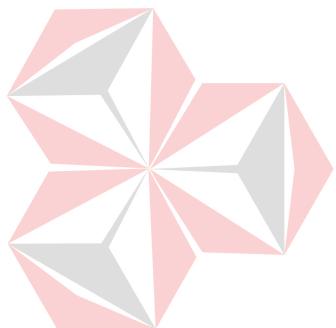


**RANCANG BANGUN PERANGKAT PENGENDALI PINTU AIR
SUNGAI JAGIR DENGAN METODE FUZZY MENGGUNAKAN
JARINGAN SENSOR NIRKABEL**

TUGAS AKHIR



Nama : Adi Priyanto
NIM : 06.41020.0005
Nama : Ifnu Fajar Pratomo
NIM : 06.41020.0025
Program : S1 (Strata Satu)
Jurusan : Sistem Komputer

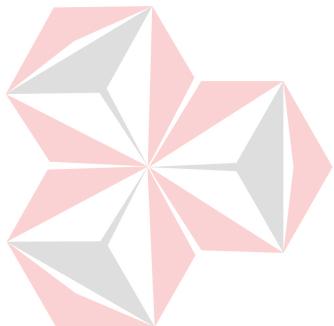
**SEKOLAH TINGGI
MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER
SURABAYA
2011**

**RANCANG BANGUN PERANGKAT PENGENDALI PINTU AIR
SUNGAI JAGIR DENGAN METODE FUZZY MENGGUNAKAN
JARINGAN SENSOR NIRKABEL**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk menyelesaikan

Program Sarjana Komputer



UNIVERSITAS
Dinamika
Oleh :

Nama : Adi Priyanto
NIM : 06.41020.0005
Nama : Ifnu Fajar Pratomo
NIM : 06.41020.0025
Program : S1 (Strata Satu)
Jurusan : Sistem Komputer

**SEKOLAH TINGGI
MANAJEMEN INFORMATIKA & TEKNIK KOMPUTER
SURABAYA
2011**

Tugas Akhir
RANCANG BANGUN PERANGKAT PENGENDALI PINTU AIR
SUNGAI JAGIR DENGAN METODE FUZZY MENGGUNAKAN
JARINGAN SENSOR NIRKABEL

dipersiapkan dan disusun oleh

Ifnu Fajar Pratomo
Adi priyanto

Telah diperiksa, diuji dan disetujui oleh Dewan Penguji
pada tanggal : 18 Maret 2011

Susunan Dewan Penguji



I. Helmy Widhyantara, S.Kom., M.Eng. _____

II. Susijanto Tri Resmana, S.Kom., MT _____

Tugas Akhir ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana

Pantjawati Sudarmaningtyas, S.Kom, OCA.
Wakil Ketua Bidang Akademik

PERNYATAAN

Dengan ini saya menyatakan dengan benar, bahwa Tugas Akhir ini adalah asli karya saya, bukan plagiat baik sebagian maupun apalagi keseluruhan. Karya atau pendapat orang lain yang ada dalam Tugas Akhir ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya. Apabila dikemudian hari ditemukan adanya tindakan plagiat pada karya Tugas Akhir ini, maka saya bersedia dilakukan pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.



Ifnu Fajar Pratomo

PERNYATAAN

Dengan ini saya menyatakan dengan benar, bahwa Tugas Akhir ini adalah asli karya saya, bukan plagiat baik sebagian maupun apalagi keseluruhan. Karya atau pendapat orang lain yang ada dalam Tugas Akhir ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya. Apabila dikemudian hari ditemukan adanya tindakan plagiat pada karya Tugas Akhir ini, maka saya bersedia dilakukan pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.

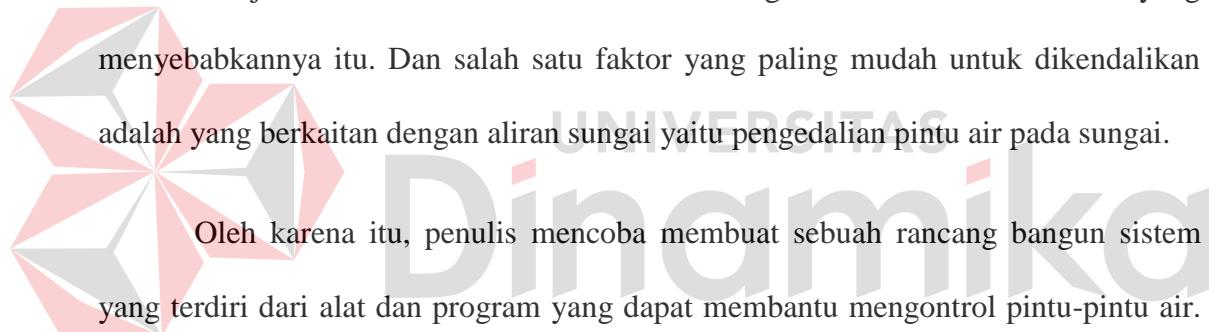


Adi Priyanto



ABSTRAKSI

Banjir adalah salah satu bencana musiman yang terjadi hampir di semua daerah khususnya daerah-daerah yang tata sungainya masih kurang baik. Ada banyak faktor yang dapat menyebabkan terjadinya banjir. Curah hujan yang tinggi, adanya kiriman aliran air yang melebihi kapasitas sungai dari daerah hulu, pengaruh pasang surut air laut dan lain-lain yang memicu terjadinya banjir. Untuk mencegah banjir, tentu menjadi suatu keharusan untuk mengendalikan faktor-faktor yang menyebabkannya itu. Dan salah satu faktor yang paling mudah untuk dikendalikan adalah yang berkaitan dengan aliran sungai yaitu pengedalian pintu air pada sungai.



Oleh karena itu, penulis mencoba membuat sebuah rancang bangun sistem yang terdiri dari alat dan program yang dapat membantu mengontrol pintu-pintu air.

Rancang bangun sistem ini diharapkan nantinya akan dapat mendeteksi kemungkinan terjadinya banjir sekaligus memberikan saran kepada operator pintu air mengenai *setting* tinggi pintu air yang optimal untuk meminimalisir banjir.

Rancang bangun sistem ini terdiri dari dua komponen utama. Komponen yang pertama adalah berupa perangkat *hardware* yang memantau kondisi sungai. Perangkat *hardware* ini sendiri merupakan perangkat elektronik yang terdiri dari mikrokontroler, sensor ultrasonik, dan perangkat komunikasi *wireless*. Sedangkan komponen yang kedua adalah berupa *desktop software* yang akan mengambil data

dari perangkat *sensing*, melakukan komputasi dan akan meng-*output*-kan *setting* ketinggian pintu air yang optimal untuk meminimalisir banjir.

Dari penerapan sistem yang dibuat oleh penulis dapat di ketahui debit air perpintu dan debit keseluruhan, dan dari debit-debit yang di ketahui dapat di olah lagi sebagai keluaran berupa saran pada petugas untuk menentukan tinggi-rendahnya bukaan pintu air, sehingga kondisi hulu dan hilir dapat di kendalikan sesuai dengan set poin yang di tentukan.



KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang maha Esa, karena berkat limpahan Rahmat dan Karunia-Nya penulis dapat menyelesaikan Tugas Akhir yang merupakan persyaratan dalam menyelesaikan Program Studi Strata Satu Jurusan Sistem Komputer di Sekolah Tinggi Manajemen Informatika dan Teknik Komputer Surabaya.

Tugas Akhir ini berjudul berjudul “ Rancang Bangun Perangkat Pengendalian Pintu Air Jagir Dengan Metode Fuzzy Menggunakan Jaringan Sensor Nirkabel ”.

Pada kesempatan ini, penulis mengucapkan terima kasih kepada:

1. Bapak dan Ibu, yang telah memberikan segalanya demi cita-cita penulis.
2. Kakak – Kakak dan Adik Tercinta , yang telah memberi dukungan moril dan doanya .
3. Bapak/Ibu Guru yang sudah mendidik selama ini.
4. Bpk Harianto S.Kom., M.Eng selaku Dosen Pembimbing I, dan Bpk. Ihyauddin, S.Kom selaku Dosen Pembimbing II, yang telah membimbing penulis dalam menyelesaikan Tugas Akhir ini.
5. Bpk. Yuwono Marthadinata, S.T, M.eng. selaku Kaprodi S1 Sistem Komputer STIKOM Surabaya telah memberikan dorongan moral kepada penulis.
7. Segenap Dosen Pengajar program studi S-1 Sistem Komputer.

8. Perpustakaan STIKOM, yang telah membantu menyediaan buku-buku bagi keperluan penulis.
9. Sahabat – sahabatku tercinta, yaitu: Riza rahardian, Achmad sugiarto, luky Harianto, Danang firmansyah, Tim Robot Stikom yang telah membantu penulis menyelesaikan Tugas Akhir ini.

Penulis berharap Tugas Akhir ini dapat berguna bagi penulis khususnya dan perkembangan teknologi Indonesia pada umumnya



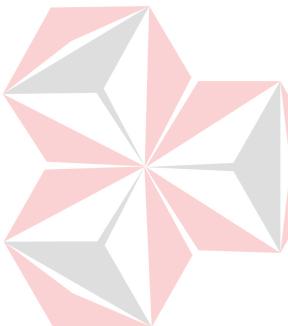
DAFTAR ISI

	Halaman
HALAMAN MOTO	iii
HALAMAN PERSEMPAHAN	iv
HALAMAN PENGESAHAN	v
ABSTRAKSI	vi
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL.....	xv
DAFTAR GAMBAR	xvi
DAFTAR LAMPIRAN	xx
DAFTAR ISTILAH DAN SINGKATAN	xxi

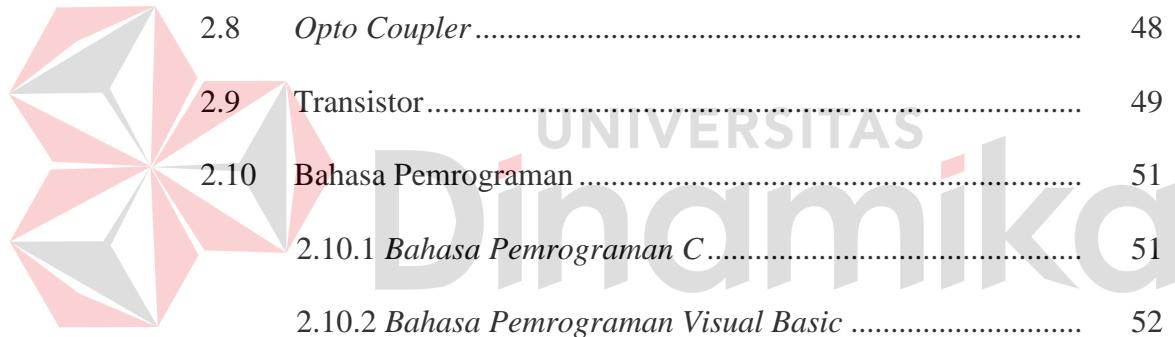
BAB I PENDAHULUAN

1.1	Latar Belakang.....	1
1.2	Perumusan Masalah.....	2
1.3	Pembatasan Masalah	2
1.4	Tujuan.....	3
1.5	Kontribusi	3
1.6	Sistematika Penulisan.....	4

BAB II LANDASAN TEORI



2.1	Pintu Air	5
2.1.1	<i>Gambaran Umum DAM Jagir.....</i>	5
2.1.2	<i>Operasi Pintu Air</i>	6
2.2	Pedoman Siaga Banjir	9
2.2.1	<i>Maksud dan Tujuan.....</i>	9
2.2.2	<i>Fungsi Pintu Air wonokromo</i>	9
2.2.3	<i>Tingkat Siaga</i>	9
2.3	Metode Fuzzy	10
2.3.1	<i>Pengantar Fuzzy</i>	10
2.3.2	<i>Dasar - Dasar Pemahaman Logika Fuzzy.....</i>	11
2.3.3	<i>Derajat Keanggotaan.....</i>	14
2.3.4	<i>Label.....</i>	14
2.3.5	<i>Fungsi Keanggotaan</i>	14
2.3.6	<i>Lingkup dan Domain.....</i>	14
2.3.7	<i>Daerah Batasan Crips</i>	14
2.3.8	<i>Fuzzyifikasi dan Fungsi Keanggotaan</i>	15
2.4	Sensor Ultrasonik	21
2.5	Komunikasi I2C.....	21
2.5.1	<i>Karakteristik sinyal I2C</i>	22
2.5.2	<i>Metode Penomoran I2C</i>	28
2.5.3	<i>IC Serial EEPROM Buatan ATMEL</i>	30
2.5.4	<i>Proses Menulis Pada serial EEPROM</i>	31



2.5.5 Proses Pembacaan Pada serial EEPROM	33
2.6 IC Modem.....	34
2.7 Modulasi FSK.....	35
2.8 Komunikasi Serial	36
2.8.1 Peralatan Komunikasi Serial	37
2.8.2 Konverter Logika RS232	39
2.9 Mikrokontroler	40
2.6.1 Port sebagai input atau output digital	44
2.6.2 Peta Memori.....	46
2.7 Walkie talkie	47
2.8 Opto Coupler.....	48
2.9 Transistor	49
2.10 Bahasa Pemrograman	51
2.10.1 Bahasa Pemrograman C	51
2.10.2 Bahasa Pemrograman Visual Basic	52

BAB III METODE PENELITIAN

3.1 Perancangan Perangkat Keras	58
3.1.1 Minimum Sistem	60
3.1.2 Modulasi FSK	64
3.1.3 Demodulasi FSK	67
3.1.4 Interface RS232	66
3.1.5 Sensor ultrasonik	67
3.2 Perancangan Perangkat Lunak	69
3.2.1 Desain fuzzy yang akan di terapkan dalam sisitem	70

3.2.2 <i>Flowchart Sistem Keseluruhan</i>	75
BAB IV PENGUJIAN SISTEM	
4.1 Pengujian <i>Minimum system Atmega16/8</i>	80
4.1.1 <i>Prosedur Pengujian</i>	80
4.1.2 <i>Hasil Pengujian</i>	81
4.1.3 <i>Analisa</i>	83
4.2 Pengujian IC MAX RS232 dan IC modem TCM3015	83
4.1.1 <i>Prosedur Pengujian</i>	83
4.1.2 <i>Hasil Pengujian</i>	85
4.1.3 <i>Analisa</i>	83
4.3 Pengujian Pengiriman Data dan Proses Swicting Chanel	86
4.1.1 <i>Prosedur Pengujian</i>	86
4.1.2 <i>Hasil Pengujian</i>	87
4.1.3 <i>Analisa</i>	87
4.4 Pengujian Sensor Ultrasound	87
4.4.1 <i>Prosedur Pengujian</i>	87
4.4.2 <i>Prosedur pengujian</i>	88
4.4.3 Hasil pengujian.....	89
4.5 Pengujian Database...	90
4.5.1 <i>Prosedur Pengujian</i>	90
4.5.2 <i>Hasil pengujian</i>	90
4.5.3 <i>Analisa</i>	92
4.6 Pengujian sistem keseluruhan.....	92
4.6.1 <i>Prosedur Pengujian</i>	92
4.6.2 <i>Hasil pengujian</i>	94

4.6.3 <i>Analisa</i>	95
BAB V PENUTUP	
5.1 Kesimpulan.....	96
5.2 Saran	97
DAFTAR PUSTAKA	98
LAMPIRAN	99
BIODATA	124



DAFTAR TABEL

	Halaman
Tabel 2.1	Susunan A0...A2
Tabel 2.2	Jenis sinyal RS232.....
Tabel 2.3	Siaga Banjir.....
Tabel 4.1	Hasil pembacaan sensor ultrasonik



DAFTAR GAMBAR

	Halaman
Gambar 2.1 Sensor Ultrasonik tipe SRF02	6
Gambar 2.2 Level Sinyal I2C.....	8
Gambar 2.3 <i>Pull-up Resistor Setting Suggestion</i>	8
Gambar 2.4 Sinyal dasar I2C	9
Gambar 2.5 Kondisi <i>Start</i>	9
Gambar 2.6 Kondisi <i>Stop</i>	10
Gambar 2.7 Kondisi <i>Restart</i>	10
Gambar 2.8 Kondisi <i>Start</i> dan <i>Stop</i>	11
Gambar 2.9 Transfer Data	12
Gambar 2.10 Kondisi dimana data dianggap valid	12
Gambar 2.11 Kondisi <i>ACK</i>	13
Gambar 2.12 Kondisi <i>NACK</i>	13
Gambar 2.13 Metode Penomoran I2C.....	15
Gambar 2.14 Serial EEPROM Buatan ATMEL	16
Gambar 2.15 Proses Menulis Pada Serial EEPROM RS232	17
Gambar 2.16 Proses pengisian data ke AT24CXX dengan metode pengalamatan 8 bit.....	17
Gambar 2.17 Proses pengisian data ke AT24CXX dengan metode pengalamatan 16 bit.....	18
Gambar 2.18 Proses Membaca Pada Serial EEPROM.....	19
Gambar 2.19 Pin out IC TCM 3105 tipe J atau N	20
Gambar 2.20 Modulasi data digital dengan metode FSK	21

Gambar 2.21	Arah komunikasi serial	22
Gambar 2.22	Port DB9 jantan	24
Gambar 2.23	Port DB9 betina	24
Gambar 2.24	Rangkaian MAX 232.....	26
Gambar 2.25	ATmega16.....	27
Gambar 2.26	ATmega8.....	28
Gambar 2.27	Sistem ALU ATmega	31
Gambar 2.28	Konfigurasi Memori Data AVR ATMega16.....	32
Gambar 2.29	<i>Walkie talkie</i>	33
Gambar 2.30	Opto Coupler	33
Gambar 2.31	<i>Contoh penggunaan opto coupler</i>	34
Gambar 2.32	Transistor through-hole	35
Gambar 2.33	<i>Skematik Transistor</i>	36
Gambar 2.34	Konvensional set dan Set fuzzy	43
Gambar 2.35	Konvensional set dan Set fuzzy	43
Gambar 2.36	Contoh pengambilan keputusan crisps	44
Gambar 2.37	Konvensional set dan Set fuzzy	44
Gambar 2.38	Fungsi keanggotaan	46
Gambar 2.39	Proses fuzzyifikasi	47
Gambar 2.40	Pendekatan konvensional dan Pendakata fuzzy	48
Gambar 2.41	Contoh pendekatan fuzzy	48
Gambar 2.42	Contoh bentuk defuzzyfikas	49
Gambar 2.43	Fungsi Keanggotaan	49
Gambar 2.44	Proses fuzzy	50

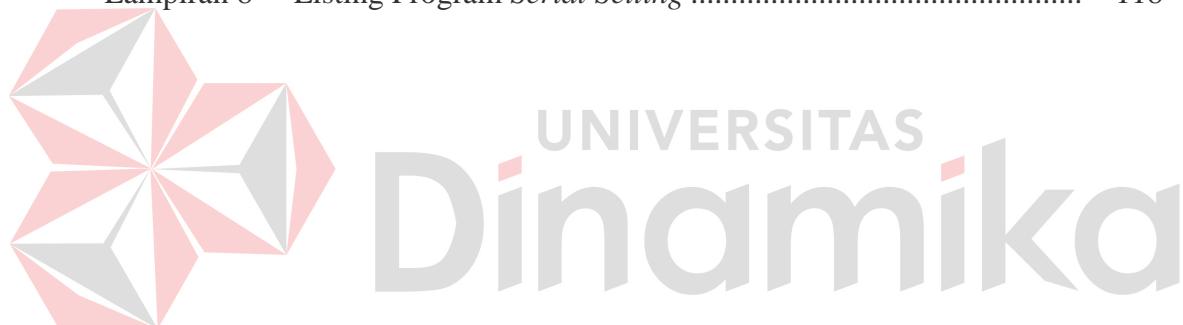
Gambar 2.45	Perilaku logika fuzzy	51
Gambar 3.1	Blok Diagram Sistem Keseluruhan	58
Gambar 3.2	Blok diagram minimum sistem <i>transmitter</i>	60
Gambar 3.3	Blok diagram minimum sistem <i>reciever</i>	61
Gambar 3.4	Minimum sistem <i>Reciever</i>	63
Gambar 3.5	Minimum sistem <i>Trasmitter</i>	64
Gambar 3.6	Rangkian Modulasi.....	66
Gambar 3.7	Rangkian Demodulasi	67
Gambar 3.8	Rangkian RS232.....	68
Gambar 3.9	Sensor Ultrasonic mode I2C.....	69
Gambar 3.10	Desain Sistem fuzzy	71
Gambar 3.11	<i>Membership Function Hulu</i>	72
Gambar 3.12	<i>Membership Function Debit</i>	72
Gambar 3.13	<i>Membership Function Pintu Kendali</i>	73
Gambar 3.14	<i>Rule Editor</i>	74
Gambar 3.15	<i>Rule Veiwer</i>	75
Gambar 3.16	Flowchart <i>transmitter</i>	76
Gambar 3.17	Flowchart <i>receiver</i>	77
Gambar 3.18	Flowchart pada program komputer	79
Gambar 4.1	Tampilan kode program	82
Gambar 4.2	Tampilan hasil program berhasil di <i>compile</i> yaitu program siap di download ke mikrokontroler.....	83
Gambar 4.3	Tampilan proses penulisan program ke mikrokontroler.....	84
Gambar 4.4	Tampilan hasil data sensor yang telah terkirim	86

Gambar 4.5	Tampilan hasil data sensor yang telah terkirim.....	88
Gambar 4.6	Tampilan hasil data sensor	89
Gambar 4.7	Tampilan setting serial port	91
Gambar 4.8	Tampilan data hulu <i>realtime</i>	92
Gambar 4.9	Tampilan data hilir <i>realtime</i>	92
Gambar 4.10	Tampilan data hulu dan hilir <i>average 1 minute</i>	93
Gambar 4.11	Tampilan <i>Software Preventing Advisor</i>	94
Gambar 4.12	Tampilan <i>Software Preventing Advisor</i> keseluruhan	95



DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Listing Program <i>Transmitter</i>	99
Lampiran 2 Listing Program <i>Swicting</i>	102
Lampiran 3 Listing Program <i>Index</i>	104
Lampiran 4 Listing Program <i>Serial Advisor</i>	110
Lampiran 5 Listing Program <i>Hilir</i>	113
Lampiran 6 Listing Program <i>Hulu</i>	114
Lampiran 7 Listing Program <i>Hulu dan Hilir Avarage 1 minute</i>	115
Lampiran 8 Listing Program <i>Serial Setting</i>	118



BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Saat ini, teknologi sudah berkembang begitu pesatnya. Hampir semua sektor kehidupan sudah menggunakan teknologi untuk membantu mengambil keputusan maupun untuk melakukan pengontrolan. Dengan menggunakan teknologi, kini komputasi untuk pengambilan keputusan dan pengontrolan dapat dilakukan dengan lebih efisien.

Sampai sekarang, banjir masih menjadi bencana yang paling sering dihadapi oleh masyarakat. Padahal usaha untuk menanganinya telah diupayakan sejak lama. Perangkat-perangkat seperti pintu-pintu air, telah banyak dibuat dan dibangun untuk mengantisipasi bencana ini. Pintu-pintu air ini bahkan telah banyak tersedia di sepanjang aliran sungai penyebab banjir. Namun, pada kenyataannya, selama ini penggunaan pintu-pintu air masih belum dilakukan optimal. Pengontrolan dan pengaturan *setting*-nya yang masih secara *trial and error* dan tanpa melalui perhitungan yang sistematis, masih membuat banjir kerap saja terjadi.

Oleh karena itu, penulis mencoba untuk menerapkan suatu teknologi komputasi untuk membuat sebuah sistem yang akan dapat membantu mengontrol pintu-pintu air. Sistem ini diharapkan nantinya akan dapat mendeteksi kemungkinan terjadinya banjir sekaligus memberikan *advise* mengenai *setting* tinggi pintu air yang optimal untuk meminimalisir banjir.

