

## **PENGATURAN LAMPU LALU LINTAS BERBASIS *FUZZY LOGIC* MENGUNAKAN KOMPUTER IBM PC-XT 8088**

Oleh :

**Mochamad Wahyudi, S.Kom**

### **ABSTRAK**

Suatu sistem peralatan elektronik yang ditangani oleh komputer, maka semuanya akan terasa lebih canggih, lebih pintar, lebih otomatis, lebih praktis, lebih efisien, lebih aman, lebih teliti dan sebagainya, hal tersebut yang menunjukkan keuntungan-keuntungan apabila dibandingkan dengan pengerjaan yang dilakukan secara manual.

Kemampuan komputer dapat diberdayakan atau ditingkatkan melalui peningkatan kemampuan unjuk kerja (*performance*) perangkat keras (*hardware*) atau pada perangkat lunak (*software*) atau perpaduan keduanya. Kemampuan inilah yang menjadikan syarat untuk mewujudkan terciptanya suatu alat yang berfungsi sebagai peralatan Sistem Pengaturan Lampu Lalu Lintas Berbasis *fuzzy logic* menggunakan Komputer IBM PC-XT 8088.

Pengatur lampu lalu lintas ini pada awalnya menggunakan teknologi rangkaian logika yang berfungsi mengatur atau mengontrol lampu berdasarkan beberapa daftar waktu tetap yang sudah diset terlebih dahulu. Metode ini disebut dengan istilah *fixed time*. Kinerja pengatur lampu lalu lintas seperti ini dalam hubungannya dengan fungsi untuk mengatur waktu tunda (*delay time*) kendaraan bermotor sangatlah ditentukan oleh survey yang diperoleh, yang sangat tergantung sekali dengan kondisi statistik lalu lintas kendaraan yang ada pada suatu persimpangan jalan tersebut. Kinerja pengontrol lampu lalu lintas ini akan dirasakan sangat optimal sekali jika setiap terjadi perubahan kondisi lalu lintas dilakukan penyesuaian dengan cara menset ulang atau penjadwalan berdasarkan jam.

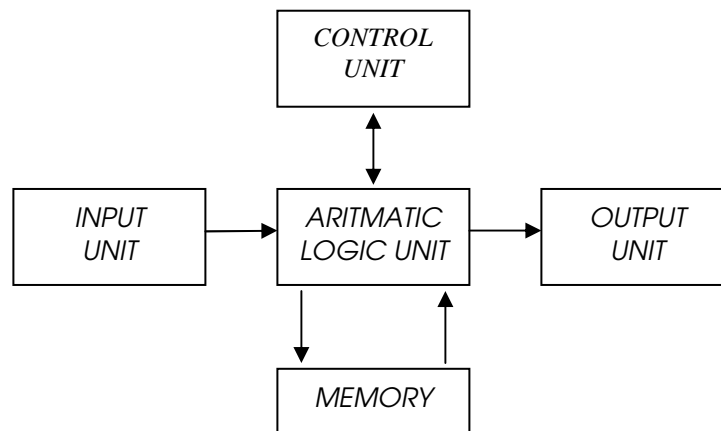
Lampu lalu lintas memegang peranan penting dalam pengaturan kelancaran lalu lintas. Sistem pengendalian lampu lalu lintas yang baik akan secara otomatis menyesuaikan diri dengan kepadatan arus lalu lintas pada jalur yang diatur. Dengan penerapan *logika fuzzy* maka sistem pengaturan lampu lalu lintas seperti sangat memungkinkan untuk dapat dilakukan.

## BAB I

### PENDAHULUAN

#### 1.1 Sistem Dasar Komputer IBM PC-XT 8088

Setiap komputer pada dasarnya memiliki lima bagian dasar, yaitu : *Arithmetic Logic Unit (ALU)*, *Control Unit (CU)*, *Memory*, *Input Unit* dan *Output Unit (I/O Unit)*. Adapun diagram blok dari sistem komputer tersebut terlihat pada gambar 1.1.

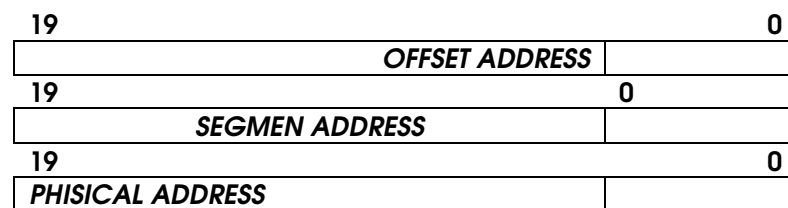


Gambar 1.1 Diagram blok sistem komputer

Komputer IBM PC-XT (*Extra Technology*) menggunakan Microprocessor 8088 keluaran Intel. *Microprocessor* ini menggunakan perintah-perintah dalam bentuk 16-bit, meskipun memiliki lebar *word* keluaran data sebesar 8-bit. *Microprocessor* ini menggunakan jalur alamat (*address bus*) sebesar 20-bit untuk menunjukkan lokasi memori, yang berarti secara langsung memori menempatkan 1.048.567 memori.

*Microprocessor* 8088 hanya menangani *word* sepanjang 16-bit, maka *microprocessor* ini memperluas alamat *word* menjadi 20-bit pada keluarannya

dengan skema segmentasi. Alamat-alamat memori secara logika dibagi-bagi lagi menjadi segmen khusus yang masing-masing berukuran 64KB. *Segmen* tersebut dialokasikan ke *register segmen* khusus di dalam *Microprocessor* 8088. *Byte* dalam sebuah *segmen* dialamatkan dengan menggunakan alamat *offset* 16-bit. Alamat fisik 16-bit disusun di dalam *Microprocessor* 8088 dengan cara menambahkan alamat *offset* 16-bit ke alamat *segmen* 16-bit dengan alamat yang digeser ke kiri sebesar satu *Hexadecimal*. (Lihat gambar 1.2)



Gambar 1.2. Peta Alamat 20-bit dari alamat *segmen* dan alamat *offset*

*Microprocessor* 8088 memeriksa memori, mengambil instruksi dari lokasi dan menginterpretasikan instruksi untuk melakukan kegiatan yang diperintahkan. CPU akan menjalankan satu perintah ke perintah lain secara urut, atau CPU akan menuju ke lokasi memori sesuai yang diperintahkan oleh program.

*Microprocessor* 8088 didukung oleh beberapa *chip* pendukung agar dapat berfungsi sebagai satu kesatuan yang utuh. *Chip* pendukung tersebut antara lain terdiri dari : *Clock Generator* 8284, *Programable Interupt Controller* (PIC) 8259, *Programable Peripheral Interface* (PPI) 8255, *Programable Interval Timer* (PIT) 8253, *Direct Memory Access* (DMA) 8237.

