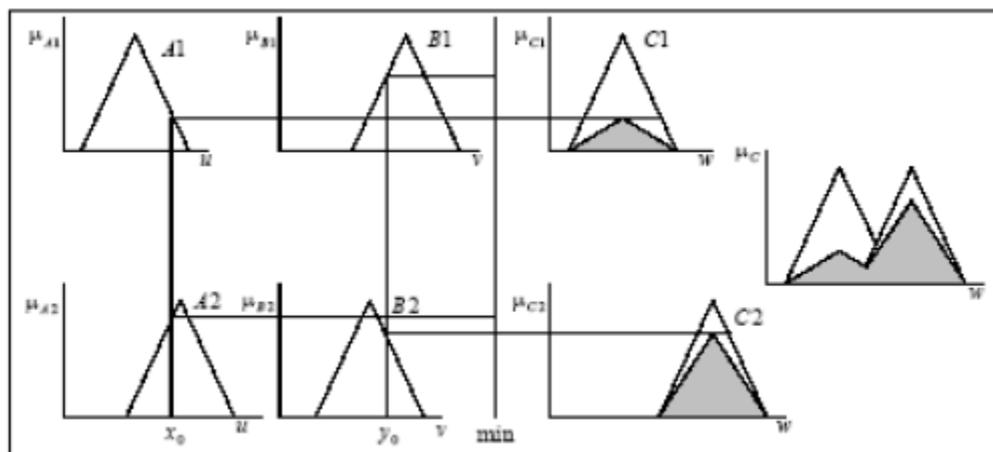
Sumber : Jun Yan, 1994 : 55

## 2. Metode INFERENSI MAX – DOT

Pada metode inferensi Max – Dot, aturan operasi *Larsen* digunakan untuk fungsi implikasi *fuzzy*. Metode MAX-DOT fungsi keanggotaan konsekuen C dinyatakan dengan:

$$\mu_c(\omega) = (\alpha_1 \cdot \mu_{c1}(\omega)) \vee (\alpha_2 \cdot \mu_{c2}(\omega))$$

Metode penalaran MAX-DOT diperlihatkan dalam Gambar 7.11.



Gambar 11 Inferensi Fuzzy dengan metode MAX-DOT

Sumber : Jun Yan, 1994 : 55

### 7.6.5 Defuzzifikasi

*Defuzzifikasi* adalah proses untuk mendapatkan nilai numerik dari data *fuzzy* yang dihasilkan dari proses *inferensi*. ( Jun Yan, 1994 : 55). Proses *defuzzifikasi* dinyatakan sebagai berikut :

$$y_0 = \text{defuzzifier}(y) \dots\dots\dots (2.10)$$

dengan:

- y : aksi kontrol *fuzzy*.
- y<sub>0</sub> : aksi kontrol *crisp*.
- Defuzzifier* : operator *defuzzifikasi*

Dua metode *defuzzifikasi* yang umum digunakan adalah :

#### 1. Metode COA(*Center Of Area*)

Metode ini didefinisikan sebagai berikut:

$$U = \frac{\sum_{i=1}^n w_i u_i}{\sum_{i=1}^n w_i}$$

dengan:

- U = *output*
- w<sub>i</sub> = bobot nilai benar w<sub>i</sub>
- u<sub>i</sub> = nilai linguistik pada fungsi keanggotaan *output*
- n = banyak derajat keanggotaan

## 2. Metode (*Mean of Maximum*)

Metode ini didefinisikan sebagai berikut :

$$Z = \frac{\sum_{i=1}^n z_i}{n}$$

dengan :

n = jumlah proposisi / *domain*

Z = *output*

Z<sub>i</sub> = *domain* yang memiliki derajat keanggotaan terbesar / maximum.

## 8. Metode Penelitian

Kajian dalam skripsi ini merupakan penelitian yang bersifat aplikatif, yaitu perencanaan dan pembuatan sistem keseimbangan otomatis pada miniatur sepeda roda dua yang dikendalikan oleh FLC dalam sistem pengendalian sudutnya. Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang akan dibuat adalah sebagai berikut:

### 8.1. Perencanaan Sistem

Perencanaan sistem dilakukan sebagai langkah awal sebelum terbentuknya suatu sistem beserta rangkaian elektronik pendukungnya, hal ini dimaksudkan agar sistem keseimbangan otomatis pada miniatur sepeda roda dua tersebut dapat berjalan sesuai dengan yang deskripsi awal yang telah direncanakan.

Perancangan sistem yang dilakukan meliputi:

- Penentuan spesifikasi sistem yang akan dibuat, meliputi:
  - a. Penentuan deskripsi kerja sistem secara keseluruhan.
  - b. FLC dan rangkaian elektronik pendukung.
- Penentuan rangkaian yang digunakan.