1.2 Perumusan Masalah

Berdasarkan latar belakang di atas, terdapat beberapa rumusan masalah sebagai berikut :

- a. Bagaimana merancang dan membuat perangkat pemantauan kondisi sungai.
- b. Bagaimana cara membuat sistem komunikasi data jarak jauh, dari sensor ultrasonik ke perangkat pengolah data dengan memanfaatkan gelombang radio.
- c. Bagaimana merancang dan membuat aplikasi yang dapat memberi *advise* mengenai *setting* ketinggian pintu air berdasarkan kondisi aktual sungai pada saat itu dengan menggunakan logika *fuzzy*.

1.3 Pembatasan Masalah

Dalam perancangan dan pembuatan rancangan bangun sistem ini, terdapat beberapa pembatasan masalah, antara lain:

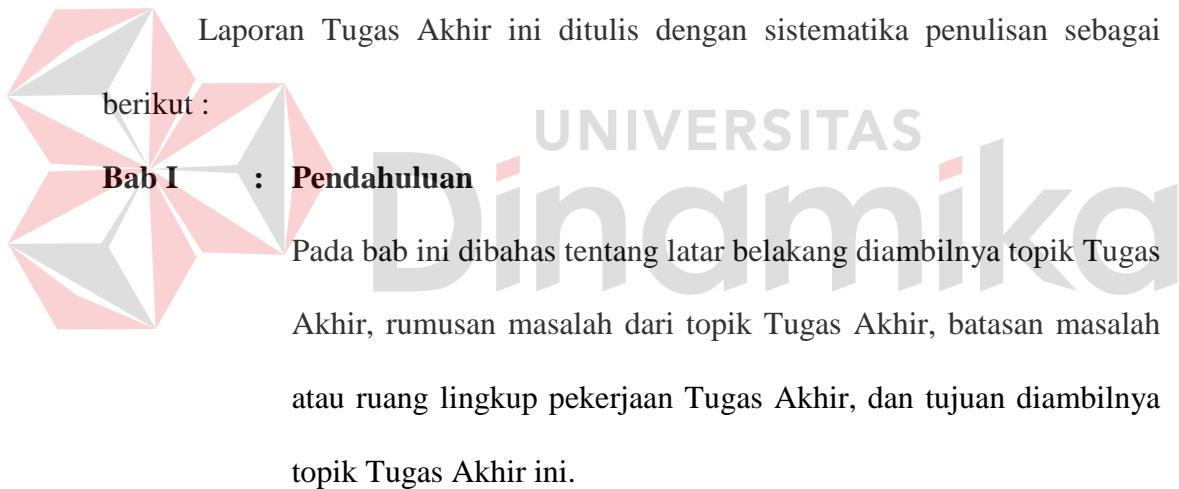
- a. Rancangan bangun ini didesain untuk penggunaan di Pintu Air Jagir.
- b. Parameter-parameter yang disensor untuk penentuan kondisi aktual sungai adalah ketinggian elevasi muka air sungai di bagian hulu yaitu sisi sungai sebelum melewati pintu air dan bagian hilir yaitu bagian sisi sungai setelah melewati pintu air.
- c. Untuk memantau ketinggian air, digunakan sensor ultrasonik yang dikontrol oleh mikrokontroler AVR.
- d. Komunikasi *wireless* dilakukan dengan menggunakan *walkie talkie* yang memiliki jarak jangkau kurang dari 1 km.
- e. Data yang dikirim melalui media *wireless* dengan *walkie talkie* masih rentan oleh derau

1.4 Tujuan

Tujuan dari pembuatan rancang bangun ini adalah:

- a. Bagaimana merancang dan membuat perangkat pemantauan kondisi sungai.
- b. Bagaimana cara membuat sistem komunikasi data jarak jauh, dari sensor ultrasonik ke perangkat pengolah data dengan memanfaatkan gelombang radio.
- c. Bagaimana merancang dan membuat aplikasi yang dapat memberi *advise* mengenai *setting* ketinggian pintu air berdasarkan kondisi aktual sungai pada saat itu dengan menggunakan logika *fuzzy*.

1.5 Sistematika Penulisan



Bab II : Landasan Teori

Pada bab ini dibahas tentang Pintu Air Jagir sebagai objek penelitian dari Tugas Akhir ini dan perangkat-perangkat yang digunakan dalam membuat Tugas Akhir ini, baik perangkat keras (*hardware*) maupun perangkat lunak (*software*).

Bab III : Metode Penelitian dan Perancangan Sistem

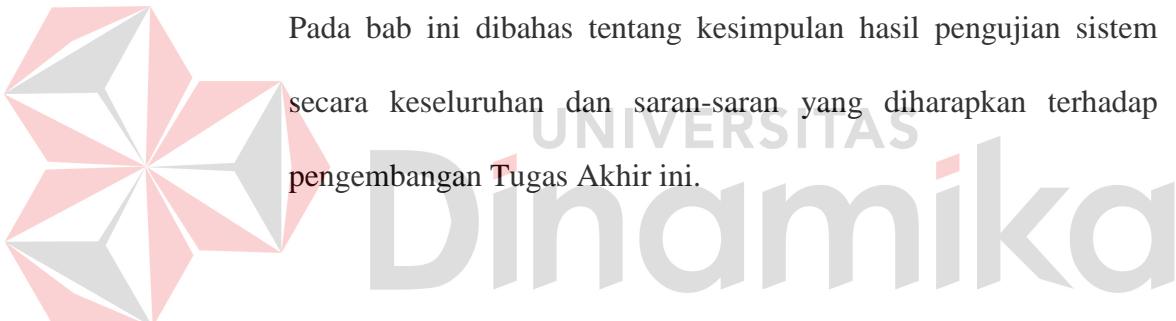
Pada bab ini dibahas mengenai perancangan perangkat keras dan perangkat lunak. Perangkat keras meliputi mikrokontroler, sensor-sensor, dan perangkat komunikasi yang digunakan. Sedangkan perangkat lunak yang digunakan meliputi pemrograman menggunakan bahasa C, bahasa Visual Basic, dan SQL.

Bab IV : Evaluasi dan Implementasi

Pada bab ini dibahas tentang pengujian sistem, baik perangkat lunak maupun perangkat lunak.

Bab V : Penutup

Pada bab ini dibahas tentang kesimpulan hasil pengujian sistem secara keseluruhan dan saran-saran yang diharapkan terhadap pengembangan Tugas Akhir ini.



BAB II

LANDASAN TEORI

2.1 PINTU AIR

2.1.1 Gambaran Umum DAM Jagir

DAM jagir terletak pada Kali Wonokromo, di kecamatan Wonokromo kota Surabaya, kurang lebih 2,5 Km di hilir Bendungan Gunung Sari Baru.

DAM Jagir ini menerima aliran dari Kali Surabaya dari Bendungan Kali Gunung sari Baru ditambah aliran air dari Kali Kedurus. Drainase kota dibuang melalui pompa Gunungsari I ke Kali Surabaya melalui pompa Gunungsari II ke Sungai Kedurus di dekat Muara Kali Kedurus.

Tepat di hulu DAM Jagir, disisi sebelah kiri sungai terdapat tempat pengambilan air untuk instalasi Pengolahan Air Minum (IPAM) Ngagel yang mengolah air baku untuk penyediaan air kota Surabaya dan dikelola oleh Perusahaan Daerah Air Minum (PDAM) Surabaya.

DAM Jagir awalnya merupakan bendungan bergerak dengan pintu kayu rangka baja dengan pemberat batu dan dibangun pada tahun 1917 oleh Pemerintahan Belanda serta dioperasikan secara manual.

Fungsi DAM Jagir yaitu untuk mengatur elevasi muka air sungai agar dapat melayani kebutuhan air di IPAM Ngagel, mengendalikan banjir Kali Surabaya dan Kali Wonokromo, mengalirkan air Kali Surabaya ke Kali Mas dan menahan intrusi air laut.

2.1.2 Operasi Pintu Air

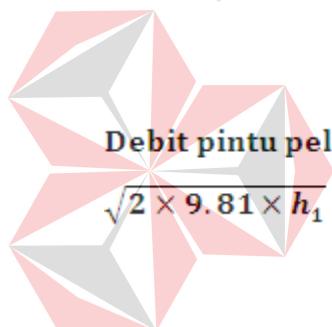
- a. Untuk melayani kebutuhan air IPAM Ngagel dan menjaga agar tidak terjadi genangan/limpasan di daerah sekitar Kali Surabaya (drainase kota, Kali Kedurus mengalir dengan lancar) khususnya pada saat kondisi banjir, maka elevasi muka air sungai di hulu Dam Jagir harus di atur pada El. 2,80 m – 3,00 m. Khusus untuk keperluan penggelontoran sampah diizinkan menaikkan elevasi muka air sesaat El. 3,35 m. Pengendalian elevasi muka air waduk tersebut dilakukan dengan membuka atau menutup Pintu Pengendali Banjir atau Pintu Pelimpah.
- b. Pada kondisi normal (debit $< 180 \text{ m}^3 / \text{det}$), elevasi muka air di atur antara El. 3,00 m – 3,30 m. Jika debit ke hilir Dam $Q \leq 7,0 \text{ m}^3 / \text{det}$, maka air dia atur melalui pintu pelimpah. Tabel hubung antara elevasi muka air hulu dan debit keluar dari pintu pelimpah pada bukaan pintu penuh dapat di lihat pada tabel 1. Jika debit keluar hilir dan Dam $Q > 7,0 \text{ m}^3 / \text{det}$. Maka air diatur melalui pintu pengendali banjir. Tabel hubungan antara elevasi muka air di hilir, bukaan pintu pengendali banjir pada berbagai ketinggian muka air di hulu
- c. Pada saat kondisi banjir yaitu saat debit mencapai $\geq 180 \text{ m}^3 / \text{det}$ atau terjadi daerah genangan di daerah hulu K. Kedurus akibat air K. Kedurus tidak dapat mengalir masuk ke K. Surabaya, maka elevasi muka air diatur antara El. 2,80 m – 3,10 m. Dalam keadaan khusus, misalnya ada pekerjaan pemeliharaan sungai hilir dam, maka untuk sementara waktu batasan debit tersebut bisa disesuaikan, Selanjutnya harus diikuti ketentuan-ketentuan dalam buku pedoman siaga banjir seperti berikut :

Ketentuan pembukaan Pintu Pengendali banjir :

- Pintu Pengendali Banjir No. 1 (P1) terletak di sisi paling kiri, diikuti dengan pintu No. 2 (P2) di bagian tengah dan pintu No. 3 (P3) disisi paling kanan.
- Pintu Pengendali Banjir yang di buka dahulu yaitu P2, diikuti oleh P1 dan P3
- Tinggi bukaan pintu P2 harus \geq tinggi bukaan pintu P1 dan P3
- Tinggi bukaan pintu P1 dan P3 harus sama

2.1.3 Rumus Pintu Air Jagir

$$\text{Debit pintu pelimpah} \left(\frac{\text{m}^3}{\text{s}} \right) = \frac{0.6}{1+0.588 \times \sqrt{\frac{a}{h_1}}} \times a \times 4.5 \times \sqrt{2 \times 9.81 \times h_1} \quad (2.1)$$



Debit pintu pelimpah yang dibuka penuh $\left(\frac{\text{m}^3}{\text{s}} \right) = 0.5 \times \frac{2}{3} \times 4.5 \times h_1 \times \sqrt{2 \times 9.81 \times h_1}$ (2.2)

h_1 = tinggi muka air di atas pintu (m)

a = bukaan pintu pelimpah (m)

$$\text{Jika } h_d \leq 0 \quad (2.3)$$

$$\text{Jika } \frac{h_d}{a} < 0.9 \quad (2.4)$$

$$\text{Debit keluar pintu pengendali banjir dam Jagir} = \frac{0.405}{\left(1 + 0.405 \times \frac{a}{h_u - h_p} \right)^{0.555}} \times a \times B \times (2 \times 9.81 \times h_u - h_p)^{0.555} \quad (2.5)$$

$$\text{Jika } \frac{h_d}{a} \geq 0.9 \quad (2.6)$$

Debit keluar pintu pengendali banjir dam Jagir = $0.796 \times a \times B \times (2 \times 9.81 \times (h_u - h_d))^{0.555}$

$$(2.7)$$

$$\text{Jika } h_d > 0 \quad (2.8)$$

$$\text{Jika } \frac{h_d}{a} < 0.9 \quad (2.9)$$

Debit keluar pintu pengendali banjir dam Jagir = $\frac{0.405}{(1+0.405 \times \frac{a}{h_u - h_p})^{0.53}} \times a \times B \times (2 \times 9.81 \times h_u - h_p)^{0.53}$

$$(3.0)$$

Debit keluar pintu pengendali banjir dam Jagir = $0.59 \times a \times B \times (2 \times 9.81 \times (h_u - h_d))^{0.53}$

$$(3.1)$$

h_u = elevasi muka air di Hulu (m)

h_d = elevasi muka air di Hilir (m)

h_p = elevasi dasar pintu (-1.96 m)

a = tinggi bukaan pintu (m)

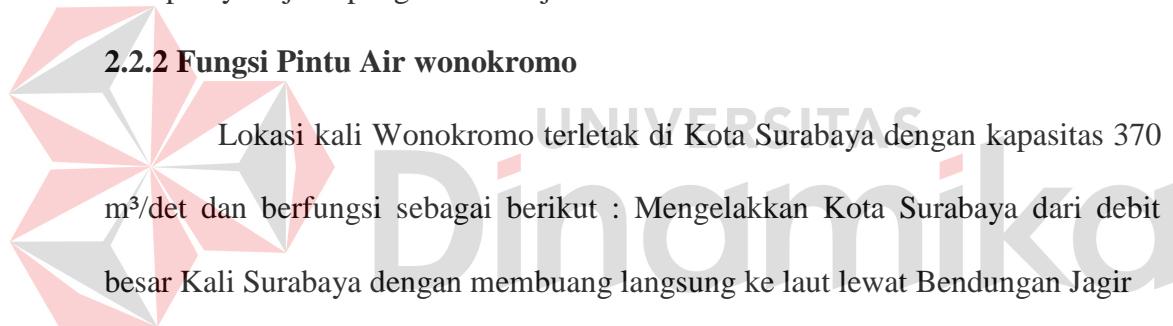
B = lebar pintu (8.30 m)

(sumber : Perum jasa tirta)

2.2 Pedoman Siaga Banjir

2.2.1 Maksud dan Tujuan

Penyusunan Pedoman siaga Banjir Musim Tahun 2009-2010 dimaksudkan untuk digunakan sebagai salah satu perangkat pendukung tercapainya tujuan pengendali banjir di DAS kali Brantas.



2.2.2 Fungsi Pintu Air wonokromo

Lokasi kali Wonokromo terletak di Kota Surabaya dengan kapasitas 370 m³/det dan berfungsi sebagai berikut : Mengelakkan Kota Surabaya dari debit besar Kali Surabaya dengan membuang langsung ke laut lewat Bendungan Jagir

2.2.3 Tingkat Siaga

Bahaya banjir dibagi dalam tiga tingkatan siaga yaitu siaga hijau, kuning dan merah. Yang secara berurutan menggambarkan tingkat bahaya yang lebih tinggi. Batas siaga untuk masing-masing tingkat siaga banjir di :

- a. Masing-masing stasiun pemantuan disungai, waduk, bendungan, pintu air dapat dilihat pada peta siaga banjir pada lampiran-1
- b. masing-masing curah hujan pada lampiran-2

Tingkat siaga tersebut diatas di tentukan antara lain berdasarkan :

- a. Buku manual operasi dan pemeliharaan bangunan yang bersangkutan. Selengkapnya dapat dilihat di lampiran-3.

- b. Kondisi aktual bangunan/lokasi di sungai sebagai hasil inspeksi
- c. Distribusi debit banjir debit banjir di DAS kali brantas yang disusun berdasarkan perencanaan dan yang sudah terjadi, selengkapnya bisa dilihat di lampiran-4

Dan dari data yang kami dapatkan tingkatan siaga banjir pada kali wonokromo dapat dilihat pada tabel 2.3.

Tabel 2.1 Siaga banjir

JENIS SIAGA	ELAVASI (SHVP,M)	FREE BOARD(M)	DEBIT(M3/DT)
Hijau	> 3,20	<1,30	>180
Kuning	> 3,25	<1,25	>200
Merah	> 3,30	<1,00	>220

(Sumber : Buku siaga banjir dari perum jasa tirta)

2.3 Metode Fuzzy

2.3.1 Pengantar Fuzzy

Fuzzy pertama kali diperkenalkan oleh Prof.Lotfi A. Zadeh . Logika Fuzzy adalah peningkatan dari logika Boolean yang berhadapan dengan konsep kebenaran sebagian. Di mana logika klasik menyatakan bahwa segala hal dapat diekspresikan dalam istilah binary (0 atau 1, hitam atau putih, ya atau tidak), logika fuzzy menggantikan kebenaran boolean dengan tingkat kebenaran .

Alasan menggunakan fuzzy adalah sebagai berikut.

1. Konsep fuzzy yg mudah dimengerti
2. Logika fuzzy sangat fleksibel
3. Logika fuzzy memiliki toleransi terhadap data yang tidak tepat

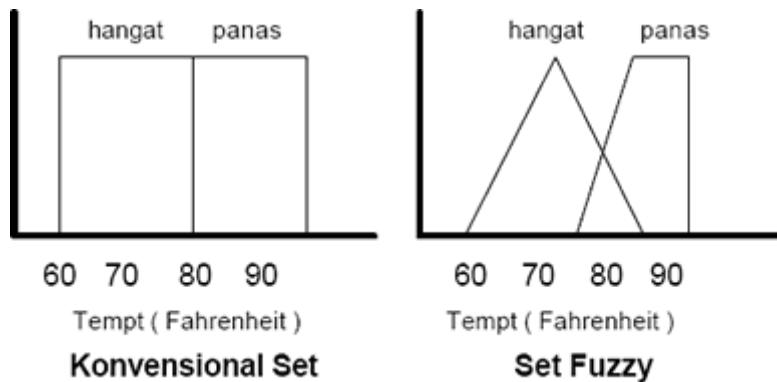
4. Logika *fuzzy* mampu memodelkan fungsi-fungsi non linier yang sangat kompleks
5. Logika *fuzzy* dapat bekerja sama dengan teknik kendali konvensional
6. Logika *fuzzy* didasarkan pada bahasa alami.

Banyak riset yang telah dilakukan oleh Prof. Lotfi A Zadeh, dan banyak juga perkembangannya di Amerika, Eropa, dan Jepang. Di Amerika, banyak penelitian *fuzzy* dalam bidang control (*fuzzy logic control*) yang dikembangkan oleh Mohammed El Hawary (Dalhousie Univ.), Malik (Calgary Univ), dan El Sharkawi (Washington Univ.). Perkembangan di Eropa dipelopori oleh Prof. Mamdani dan Miranda. Sedangkan di Jepang, *fuzzy* banyak dikembangkan oleh Prof. Sugeno dan Mizumoto (Tokyo Univ.), Tanaka (Osaka Perfecture Univ.), Hiyama (Kumamoto Univ.), Yamakawa (Kyushu Institute of Tech.) dan Nishimura Katsumi (Tottori Univ.)[Desain Sistem Tenaga Modern, Prof. Imam Robandi].

2.3.2 Dasar - Dasar Pemahaman Logika Fuzzy

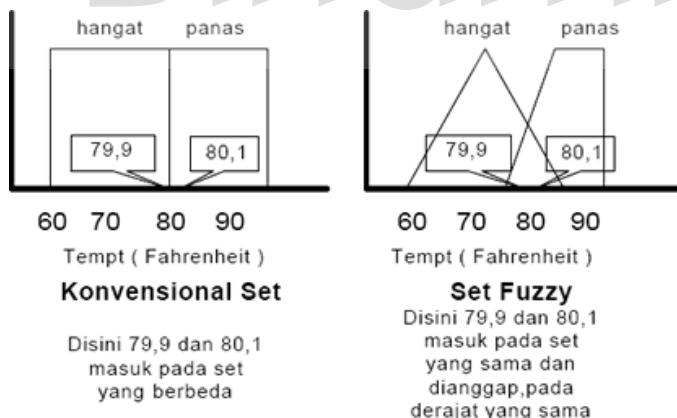
Sebagai contoh, apakah 80 derajat *fahrenheit* tergolong hangat atau panas. Dalam logika *fuzzy*, dan dalam dunia nyata, “kedua-duanya benar” mungkin merupakan jawabannya. Seperti yang anda lihat pada grafik *fuzzy* pada gambar 2.34, 80 derajat adalah sebagian hangat dan sebagian panas dalam gambaran set *fuzzy*. Sementara hal ini dapat dibenarkan bahwa tumpang tindih antara set dapat terjadi dalam logika boolean, transisi dari set ke set terjadi seketika itu juga (yaitu elemen yang dapat menjadi anggota set atau tidak).

Dengan logika *fuzzy*, sementara itu, transisi dapat bertingkat – tingkat (yaitu elemen dapat memiliki sebagian keanggotaan dalam sejumlah set)



Gambar 2.1 Konvensional set dan Set fuzzy

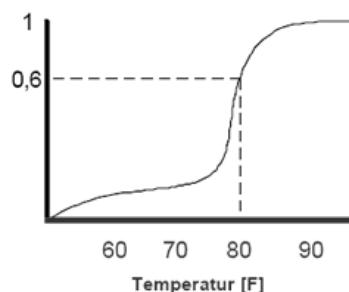
Dalam logika klasik menggunakan set konvensional yang ditunjukkan pada gambar 2.35, 79,9 derajat dapat diklasifikasikan sebagai hangat, dan 80,1 derajat dapat diklasifikasikan sebagai panas. Perubahan kecil dalam sistem dapat menyebabkan perbedaan reaksi yang berarti. Dalam sistem *fuzzy*, perubahan kecil temperatur akan memberikan hasil perubahan yang tidak jelas pada kinerja sistem.



Gambar 2.2 Konvensional set dan Set fuzzy

Logika *Fuzzy* mengenali tidak hanya *clear cut*, alternatif hitam dan putih, tapi juga tingkatan tak terbatas antara keduanya. Hal ini mungkin kelihatan tidak jelas, namun logika *fuzzy* menghilangkan banyak keraguan dengan menentukan

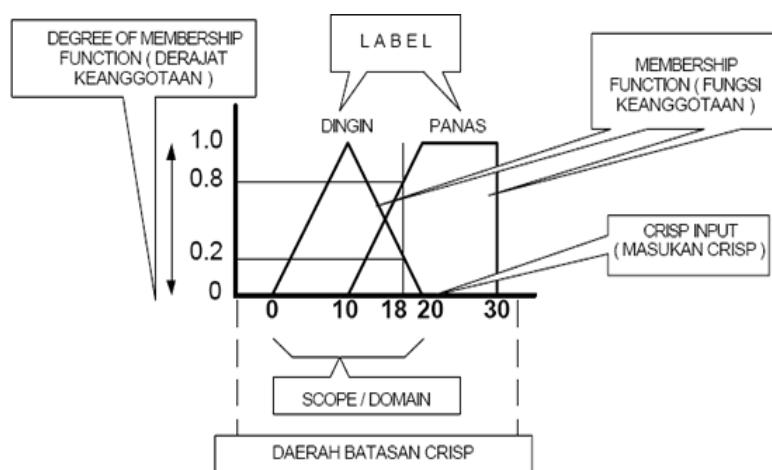
nilai tertentu pada tingkatan tersebut. Sebagai contoh temperatur 80 derajat *fahrenheit* dapat diklasifikasikan antara daerah nol sampai satu sebagai panas pada tingkat 0,6. Nilai ini kemudian digunakan untuk menurunkan kepastian atau penyelesaian *crisp* terhadap masalah perancangan. Walaupun kelihatan kontradiksi dari namanya, logika *fuzzy* memberikan metoda ketepatan yang dapat diandalkan dari persoalan pengambilan keputusan *crisp*.