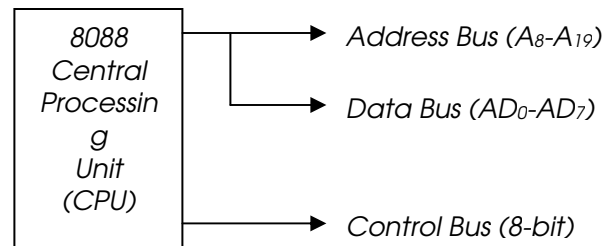
Tabel 1.1. Peta Alamat Memori

ALAMAT PORT	FUNGSI
000H-00FH	DMA Controller 8237A-5
020H-021H	8259 Programable Interup Controller
040H-043H	8253 Programable Interval Timer
060H-063H	8255A Programable Peripheral Interface
080H-083H	DMA Page Register
00AH	NMI Mask
00CH	Cadangan
00EH	Cadangan
200H-20FH	Game Control Adapter
210H-217H	Expansion Unit
220H-24FH	Cadangan
278H-27FH	Cadangan
2F0H-2F7H	Cadangan
2F8H-2FFH	Asynchronous Communication
300H-31FH	Prototyp Card
320H-32FH	Hard Disk Controller
378H-37FH	Pararel Printer Controller
380H-38FH	SDLC Comunication Controller
3A0H-3AFH	Cadangan
3B0H-3BFH	IBM Monochrome Display/Printer
3C0H-3CFH	Color Graphic Adapter
3E0H-3E7H	Cadangan
3F0H-3F7H	Floppy Disk Drive Controller
3F8H-3FFH	Asynchronous Communication

Sumber : *Technical Reference for IBM PC-XT 8088* (Diterjemahkan ke dalam bahasa Indonesia)

## 1.2 Sistem Bus IBM PC-XT 8088

Menurut buku *Technical Reference for IBM PC-XT 8088* sinyal kontrol, alamat dan data dihubungkan antara CPU dengan sistem yang lainnya melalui bus pada *motherboard (mainboard)*. IBM PC-XT 8088 memiliki rancangan bus yang telah maju dengan semua jalur keluaran data dan alamat yang di-*buffer* separuhnya. Adapun bus yang dimiliki oleh IBM PC-XT 8088 yaitu : *data bus*, *address bus*, dan *control bus*.



Gambar 1.3. Bus Utama pada IBM PC-XT 8088

Pada *Slot Motherboard* akan dijumpai semua *bus* ini, kecuali *bus* tersebut pada *Slot* juga dijumpai juga jalur daya dan jalur pengatur waktu yang kesemuanya berjumlah 62 jalur.

Penjelasan mengenai sinyal yang terdapat pada *slot card* pada IBM PC-XT 8088 menurut buku *Technical Reference for IBM PC-XT 8088* dapat dilihat pada tabel 1.2.

Tabel 1.2. Definisi Sinyal I/O pada *Expansion Slot Card* IBM PC-XT 8088

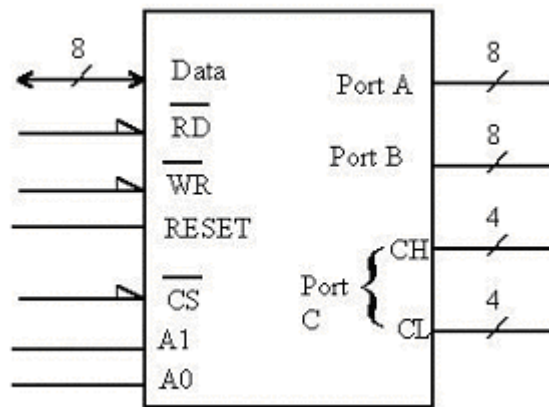
NO	SINYAL	I/O	KETERANGAN
1	OSC	O	Oscilator periode 70 ns (14,31818 MHz) dengan <i>Duty Cycle</i> 50%.
2	CLK	O	Hasil pembagian dari tiga sinyal OSC, periodanya 210 ns (4,77 MHz) <i>Duty Cycle</i> .
3	RESET DRV	O	Sinyal ini digunakan untuk mereset atau inisialisasi <i>logic</i> sistem catu daya sistem mulai dihidupkan. Sinyal ini disinkronkan dengan saat tepi naik sinyal CLK, dan sinyal ini aktif lagi.
4	A <sub>0</sub> - A <sub>19</sub>	O	bit alamat 0-19, sinyal ini digunakan untuk mengamati lokasi memori dan peralatan I/O dalam sistem. Dengan 20 jalur ini dapat dibentuk sampai 1 MB lokasi memori. Hanya A <sub>0</sub> -A <sub>15</sub> saja yang aktif saat Microprocessor menginginkan berhubungan dengan I/O. A <sub>0</sub> adalah Least Significant bit (LSB) dan A <sub>19</sub> adalah <i>Most Significant bit</i> (MSB). Sinyal aktif <i>high</i> .
5	D0-D7	O	bit data 0-7, sinyal ini menyediakan Data bus bit 0-7 untuk Microprocessor, Memori dan Peralatan I/O, bit D0 adalah LSB dan D7 adalah MSB. Sinyal ini aktif <i>high</i> .
6	ALE	O	<i>Address Lact Enable</i> . Sinyal ini disediakan oleh Chip 8288 <i>Control bus</i> dan digunakan untuk memberitahu sistem <i>board</i> digunakan untuk menahan data alamat yang sah dari Microprocessor. Data dari Microprocessor ditahan pada sisi naik ALE.
7	I/O CH CK	I	<i>I/O Chanel Check</i> . Sinyal ini menyediakan informasi <i>Parity (Error)</i> pada memori atau peralatan I/O bagi Microprocessor. Ketika sinyal ini <i>low</i> , menyatakan adanya kesalahan.
8	I/O CH RDY	I	<i>I/O Chanel Ready</i> . Sinyal ini berfungsi untuk memperpanjang siklus <i>Bus</i> . Ketika memori atau peralatan atau peralatan I/O bekerja kurang cepat bila dibandingkan dengan siklus <i>Bus</i> normal, maka peralatan ini dapat meminta pada <i>Control bus</i> untuk memperpanjang siklusnya, sehingga pada siklus tersebut dapat diperpanjang sampai dengan 10X siklus <i>Clock</i> (1 Siklus = 21 ns).
9	IRQ <sub>2</sub> -	I	<i>Interupt Request 2-7</i> . Sinyal ini digunakan oleh peralatan I/O untuk meminta

	IRQ <sub>7</sub>		pelayanan. Pelayanan akan ditanggapi berdasarkan urutan Prioritas paling tinggi.
10	IOR	○	I/O <i>Read</i> . Sinyal ini mengharuskan peralatan I/O yang ditunjuk, untuk menyerahkan datanya ke <i>Data bus</i> . Sinyal ini dikontrol oleh Microprocessor atau pengontrol DMA. Sinyal ini aktif <i>low</i> .
11	IOW	○	I/O <i>Write</i> . Sinyal ini mengharuskan peralatan I/O yang ditunjuk, untuk membaca data yang terdapat pada <i>Data bus</i> . Sinyal ini dikontrol oleh Microprocessor atau pengontrol DMA. Sinyal ini aktif <i>low</i> .
12	MEMR	○	Memori <i>Read</i> . Sinyal ini mengharuskan memori yang ditunjuk, untuk menyerahkan datanya ke <i>Data bus</i> . Sinyal ini dikontrol oleh Microprocessor atau pengontrol DMA. Sinyal ini aktif <i>low</i> .
13	MEMW	○	Memori <i>Write</i> . Sinyal ini mengharuskan memori yang ditunjuk, untuk membaca data yang terdapat pada <i>Data bus</i> . Sinyal ini dikontrol oleh Microprocessor atau pengontrol DMA. Sinyal ini aktif <i>low</i> .
14	DRQ <sub>1</sub> – DRQ <sub>3</sub>	I	DMA Request1-3. Sinyal ini dipergunakan oleh <i>Interface</i> untuk meminta pelayanan <i>Control DMA</i> . Permintaan ini dilakukan jika diinginkan terjadi pemindahan data dari <i>Interface</i> ke memori tanpa harus melalui Microprocessor, digunakan untuk mempersingkat waktu pemindahan karena datanya dalam jumlah yang besar. Sinyal ini juga mempunyai prioritas pelayanan. DRQ <sub>1</sub> mempunyai prioritas yang paling tinggi dan mempunyai DRQ <sub>3</sub> prioritas paling rendah.
15	DACK <sub>0</sub> – DACK <sub>3</sub>	○	DMA <i>Acknowledge</i> 0-3. Sinyal ini dipergunakan sebagai pemberitahuan pelayanan DMA karena DRQ <sub>1</sub> -DRQ <sub>3</sub> . Sedangkan DACK <sub>0</sub> dipergunakan sebagai sinyal penyegar data pada <i>Dynamic Memory</i> (Memori Dinamik). Sinyal ini aktif <i>low</i> .
16	AEN	○	<i>Address Enable</i> . Sinyal ini dipergunakan untuk memisahkan Microprocessor dan menghubungkan DMA dengan peralatan lainnya. Pada saat sinyal ini aktif ( <i>high</i> ), pengontrol DMA mengontrol <i>Data bus</i> dalam pelaksanaan perintah baca dan tulis antara memori dan peralatan I/O.
17	T/C	○	Terminal Count.
18	CARD SLCT	I	<i>Card Select</i> .