## 8.2. Perangkat Lunak

Pembuatan perangkat lunak (*software*) dilakukan dengan pembuatan *flowchart* terlebih dahulu, kemudian dilanjutkan dengan pembuatan programnya. Bahasa pemrogramannya menggunakan *software* program CodeVision AVR.

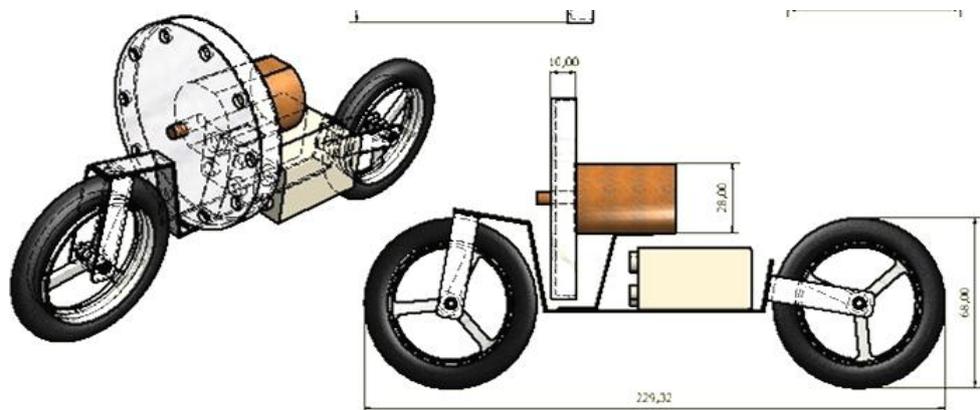
## 8.3. Model Mekanik

Referensi model mekanik “Motor bike robot” dari Budapest University of Technology oleh Kamran Ghaffari Toiserkan.



Gambar 12 referensi model mekanik

Rencana desain mekanik yang akan dibuat.



Gambar 13 desain mekanik

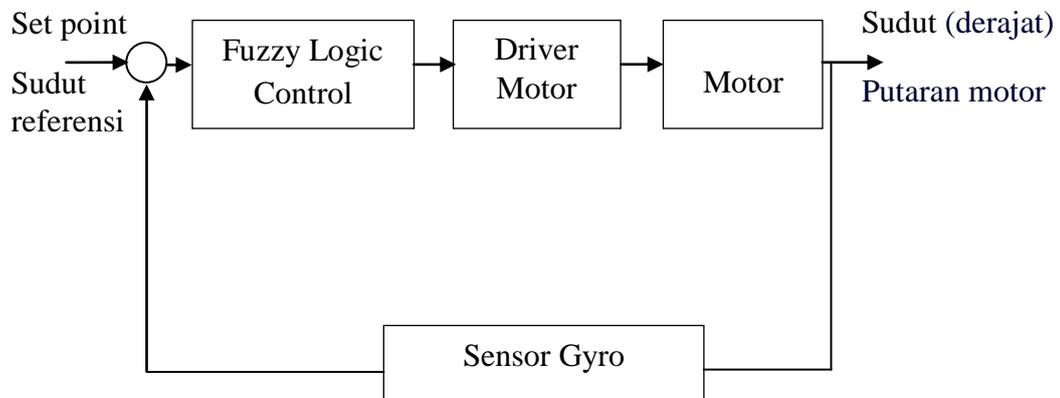
#### 8.4. Pengujian Sistem

Pengujian ini dilakukan untuk mengetahui apakah sistem keseimbangan otomatis pada miniatur sepeda roda dua secara keseluruhan telah bekerja sesuai dengan yang telah direncanakan sebelumnya atau tidak.

#### 8.5. Pengambilan Kesimpulan

Kesimpulan diambil berdasarkan data yang diperoleh dari pengujian sistem secara keseluruhan. Jika hasil yang didapatkan telah sesuai dengan yang direncanakan sebelumnya, maka sistem kendali tersebut telah berhasil memenuhi harapan dan tentunya memerlukan pengembangan lebih lanjut untuk penyempurnaannya.

#### 8.6. Blok Diagram





## DAFTAR PUSTAKA

Donald Coughanowr, 1991. *Process Systems Analysis And Control*

Bishop, O., 2002, *Dasar-Dasar Elektronika*, Erlangga, Jakarta

Ogata, Katsuhiko. 1997. *Teknik Kontrol Automatik (Sistem Pengaturan)*.  
Erlangga. Jakarta.

Petruzella, F. D., 1996, *Elektronik Industri*, ANDI, Yogyakarta