Gambar 2.3 Contoh pengambilan keputusan *crisp*



Untuk mengerti sistem *fuzzy*, anda harus mengenal konsep dasar yang berhubungan dengan logika *fuzzy*.



Gambar 2.4 Konvensional set dan Set *fuzzy*

2.3.3 Derajat Keanggotaan

Adalah derajat dimana nilai *crisp* compatible dengan fungsi keanggotaan (dari 0 sampai 1), juga mengacu sebagai tingkat keanggotaan, nilai kebenaran, atau masukan *fuzzy*.

2.3.4 Label

Adalah nama deskriptif yang digunakan untuk mengidentifikasi sebuah fungsi keanggotaan.

2.3.5 Fungsi Keanggotaan

Adalah mendefinisikan *fuzzy* set dengan memetakan masukan *crisp* dari domainnya ke derajat keanggotaan. MASUKAN *CRISP* adalah masukan yang tegas dan tertentu.

2.3.6 Lingkup atau Domain

Adalah lebar fungsi keanggotaan. Jangkauan konsep, biasanya bilangan, tempat dimana fungsi keanggotaan dipetakkan. Disini domain dari *fuzzy* set (fungsi keanggotaan) adalah dari 0 sampai 20 derajat dan lingkupnya adalah 20 derajat.

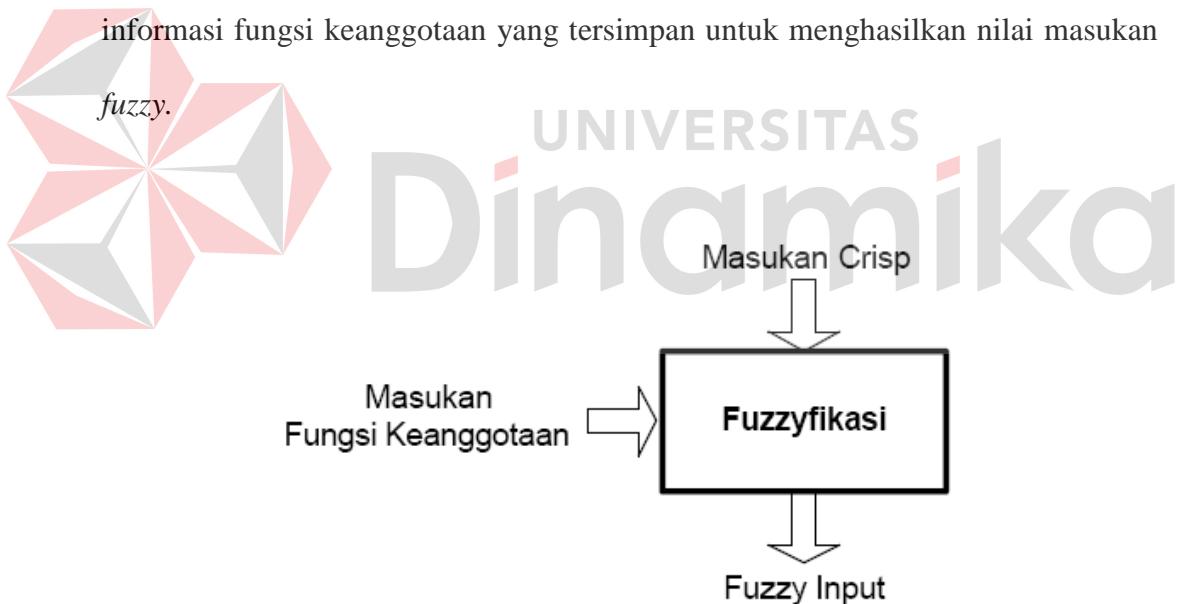
2.3.7 Daerah Batasan Crips

Adalah jangkauan seluruh nilai yang mungkin dapat diaplikasikan pada variabel sistem. Menggunakan logika *fuzzy* untuk mencapai penyelesaian crisp pada masalah khusus biasanya melibatkan tiga langkah : *fuzzyifikasi*, evaluasi rule, dan defuzzyifikasi. Jika anda tidak atau belum mengetahui bagaimana logika *fuzzy* itu bekerja kami sarankan agar anda benar benar memahami betul tiap langkah dalam bagian berikutnya.

2.3.8 Fuzzyifikasi dan Fungsi Keanggotaan

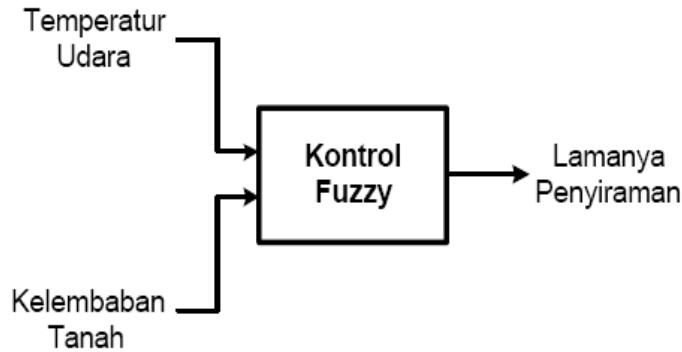
Langkah pertama dalam memproses logika *fuzzy* mengandung transformasi domain yang dinamakan *fuzzyifikasi*. Masukan *crisp* ditransformasikan kedalam masukan *fuzzy*. Sebagai contoh, masukan crisp 78 derajat akan ditransformasikan sebagai hangat dalam bentuk *fuzzy*. 90 mph akan ditransformasikan menjadi cepat, dan sebagainya. Untuk mengubah bentuk masukan *crisp* kedalam masukan *fuzzy*, fungsi keanggotaan pertama kali harus ditentukan untuk tiap masukan.

Sekali fungsi keanggotaan ditentukan, *fuzzyifikasi* mengambil nilai masukan secara *realtime*, seperti temperatur, dan membandingkannya dengan informasi fungsi keanggotaan yang tersimpan untuk menghasilkan nilai masukan



Gambar 2.5 Fungsi keanggotaan

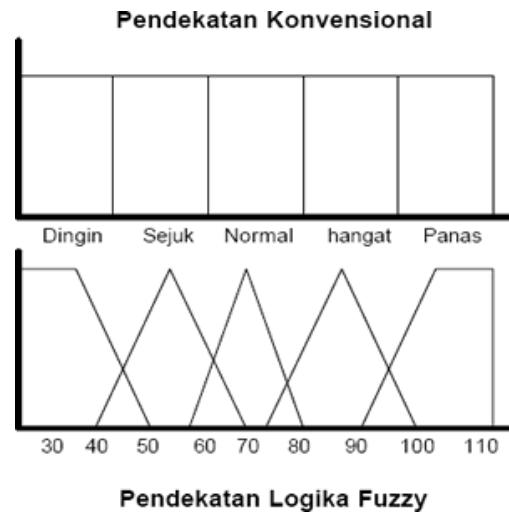
Untuk menggambarkan proses *fuzzyifikasi*, mari kita lihat sistem penyiraman tanaman. Pengendali *fuzzy* dalam sistem ini akan menggunakan dua masukan, temperatur udara luar dan kondisi tanah, untuk mengkalkulasi lamanya penyiraman



Gambar 2.6 Proses fuzzyfikasi

Langkah pertama dalam *fuzzyifikasi* adalah menentukan label label *fuzzy* pada daerah batasan crisp dari setiap dapat menentukan daerah label misalkan kita bagi menjadi 5 label, Dingin, Sejuk, Normal, Hangat, Panas. Dan untuk kondisi tanah kita bagi menjadi 3 label, Kering, Lembab, Basah. Untuk pembahasan selanjutnya dari bagian ini kita akan berkonsentrasi pada masukan temperature masukan *crisp*.

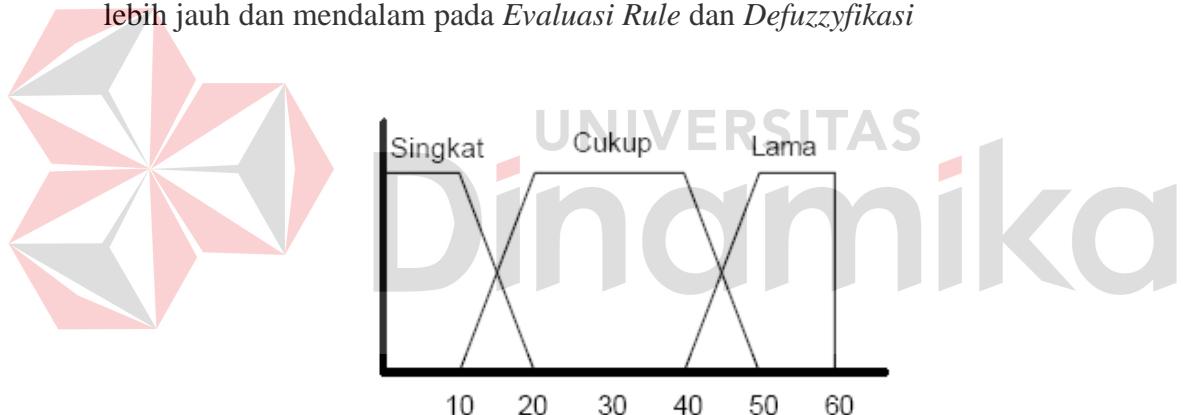
Berikutnya, fungsi keanggotaan dinyatakan untuk memberi arti numerik pada tiap label. Setiap fungsi keanggotaan mengidentifikasi daerah nilai masukan yang berkorespondensi dengan label. Tidak seperti logika boolean, fungsi keanggotaan label tidak mendefinisikan batas-batas dimana label diaplikasikan secara penuh pada satu sisi *cutoff* dan tidak pada semua sisi lain *cutoff*. Walaupun ada daerah dimana nilai masukan berubah secara bertahap dari keadaaan dapat diaplikasikan ke keadaan tak dapat diaplikasikan.



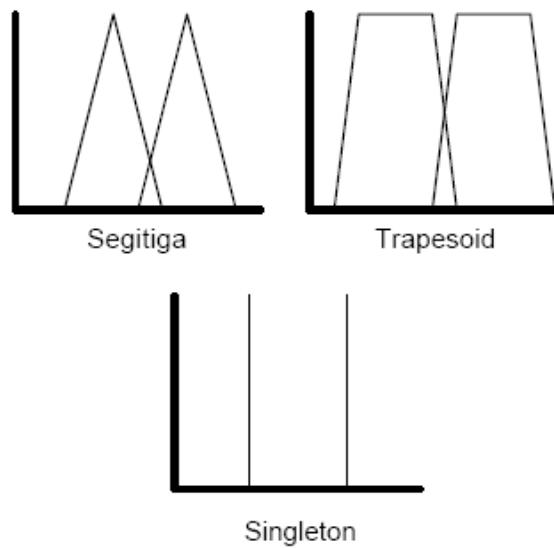
Gambar 2.7 Pendekatan konvensional dan Pendakata *fuzzy*

Keluaran fuzzy juga punya fungsi keanggotaan. Hal ini akan dibahas

lebih jauh dan mendalam pada *Evaluasi Rule* dan *Defuzzyifikasi*

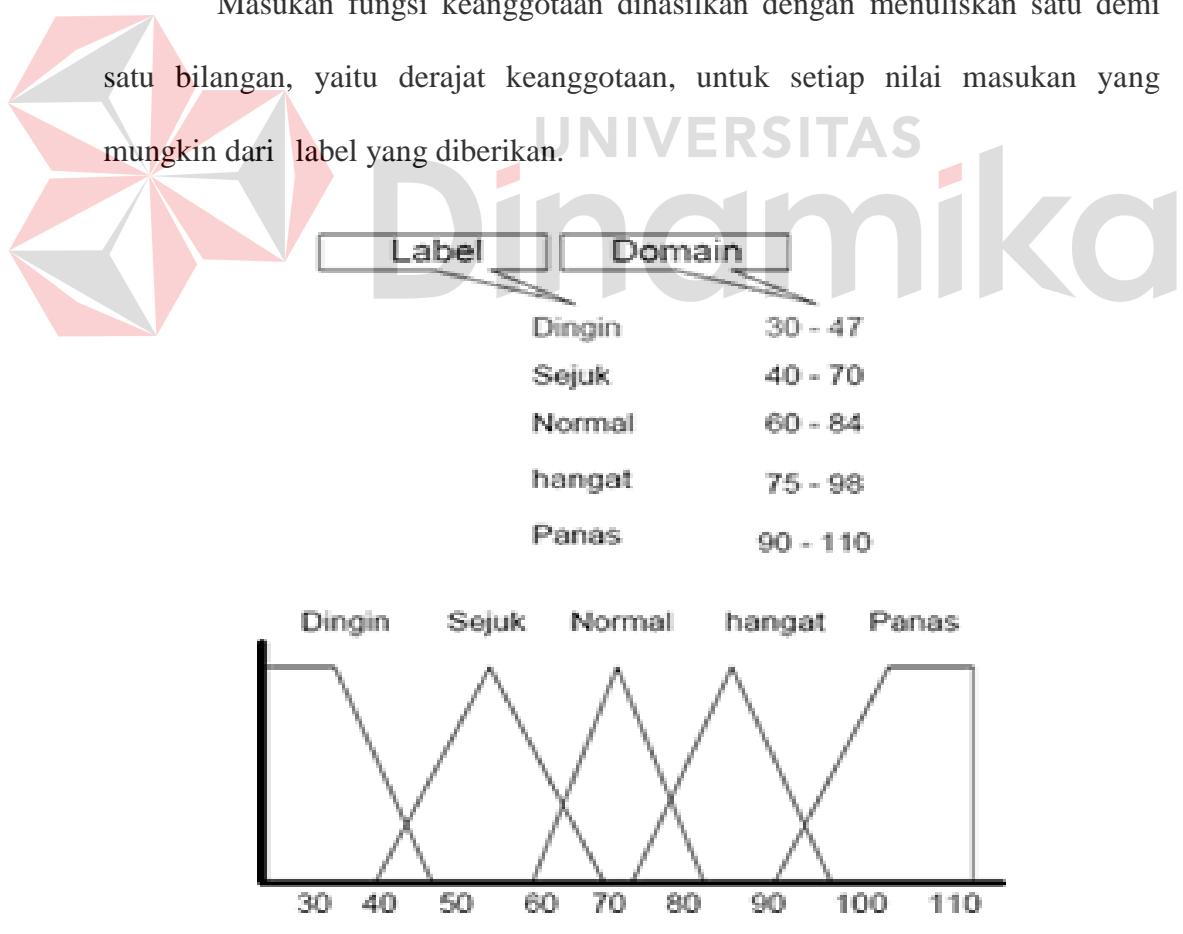


Gambar 2.8 Contoh pendekatan *fuzzy*



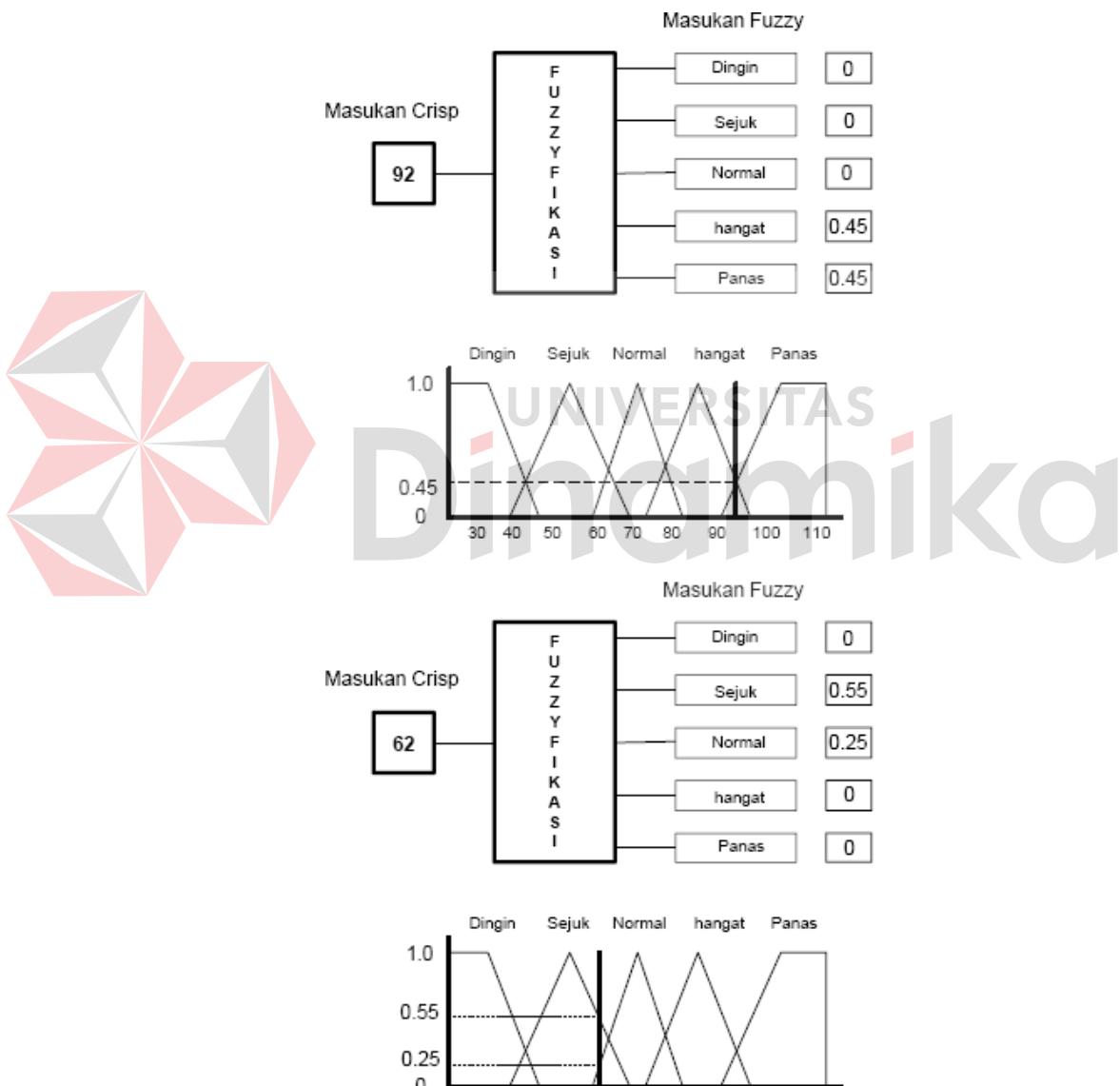
Gambar 2.9 Contoh bentuk *defuzzyfikas*

Masukan fungsi keanggotaan dihasilkan dengan menuliskan satu demi satu bilangan, yaitu derajat keanggotaan, untuk setiap nilai masukan yang mungkin dari label yang diberikan.



Gambar 2.10 Fungsi Keanggotaan

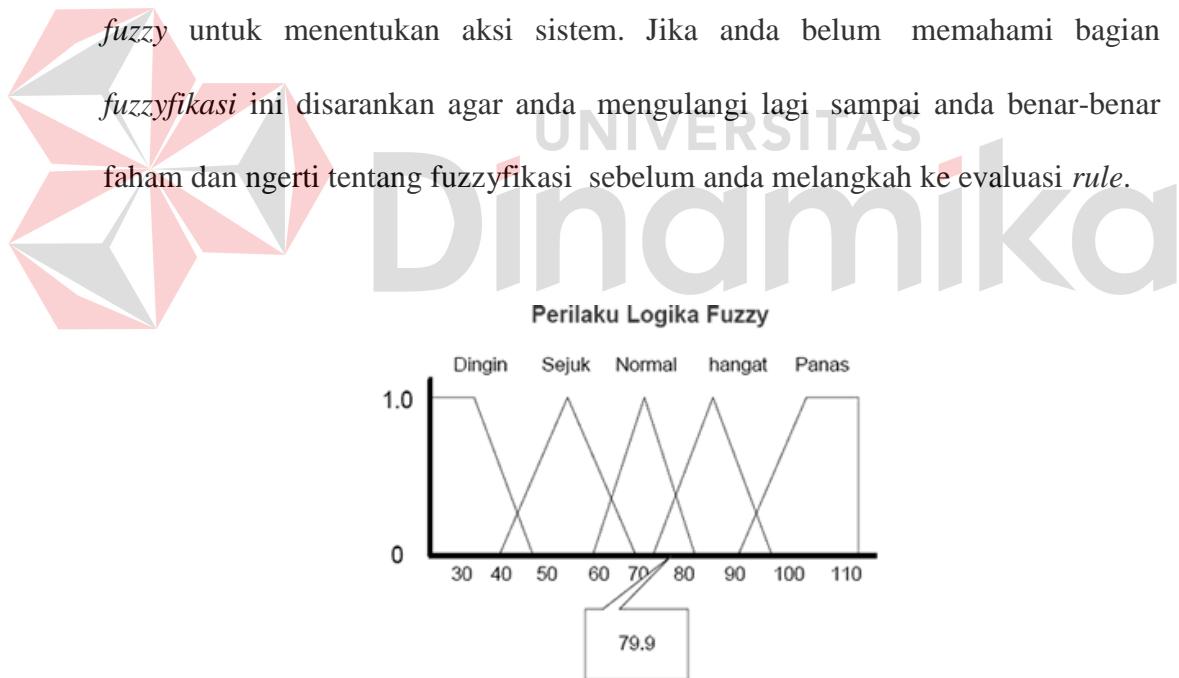
Harga (m) pada sumbu y mengacu pada derajat dimana nilai masukan *crisp* (temperatur) dipasang pada tiap label fungsi keanggotaan { sejuk, hangat, dan sebagainya}. Seperti yang anda lihat, nilai masukan dapat menjadi milik lebih dari satu set fuzzy. 92 derajat termasuk dalam set hangat dan juga set panas yang telah didefinisikan oleh fungsi keanggotaan.



Gambar 2.11 Proses fuzzy

Menguraikan masukan *crisp* dalam bentuk *fuzzy* mengijinkan sistem merespon secara halus perubahan dalam temperatur masukan. Sebagai contoh, reaksi sistem *fuzzy* terhadap aturan, “jika temperatur luar panas, maka lamanya penyiraman singkat, jika temperatur luar hangat maka lamanya penyiraman agak lama” akan berubah hanya sedikit jika temperatur luar bergerak dari 80 derajat ke 79,9 derajat. Respon sistem dalam bentuk lamanya penyiraman akan dihitung pada derajat keanggotaan temperatur masukan pada tiap set, pada keadaan ini, 80 derajat dan 79,9 derajat adalah normal sekaligus hangat.

Pada bagian atau langkah berikutnya setelah *fuzzyifikasi* yaitu evaluasi rule, kita akan mengetahui bagaimana aturan-aturan menggunakan masukan *fuzzy* untuk menentukan aksi sistem. Jika anda belum memahami bagian *fuzzyifikasi* ini disarankan agar anda mengulangi lagi sampai anda benar-benar faham dan ngerti tentang *fuzzyifikasi* sebelum anda melangkah ke evaluasi *rule*.



Perilaku Kode Konvensional

IF temperatur $\geq 80^\circ$, **THEN** lamanya penyiraman 50 menit.

IF temperatur $< 80^\circ$, **THEN** lamanya penyiraman 30 menit.

Gambar 2.12 Perilaku logika *fuzzy*

2.4 Sensor Ultrasonik

Sensor Ultrasonik sudah sering kita dengar, sensor ini memiliki frekuensi sebesar 40 kHz. Sensor ini terdiri dari pemancar dan penerima gelombang Ultrasonik.