Sumber : *Technical Reference for IBM PC-XT 8088* (Diterjemahkan ke dalam bahasa Indonesia)

### 1.3 Chip Proramable Peripheral Interface (PPI) 8255

*Chip Programable Peripheral Interface* (PPI) 8255 adalah perangkat perantara yang dapat diprogram untuk memasukan/keluaran (*input/output*). *Chip* ini dirancang untuk mendukung *Microprocessor* 8088, *Chip* 8255 memiliki 40 pin atau kaki, piranti ini mempunyai 24 pin masukan/keluaran serta terbagi menjadi tiga kelompok atau *port*, yaitu : *port* A (8 pin), *port* B (8 Pin) dan *port* C (8 Pin). Pada *port* C terbagi atas dua bagian, yaitu : *port* C atas dan *port* C bawah yang masing-masing terdiri dari empat pin. Berikut ini adalah gambar dari skema pin *Chip* PPI 8255.



Gambar 1.4. Skema Pin *Chip* PPI 8255

Adapun penjelasan mengenai fungsi dari masing-masing *Pin* menurut buku *Technical Reference for IBM PC-XT 8088* adalah sebagai berikut :

- a. Pin RD (*Read=Baca*) berfungsi untuk memberikan perintah ke Chip 8255 pada operasi baca CPU, jika Pin ini diberikan sinyal rendah (0), maka akan memungkinkan *Chip* 8255 untuk mengirimkan data ke CPU melalui *Data bus*, pada prinsipnya ini memungkinkan CPU membaca dari 8255.
- b. Pin WR (*Write=Tulis*) berfungsi untuk memberikan perintah pada operasi tulis pada CPU, jika Pin ini diberikan sinyal rendah (0). Pin A<sub>1</sub> dan A<sub>0</sub> berfungsi sebagai kontrol untuk memilih satu dari tiga gerbang (*port*) atau *Register Control Word* (CW).
- c. *Pin* A<sub>1</sub> dan A<sub>0</sub> secara normal dihubungkan dengan LSB (*Least Signifucan bit*) dari *bus* alamat.

Adapun alamat dari masing-masing gerbang dalam *chip* 8255 ditunjukkan dalam tabel 1.3 berikut ini :

Tabel 1.3. Alamat dari gerbang masukan/keluaran pada *chip* 8255

A <sub>1</sub>	A <sub>0</sub>	Gerbang
0	0	<i>port A</i>
0	1	<i>port B</i>
1	0	<i>port C</i>
1	1	<i>Control Word</i>

Sumber : *Technical Reference for IBM PC-XT 8088* (Diterjemahkan ke dalam bahasa Indonesia)

- d. *Pin CS (Chip Select)* berfungsi untuk komunikasi antara 8255 dan CPU. *Pin* ini aktif pada kondisi rendah (*low*).
- e. *Pin Reset* berfungsi untuk mereset isi dari *Register Control Word* dan semua *port* diset untuk *mode* masukan (*input*). *Pin* ini aktif apabila diberikan sinyal tinggi (*high*).
- f. *Bus Data* berfungsi sebagai perantara dari 8255 ke sistem *bus* data pada *Microprocessor*. Data dikirim atau diterima melalui *bus* data setelah perintah instruksi masukan/keluaran oleh CPU. *Control Word* dan informasi juga dikirim melalui *bus data*. *Pin* ini besarnya 8 bit dan merupakan *bus* tiga keadaan (*3-State*).
- g. *Pin PA<sub>0</sub>-PA<sub>7</sub>* merupakan *port A* yang besarnya *8-bit*. *port A* merupakan tempat masukan/keluaran data.
- h. *Pin PB<sub>0</sub>-PB<sub>7</sub>* merupakan *port B* yang besarnya *8-bit*. *port B* merupakan tempat masukan/keluaran data.
- i. *Pin PC<sub>0</sub>-PC<sub>7</sub>* merupakan *port C* yang besarnya *8-bit*. *port C* dibagi menjadi dua bagian, yaitu : *port C* atas (*PC<sub>4</sub>-PC<sub>7</sub>*) dan *port C* bawah (*PC<sub>0</sub>-PC<sub>3</sub>*) yang masing-masing besarnya *4-bit*.

Pada *Chip Intel PPI (Programmable Peripheral Interface) 8255* mempunyai tiga mode kerja, yaitu :

a. Mode 0

Pada mode ini *port A (PA<sub>0</sub>-PA<sub>7</sub>)*, *port C atas (PC<sub>4</sub>-PC<sub>7</sub>)*, *port C bawah (PC<sub>0</sub>-PC<sub>3</sub>)* dan *port B (PB<sub>0</sub>-PB<sub>7</sub>)* berfungsi sebagai masukan atau keluaran (*Input/Output*).

b. Mode 1

Pada mode ini *port A (PA<sub>0</sub>-PA<sub>7</sub>)* dan *port B (PB<sub>0</sub>-PB<sub>7</sub>)* berfungsi sebagai masukan dan keluaran (*Input/Output*). Sedangkan *port C* sebagai kontrol. *Port C atas (PC<sub>4</sub>-PC<sub>7</sub>)* untuk mengontrol *port A* dan *port C bawah (PC<sub>0</sub>-PC<sub>3</sub>)* untuk mengontrol *port B*.

c. Mode 2

Pada mode ini *port A (PA<sub>0</sub>-PA<sub>7</sub>)* bersifat dua arah (*bi-directional*), sedangkan *port C atas (PC<sub>4</sub>-PC<sub>7</sub>)*, satu *port C bawah (PC<sub>0</sub>-PC<sub>3</sub>)* digunakan sebagai kontrol melayani transfer data pada *port A*. Pada mode ini *port B* dapat dibuat seperti pada mode 0 atau mode 1.

Pemilihan mode dan apakah suatu *port* berlaku sebagai masukan atau keluaran ditentukan oleh isi dari *register Control Word*. *Register* ini terdiri dari 8-bit, dan isinya dapat ditulis oleh CPU dengan instruksi *out address port* dengan *address port* adalah suatu alamat yang membuat  $A_1=1$  dan  $A_0=1$ . *Register Control Word* ini hanya dapat ditulis (*Write Only*), tidak dapat dibaca (*Read*).