Rangkaian pemancar akan memancarkan gelombang Ultrasonik dalam waktu 200 μ s. Gelombang ini melalui udara dengan kecepatan kurang lebih 344 meter/detik. Jika gelombang ini mengenai suatu objek, maka gelombang ini akan dipantulkan kembali ke penerima dari sensor Ultrasonik. Dengan mengatur lamanya waktu penerimaan gelombang Ultrasonik, kita dapat menentukan jarak dari suatu objek yang kita siapkan. (Sumber : <http://www.electroniclab.com>)



Gambar 2.13 Sensor Ultrasonik tipe SRF02

2.5 Komunikasi Data I2C

I2C termasuk jenis komunikasi serial dengan tipe protokol *Synchronous*, dimana setiap *bit* data masuk atau data keluar seirama atau sinkron dengan perubahan *clock*. Hal ini sangat beda dengan RS-232 dan tipe *protocol asynchronous* yang lain yang tidak mempunyai *clock*, akan tetapi pada tipe *asynchronous* data masuk dan data keluar berdasarkan pada kecepatan tertentu yang konstan. I2C versi 1.0 dikenalkan oleh Philips pada tahun 1992, direvisi

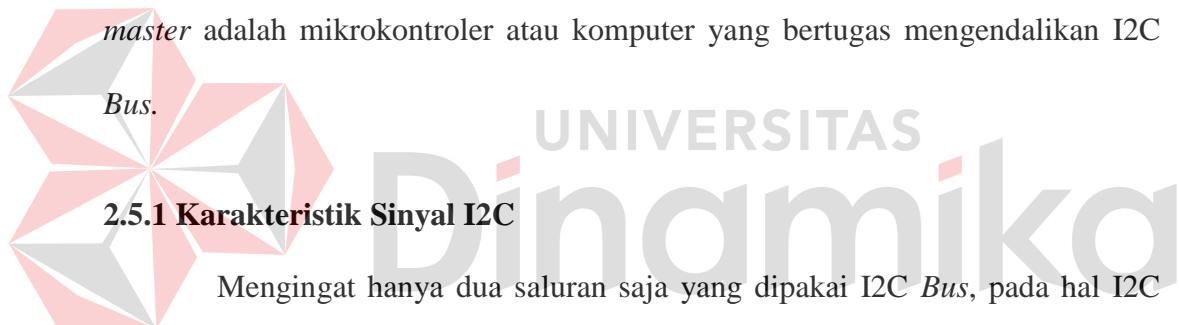
menjadi versi 2.0 pada tahun 1998, setahun kemudian direvisi lagi menjadi versi 2.1 Komunikasi data secara I2C dilakukan melalui dua saluran, masing-masing adalah :

saluran data secara seri (SDA) = *Serial Data*

saluran *clock* (SCL) = *Serial Clock*

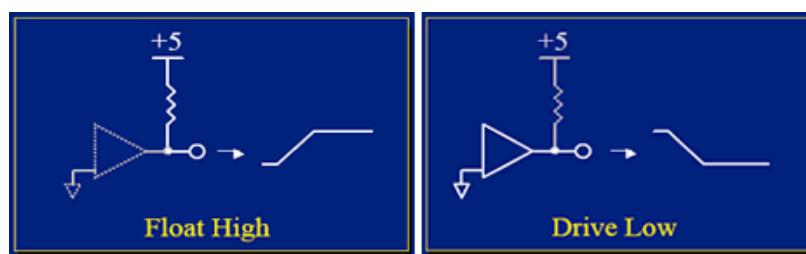
kedua saluran ini dikenal sebagai I2C *Bus* yang dipakai menghubungkan banyak IC I2C untuk berbagai macam keperluan.

IC-IC I2C itu dibedakan menjadi induk (*master*) dan anak buah (*slave*), yang dimaksud dengan induk adalah peralatan I2C yang memulai *transfer* data dan yang membangkitkan *clock* (SCK). Yang bertindak sebagai *master* adalah mikrokontroler atau komputer yang bertugas mengendalikan I2C



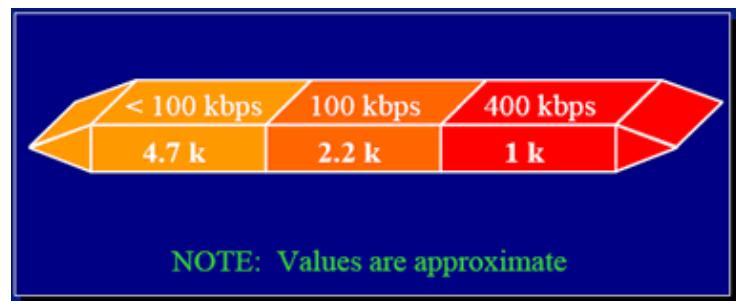
Mengingat hanya dua saluran saja yang dipakai I2C *Bus*, pada hal I2C *Bus* diharapkan bisa dipakai membentuk jaringan kecil dengan banyak peralatan I2C.

Maka dalam konsep I2C ditentukan sinyal dan tatacara dasar untuk memperlancar komunikasi antar peralatan I2C tersebut.

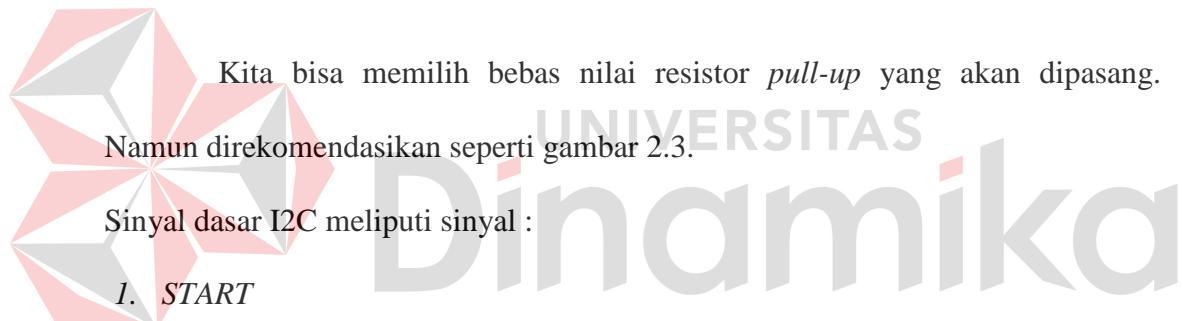


Gambar 2.14 Level sinyal I2C

Jalur I2C hanya mempunyai dua keadaan (Secara *Electrical*) yaitu ***Float High*** dan ***Drive Low***. I2C bekerja dengan Resistor *Pull-up*. Jika tidak ada *device* yang terkoneksi maka jalur ini menjadi “*Float High*”, oleh karena itu diperlukan resistor *pull-up* pada I2C.



Gambar 2.15 I2C *Pull-up Resistor Setting Suggestion*



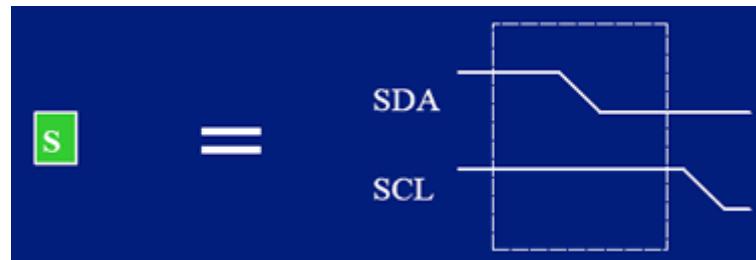
5. Kondisi dimana Data dianggap valid
6. Kondisi ACK
7. Kondisi NACK



Gambar 2.16 Sinyal dasar I2C

A. START CONDITION

1. Inisialisasi ke jalur I2C
2. SDA diturunkan ke nilai “0” saat SCL bernilai “1”



Gambar 2.17 Kondisi *Start*

Elemen yang pertama adalah “*START*”. Kondisi *START* mengindikasikan bahwa *device* akan memulai *transfer* data pada bus I2C.

Pertama SDA menjadi *LOW* diikuti kemudian SCL.

B. STOP CONDITION

1. Release ke jalur I2C
2. SDA di *release* saat SCL bernilai “1”



Gambar 2.18 Kondisi *Stop*

Elemen berikutnya yaitu “*STOP*” *condition*. *STOP* mengindikasikan bahwa *device* telah selesai melakukan transfer data dan akan *release* dari *Bus*.

Sinyal yang dibentuk dari STOP ini pertama adalah jalur SCL *release* diikuti dengan SDA yang juga *release*.

Perlu diingat bahwa ketika jalur *release* berarti *driver* dalam keadaan OFF sehingga pada jalur tinggal resistor *pull-up*, dan akhirnya masuk kedalam keadaan “*Float High*”. Setelah STOP, jalur I2C semua menjadi HIGH dan disebut dengan kondisi *idle*. Pada saat *idle* ini kondisi START dapat dimulai lagi untuk memulai mengirim data.

C. RESTART CONDITION

1. Menginisialisasi ulang jalur I2C
2. Dipakai *START* tidak mengikuti *STOP*



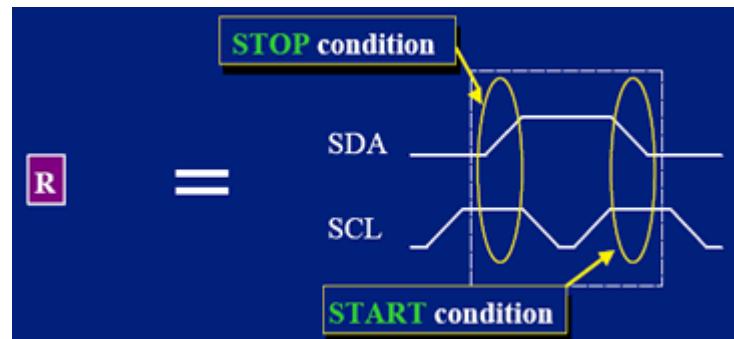
Gambar 2.19 Kondisi *Stop*

Kondisi *RESTART* mengindikasikan bahwa *device* ingin mengirimkan data lebih banyak lagi.

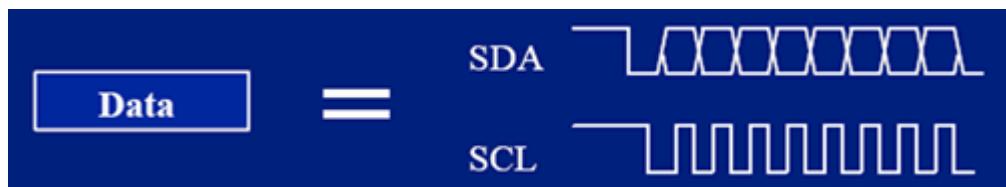
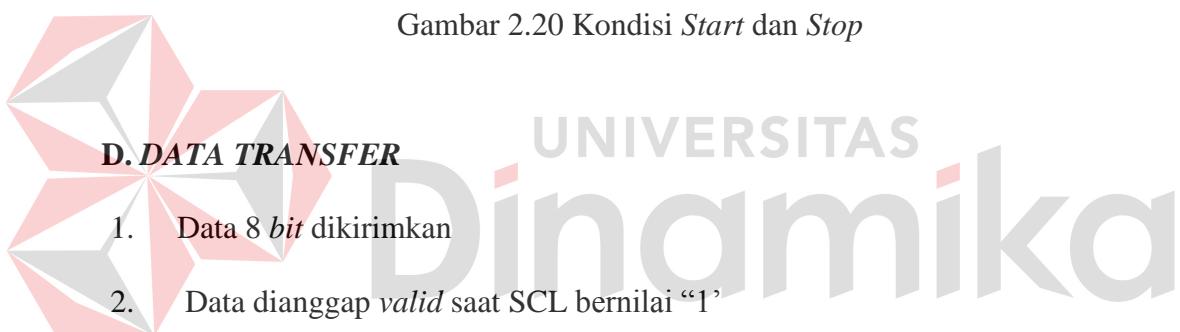
Biasanya kondisi *RESTART* dilakukan ketika *START* telah terkirim tetapi kondisi *STOP* tidak kunjung terjadi. Sebagai contoh misalnya, jika anda berkomunikasi dengan suatu *device* serial EEPROM, kita mungkin tidak menginginkan terjadi interupsi ketika mengirimkan sejumlah data ke serial EEPROM tersebut, *RESTART* bisa membantu untuk meng-*handle* permasalahan

ini. Kondisi *RESTART* bisa disimpulkan bahwa mula-mula jalur I2C pada kondisi *STOP* kemudian diikuti kondisi *START*.

1. *STOP* = SDA dinaikkan ke nilai “1” saat SCL bernilai “1”
2. *START* = SDA diturunkan ke nilai “0” saat SCL bernilai “1”



Gambar 2.20 Kondisi *Start* dan *Stop*



Gambar 2.21 Transfer Data

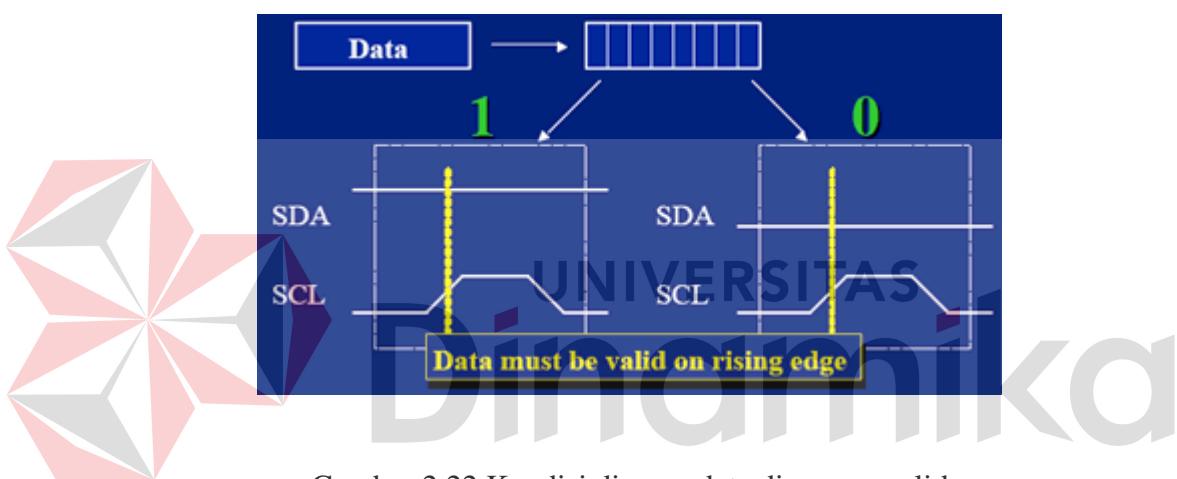
Blok data pada gambar 2.21 mempresentasikan proses transfer informasi data sebanyak 8 bit. Data dikirim melalui jalur SDA, sedangkan SCL tetap menghasilkan *clock* sebagai pendorong pergerakan setiap *bit* data.

Data pada jalur SDA *valid* dan boleh dibaca ketika jalur *clock SCL high*.

Karena pada saat SCL tidak *high (LOW)* data bisa berubah sesuai dengan nilainya.

Data bit ini digunakan untuk mentransfer semua jenis informasi data, bisa jadi data tersebut sebagai *control code*, *address code* atau data *code* itu sendiri tergantung dari *protocol* yang ingin dibentuk.

E. DATA STATES

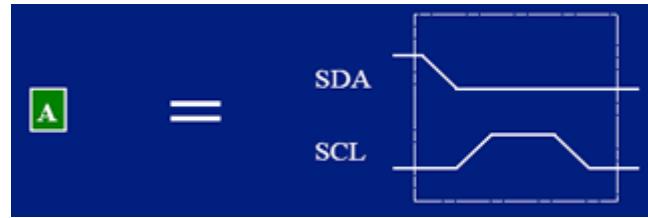


Gambar 2.22 Kondisi dimana data dianggap valid

Sebuah blok data pada gambar 2.10 bisa dilihat, bahwa sebuah data *valid* boleh dibaca hanya ketika SCL transisi dari LOW ke HIGH, dan masih *valid* sampai SCL *high*.

F. ACKNOWLEDGE CONDITION

1. Mengetahui terjadinya Transfer Data
2. Kondisi ACK terjadi disaat penerima membuat SDA bernilai “0”



Gambar 2.23 Kondisi *ACK*

Sebuah *device* dapat mengirimkan sinyal “ACK” (Sebagai tanda bahwa data telah diterima dengan sempurna) dengan cara men-*drive* jalur SDA menjadi *Low* pada *clock* yang ke-9 dari SCL.

G. NACK CONDITION

1. Berkebalikan dengan ACK, yaitu mengacuhkan terjadinya Transfer Data
2. NACK terjadi saat penerima tidak membuat SDA bernilai “0”



Gambar 2.24 Kondisi *NACK*

Kebalikan dari ACK, NACK merupakan tanda bahwa data tidak bisa diterima oleh *device*, sehingga tidak ada respon dari *device* yang membuat SDA dibiarkan dalam kondisi “*Float High*”.

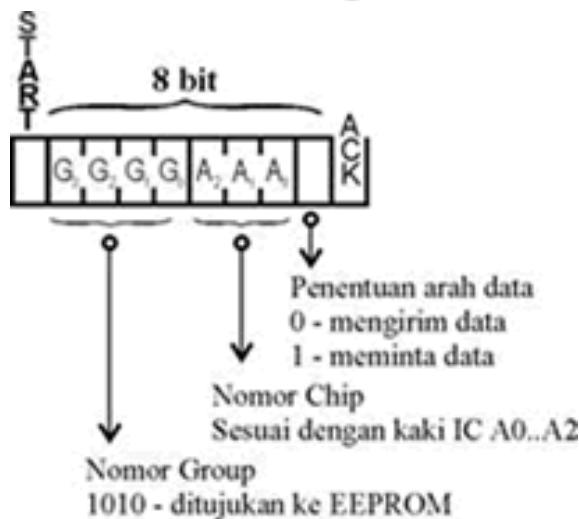
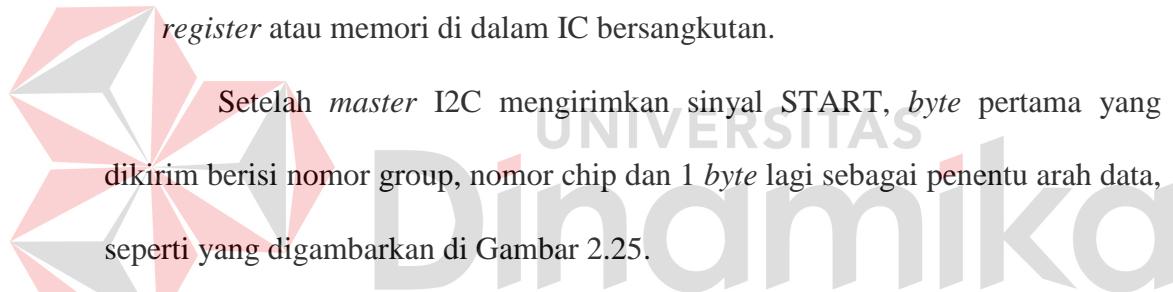
2.5.2 Metode Penomoran I2C

Karena IC I2C hanya dikendalikan lewat kaki SDA dan SCK saja, tidak ada sarana lainnya dari *master* yang bisa dipakai untuk mengendalikan I2C,

maka alamat yang dipakai untuk memilih isi IC I2C dikirimkan secara serial pula, persis seperti halnya pengiriman data.

Pengalamat dasar IC I2C dilakukan dengan nomor group dan nomor *Chip*.

- Nomor group adalah nomor yang diberikan oleh Philips (sebagai pencipta I2C) pada kelompok-kelompok IC I2C. Sebagai contoh nomor group untuk serial EEPROM adalah 1010 (biner).
- Nomor *chip* adalah nomor yang diberikan pada masing-masing *chip* lewat kaki A0, A1 dan A2 dari masing-masing IC. Dalam IC I2C tertentu, A0..A2 tidak dihubungkan ke kaki IC, tapi dipakai didalam IC untuk menomori *register* atau memori di dalam IC bersangkutan.



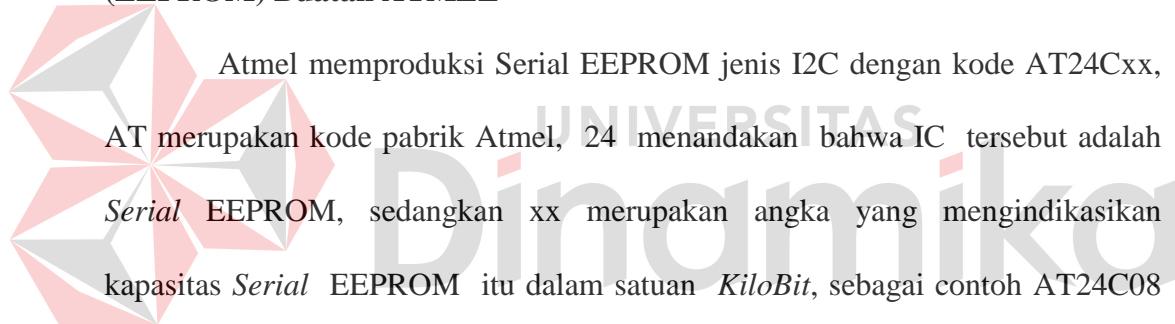
Gambar 2.25 Metode Penomoran I2C

Mekanisme kerja *byte* pertama tersebut bisa dijelaskan sebagai berikut:

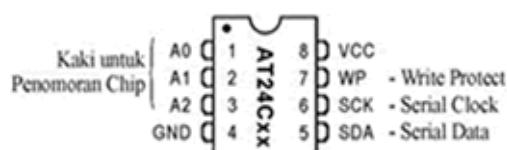
1. IC-IC pada I2C *Bus* yang mempunyai nomor Group sama dengan nomor group dalam *byte* pertama tersebut akan ter panggil
2. Berikutnya IC-IC dengan nomor group sama tersebut akan membandingkan nomor *chip* dalam *byte* pertama, nomor *chip* tidak mungkin sama.
3. Bit Penentu Arah Data dipakai untuk memberi tahu IC I2C arah data yang dikehendaki, apakah *master* akan mengirim data atau master menghendaki kiriman data.

2.5.3 IC Serial *Electronic Eraseble Programmable Read Only Memory*

(EEPROM) Buatan ATMEL



Atmel memproduksi Serial EEPROM jenis I2C dengan kode AT24Cxx, AT merupakan kode pabrik Atmel, 24 menandakan bahwa IC tersebut adalah *Serial EEPROM*, sedangkan xx merupakan angka yang mengindikasikan kapasitas *Serial EEPROM* itu dalam satuan *KiloBit*, sebagai contoh AT24C08 merupakan IC EEPROM I2C berkapasitas 8 *KiloBit* (1 *KiloByte*).



Type IC	Kapasitas (Byte)	Penomoran Chip	Max. Chip per Bus
AT24C01	128	Tidak ada	1
AT24C01A	128	A0,A1,A2	8
AT24C02	256	A0,A1,A2	8
AT24C04	512	A1,A2	4
AT24C08	1024	A2	2
AT24C16	2048	Tidak ada	1
AT24C164	2048	A0,A1,A2	8
AT24C32	4096	A0,A1,A2	8
AT24C64	8192	A0,A1,A2	8

Gambar 2.26 IC Serial EEPROM Buatan ATMEL

Kaki **SDA** (kaki nomor 5) dan kaki **SCK** (kaki nomor 6) merupakan kaki baku IC jenis I2C, kedua kaki inilah yang membentuk I2C *Bus*. Kaki nomor 7

(WP –Write Protect) merupakan kaki yang dipakai untuk melindungi isi yang disimpan di dalam IC Serial EEPROM, jika kaki ini diberi tegangan ‘1’ maka IC dalam keadaan ter-proteksi, isinya tidak dapat diganti. Agar bisa menuliskan informasi ke dalam IC ini, kaki ini harus diberi tegangan ‘0’.

Kaki nomor 1 sampai dengan nomor 3 (A0, A1 dan A2) merupakan fasilitas untuk penomoran chip, hal ini diperlukan kalau dalam satu rangkaian dipakai lebih dari satu IC EEPROM sejenis. Misalnya dalam satu rangkaian dipakai 3 chip AT24C02, SDA dan SCK ketiga IC ini masing-masing dihubungkan jadi satu membentuk I2C Bus, agar ketiga IC ini bisa dipakai secara terpisah kaki A0..A2 (kaki nomor 1 sampai nomor 3) masing-masing AT24C02

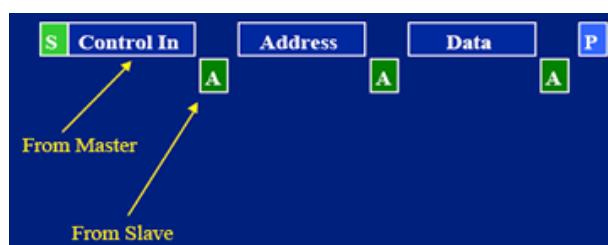
diberi level tegangan seperti terlihat pada Tabel 2.2.

Tabel 2.2 susunan A0...A2
Contoh Susunan A0..A2

	A2	A1	A0
AT24C02 No. 1	0	0	0
AT24C02 No. 2	0	0	1
AT24C02 No. 3	0	1	0

2.2.4 Proses Menulis Pada Serial EEPROM

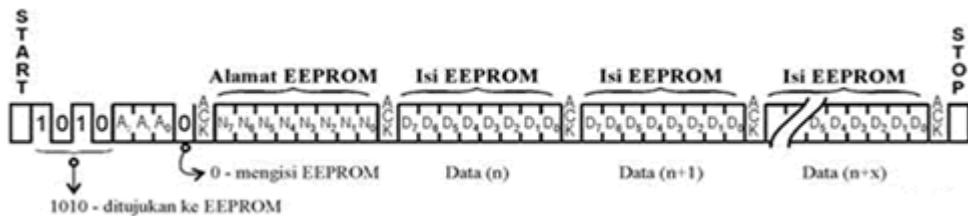
Secara umum proses pengisian atau penulisan data digambarkan pada ilustrasi pada gambar 2.27 :



Gambar 2.27 Proses Menulis Pada Serial EEPROM

Control in = 1010 000

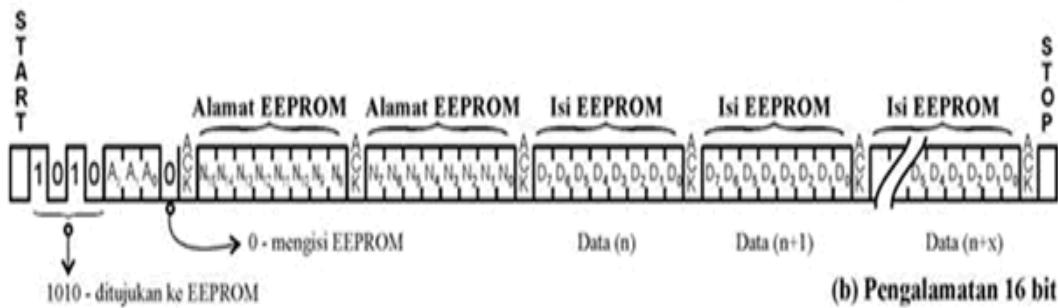
Secara lebih detail gambar 2.15 memperlihatkan komunikasi data antara *Master* dan AT24Cxx dalam proses pengisian atau penulisan data ke AT24C. Gambar 2.28 menggambarkan proses pengisian data ke AT24Cxx dengan metode pengalamatan 8 bit



Gambar 2.28
Proses pengisian data ke AT24CXX dengan metode pengalamatan 8 bit.

Dalam satu proses pengisian data SEEPROM sinyal **START** dan sinyal **STOP** masing-masing cukup dikirim satu kali saja, yakni sinyal **START** dipakai untuk mengawali proses dan sinyal **STOP** dipakai untuk mengakhiri proses. Kedua sinyal itu bukanlah awalan dan akhiran dari pengiriman data 1 byte.

Setelah mengirimkan alamat SEEPROM yang akan diisi, *Master* mengirim data yang diisikan ke AT24Cxx, setiap kali selesai menyimpan data AT24Cxx dengan sendirinya menaikkan alamat SEEPROM yang disimpannya, dengan demikian kiriman data selanjutnya akan disimpan ke memori berikutnya, proses pengisian ini akan berhenti setelah master menutup komunikasi ini dengan sinyal **STOP**. Gambar 2.29 menggambarkan proses pengisian data ke AT24CXX dengan metode pengalamatan 16 bit.



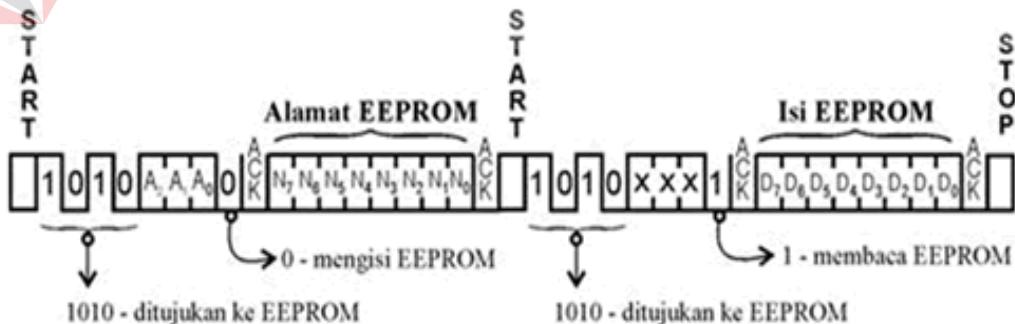
Gambar 2.29

Proses pengisian data ke AT24CXX dengan metode pengalamatan 16 bit.

2.2.5 Proses Membaca Pada Serial EEPROM

Proses pembacaan data dari AT24Cxx dilakukan seperti gambar 2.30

Mula-mula *Master* mengirimkan alamat EEPROM yang akan dibaca isinya, proses ini mirip dengan bagian awal pengisian EEPROM yang dibahas di atas, setelah itu *Master* mengirim sinyal START sekali lagi, disusul dengan perintah untuk membaca isi EEPROM dan selanjutnya disusul dengan pembacaan isi EEPROM yang sesungguhnya.



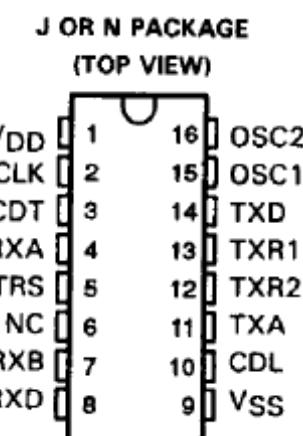
Gambar 2.30 Proses Membaca Pada Serial EEPROM

Selesai membaca isi EEPROM *Master* menutup komunikasi dengan mengirimkan sinyal STOP.

2.6 IC Modem

Modem singkatan dari *MODulator-DEModulator* adalah alat yang digunakan untuk melakukan modulasi dari sinyal digital ke analog, kemudian dikembalikan lagi dari sinyal analog ke digital. Alat ini dibutuhkan karena komputer adalah peralatan digital (hanya mengenal 0 dan 1 misalnya dalam bentuk tegangan 0-5 Volt). Sedang peralatan radio adalah peralatan analog yang hanya mengenal frekuensi suara saja.

IC modem yang digunakan memang dirancang khusus untuk disambungkan dengan peralatan radio komunikasi. IC modem ini disambungkan ke *Mic*, *Speaker* dan *Push-To-Talk* (PTT). Sedangkan modem telepon dirancang khusus untuk komunikasi menggunakan kabel telepon, yang harus mampu memberikan sinyal *dial tone* dan lain-lain. Jadi tidak bisa menggunakan modem telepon pada jaringan radio, karena memang peralatan fisiknya berbeda. Konfigurasi pin-nya dapat dilihat pada gambar 2.31.



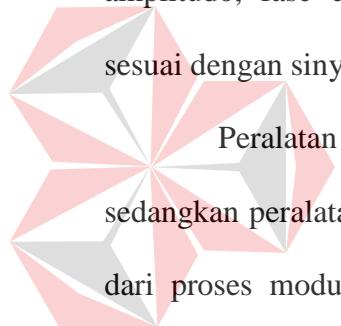
Gambar 2.31 Pin out IC TCM 3105 tipe J atau N

TCM 3105 ini dapat mentransmisikan data hasil modulasi maupun

demodulasi secara *serial* dengan komunikasi secara *half duplex* maupun *full duplex* dengan *baudrate* sampai 1200 bps. (Sumber : [Texas instruments, 1990, Dallas](#))

2.7 Modulasi *Frequency shift keying* (FSK)

Modulasi adalah proses perubahan suatu gelombang periodik sehingga menjadikan suatu sinyal mampu membawa suatu informasi. Dengan proses modulasi, suatu informasi (biasanya berfrekuensi rendah) bisa dimasukkan ke dalam suatu gelombang pembawa, biasanya berupa gelombang sinus berfrekuensi tinggi. Terdapat tiga parameter kunci pada suatu gelombang sinusiodal yaitu : amplitudo, fase dan frekuensi. Ketiga parameter tersebut dapat dimodifikasi sesuai dengan sinyal informasi untuk membentuk sinyal yang termodulasi.



Peralatan untuk melaksanakan proses modulasi disebut modulator, sedangkan peralatan untuk memperoleh informasi-informasi awal (kebalikan dari dari proses modulasi) disebut demodulator dan peralatan yang melaksanakan kedua proses tersebut disebut modem.

Informasi yang dikirim bisa berupa data analog maupun digital sehingga terdapat dua jenis modulasi, yaitu :

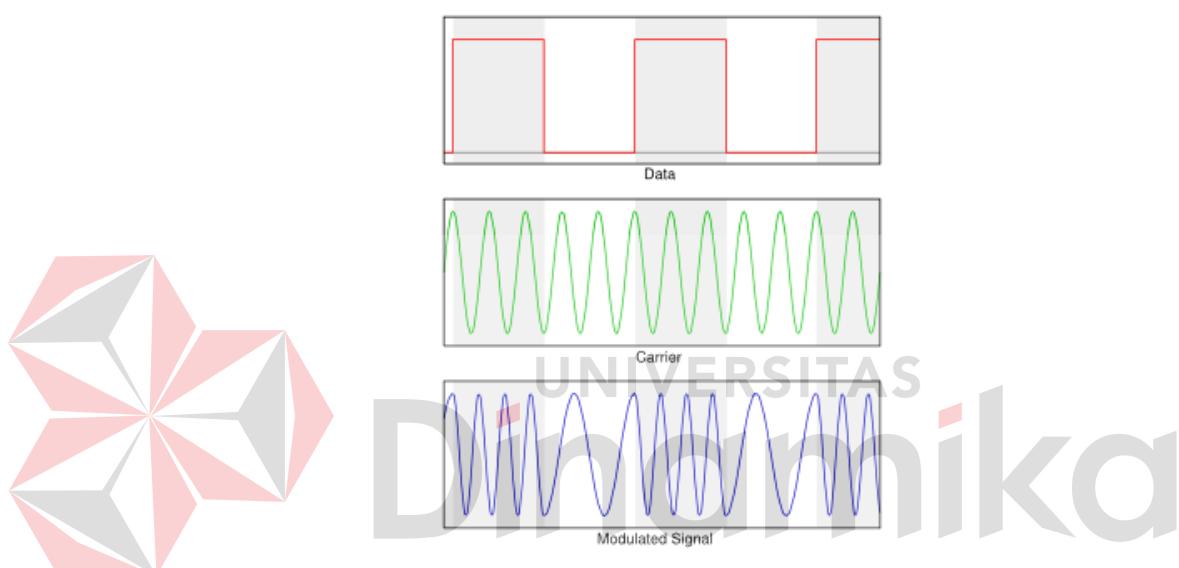
a. Modulasi analog :

teknik modulasi dimana informasi yang dikirimkan berupa *analog low pass signal* seperti sinyal audio, sinyal siaran televisi melalui pemancar sinyal analog.

b. Modulasi digital :

teknik modulasi dimana informasi yang dikirimkan berupa data digital melalui pemancar sinyal analog.

Pada sub bab ini hanya akan dibahas mengenai modulasi digital FSK saja. Modulasi FSK adalah salah satu teknik modulasi digital yang merepresentasikan *bit-bit* digital kedalam *output* frekuensi yang digeser dan ditumpangkan pada frekuensi *carrier*. Bit-bit digital diwujudkan dalam bentuk *mark* frekuensi untuk *logic '1'* dan *space* frekuensi untuk *logic '0'* (Sumber : http://en.wikipedia.org/wiki/Frequency-shift_keying, 2011).



Gambar 2.32 Modulasi data digital dengan metode FSK

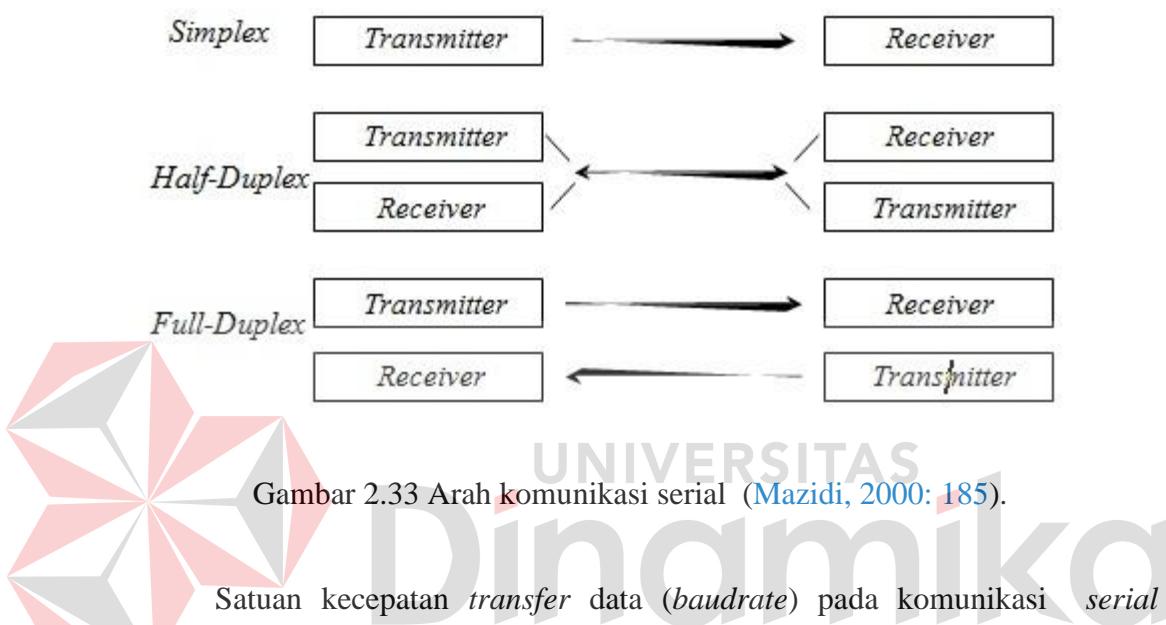
2.8 Komunikasi Serial

Transmisi data secara *serial* adalah transmisi data dimana data tersebut akan dikirimkan tiap *bit* dalam satu waktu (Mazidi, 2000: 184). Terdapat dua cara dalam mentransmisikan data secara *serial*, yaitu secara *Synchronous* dan *Asynchronous*. Dari kedua cara tersebut yang membedakan adalah sinyal *clock* yang dipakai untuk sinkronisasi dalam mengirim data.

Transmisi secara *synchronous* yaitu pengiriman data *serial* bersamaan dengan sinyal *clock*, sedangkan *asynchronous* yaitu pengiriman data

serial tidak bersamaan dengan sinyal *clock* sehingga *receiver* harus membangkitkan sinyal *clock* sendiri (tidak perlu sinkronisasi) (Nalwan, 2003).

Berdasarkan arah proses komunikasi *serial* terdapat 3 metode, yaitu *Simplex*, *Half-Duplex* dan *Full-Duplex* untuk lebih jelasnya dapat dilihat pada gambar 2.33.



Gambar 2.33 Arah komunikasi serial (Mazidi, 2000: 185).

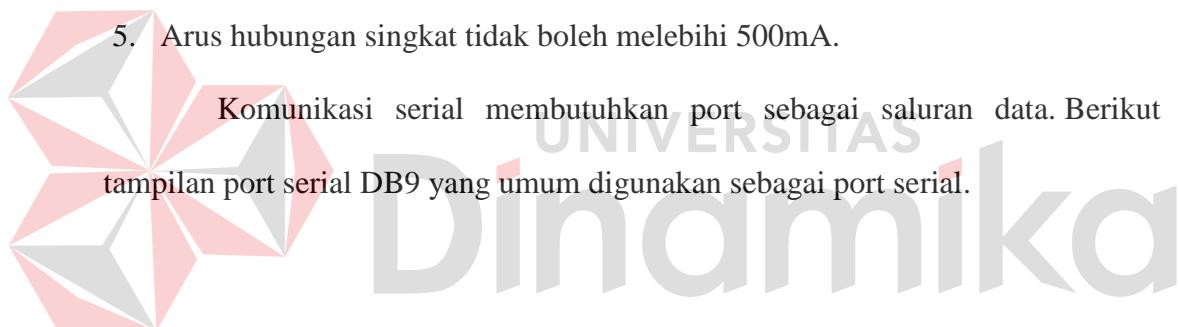
Satuan kecepatan *transfer* data (*baudrate*) pada komunikasi *serial* adalah *bit per second* (bps). Untuk menjaga kompatibilitas dari beberapa peralatan komunikasi data yang dibuat oleh beberapa pabrik, pada tahun 1960 *Electronics Industries Association* (EIA) menstandarkan antarmuka *serial* dengan nama RS232.

Output yang dihasilkan oleh RS232 tidak sesuai dengan *output* TTL yang sudah ada. Dalam RS232, logika 1 direpresentasikan dengan tegangan -3 sampai dengan -25 volt sedangkan logika 0 direpresentasikan dengan tegangan +3 sampai dengan +25 volt. Hasil tak terdefinisi jika berada diantara tegangan -3 sampai dengan +3 volt.

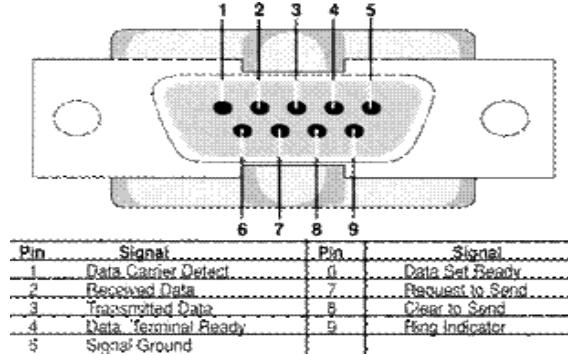
2.8.1 Peralatan Komunikasi Serial

Device pada komunikasi serial port dibagi menjadi 2 (dua) kelompok yaitu *Data Communication Equipment (DCE)* dan *Data Terminal Equipment (DTE)*. Contoh dari DCE ialah modem, plotter, scanner dan lain-lain sedangkan contoh dari DTE ialah terminal di komputer. Spesifikasi elektronik dari serial port merujuk pada *Electronic Industry Association (EIA)* :

1. “Space” (logika 0) ialah tegangan antara + 3 hingga +25 V.
2. “Mark” (logika 1) ialah tegangan antara –3 hingga –25 V.
3. Daerah antara + 3V hingga –3V tidak didefinisikan /tidak terpakai
4. Tegangan open circuit tidak boleh melebihi 25 V.
5. Arus hubungan singkat tidak boleh melebihi 500mA.



Gambar 2.34 Port DB9 jantan



Gambar 2.35 Port DB9 betina

Konektor port serial terdiri dari dua jenis, yaitu konektor 25 pin (DB25) dan 9 pin (DB9) yang berpasangan (jantan dan betina). Bentuk dari konektor DB-25 sama persis dengan port paralel. Umumnya COM1 berada di alamat 3F8H, sedangkan COM2 di alamat 2F8H.

Tabel 2.3 Jenis sinyal RS232

Nama Sinyal	Arah Sinyal	Nomor Kaki Konektor	
		DB9	DB25
Signal Common	-	5	7
Transmitted Data (TD)	Ke DCE	3	2
Received Data (RD)	Dari DCE	2	3
Request to Send (RTS)	Ke DCE	7	4
Clear to Send (CTS)	Dari DCE	8	5
DCE Ready (DSR)	Dari DCE	6	6
DTE Ready (DTR)	Ke DCE	4	20
Ring Indicator (RI)	Dari DCE	9	22
Data Carrier Detect (DCD)	Dari DCE	1	8

2.8.2 Konverter Logika RS-232

Jika peralatan menggunakan logika TTL maka sinyal serial port harus dikonversikan dahulu ke pulsa TTL sebelum kita gunakan, dan sebaliknya sinyal dari peralatan harus dikonversikan ke logika RS-232 sebelum di-inputkan ke

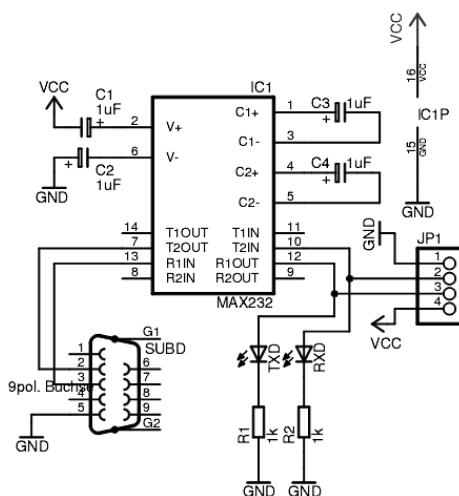
serial port. Konverter yang paling mudah digunakan adalah MAX-232. Di dalam IC ini terdapat *Charge Pump* yang akan membangkitkan +10 Volt dan -10 Volt dari sumber +5 Volt tunggal. Dalam IC DIP (Dual In-line Package) 16 pin (8 pin x 2 baris) ini terdapat 2 buah *transmiter* dan dua *receiver*. Sering juga sebagai *buffer serial* digunakan chip DS275.

Untuk menghubungkan antara 2 buah PC, biasanya digunakan format *null mode*, dimana pin TxD dihubungkan dengan RxD pasangan, pin *Sinyal ground* (5) dihubungkan dengan SG, dan masing-masing pin DTR, DSR dan CD dihubung singkat, dan pin RTS dan CTS dihubung singkat di setiap *device*.

IC MAX232 adalah komponen untuk mengubah sinyal dari RS232 ke

sinyal TTL yang bisa diolah oleh mikrokontroler.