### 1.4. Register Control Word

Menurut buku *Technical Reference for IBM PC-XT 8088 Register Control Word* padat ditulis oleh CPU dengan instruksi *out control word*, dengan *Control Word* menyatakan alamat dengan  $A_0=1$  dan  $A_1=1$ . Dengan instruksi ini maka isi *Accumulator* akan disalin (dikopi) ke *Control Word*. Arti setiap *bit* dalam *Register Control Word* dapat dilihat pada gambar 1.5 berikut ini :

D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

*Keterangan :*

D7	: <i>Flag mode.</i>	1	= Mode aktif,	0	= <i>Bit set/Reset port C</i>
D6-D5	: <i>Petunjuk mode.</i>	00	= Mode 0,	01	= Mode 1,
				1X	= Mode 2
D4	: <i>Petunjuk pada port A.</i>	1	= Masukan,	0	= Keluaran
D3	: <i>Petunjuk pada port C atas.</i>	1	= Masukan,	0	= Keluaran
D2	: <i>Petunjuk mode pada port B.</i>	1	= Masukan,	0	= Keluaran
D1	: <i>Petunjuk pada port B .</i>	1	= Masukan,	0	= Keluaran
D0	: <i>Petunjuk pada port C bawah.</i>	1	= Masukan,	0	= Keluaran

Gambar 1.5. *Register Control Word*

### 1.5 Teknologi Sistem Fuzzy

Pada generasi pertama teknologi *fuzzy*, terdapat beberapa kendala yang ditemui untuk mengembangkan pada industri-industri atau sistem kendali yang telah ada. Saat itu belum ada metodologi yang sistematis tentang aplikasi pengendali *fuzzy*, penentuan rancang bangun yang tepat, analisa permasalahan, dan bagaimana pengaruh perubahan parameter sistem terhadap kualitas unjuk kerja sistem. Jadi tidak bisa diharapkan suatu rancang bangun yang *universal* dan strategi optimasi *fuzzy* dapat segera digunakan secara praktis.

Saat ini logika *fuzzy* telah berhasil menerobos kendala-kendala yang dahulu pernah ditemukan dan segera menjadi basis teknologi tinggi. Penerapan teori logika

ini dianggap mampu menciptakan sebuah revolusi dalam teknologi. Sebagai contoh, mulai tahun 90-an para industri manufaktur, yang bergerak pada bidang *Distributed Control System* (DCS), *Programmable Controllers* (PLC), dan *Microcontrollers* (MCU) telah menyatukan sistem logika *fuzzy* pada barang produksi mereka dan memiliki prospek ekonomi yang baik.

Menurut Bartos, Frank J (Juli, 1992), sebuah perusahaan *microprosesor* terkemuka, Motorola, dalam sebuah jurnal teknologi, pernah menyatakan "bahwa logika *fuzzy* pada masa-masa mendatang akan memainkan peranan penting pada sistem kendali digital". Pada saat yang bersamaan, pertumbuhan yang luar biasa terjadi pada industri perangkat lunak yang menawarkan kemudahan penggunaan logika *fuzzy* dan penerapannya pada setiap aspek kehidupan sehari-hari.

Menurut Hellendoorn, Hans dan Palm, Rainer (1994), Perusahaan Jerman Siemens yang bergerak diberbagai bidang teknik, seperti : otomatisasi industri, pembangkit tenaga, semikonduktor, jaringan komunikasi umum dan pribadi, otomotif dan sistem transportasi, sistem audio dan video, dan lain sebagainya, beberapa tahun belakangan ini telah membentuk kelompok riset khusus tentang *fuzzy*. Tujuannya untuk melakukan penelitian dan pengembangan yang sistematis tentang logika *fuzzy* pada setiap aspek teknologi. Ada dua alasan utama yang mendasari pengembangan teknologi berbasis sistem *fuzzy* :

- a. Menjadi *state-of-the-art* dalam sistem kendali berteknologi tinggi. Jika diamati pengalaman pada negara-negara berteknologi tinggi, khususnya di negara Jepang, pengendali *fuzzy* sudah sejak lama dan luas digunakan pada industri-industri dan alat-alat elektronika. Daya gunanya dianggap melebihi dari pada teknik kendali yang pernah ada. Pengendali *fuzzy* terkenal karena

keandalannya, mudah diperbaiki, dan yang lebih penting lagi pengendali *fuzzy* memberikan pengendalian yang sangat baik dibandingkan teknik lain, yang biasanya membutuhkan usaha dan dana yang lebih besar.

- b. Dalam perspektif yang lebih luas, pengendali *fuzzy* ternyata sangat bermanfaat pada aplikasi-aplikasi sistem identifikasi dan pengendalian *ill-structured*, dimana linieritas dan invariansi waktu tidak bisa ditentukan dengan pasti, karakteristik proses mempunyai faktor *lag*, dan dipengaruhi oleh *derau* acak. Bentuk sistem seperti ini jika dipandang sistem konvensional sangat sulit untuk dimodelkan.

Menurut Hellendoorn, Hans dan Palm, Rainer (1994), beberapa proyek teknologi yang dinilai digunakan dan memiliki prospek ekonomi yang cerah seperti :

- a. Dalam teknologi otomotif  
Sistem transmisi otomatis *fuzzy* dan pengendali kecepatan *idle fuzzy*.
- b. Dalam teknologi transportasi  
Pengendali *fuzzy* anti selip untuk kereta listrik, sistem pengaturan dan perencanaan perparkiran, sistem pengaturan lampu lalu lintas (*traffic light*), dan pengendalian kecepatan kendaraan di jalan bebas hambatan (jalan tol).
- c. Dalam peralatan sehari-hari  
Mesin cuci *fuzzy* dan *vacum cleaner fuzzy*, dan lain-lain.
- d. Dalam aplikasi industri  
Di antaranya adalah : industri kimia, sistem pengolahan kertas, dan lain-lain.
- e. Dalam *power stations*  
Sistem diagnosis kebocoran-H<sub>2</sub>

Masih banyak aplikasi lainnya yang sudah beredar sebagai alat kendali dan barang-barang elektronik berteknologi tinggi.

### **1.5.1 Kendali Perkembangan Teknologi Sistem Fuzzy**

Keberhasilan penerapan teknologi *fuzzy* seperti yang telah dijelaskan sebelumnya, dapat direalisasikan jika terdapat penelitian dan strategi pengembangan riset dan desain oleh sebuah industri untuk menemukan teknik terbaik untuk produknya. Hal tersebut tentunya tidak terlepas dari kesulitan-kesulitan yang ditemui dalam menggunakan dan pengembangan teknologi ini. Menurut Hellendoorn, Hans dan Palm, Rainer (1994), secara garis besar beberapa kesulitan yang ditemui oleh industri-industri elektronika adalah sebagai berikut :

- a. Para *engineer* dan ilmuwan generasi sebelumnya dan sekarang banyak yang tidak mengenal teori kendali *fuzzy*, meskipun secara teknik praktis mereka memiliki pengalaman untuk menggunakan teknologi dan peralatan kontrol yang sudah ada.
- b. Masih belum banyak terdapat tempat kursus atau balai pendidikan dan buku-buku bacaan (buku referensi) yang menjangkau setiap tingkat pendidikan (*undergraduate, postgraduate, dan on site training*)
- c. Sampai dengan saat ini belum ada pengetahuan yang sistematis yang sudah baku (*standard*) dan seragam tentang metode pemecahan masalah kendali (*control*) menggunakan pengendali *fuzzy*.
- d. Belum adanya metode umum (*general*) untuk mengembangkan dan implementasi pengendali *fuzzy*.

Kendala pertama dan kedua dapat diatasi dengan cara sering diadakannya kursus dan balai pendidikan, memperbanyak penulisan karya-karya ilmiah dan juga pengadaan buku-buku tentang *fuzzy* pada setiap perguruan tinggi atau institusi pendidikan lainnya atau bahkan dijual di toko-toko buku. Kendala ke tiga dan ke empat dapat diatasi dengan cara membuat suatu metode untuk merancang dan mengembangkan sistem *fuzzy*. Metode ini mencakup fasilitas-fasilitas yang terdapat dalam teori sistem kendali *fuzzy* seperti : pemilihan fungsi keanggotaan, operator, penggunaan faktor skala, pengembangan basis pengetahuan, penurunan basis aturan, uji coba, dan simulasi sistem.

Menurut Bartos, Frank J (Juli, 1992), perusahaan elektrik, Omron selain menjual produknya, kini mereka juga tengah mengembangkan metode pendidikan dan pelatihan tentang teknik logika *fuzzy*. Asisten manajer Omron FA System Div, Jim Krill menyatakan "*Educating potential customers about the benefits of fuzzy logic and where it can be applied is important for proper development of this technology*". Jadi cara terbaik untuk mencapai teknologi ini menurutnya adalah melalui program pelatihan, seminar, dan pemakaian perangkat lunak (*software*) simulasi sistem *fuzzy* yang efektif.

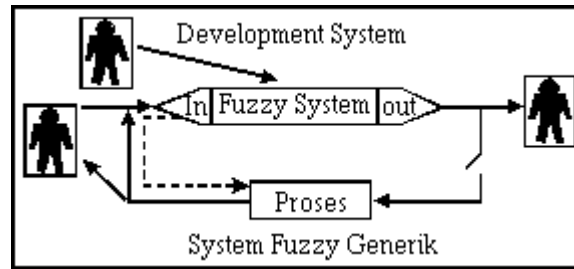
Menurut Hellendoorn, Hans dan Palm, Rainer (1994), sampai dengan saat ini, *software* pengembangan logika *fuzzy* sudah tidak terhitung banyaknya, mulai dari simulasi sistem yang sederhana hingga sistem yang sangat kompleks dan rumit. Masing-masing produk menawarkan berbagai kelebihan dan kemudahan pemakaian seperti : *User friendly editor, sistem on-line dan off-line debugging, compilers* untuk setiap bahasa pemrograman termasuk bahasa rakitan *micro*

*controler*, tampilan tiga dimensi (3D) dan berbagai macam proyek simulasi yang bisa dilakukan.

### **1.5.2 Kendali *Fuzzy*, Klasifikasi *Fuzzy*, dan *Diagnosis Fuzzy***

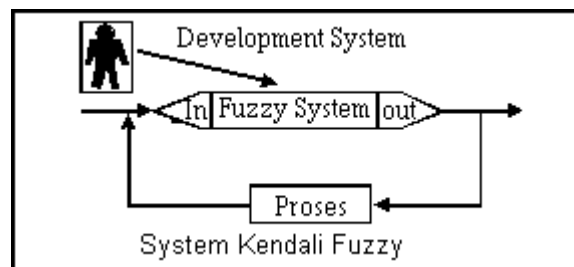
Aplikasi yang menggunakan logika *fuzzy*, selalu identik dengan pengendalian *fuzzy*. Walaupun sebenarnya aplikasi itu tergolong dalam klasifikasi *fuzzy* atau *diagnosis fuzzy*. Kejadian ini bukanlah masalah yang dominan dan pelik dalam sistem *fuzzy*, karena istilah "*fuzzy*" sebenarnya sudah kabur dan sering disamakan dengan istilah-istilah yang ada pada teori himpunan *fuzzy*, *topologi fuzzy*, atau dalam pengertian yang lebih sempit lagi sering disebut sebagai *approximate reasoning* dalam logika keputusan. Dengan cara pandang yang sama sistem kendali *fuzzy* sering sekali dinyatakan sebagai bagian teori himpunan *fuzzy* yang digunakan pada aplikasi-aplikasi dalam bentuk sistem lingkaran tertutup. Namun tujuan utama tulisan ini adalah membedakan antara sistem kendali *fuzzy* dengan sistem klasifikasi *fuzzy* dan sistem *diagnosis fuzzy*. Pada ruang lingkup yang lebih luas lagi, masih ada sistem lainnya yang cukup sukses digunakan seperti sistem pakar *fuzzy*, sistem analisa data *fuzzy*, sistem pengolahan citra *fuzzy*, dan berbagai ragam aplikasi sistem *fuzzy* yang sudah ada.

Pada dasarnya penggunaan istilah klasifikasi dan diagnosis bukanlah merupakan penamaan yang baku, karena keduanya mempunyai pengertian atau makna yang hampir sama dan batas-batas perbedaannya juga tidak begitu jelas. Namun yang teramat penting adalah kedua istilah tadi menunjukkan perbedaan antara kedua sistem aplikasi berbasis logika *fuzzy* (*fuzzy logic*). Sistem *fuzzy* secara umum dapat dilihat pada Gambar 1.6 sebagai berikut :



Gambar 1.6. Sistem *fuzzy* secara umum

Pada gambar 1.6 tersebut terdapat blok proses, sistem *fuzzy*, dan sistem pengembangan (*development system*). Pihak *developer* diletakkan paling atas pada gambar ini. Selain itu, terdapat dua operator, yaitu seorang yang bertanggung jawab atas masukan untuk sistem *fuzzy* dan keluaran dari proses, dan seorang lagi bertugas membawa masukan ke dalam proses dan menentukan keluaran dari sistem *fuzzy*. Operator ini sebenarnya tidak mesti seorang operator manusia, biasanya sistem *fuzzy* atau non-*fuzzy* yang berfungsi mengantarkan masukan atau keluaran sinyal proses. Dari gambar ini dapat diturunkan beberapa sistem sistem *fuzzy*, seperti pengendali *fuzzy*, klasifikator *fuzzy*, dan sistem pendiagnosaan *fuzzy*.



Gambar 1.7 Kendali *fuzzy*

Sebuah kendali *fuzzy* yang digambarkan pada Gambar 1.7, merupakan suatu sistem lingkaran tertutup, dimana tidak terdapat operator yang menjadi bagian dari sistem lingkaran kendali (*control loop*). Contoh dari sistem kendali ini adalah :

a. *Vacuum cleaner*

Sistem pada alat ini mengatur daya motor penghisap tergantung pada banyaknya debu pada lantai atau karpet.

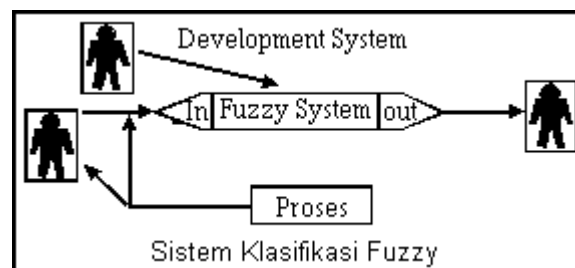
b. Optimisasi torsi dalam sistem anti slip yang digunakan kereta listrik.

c. Sistem kereta bawah tanah

Masukan sistem kendali berupa kecepatan kereta dan koefisien resistansi rel.

Pada sistem klasifikasi *fuzzy* tidak terdapat sistem *loop* tertutup atau *closed loop*.

Perhatikanlah gambar 1.8 berikut ini :



Gambar 1.8. Sistem klasifikasi *fuzzy*

Menurut Hellendoorn, Hans dan Palm, Rainer (1994), sistem ini hanya menerima masukan (*input*) dan keluaran (*output*) dari proses untuk selanjutnya memberikan informasi berupa kondisi (*state*) dari proses tadi. Informasi kondisi ini dapat digunakan untuk mengendalikan sistem atau memberikan tanggung jawab kendali kepada operator. Secara matematis, sistem klasifikasi lebih dekat pada teori

himpunan daripada teori fungsi. Pada sistem ini, sifat kesamaan (*Vagueness*) sering ditemui pada opini pakar dan jarang menggunakan model relasi *fuzzy*. Contoh dari sistem klasifikasi *fuzzy* adalah :

a. Mesin cuci *fuzzy*

Beberapa variabel atau parameter mesin cuci ditentukan berdasarkan jumlah dan jenis pakaian. Keluaran atau informasi dari sistem klasifikasi ini digunakan untuk menentukan jenis *spin-dry* serta lembut atau kasar gesekan pakaian yang optimal.