IC ini berguna untuk membuat komunikasi data antara komputer (atau alat lain yang menggunakan RS232) dengan mikrokontroler. Dapat dilihat pada gambar 2.36 :



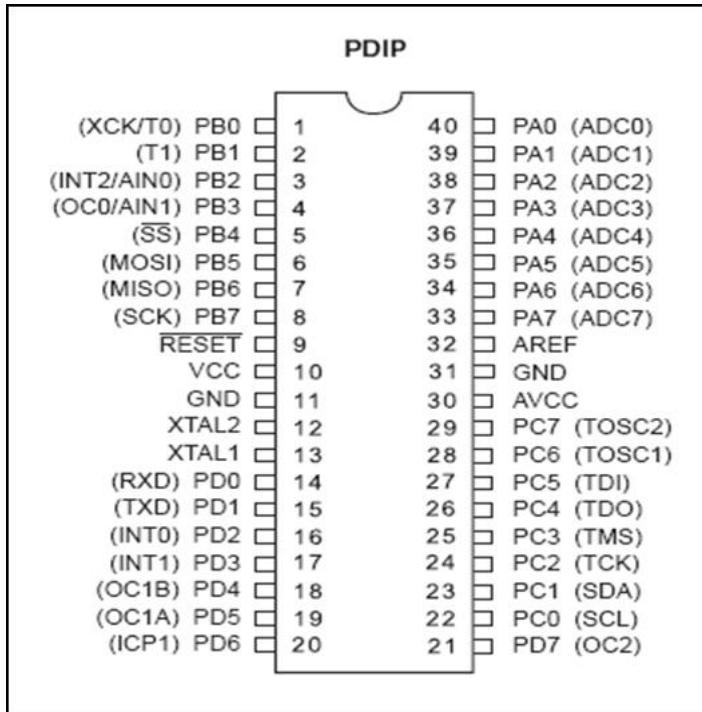
Gambar 2.36 Rangkaian MAX 232

2.9 Mikrokontroler

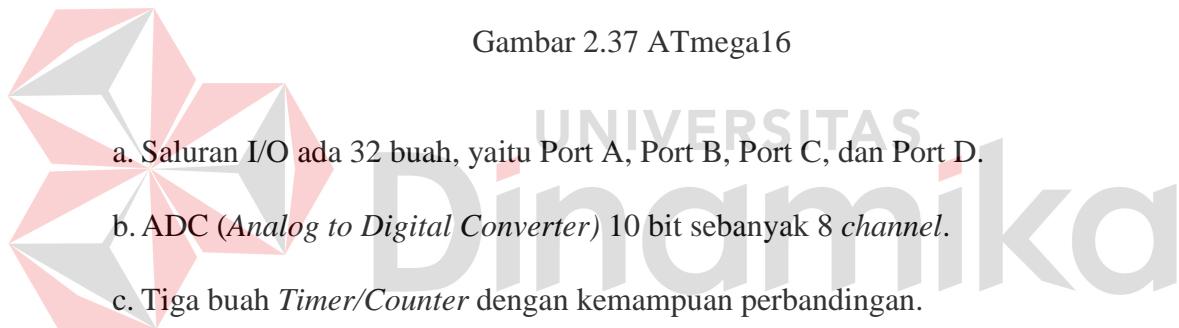
Mikrokontroler adalah suatu mikroprosesor plus. Mikrokontroler adalah otak dari suatu sistem elektronika seperti halnya mikroprosesor sebagai otak komputer. Nilai plus bagi mikrokontroler adalah terdapatnya memori dan port Input/Output dalam suatu kemasan IC yang kompak. Kemampuannya yang programmable, fitur yang lengkap (ADC internal, EEPROM internal, Port I/O, komunikasi serial, dan lain-lain), dan juga harga yang terjangkau yang memungkinkan mikrokontroler digunakan pada berbagai sistem elektronis, seperti pada robot, otomatisasi industri, sistem alarm, peralatan telekomunikasi, hingga peralatan rumah tangga. Salah satunya jenis mikrokontroler AVR yang terpopuler dan terlengkap fasilitasnya ([Lingga, 2006](#)).



Mikrokontroler AVR memiliki arsitektur RISC 8 bit, dimana semua instruksi dikemas dalam kode 16-bit (16-bit word) dan sebagian besar instruksi dieksekusi dalam 1 siklus clock sedangkan MCS51 membutuhkan 12 siklus clock ini dikarenakan AVR berteknologi RISC (*Reduced Instruction Set Computing*). Di dalam mikrokontroler Atmega16 terdiri dari beberapa *I/O* dapat dilihat pada gambar Gambar 2.37



Gambar 2.37 ATmega16



- a. Saluran I/O ada 32 buah, yaitu Port A, Port B, Port C, dan Port D.
- b. ADC (*Analog to Digital Converter*) 10 bit sebanyak 8 channel.
- c. Tiga buah *Timer/Counter* dengan kemampuan perbandingan.
- d. CPU yang terdiri dari 32 register.
- e. 131 intruksi andal yang umumnya hanya membutuhkan 1 siklus *clock*.
- f. *Watchdog Timer* dengan oscilator internal.
- g. Dua buah *Timer/Counter* 8 bit.
- h. Satu buah *Timer /Counter* 16 bit.
- i. Tegangan operasi 2.7 V - 5.5 V pada Atmega16.
- j. Internal *SRAM* sebesar 1KB.
- k. *Memory Flash* sebesar 16KB dengan kemampuan *Read While Write*.
- l. Unit interupsi internal dan eksternal.
- m. *Port* antarmuka *SPI*.

n. *EEPROM* sebesar 512 byte dapat diprogram saat operasi.

o. Antar muka komparator analog.

p. 4 channel *PWM*.

q. 32x8 general *purpose register*.

r. Hampir mencapai 16 MIPS pada Kristal 16 MHz.

s. *Port USART* programmable untuk komunikasi serial.

Mikrokontroler digunakan sebagai komunikasi antara komputer dengan *Plant*, menggunakan komunikasi serial RS232 sebagai komunikasi antara Mikrokontroler dengan Komputer. ([Wardhana.L ,2006](#)).



Mikrokontroler AVR ATMega8

Arduino Pin Mapping

www.arduino.cc

digital pin 0 (RX)	(RESET) PC6	1	28	PC5 (ADC5/SCL)	analog input 5
digital pin 1 (TX)	(RxD) PD0	2	27	PC4 (ADC4/SDA)	analog input 4
digital pin 2	(TxD) PD1	3	26	PC3 (ADC3)	analog input 3
digital pin 3	(INT0) PD2	4	25	PC2 (ADC2)	analog input 2
digital pin 4	(INT1) PD3	5	24	PC1 (ADC1)	analog input 1
	(XCK/T0) PD4	6	23	PC0 (ADC0)	analog input 0
	VCC	7	22	GND	
	GND	8	21	AREF	
	(XTAL1/TOSC1) PB6	9	20	AVCC	
	(XTAL2/TOSC2) PB7	10	19	PB5 (SCK)	digital pin 13 (LED)
digital pin 5	(T1) PD5	11	18	PB4 (MISO)	digital pin 12
digital pin 6	(AIN0) PD6	12	17	PB3 (MOSI/OC2)	digital pin 11 (PWM)
digital pin 7	(AIN1) PD7	13	16	PB2 (SS/OC1B)	digital pin 10 (PWM)
digital pin 8	(ICP1) PB0	14	15	PB1 (OC1A)	digital pin 9 (PWM)

ATmega8

Gambar 2.38 ATmega8

Di dalam mikrokontroler Atmega8 terdiri dari:

- a. Saluran I/O ada 32 buah, yaitu Port B, Port C, dan Port D.
- b. ADC (*Analog to Digital Converter*) 10 bit sebanyak 8 *channel*.
- c. Tiga buah *Time /Counter* dengan kemampuan perbandingan.
- d. CPU yang terdiri dari 32 register.
- e. 131 intruksi andal yang umumnya hanya membutuhkan 1 siklus *clock*.
- f. *Watchdog Timer* dengan oscilator internal.
- g. Dua buah *Timer/Counter* 8 bit.
- h. Satu buah *Timer/Counter* 16 bit.
- i. Tegangan operasi 2.7 V - 5.5 V pada Atmega8
- j. Internal *SRAM* sebesar 1KB.
- k. *Memory Flash* sebesar 8KB dengan kemampuan *Read While Write*.
- l. Unit interupsi internal dan eksternal.
- m. *Port* antarmuka *SPI*.
- n. *EEPROM* sebesar 512 byte dapat diprogram saat operasi.
- o. Antar muka komparator analog.
- p. 3 channel *PWM*.
- q. 32x8 general purpose register.
- r. Hampir mencapai 16 MIPS pada Kristal 16 MHz.
- s. *Port USART* programmable untuk komunikasi serial.

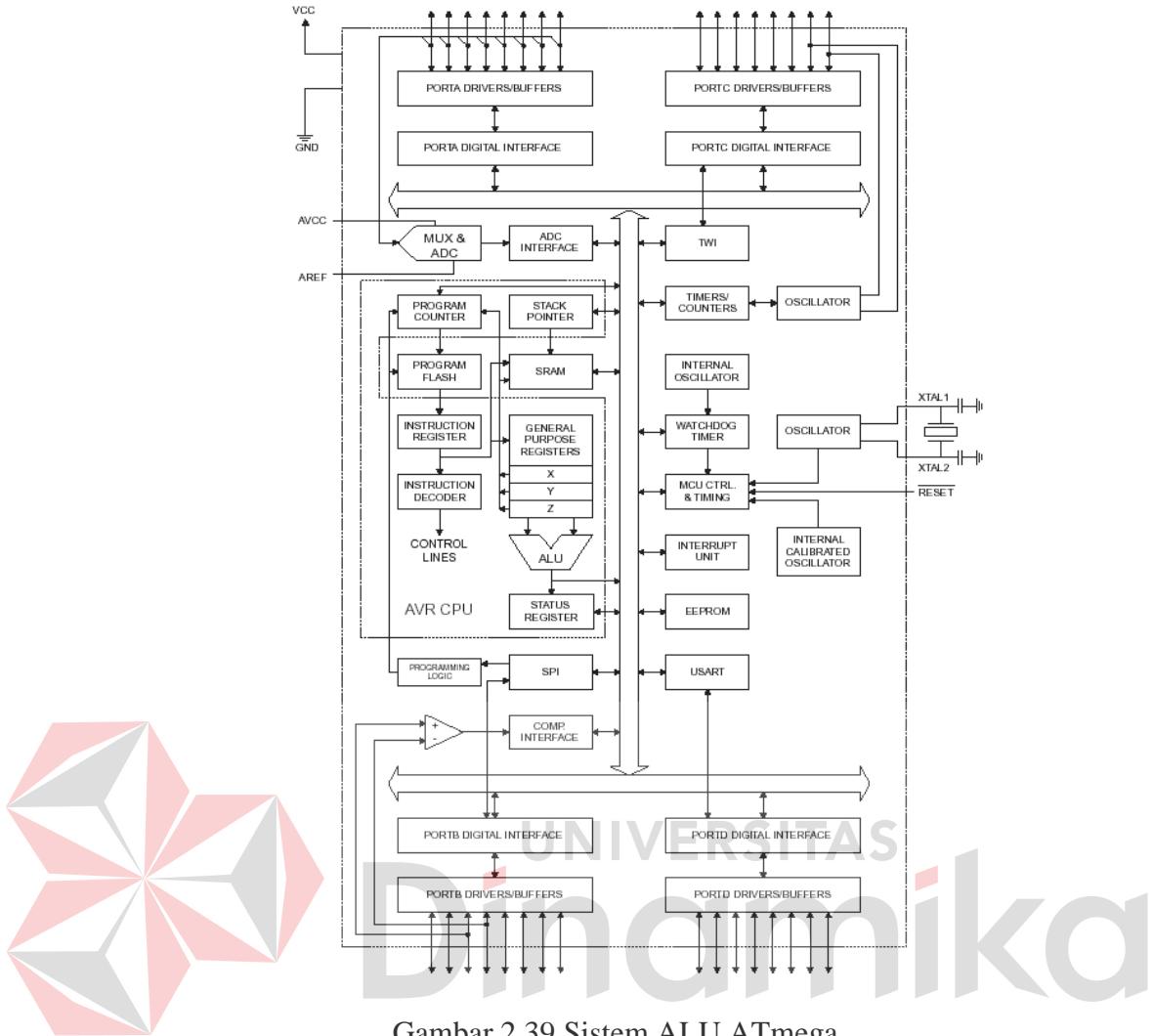
2.9.1 Port sebagai input atau output digital

ATMega16 mempunyai empat buah *port* yang bernama *PortA*, *PortB*, *PortC*, dan *PortD*. Keempat *port* tersebut merupakan jalur *bidirectional* dengan pilihan *internal pull-up*. Tiap *port* mempunyai tiga buah *register bit*, yaitu *DDxn*, *PORTxn*, dan *PINxn*. Huruf ‘x’ mewakili nama huruf dari *port* sedangkan huruf ‘n’ mewakili nomor bit. Bit *DDxn* terdapat pada I/O *address* *DDR_x*, bit *PORTxn* terdapat pada I/O *address* *PORTx*, dan bit *PINxn* terdapat pada I/O *address* *PIN_x*. Bit *DDxn* dalam register *DDR_x* (*Data Direction Register*) menentukan arah pin. Bila *DDxn* diset 1 maka *Px* berfungsi sebagai pin *output*. Bila *DDxn* diset 0 maka *Px* berfungsi sebagai pin *input*. Bila *PORTxn* diset 1 pada saat pin terkonfigurasi sebagai pin *input*, maka resistor *pull-up* akan diaktifkan. Untuk mematikan resistor *pull-up*, *PORTxn* harus diset 0 atau pin dikonfigurasi sebagai pin *output*.

Pin *port* adalah tri-state setelah kondisi reset.

Bila *PORTxn* diset 1 pada saat pin terkonfigurasi sebagai pin *output* maka pin *port* akan berlogika 1. Dan bila *PORTxn* diset 0 pada saat pin terkonfigurasi sebagai pin *output* maka pin *port* akan berlogika 0. Saat mengubah kondisi *port* dari kondisi *tri-state* (*DDxn*=0, *PORTxn*=0) ke kondisi *output high* (*DDxn*=1, *PORTxn*=1) maka harus ada kondisi peralihan apakah itu kondisi *pull-up enabled* (*DDxn*=0, *PORTxn*=1) atau kondisi *output low* (*DDxn*=1, *PORTxn*=0).

Biasanya, kondisi *pull-up enabled* dapat diterima sepenuhnya, selama lingkungan impedansi tinggi tidak memperhatikan perbedaan antara sebuah *strong high driver* dengan sebuah *pull-up*. Untuk lebih jelasnya, dapat dilihat pada gambar 2.39



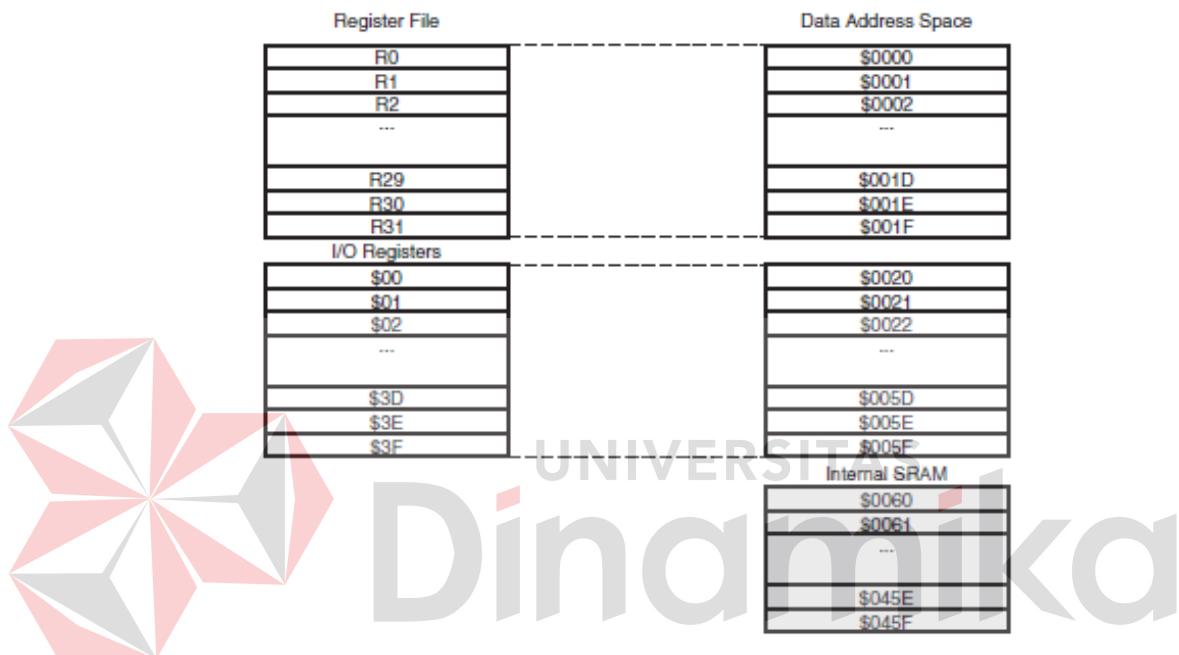
Gambar 2.39 Sistem ALU ATmega

2.9.2 Peta Memori

AVR ATMega16 memiliki ruang pengalamanan memori data dan memori program yang terpisah. Memori data terbagi menjadi 3 bagian, yaitu 32 buah register umum, 64 buah register I/O, dan 1kb SRAM internal.

Register keperluan umum menempati space data pada alamat terbawah, yaitu \$00 sampai \$1F. Sementara itu, register khusus untuk menangani I/O dan kontrol terhadap mikrokontroler menempati 64 alamat berikutnya, yaitu mulai

dari \$20 hingga \$5F. Register tersebut merupakan register yang khusus digunakan untuk mengatur fungsi terhadap berbagai periferal mikrokontroler, seperti kontrol register, *timer* atau *counter*, fungsi – fungsi I/O, dan sebagainya. Alamat memori berikutnya yang digunakan untuk SRAM 1kb, yaitu pada lokasi \$60 sampai dengan \$45F. Konfigurasi memori data ditunjukan pada gambar 2.40



Gambar 2.40 Konfigurasi Memori Data AVR ATmega16

Mikrokontroler digunakan sebagai komunikasi antara komputer dengan *Plant*, menggunakan komunikasi serial RS232 sebagai komunikasi antara Mikrokontroler dengan Komputer. ([Wardhana.L ,2006](#)). Mikrokontroler AVR ATmega8

2.10 Walkie Talkie

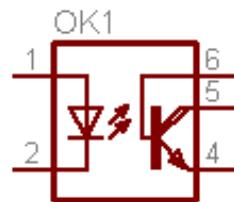
Walkie talkie adalah sebuah alat komunikasi genggam yang dapat mengkomunikasikan dua orang atau lebih dengan menggunakan gelombang radio. Kebanyakan *walkie talkie* digunakan untuk melakukan kedua fungsinya yaitu

berbicara ataupun mendengar. *Walkie Talkie* dikenal dengan sebutan *Two Way Radio* ataupun radio dua arah, yang dapat melakukan pembicaraan dua arah, berbicara dan mendengar lawan bicara secara bergantian. *Walkie talkie* dapat digunakan dalam jarak 0,5 Km sampai dengan 2,5 Km tanpa menggunakan biaya pulsa seperti menelpon. *Walkie talkie* merupakan *transceiver*, yang dikarenakan ia memiliki *two way radio* tersebut, alat ini memiliki radio *transmitter* dan sinyal penerima komunikasi radio.



Gambar 2.41 *Walkie talkie*

2.11 *Opto Coupler*



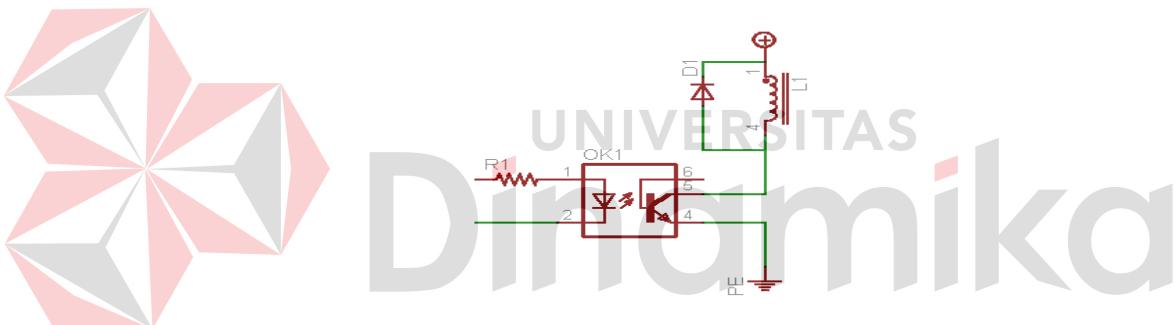
Gambar 2.42 *Opto Coupler*

Opto coupler merupakan komponen yang digunakan sebagai isolator antara rangkaian power dengan rangkaian kontrol, contohnya sebagai pemisah

antara mikrokontroler dengan peralatan yang mempunyai tegangan besar dan dapat mengganggu kinerja mikrokontroler.

Seperti ditunjukkan pada gambar 2.30 *Opto coupler* jenis transistor (*opto transistor*) terdiri dari LED dan *photo transistor* yang memiliki prinsip kerja mirip dengan *transistor* normal cuma yang membedakan ialah cara pemberian sinyal, jika pada *transistor* menggunakan tegangan atau arus sedangkan pada *opto transistor* menggunakan cahaya, karena menggunakan cahaya maka masukan dengan keluaran akan terisolasi oleh cahaya tersebut. Gambar 2.43 merupakan contoh dari penggunaan *opto coupler*.

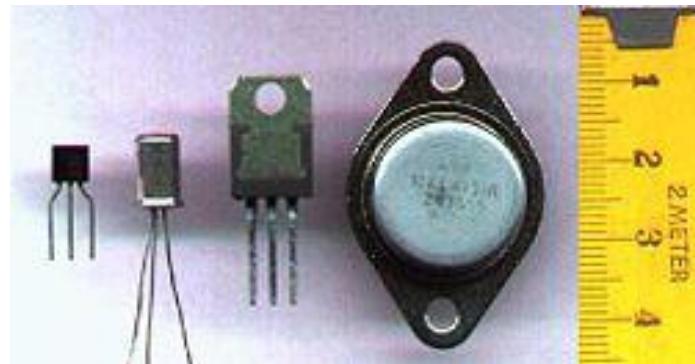
(Sumber : <http://elektronikaelektronika.blogspot.com>)



Gambar 2.43 Contoh penggunaan opto coupler

2.12 Transistor

Transistor adalah alat semikonduktor yang dipakai sebagai penguat, sebagai sirkuit pemutus dan penyambung (*switching*), stabilisasi tegangan, modulasi sinyal atau sebagai fungsi lainnya. Transistor dapat berfungsi semacam kran listrik, dimana berdasarkan arus inputnya (BJT) atau tegangan inputnya (FET), memungkinkan pengaliran listrik yang sangat akurat dari sirkuit sumber listriknya. (Sumber : <http://id.wikipedia.org/wiki/transistor>)

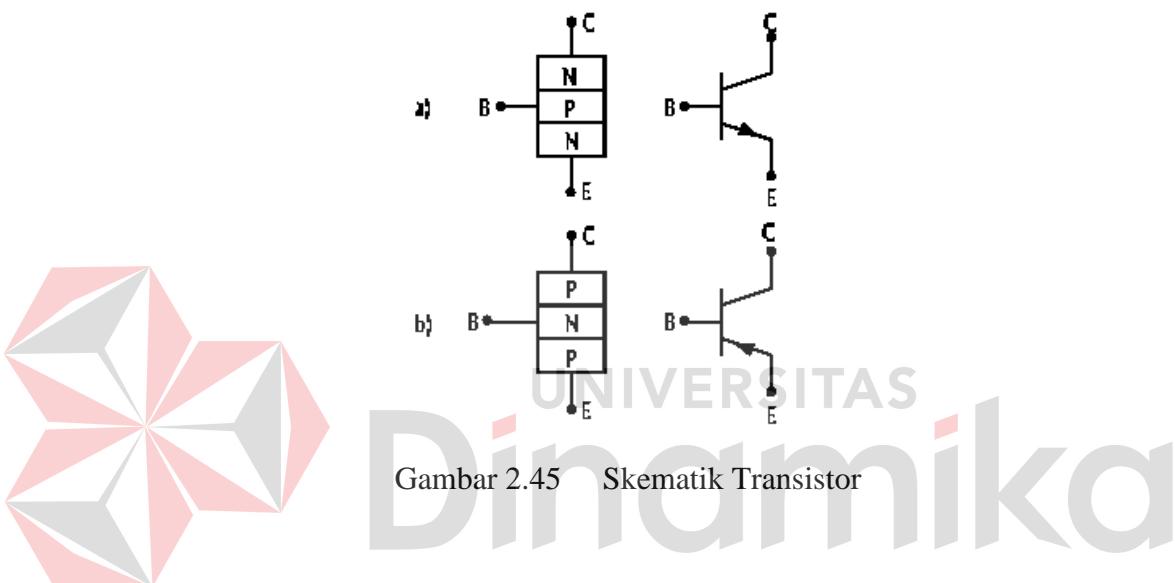


Gambar 2.44 *Transistor through-hole* (dibandingkan dengan pita ukur centimeter)

Pada umumnya, transistor memiliki 3 terminal. Tegangan atau arus yang dipasang di satu terminalnya mengatur arus yang lebih besar yang melalui dua terminal lainnya. Transistor adalah komponen yang sangat penting dalam dunia elektronik modern. Dalam rangkaian analog, transistor digunakan dalam amplifier (penguat). Rangkaian analog melengkapi pengeras suara, sumber listrik stabil, dan Penguat sinyal radio. Dalam rangkaian-rangkaian digital, *transistor* digunakan sebagai saklar berkecepatan tinggi. Beberapa transistor juga dapat dirangkai sedemikian rupa sehingga berfungsi sebagai *logic gate*, memori, dan komponen-komponen lainnya.