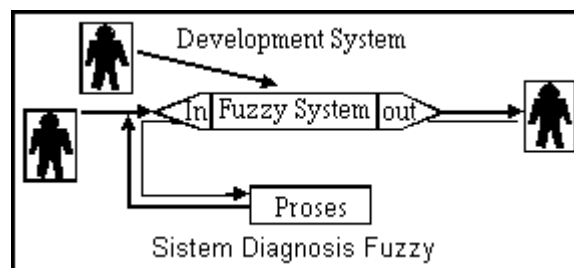
b. Sistem transmisi otomatis *fuzzy*

Sistem ini menggunakan beberapa sensor yang ditaruh pada sistem ABS.

c. Sistem *power steering* pada mobil

d. Sistem kendali motor, dan bagian penting lainnya

Selama kendaraan berjalan, sistem ini akan terus memantau dan menilai kondisi mobil tersebut, seperti beban kendaraan, kondisi mobil pada saat melewati jalan yang menanjak atau menurun dan kondisi-kondisi lainnya. Pada gambar 1.8, gambar operator manusia pada kiri dan kanan sistem klasifikasi *fuzzy*, biasanya merupakan suatu sistem khusus yang bertugas memberikan informasi yang diperlukan untuk kemudian diproses.



Gambar 1.9. Sistem *diagnosis fuzzy*

Perhatikanlah gambar 1.9 di atas. Pada sistem *diagnosis fuzzy* peranan manusia atau operator lebih dominan. Pengiriman data dilaksanakan oleh operator ke dalam sistem, ketika sistem memerlukan data tambahan. Selain itu operator dapat meminta atau menanyakan informasi dari sistem diagnosis berupa hasil konklusi diagnosis atau prosedur detail hasil diagnosis oleh sistem. Dari sifat sistem ini, sistem *diagnosis fuzzy* dapat digolongkan pada sistem pakar *fuzzy*. Sistem pakar *fuzzy* adalah sistem pakar yang menggunakan notasi *fuzzy* pada aturan-aturan dan proses *inferensi* (logika keputusan). Salah satu kelebihan sistem pakar *fuzzy* dibandingkan sistem pakar konvensional adalah jumlah aturan lebih sedikit, sehingga sistem lebih transparan untuk dianalisa. Kekurangannya adalah kehandalan sistem sangat tergantung pada baik-buruknya proses pengumpulan aturan seperti prosedur pertanyaan dan komponen-komponen kuisioner, serta sering terjadi kesulitan untuk menyimpulkan suatu pernyataan tertentu oleh operator.

Bidang aplikasi sistem diagnosis ini biasanya suatu proses yang besar dan kompleks, sehingga sangat sulit dianalisa menggunakan algoritma eksak dan dimodelkan dengan model matematika biasa. Pada permulaan persiapan sistem, jumlah aturan yang digunakan ini biasanya sangat banyak. Namun pada tahap akhir, jumlah aturan akan lebih sedikit dan mudah dibaca. Ini merupakan sifat sistem pakar *fuzzy*. Menurut Hellendoorn, Hans dan Palm, Rainer (1994), Prof. Zadeh menyatakan, bahwa sistem pakar *fuzzy* akan menggunakan aturan-aturan yang lebih sedikit dibandingkan sistem pakar konvensional sehingga mudah dibaca dan membantu menghindarkan inkonsistensi dan inkomplit sistem pengendali. Contoh dari sistem pakar *fuzzy* ini adalah proyek diagnosa kebocoran-H<sub>2</sub> pada sistem

pendingin *high-performance generator*. Salah satu contoh aturan sistem diagnostik ini adalah : "Jika konsumsi H2 tinggi dan daya yang tersedia rendah dan suhu gas rendah dan tekanan H2 generator tidak rendah atau menurun, maka tingkatkan konsumsi H2 (untuk menurunkan temperatur)"

Yang perlu diperhatikan pada sistem diagnostik ini adalah, tidak berlakunya proses *defuzzifikasi*, karena sistem ini hanya menghasilkan sifat keluaran berupa *aproksimasi linguistik* yang merupakan suatu pernyataan atau jawaban yang mudah dipahami oleh operator.

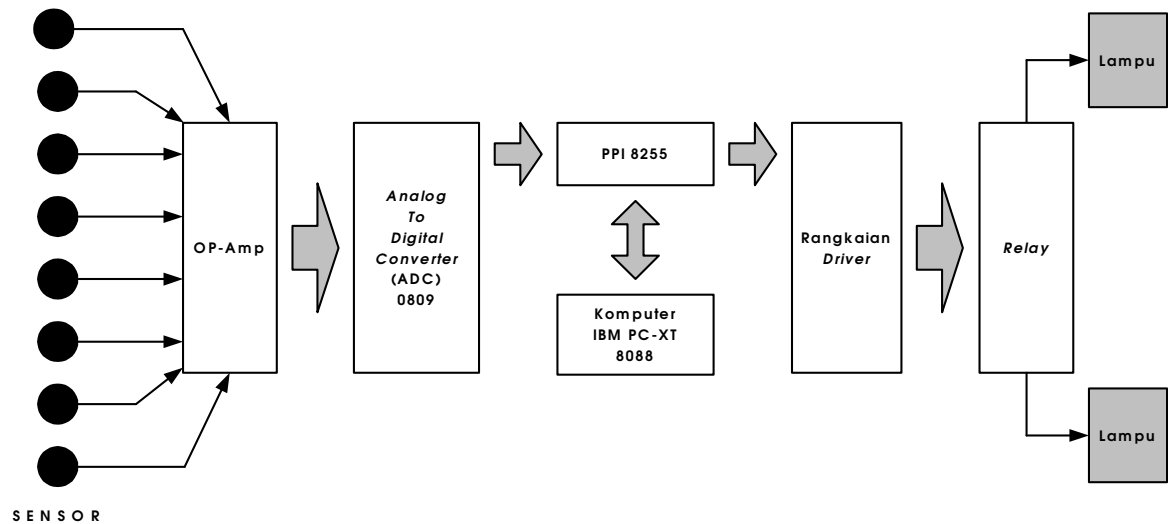
## **BAB II**

### **PEMBAHASAN**

#### **2.1 Diagram Blok Sistem**

Sistem pengatur lalu lintas yang dirancang ini, juga mempertimbangkan masukan interupsi (*interrupt*) sebagai prioritas utama, sehingga pengaturan lalu lintas yang sedang berjalan akan dihentikan sementara untuk melayani jalur yang menyela. Fasilitas ini digunakan untuk keadaan darurat atau mendesak, misalnya seperti pelayanan mobil pemadam kebakaran, mobil *ambulance*, konvoi kendaraan militer/pejabat, pawai/ karnaval dan pejalan kaki. Pendeteksian interupsi dilakukan secara terus menerus (*residen*). Jika lebih dari satu jalur memberi interupsi (*interrupt*), maka yang dilayani dulu adalah yang pertama menekan tombol interupsi (*interrupt*) itu.

Adapun diagram blok Sistem Pengaturan Lampu Lalu Lintas Berbasis fuzzy logic menggunakan Komputer IBM PC-XT 8088 adalah sebagai berikut :



**Gambar 2.1. Diagram blok**

Pada diagram tersebut terdapat blok untuk : delapan buah sensor, OP-Amp, ADC 0809, Komputer IBM PC-XT 8088, PPI 8255, Rangkaian Pengendali (*Driver*), Relay dan Lampu Lalulintas.

Untuk mendapatkan data digital dari masukan yang berbentuk analog, dibutuhkan suatu pengubah atau konverter dari sinyal analog ke sinyal digital (*Analog to Digital Converter*). Sebaliknya, setelah CPU selesai memproses data, diperlukan suatu konversi dari jawaban yang berupa sinyal digital ke dalam sinyal analog. Proses perubahan atau konversi ini membutuhkan sebuah alat yang disebut konverter digital ke analog (*A/D Converter*). Pada sistem, yang berfungsi sebagai masukan (*input*) adalah sensor dan keluarannya (*output*) adalah berupa nyala lampu pengatur lalu lintas (lampu).

## 2.2 Logika Fuzzy untuk Sistem Pengaturan Lalulintas

Berikut ini adalah beberapa istilah yang sering digunakan dalam pengendalian lampu lalu lintas, antara lain :

a. Sebaran banyaknya kendaraan di jalan raya

Adapun sebaran banyaknya kendaraan di jalan raya adalah :

1. Tidak Padat (TP).
2. Kurang Padat (KP).
3. Cukup Padat (CP).
4. Padat (P).
5. Sangat Padat (SP).

b. Lama lampu lalu lintas menyala

Adapun lama lampu lalu lintas menyala adalah :

1. Cepat (C).
2. Agak Cepat (AC).
3. Sedang (S).
4. Agak Lama (AL).
5. Lama (L).

Jelas istilah-istilah tersebut dapat menimbulkan kemenduaan (*ambiguity*) dalam pengertiannya. Logika *fuzzy* dapat mengubah kemenduaan tersebut ke dalam model matematis sehingga dapat diproses lebih lanjut untuk dapat diterapkan dalam sistem kendali. Menggunakan teori himpunan *fuzzy*, logika bahasa dapat diwakili oleh sebuah daerah yang mempunyai jangkauan tertentu yang menunjukkan derajat keanggotaannya. Untuk permasalahan disini, sebut saja derajat keanggotaan itu adalah  $u(x)$  untuk  $x$  adalah jumlah kendaraan. Derajat

keanggotaan tersebut mempunyai nilai yang bergradasi sehingga mengurangi lonjakan pada sistem.

Sistem pengendalian *fuzzy* yang dirancang mempunyai dua masukan (*input*) dan satu keluaran (*output*). Masukan adalah jumlah kendaraan pada suatu jalur yang sedang diatur dan jumlah kendaraan pada jalur lain, dan keluaran berupa lama nyala lampu hijau pada jalur yang diatur. Penggunaan dua masukan dimaksudkan supaya sistem tidak hanya memperhatikan sebaran kendaraan pada jalur yang sedang diatur saja, tetapi juga memperhitungkan kondisi jalur yang sedang menunggu. Pencuplikan dilakukan pada setiap putaran (melalui kedelapan sensor yang dipasang pada semua jalur). Satu putaran dianggap selesai apabila semua jalur telah mendapat pelayanan lampu.

Masukan berupa himpunan kepadatan kendaraan oleh logika *fuzzy* diubah menjadi fungsi keanggotaan masukan dan fungsi keanggotaan keluaran (lama lampu hijau). Bentuk fungsi keanggotaan dapat diatur sesuai dengan distribusi data kendaraan. Menerapkan logika *fuzzy* dalam sistem pengendalian, membutuhkan tiga langkah, yaitu :

- a. Fusifikasi (*Fuzzyfication*).
- b. Evaluasi kaidah.
- c. Defusifikasi (*Defuzzyfication*).

Fusifikasi adalah proses mengubah masukan eksak berupa jumlah kendaraan menjadi masukan *fuzzy* berupa derajat keanggotaan  $u(x)$ . Setelah fusifikasi adalah evaluasi kaidah. Kaidah-kaidah yang akan digunakan untuk mengatur lalu lintas ditulis secara subyektif dalam *Fuzzy Associate Memory* (FAM), yang terdiri dari hubungan antara kedua masukan yang menghasilkan keluaran tertentu. Kaidah-

kaidah ini terlebih dahulu dikonsultasikan kepada mereka yang berpengalaman dalam bidang yang akan dikendalikan, yaitu misalnya kepada pihak Kepolisian Republik Indonesia (POLRI) khususnya bagian lalu lintas (Polisi Lalulintas) dan pihak Dinas Lalu Lintas dan Anggukan Jalan Raya (DLLAJR). Disini dipakai kaidah hubungan sebab akibat dengan dua masukan yang digabung menggunakan operator AND, yaitu : Jika masukan-1 AND masukan-2, maka keluaran, dan dibuat dalam tabel dalam Tabel *Fuzzy Associate Memory* (FAM).

Sebagai contoh, jika TP(0,25) dan KP(0,75), maka AC(0,25). Di sini keluaran *fuzzy* adalah Agak Cepat yaitu AC(0,25).

Tabel 2.1 *Fuzzy Associate Memory* (FAM) untuk kepadatan Lalulintas

Masukan-1	TP	KP	CP	P	SP
Masukan-2					
TP	C	AC	S	AL	L
KP	C	AC	S	AL	L
CP	C	AC	S	AL	AL
P	C	AC	S	S	AL
SP	C	AC	AC	S	S

Keterangan :  
 Masukan-1 adalah jumlah kendaraan pada jalur yang diatur  
 Masukan-2 adalah jumlah kendaraan pada jalur lain

Sumber : Majalah Elektro Indonesia. Nomor 27, Tahun VI, Agustus 1999.

Setelah diperoleh keluaran *fuzzy*, proses diteruskan pada defusifikasi. Proses ini bertujuan untuk mengubah keluaran *fuzzy* menjadi keluaran eksak (lama nyala lampu hijau). Karena keluaran *fuzzy* biasanya tidak satu untuk selang waktu tertentu, maka untuk dihasilkan keluaran eksaknya dipilih keluaran dengan harga yang

terbesar. Apabila terdapat dua buah derajat keanggotaan berbeda pada akibat yang sama, diambil harga yang terbesar.

### 2.3 Program Kontrol

Perancangan perangkat lunak (*software*) yang dibuat sebagai program yang berfungsi untuk mengontrol Sistem Pengaturan Lampu Lalu Lintas Berbasis *fuzzy logic* menggunakan Komputer IBM PC-XT 8088 ini dibagi menjadi beberapa bagian besar, antara lain :

- a. Meliputi algoritma pengambilan dan masukan.
- b. Pengiriman data keluaran.
- c. Pengolahan data secara *fuzzy*.
- d. Proses kendalinya.

Perangkat lunak ini dapat dibuat dengan menggunakan bahasa pemrograman Pascal. Algoritma program utama mengikuti proses sebagai berikut : mula-mula chip PPI 8255 diinisialisasi dengan mengirimkan *control word* ke *register* kendali PPI 8255. Dengan mengirimkan nilai 90 H ke *register* kendali PPI 8255, maka *port* A akan berfungsi sebagai masukan (*input*) dan *port* B serta *port* C akan berfungsi sebagai keluaran (*output*).

Selanjutnya akan dikirimkan pulsa *reset* ke semua masukan rangkaian *Analog to Digital Converter* (ADC), pada saat awal seluruh jalur akan diberi lampu merah. Setelah proses ini, program melakukan proses yang berulang-ulang, yaitu proses pengambilan data pada tiap sensor, pengolahan data dan proses pengaturan *fuzzy* menggunakan prinsip-prinsip yang telah dibahas di atas dan menjalankan pengaturan sesuai dengan tabel kendali yang telah dibuat.

## BAB III

### KESIMPULAN

#### 3.1 Kesimpulan

Komponen dari Sistem Pengaturan Lampu Lalu Lintas Berbasis fuzzy logic menggunakan Komputer IBM PC-XT 8088 adalah : Sensor, Op-Amp, *Analog to Digital Converter* (ADC) 0809, *Programable Peripheral Interface* (PPI) 8255, Rangkaian Pengendali (*Driver*), *Relay* dan lampu lalu lintas.

Pada sistem, yang berfungsi sebagai masukan (*input*) adalah sensor dan keluarannya (*output*) adalah berupa nyala lampu pengatur lalu lintas (lampu).