Terdapat dua jenis kontruksi dasar BJT, yaitu jenis *n-p-n* dan jenis *p-n-p*. Untuk jenis *n-p-n*, BJT terbuat dari lapisan tipis semikonduktor tipe-*p* dengan tingkat doping yang relatif rendah, yang diapit oleh dua lapisan semikonduktor tipe-*n*. Karena alasan sejarah pembuatannya, bagian di tengah disebut “basis” (*base*), salah satu bagian tipe-*n* (biasanya mempunyai dimensi yang kecil) disebut “emitor” (*emitter*) dan yang lainnya sebagai “*kolektor*” (*collector*). Secara skematik kedua jenis transistor diperlihatkan pada gambar 2.35. Tanda panah pada gambar 2.33 menunjukkan kaki emitor dan titik dari material tipe-*p* ke material tipe-*n*. Perhatikan bahwa untuk jenis *n-p-n*, transistor terdiri dari dua

sambungan *p-n* yang berperilaku seperti diode. Setiap diode dapat diberi panjar maju atau berpanjar mundur, sehingga transistor dapat memiliki empat modus pengoperasian. Salah satu modus yang banyak digunakan disebut “modus normal”, yaitu sambungan emitor-basis berpanjar maju dan sambungan kolektor-basis berpanjar mundur. Modus ini juga sering disebut sebagai pengoperasian transistor pada “daerah aktif”.(Sumber : <http://id.wikipedia.org/wiki/transistor>)



Gambar 2.45 Skematik Transistor

2.13 Bahasa Pemrograman

2.13.1 Bahasa Pemrograman C

Bahasa pemrograman C merupakan salah satu bahasa pemrograman komputer. Dibuat pada tahun 1972 oleh Dennis Ritchie untuk Sistem Operasi Unix di Bell Telephone Laboratories.

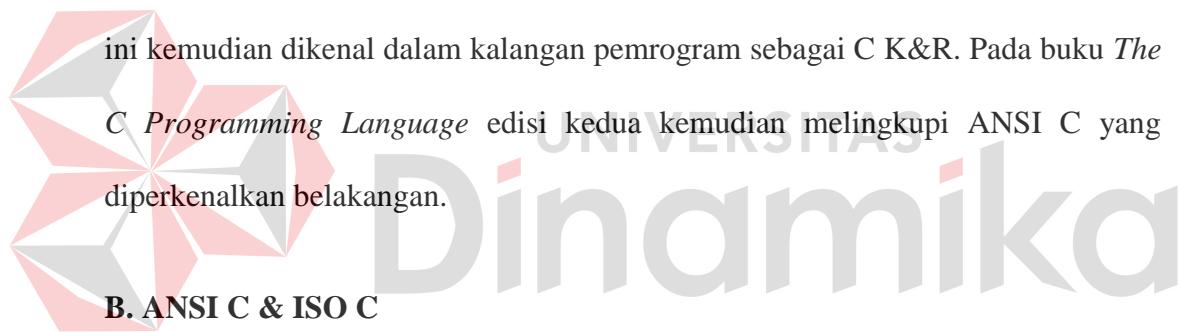
Meskipun C dibuat untuk memprogram sistem dan jaringan komputer namun bahasa ini juga sering digunakan dalam mengembangkan software aplikasi. C juga banyak dipakai oleh berbagai jenis platform sistem operasi dan arsitektur komputer, bahkan terdapat beberapa compiler yang sangat populer telah

tersedia. C secara luar biasa mempengaruhi bahasa populer lainnya, terutama C++ yang merupakan extensi dari C.

Versi Bahasa C

A. C K&R

Pada tahun 1978, *Dennis Ritchie* dan *Brian Kernighan* menerbitkan edisi pertama dari buku yang berjudul *The C Programming Language*. Buku ini hingga sekarang diakui sebagai kitab suci bahasa C dan merupakan referensi utama seorang pemrogram yang ingin mengetahui tentang bahasa C, terutama karena begitu lengkapnya cakupan buku ini tentang bahasa C dan mudahnya program yang dicontohkan dalam buku ini. Versi bahasa C yang ditampilkan dalam buku



B. ANSI C & ISO C

Pada perkembangannya, muncul versi-versi C lain yang pada akhirnya membuat kebingungan di kalangan pemrogram. Karena itu, pada tahun 1983, *American National Standards Institute* (ANSI) membuat sebuah komite untuk membuat sebuah versi standar dari bahasa C. Setelah melalui proses yang panjang dan sengit, pada tahun 1989, telah berhasil disahkan standar yang dinamakan ANSI X3.159-1989, versi ini seringkali dinamakan ANSI C, atau kadang-kadang C89.

Pada 1990, versi ANSI C diadopsi oleh *Organization for Standardization* (ISO) dengan sedikit perubahan dengan nama ISO/IEC 9899:1990. Versi ini seringkali dinamakan ISO C atau C90. Karena versi ANSI C dan ISO C hanya

memiliki sedikit perbedaan, pemanggilan C90 dan C89 merujuk pada bahasa yang sama.

C. C99

Versi C99 dibuat oleh ISO C pada tahun 1999. Versi ini dimaksudkan terutama untuk memperbanyak dukungan kepada pemrograman berorientasi objek, terutama setelah C++, yang dibuat berdasarkan bahasa ini mendapat tempat yang istimewa di kalangan pemrogram.

2.13.2. Bahasa Pemrograman Visual Basic

VB 1.0 dikenalkan pada tahun 1991, pendekatan yg dilakukan untuk menghubungkan bahasa pemrograman dengan GUI berasal dari prototype yg dikembang oleh “Alan Cooper” yg di sebut TRIPOD, Kemudian Microsoft mengontrak copper dan asosiasinya utk mengembangkan tripod agar dapat digunakan di windows 3.0 dibawah nama kode Ruby.Berikut Perjalanan Visual Basic (VB 1.0 Sampai VB 10) :Perjalanan dari Visual Basic (VB1 to VB 10):

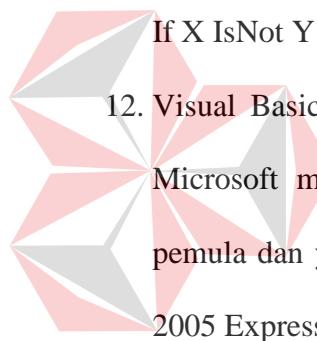
1. Proyek “Thunder” dirintis
2. Visual Basic 1.0 (May 1991) di rilis untuk windows pada [COMDEX](#)/Windows Wordltrade yg dipertunjukan di Atlanta , Georgia
3. Visual Basic 1.0 untuk DOS dirilis pada bulan September 1992. Bahasa ini tidak kompatibel dengan Visual Basic For Windows. VB 1.0 for DOS ini pada kenyataanya merupakan versi kelanjutan dari compiler BASIC, QuickBasic dan BASIC Professional Development System.
4. Visual Basic 2.0 dirilis pada November 1992, Cakupan pemrogramannya cukup mudah untuk digunakan dan kecepatannya juga telah di modifikasi.

Khususnya pada Form yg menjadikan object dapat dibuat secara seketika, serta konsep dasar dari Class modul yg berikutnya di implementasikan pada VB 4

5. Visual Basic 3.0 , dirilis pada musim panas 1993 dan dibagi menjadi versi standard dan professional. VB 3 memasukan Versi 1.1 dari Microsoft Jet Database Engine yg dapat membaca serta menulis database Jet (atau [access](#)) 1.x
6. Visual Basic 4.0 (Agustus 1995) merupakan versi pertama yg dapat membuat windows program 32 bit sebaik versi 16 bit nya. VB 4 juga memperkenalkan kemampuan untuk menulis non-GUI class pada Visual Basic
7. Visual Basic 5.0 (February 1997), Microsoft merilis secara eksklusif Visual basic untuk versi windows 32 bit . Programmer yg menulis programnya pada versi 16 bit dapat dengan mudah melakukan import programnya dari VB4 ke VB5. dan juga sebaliknya, program VB5 dapat diimport menjadi VB4. VB 5 memperkenalkan kemampuan untuk membuat User Control.
8. Visual Basic 6.0 (pertengahan 1998) memperbaiki beberapa cakupan, termasuk kemampuannya untuk membuat Aplikasi Web-based . Visual Basic 6 dijadwalkan akan memasuki Microsoft “fase non Supported” dimulai pada maret 2008
9. Visual Basic .NET, dirilis pada tahun 2002, Beberapa yang mencoba pada versi pertama .NET ini mengemukakan bahwa bahasa ini sangat powerful tapi bahasa yg digunakan sangat berbeda dengan bahasa sebelumnya, dengan kekurangan diberbagai area, termasuk runtime-nya yang 10 kali lebih besar dari paket runtime VB6 serta peningkatan penggunaan memory.

10. Visual Basic .NET 2003, dirilis menggunakan NET framework versi 1.1

11. Visual Basic 2005, merupakan integrasi selanjutnya dari Visual Basic .NET. dan Microsoft memutuskan untuk menghilangkan kata kata .NET pada judulnya. Pada Rilis ini , Microsoft memasukan beberapa fitur baru, diantaranya : Edit and Continue , mungkin inilah kekurangan fitur terbesar dari VB .NET. pada VB 2005 ini kita diperbolehkan melakukan perubahan kode pada saat program sedang dijalankan Perbaikan pada Konversi dari VB ke VB .NET12 Visual Basic .NET 2003 (VB 7.1) , dirilis dengan menggunakan NET Kerangka kerja versi 1.1. IsNot Patent, merupakan salah satu fitur dari Visual Basic 2005 merupakan konversi If=Not X Is Y menjadi

- 
- If X IsNot Y
12. Visual Basic 2005 Express , merupakan bagian dari Product Visual Studio. Microsoft membuat Microsoft Visual Studio 2005 Express edition untuk pemula dan yg gemar dengan VB, salah satu produknya adalah Visual Basic 2005 Express yg merupakan produk gratis dari Microsoft
13. Visual Basic “Orcas”, dijadwalkan akan dirilis pada tahun 2007 dan dibangun diatas .NET 3.5. Pada rilis ini , Microsoft menambahkan beberapa fitur , diantaranya : True Ternary operator , yaitu fungsi If(boolean,value, value) yg digunakan untuk menggantikan fungsi IIF, LINQ Support, Ekspresi Lambda, XML Literals, Nullable types, Type Inference
14. Visual Basic ‘VBx’, Visual Basic 10, yang juga dikenal dengan nama VBx, akan menawarkan dukungan untuk Dynamic Language Runtime. VB 10 direncanakan akan menjadi bagian dari SilverLight 1.1

(Sumber : http://id.wikipedia.org/wiki/bahasa_pemrograman)



BAB III

METODE PENELITIAN

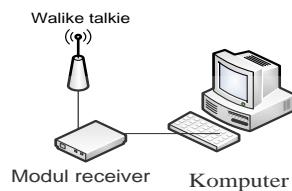
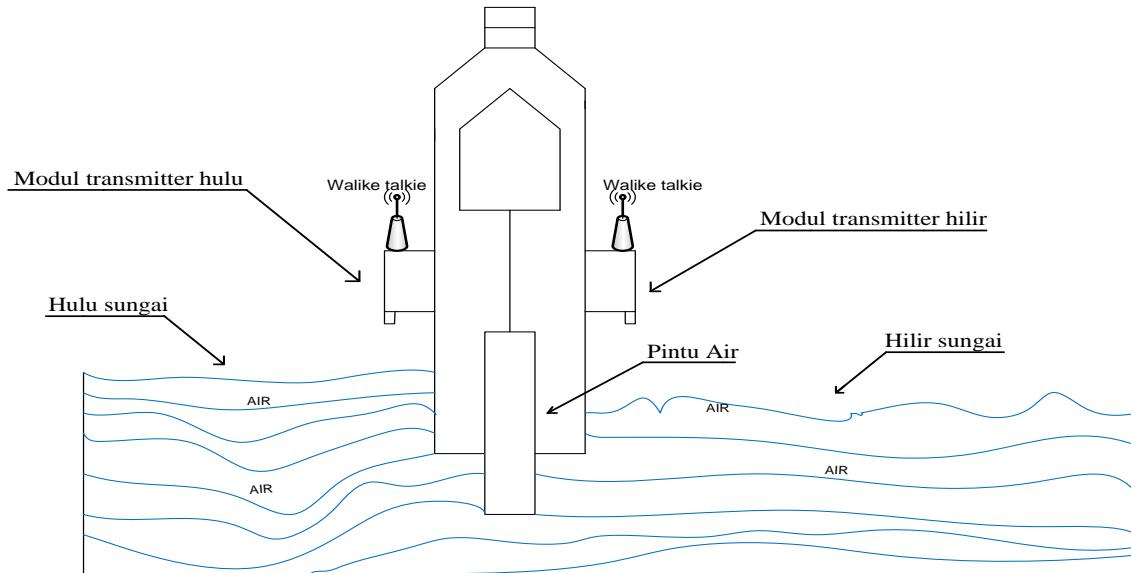
Metode penelitian yang digunakan pada pembuatan perangkat keras dan perangkat lunak yaitu dengan studi kepustakaan. Dengan cara ini penulis berusaha untuk mendapatkan dan mengumpulkan data-data, informasi, konsep-konsep yang bersifat teoritis dari buku bahan-bahan kuliah dan referensi dari internet yang berkaitan dengan permasalahan.

Dari data-data yang diperoleh maka dilakukan perencanaan rangkaian perangkat keras. Dalam perangkat keras ini, penulis akan melakukan pengujian perangkat keras dengan program-program yang telah dibuat, Pembuatan perangkat lunak adalah tahap selanjutnya. Terakhir adalah penggabungan perangkat keras dengan kerja perangkat lunak yang telah selesai dibuat.

Pada bab ini akan dibahas mengenai masalah yang timbul dalam perencanaan dan pembuatan perangkat keras (*hardware*) maupun perangkat lunak (*software*). Dari kedua bagian tersebut akan dipadukan agar dapat bekerja sama untuk menjalankan sistem yang baik.

Perencanaan ini diperlukan sebelum proses pembuatan sistem tersebut, Perancangan ini berguna agar pelaksanaan tahapan selanjutnya berjalan dengan lancar. Tahapan-tahapannya meliputi tahap pembuatan perangkat keras, perangkat lunak dan penggabungannya.

Dalam perancangan sistem alat pendekripsi banjir pada sungai dengan metode fuzzy menggunakan jaringan sensor nirkabel, penulis menggunakan diagram blok seperti gambar 3.1.



Gambar 3.1 Blok diagram alat yang terpasang pada pintu air



Dari blok diagram pada gambar 3.1 kita dapat melihat sistem pendeksi banjir pada sungai menggunakan metode fuzzy dengan jaringan sesnor nirkabel. Dapat kita lihat bagian input yaitu sensor ultrasonik (sensor untuk mengukur jarak elavasi sungai), Bagian pemroses yaitu mikrokontroler ATmega16 di bagian transmitter dan ATmega8 di bagian *reciever* dan output berupa data serial yang akan di teruskan pada komputer melalui kabel serial yang selanjutnya akan di olah lagi untuk menghasilkan data yang akurat untuk pemantauan banjir. Berikut ini adalah penggunaan dari bagian-bagian tersebut :

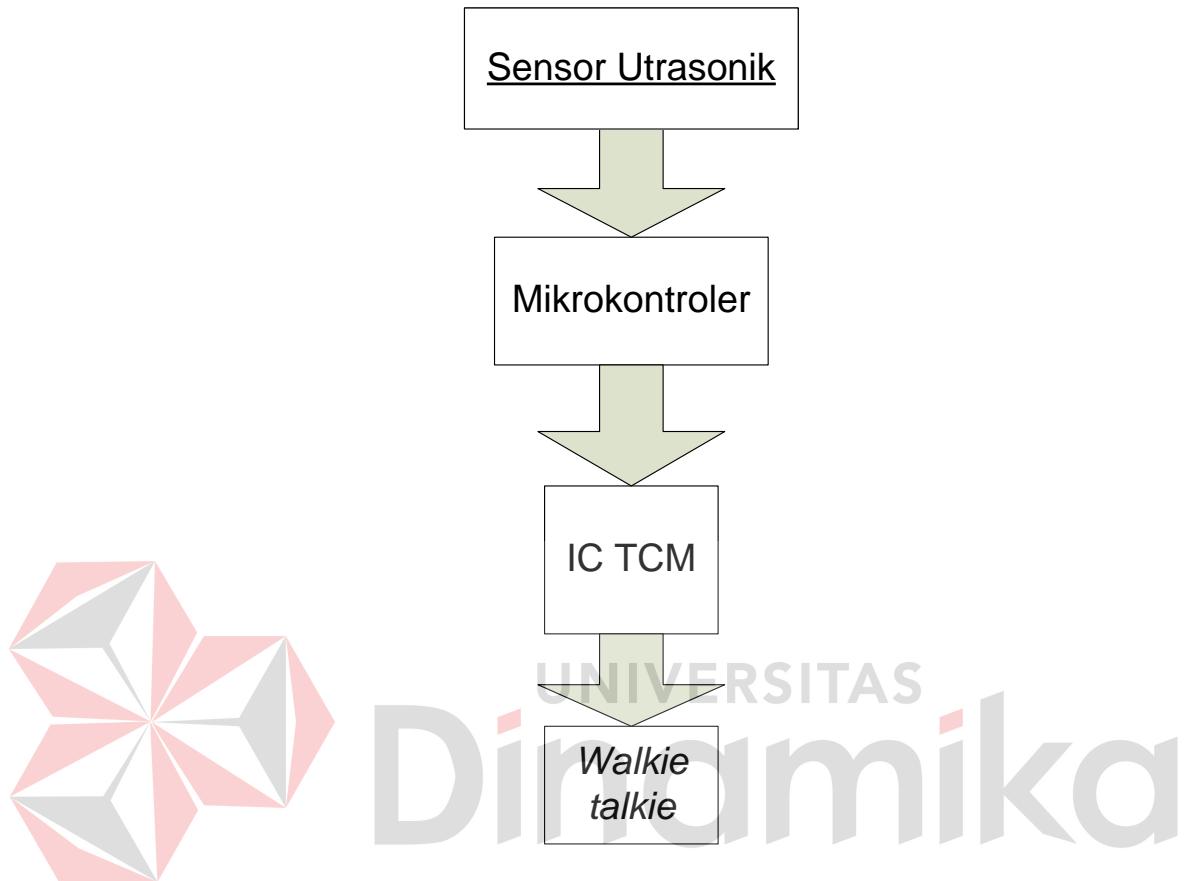
1. Masukan Sensor Ultrasonik Tipe SRF02
 - a. Sensor ultrasonik berfungsi untuk mengetahui jarak elavasi pada sungai di daerah hilir maupun hulu.
2. Pemroses Mikrokontroler ATmega16 dan ATmega8
 - a. ATmega16 di gunakan pada alat pengukur elavasi sungai yang di hilir maupun di hulu dan mengirikan pembacaan data sensor ultrasonik ke IC TCM yang kemudian di teruskan ke *walkie talkie* untuk dikirim kesisi alat penerima
 - b. ATmega8 digunakan untuk switching saluran pada *walkie talkie* agar dapat berpindah-pindah agar data dari sisi pengirim hulu dan hilir bisa di terima ke komputer.
3. Keluaran
 - a. Komputer berfungsi sebagai pengolah data yang didapatkan dari alat pengirim data dari sisi hulu maupun hilir dan akhirnya diterima di sisi alat penerima yang akhirnya diteruskan ke komputer. Setelah data valid dan diterima oleh komputer maka data akan diproses sedemikian rupa sesuai aturan-aturan yang ada untuk mendeteksi banjir sehingga data yang telah di proses dapat dioutputkan di layar monitor sehingga dapat untuk monitoring sungai dan mengetahui jika akan terjadi banjir.

3.1 Perancangan Perangkat Keras

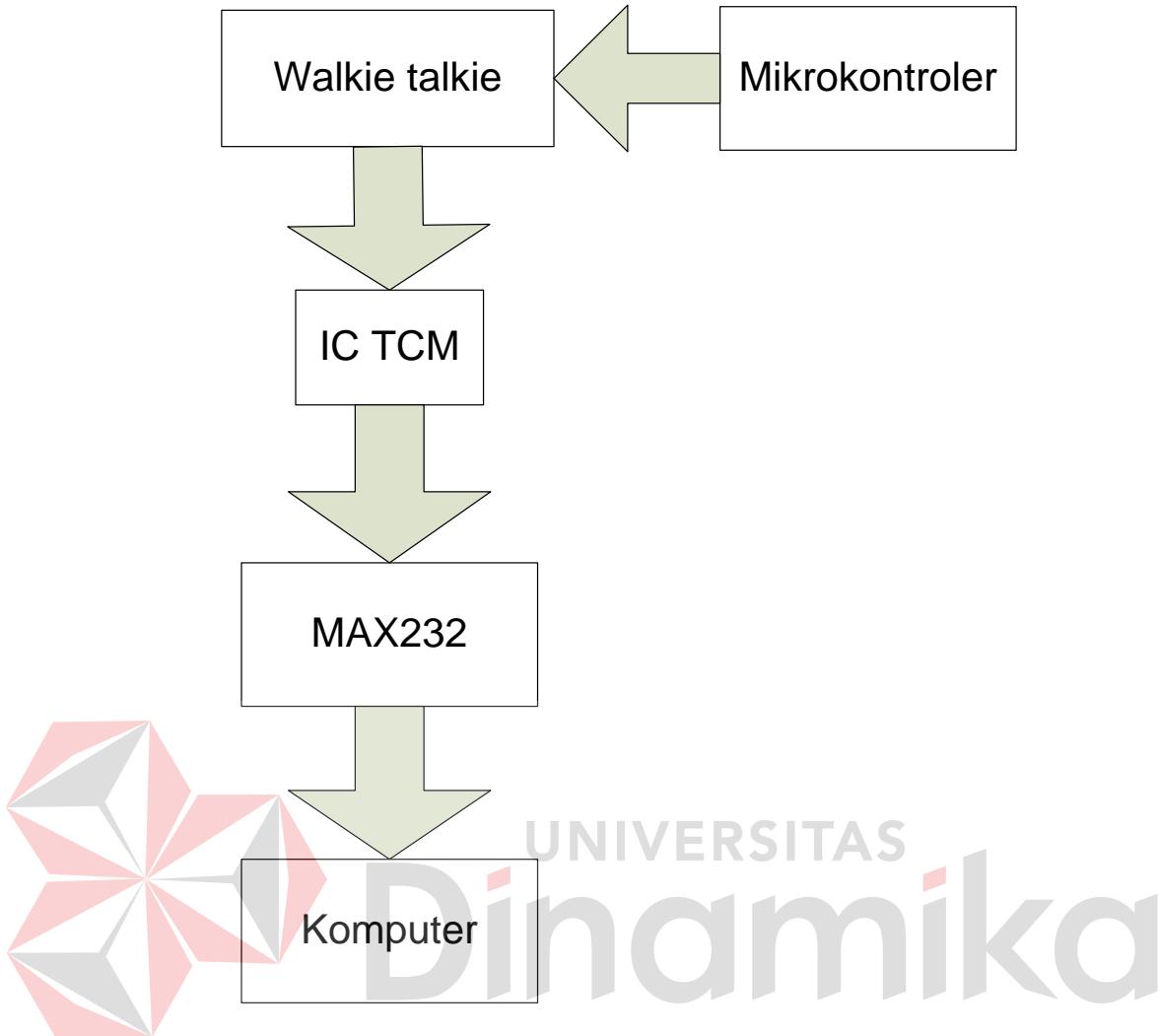
Perancangan perangkat keras minimum sistem untuk *transmitter* dan minimum sistem untuk *reciever*.