Perangkat lunak (*software*) yang dapat dipergunakan untuk mengontrol kerja dari sistem ini dapat dibuat dengan menggunakan bahasa pemrograman Pascal. Algoritma program utama mengikuti proses sebagai berikut : mula-mula chip PPI 8255 diinisialisasi dengan mengirimkan *control word* ke *register* kendali PPI 8255. Dengan mengirimkan nilai 90 H ke *register* kendali PPI 8255, maka *port A* akan berfungsi sebagai masukan (*input*) dan *port B* serta *port C* akan berfungsi sebagai keluaran (*output*). Selanjutnya akan dikirimkan pulsa *reset* ke semua masukan rangkaian *Analog to Digital Converter* (ADC), pada saat awal seluruh jalur akan diberi lampu merah. Setelah proses ini, program melakukan proses yang berulang-ulang, yaitu proses pengambilan data pada tiap sensor, pengolahan data dan proses pengaturan *fuzzy* menggunakan prinsip-prinsip yang telah dibahas di atas dan menjalankan pengaturan sesuai dengan tabel kendali yang telah dibuat.

Dari hasil pengamatan logika *fuzzy* ini terbukti dapat digunakan untuk memenuhi tujuan pengaturan lalu lintas secara optimal. Sistem yang dihasilkan relatif sederhana dan mempunyai fleksibilitas tinggi. Sistem ini dapat diterapkan pada kondisi jalan yang berbeda, yaitu melalui penyesuaian ranah (domain) himpunan fungsi keanggotaan masukan dan keluaran dan kaidah-kaidah kendali pada *Fuzzy Associative Memory* (FAM).

### **3.2 Saran-saran**

Sistem Pengaturan Lampu Lalu Lintas Berbasis fuzzy logic menggunakan Komputer IBM PC-XT 8088 dapat dikembangkan (diperluas) lagi, misalnya dengan membuat sistem komputerisasi yang terpusat (*centralisasi*) dengan tugas mengkoordinasi beberapa persimpangan jalan (yang tidak hanya persimpangan jalan tersebut berjumlah empat buah), terutama yang berdekatan, dengan tujuan supaya sistem-sistem saling membantu dan memperlancar sebaran kendaraan pada suatu daerah. Sistem ini dikembangkan lagi ke arah Sistem Pengaturan Lampu Lalu Lintas Berbasis fuzzy logic menggunakan Komputer IBM PC-XT 8088 yang adaptif, yaitu apabila kondisi kepadatan lalu lintas di jalan raya berubah, maka sistem akan melakukan perubahan bentuk fungsi keanggotaan masukan dan keluaran, serta tabel *Fuzzy Associative Memory* (FAM) secara otomatis. Untuk alternatif lain pengganti sebuah *Personal Computer* (PC) yang berfungsi sebagai pengendali utama peralatan ini juga dapat dipergunakan sistem minimum yang salah satunya bisa berupa suatu peralatan yang mempergunakan aplikasi dari *Microcontroler* 8031, sehingga Sistem Pengaturan Lampu Lalulintas Berbasis *fuzzy logic* ini tidak lagi

tergantung pada penyediaan perangkat komputer sebagai komponen utama dalam sistem pengendali lampu lalu lintas tersebut.

## DAFTAR PUSTAKA

- B. Kosko. *Neural Network and Fuzzy System*. Chapter 8. Prentice Hall. 1992.
- Bartos, Frank J. *Fuzzy Logic is Clearly Here to Stay*. McGraw-Hill Pub. Control Engineering. 1992.
- Duda, Walter H, Dipl-Ing. *Cement-Data-Book*. Volume 2. Bauverlag GmbH. Weisbaden Und Berlin. 1991.
- E. Cox. *Fuzzy Fundamentals*. Spectrum IEEE. 1992.
- Erdman, Denise. *Fuzzy Logic more than a play on words, Chemical Engineering*. McGraw-Hill Pub. 1993.
- Goldsbrough, Paul F. *Microcomputer Interfacing with the 8255 PPI chip*. Howard W. Sam & Co. 1979.
- Hall, Douglas V. *Microprocessor and Interfacing Programming and Hardware*. McGraw-Hill Book Company. 1986.
- Hellendoorn, dkk. *Fuzzy system technologies at Seimens R & D. Fuzzy Sets and System* 63. North-Holland. 1994.
- J.P.M, Steeman. *Data Sheet Book 2*. Elex Media Komputindo. Jakarta. 1990.
- J.W, Lea. *Sistem Pengaturan Lampu Lalulintas dengan Menggunakan Teori Himpunan Fuzzy*. 1994.
- Komisi Teknik Komputer. *Transparansi Mata Kuliah Rangkaian Digital*. AMIK BSI. Jakarta. 2000.
- S. Marsh. et al. *Fuzzy Logic Education Program. Center of Emerging Computer Technologies*. Motorola Inc. 1992.
- Setianto, S.Si. Basuki Rahmat, S.Si. *Pengaturan Lampu Lalulintas Berbasis Fuzzy Logic*. Majalah Elektro Indonesia. Nomor 27, Tahun VI, Agustus 1999. Jakarta.
- Technical Reference for IBM PC-XT.
- Wahyudi, Mochamad. S.Kom. *Perancangan Sistem Pengembang Film Menggunakan IBM PC-XT 8088*. Paradigma, Jurnal Ilmiah Komputer & Informatika Bina Sarana Informatika, Volume IV No. 1, Januari 2001. Lembaga Penelitian dan Pengabdian Pada Masyarakat Akademi Manajemen Informatika & Komputer Bina Sarana Informatika. Jakarta. 2001.

## DAFTAR RIWAYAT HIDUP

### I. Identitas Diri

Nama : Mochamad Wahyudi, S.Kom  
 Tempat lahir, umur : Jakarta, 27 tahun  
 Jenis Kelamin : Laki-laki  
 Status : Menikah  
 Agama : Islam  
 Alamat : Jl. Melati Indah I/L No.19 RT. 003/014  
 Kel. Kapuk – Kec. Cengkareng  
 Jakarta Barat 11720  
 Telepon : 021-545 0350, Hp. 0811-830311

### II. Pendidikan

1. Sekolah Tinggi Manajemen Informatika dan Komputer Budi Luhur Jakarta jurusan Teknik Komputer (1993-1997).
2. Mahasiswa Program Pasca Sarjana pada Sekolah Tinggi Ilmu Ekonomi Budi Luhur Jakarta bidang konsentrasi Sistem Informasi Manajemen.

### III. Pengalaman Kerja

1. PT. Telekomunikasi Indonesia (TELKOM) Kandatel Jakarta Barat (1996 s/d 1998)
2. PT. Shada, Inc (Citibank Card Center) Jakarta (1997 s/d 1999)
3. Akademi Manajemen Informatika dan Komputer Bina Sarana Informatika (AMIK BSI) Jakarta (1998 s/d sekarang)

### IV. Tulisan Ilmiah yang dipublikasikan dalam jurnal

Perancangan Sistem Pengembang Film Menggunakan IBM PC-XT 8088. Pada Paradigma, Jurnal Ilmiah Komputer & Informatika Bina Sarana Informatika, Volume IV No. 1, Januari 2001. Diterbitkan oleh Lembaga Penelitian dan Pengabdian Pada Masyarakat Akademi Manajemen Informatika & Komputer Bina Sarana Informatika. Jakarta.

### V. Seminar

Sebagai pembicara seminar Pemanfaatan Teknologi Jaringan Lokal (*Local Area Network*) dan Internet untuk Keperluan Pendidikan/Warung Internet (Warnet). Disampaikan pada seminar yang diselenggarakan oleh Himpunan Mahasiswa Jurusan Teknik Komputer AMIK BSI di Purwakarta pada Minggu, 29 Juli 2001.