Funngsi minimum sistem *trasnmitter* adalah sebagai pembaca data sensor ultrasonik dan mengirimkan datanya ke *reciever*, dan minimum sistem

receiver berfungsi untuk menerima data kiriman data sensor ultrasonik dari minimum sistem *transmitter*.



Gambar 3.2 Blok diagram minimum sistem *transmitter*



Gambar 3.3 Blok diagram minimum sistem *reciever*

3.1.1 Minimum Sistem

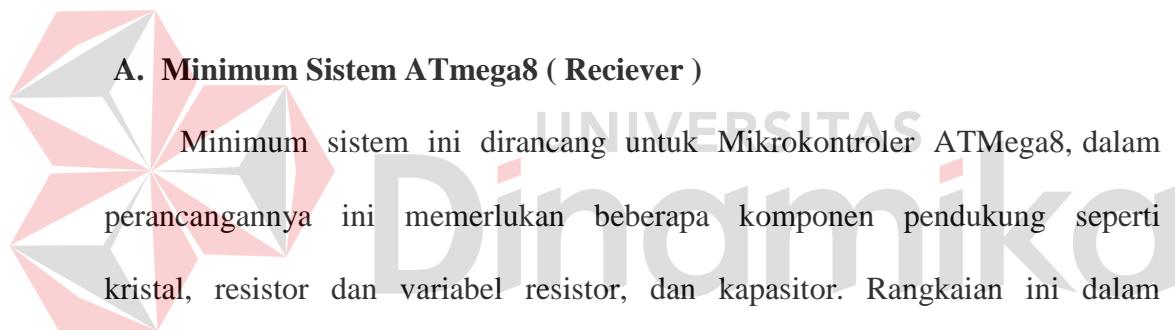
Rangkaian minimum sistem dibuat untuk mendukung kerja dari microchip ATmega dimana microchip tidak bisa berdiri sendiri alias harus ada rangakaian da komponen pendukung seperti halnya rangakaian catu daya, kristal dan lain sebagainya yang biasanya di sebut minimum sistem.

Microchip berfungsi sebagai otak dalam mengolah semua instruksi baik input maupun output seperti halnya proses pembacaan sensor ultrasonik sampai

pengiriman data secara serial ke IC Modem TCM maupun pengiriman data serial ke komputer melalui RS232.

Pada TA ini, penulis membuat dua buah minimum sistem yang berbeda karena ada perbedaan dalam fungsinya jadi desain minimum sistem pada alat yang dipasang sebagai *transmitter* berbeda dengan alat yang dipasang sebagai *reciever*. Untuk alat di transmitter menggunakan ATmega16 dan untuk alat di receiver menggunakan ATmega8.

Penggunaan seri microchip yang berbeda karena menyesuaikan penggunaan PORT sesuai kebutuhan jadi lebih efisien dalam perancangan maupun biaya.

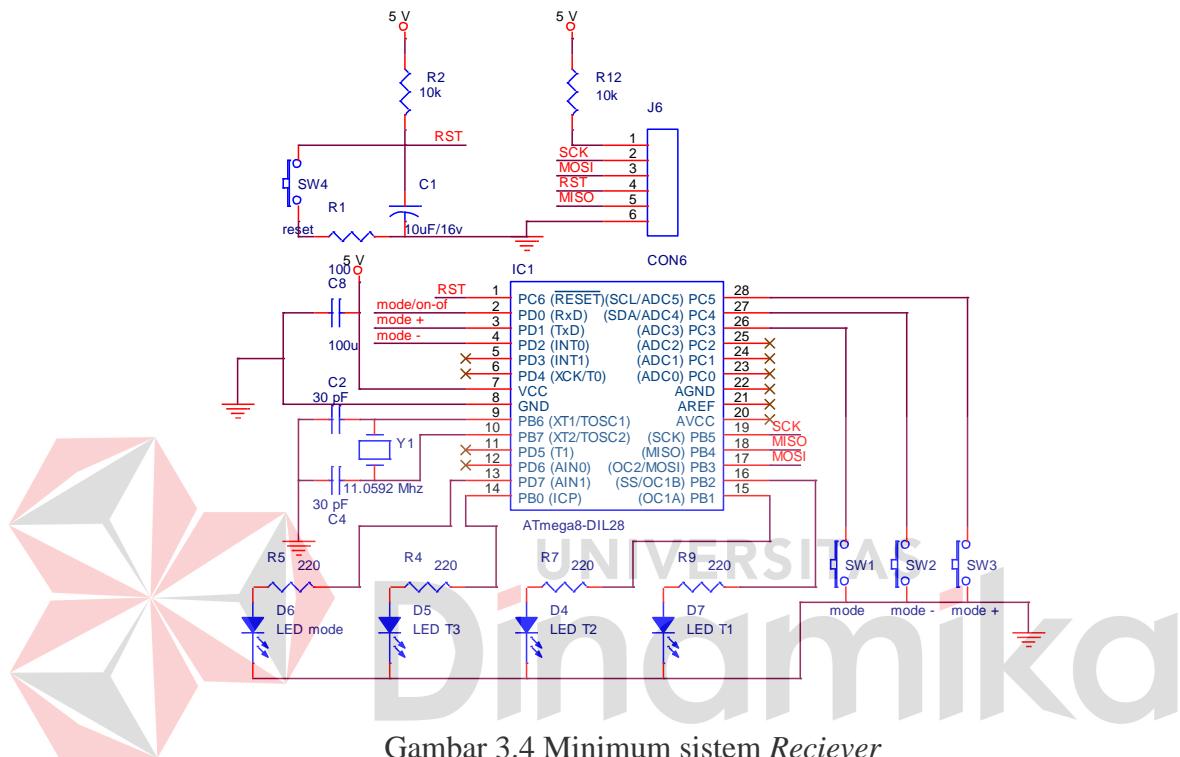


Minimum sistem ini dirancang untuk Mikrokontroler ATMega8, dalam perancangannya ini memerlukan beberapa komponen pendukung seperti kristal, resistor dan variabel resistor, dan kapasitor. Rangkaian ini dalam istilah lainnya disebut *Minimum system* ATMega8. Mikrokontroler berfungsi untuk mengatur perpindahan saluran frekuensi pada walkie talkie sehingga saluran *walkie talkie* dapat berpindah-pindah sesuai dengan keinginan secara otomatis. dalam rangkaian dibawah juga terdapat rangkaian demodulasi sinyal yang digabung menjadi satu supaya lebih praktis dan efisien .

- a. PortD.0 - PortD.2 digunakan sebagai output untuk mengubah-ubah saluran walkie talkie agar data dari sisi pengirim hulu dan sisi pengirim hilir bisa di terima secara bergantian
- b. PortB.0 - PortB.2 dan PortD7 digunakan sebagai output untuk Led sebagai indikator apakah system switching berjalan atau tidak.

- c. PortC.3 – PortC.5 digunakan sebagai input tombol mode untuk mengaktifkan sistem

Gambar *minimum system* ATMega8 dan rangkaian demodulasi sinyal dapat dilihat pada Gambar 3.4.



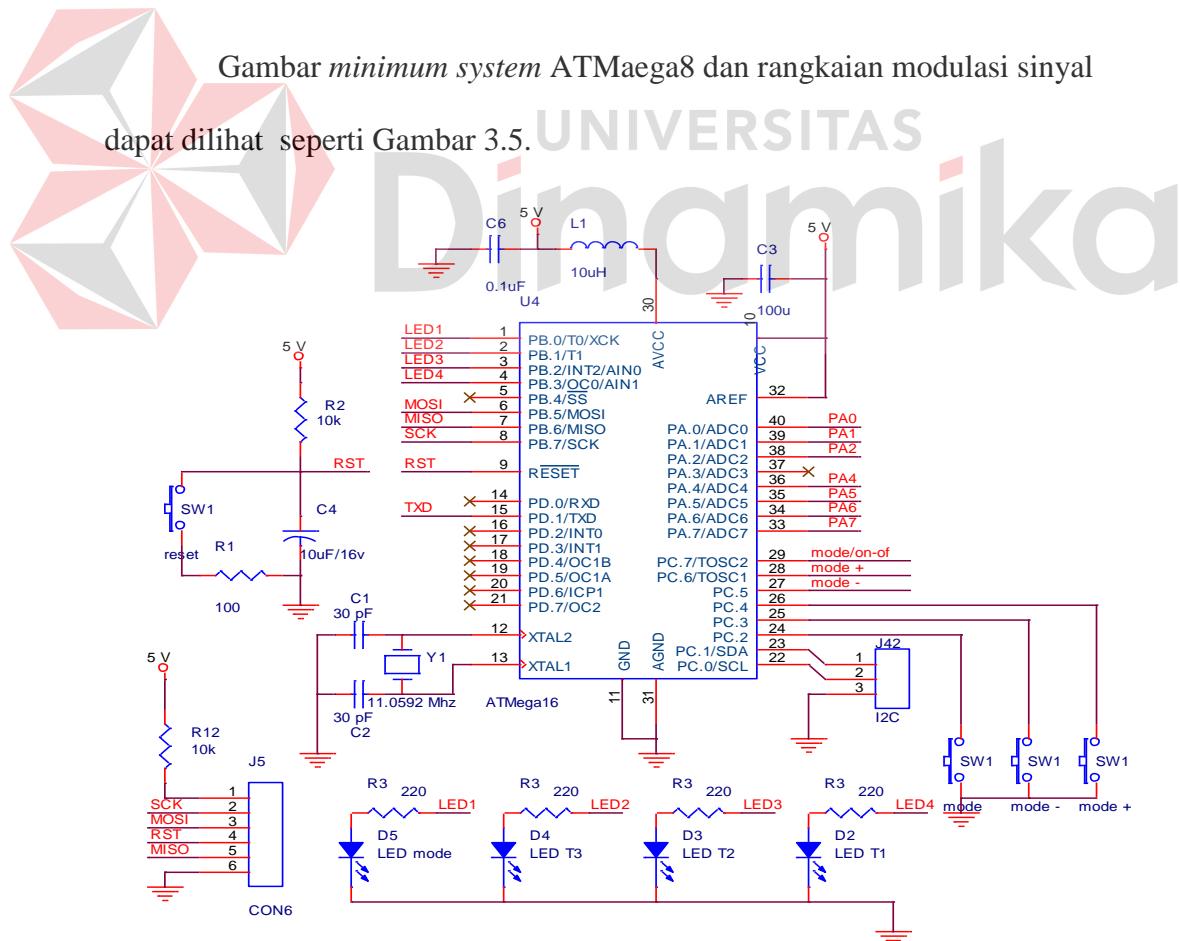
Gambar 3.4 Minimum sistem Reciever

B. Minimum Sistem ATmega16 (Transmitter)

Minimum sistem ini dirancang untuk Mikrokontroler ATMega16, dalam perancangannya ini memerlukan beberapa komponen pendukung seperti kristal, resistor dan variabel resistor, dan kapasitor. Rangkaian ini dalam istilah lainnya disebut *Minimum system* ATMega16. Mikrokontroler berfungsi untuk mengatur perpindahan saluran frekuensi pada *walkie talkie* sehingga saluran *walkie talkie* dapat berpindah-pindah sesuai dengan keinginan secara otomatis. dalam rangkaian dibawah juga terdapat rangkaian modulasi sinyal yang digabung menjadi satu supaya lebih praktis dan efesien. Untuk

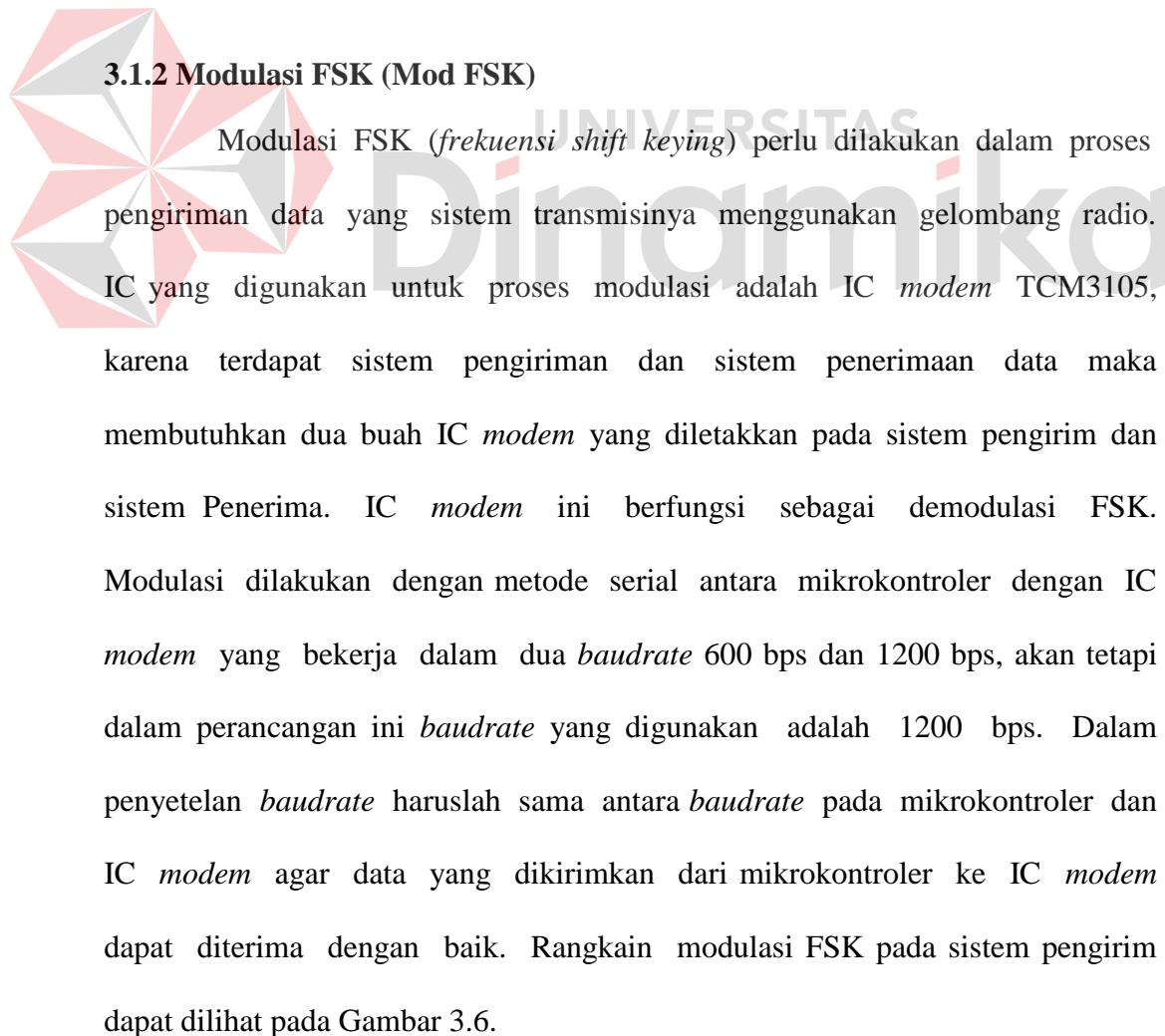
penggunaan Port pada minimum system dibawah ini adalah sebagai berikut :

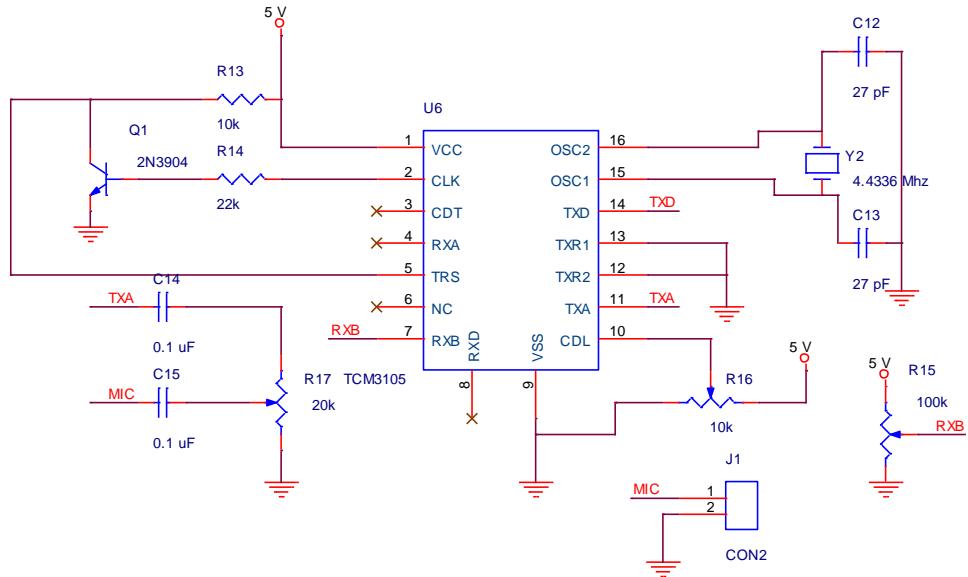
- PortB.0 - PortB.3 digunakan sebagai output lampu led sebagai indikator apakah program berjalan dengan normal atau sedang mengalami error.
- PortC.5 – PortC.7 digunakan untuk tombol input Mode tapi saat berjalan riset tombol mode tidak digunakan karena fungsionalnya tidak efisien.
- PortD.1 digunakan sebagai output untuk mengerimkan data dari sensor dan akan diteruskan ke IC Modem TCM untuk di modulasi menjadi sinyal analog agar bisa dikirimkan ke sisi receiver.
- PortC.0 – PortC.1 digunakan sebagai input sensor ultrasound dengan komunikasi I2C.



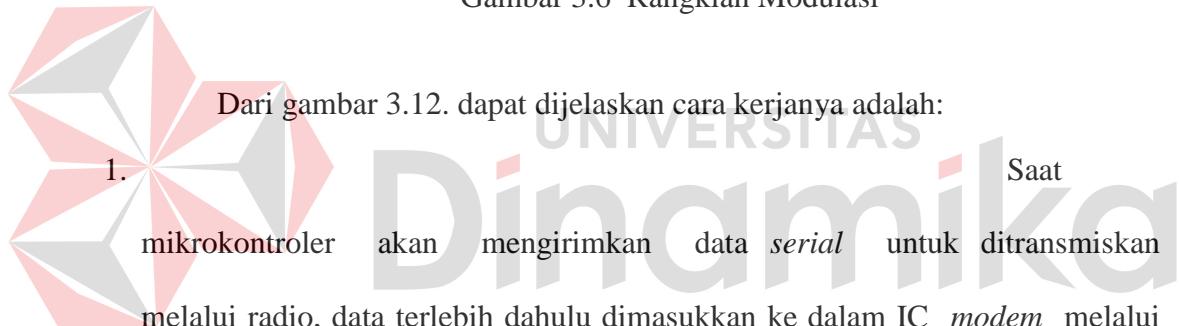
Gambar 3.5 Minimum sistem Trasmitter

Pin *VCC* diberi masukan tegangan operasi berkisar antara 4,5 Volt sampai dengan 5,5 Volt. Pin *RESET* berfungsi untuk masukan reset program secara Otomatis atau manual. Sedangkan pin *MOSI*, *MISO*, dan *SCK* digunakan untuk keperluan pemrograman mikrokontroler. Nilai kapasitor yang digunakan adalah 30 pF. Frekuensi kristal yang dipakai adalah 11,0592 MHz dengan pertimbangan bahwa kristal dengan frekuensi 11,0592 MHz mampu menghasilkan nilai bit TH1, yang digunakan untuk mengatur besarnya nilai *baudrate* yang bulat sehingga *baudrate* yang dihasilkan sama dengan nilai *baudrate* komputer. Untuk melakukan proses *downloading* program dari komputer ke dalam memory program internal mikrokontroler





Gambar 3.6 Rangkian Modulasi

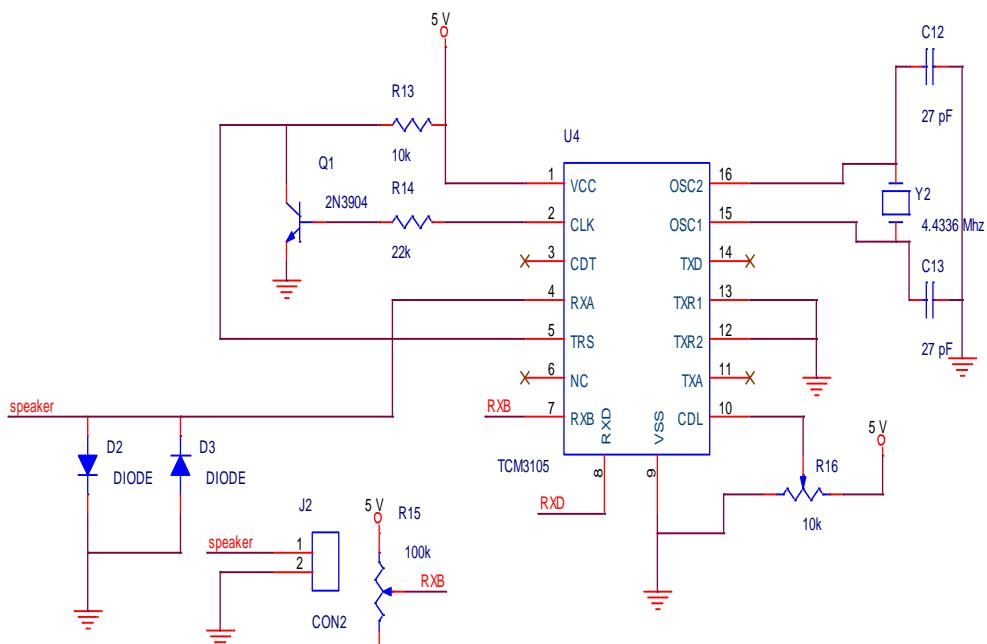


2. Setelah data tersebut masuk ke TXD, data melewati FSK modulator yang berfungsi untuk menciptakan modulasi frekuensi yang nantinya akandisisipkan pada frekuensi radio pada saat proses transmisi data. Selanjutnya data diubah menjadi data *analog* dengan rangakaian *digital to analog converter* yang ada di dalam *modem* dan dilewatkan ke rangkaian *filter* yang bertujuan untuk mengurangi *noise* dari sinyal *analog* yang dihasilkan.
3. Kemudian sinyal *analog* yang dihasilkan dikirimkan ke radio melewati *single tone* yang dihubungkan dengan *microphone*. *Single tone*

ini berfungsi untuk mengatur besarnya tegangan *analog* yang akan dikirimkan melewati *microphone* sebab tiap-tiap *microphone* belum tentu memiliki tegangan *input analog* yang sama.

3.1.3 Demodulasi FSK (Demod FSK)

Rangkaian demodulasi FSK ini digunakan untuk proses penerimaan data dan dalam perancangannya kurang lebihnya sama seperti rangkaian modulasi FSK. IC *modem* yang digunakan juga sama yang berbeda hanya dalam penggunaan pinnya saja. Untuk proses penerimaan data dilakukan dengan cara melewatkannya sinyal *analog* yang keluar dari *speaker* HT dihubungkan ke pin RXA pada IC *modem*. Selanjutnya *output*-nya terdapat pada pin RXD pada IC *modem*, pin RXD ini dihubungkan pin T1IN (pin no. 11) pada RS232. Karena ini menyangkut komunikasi data antara komputer dengan komponen *digital* maka perlu digunakan *interface* RS232 untuk mengganti tegangan TTL menjadi level tegangan RS232. Untuk lebih jelasnya dapat dilihat pada gambar 3.7.

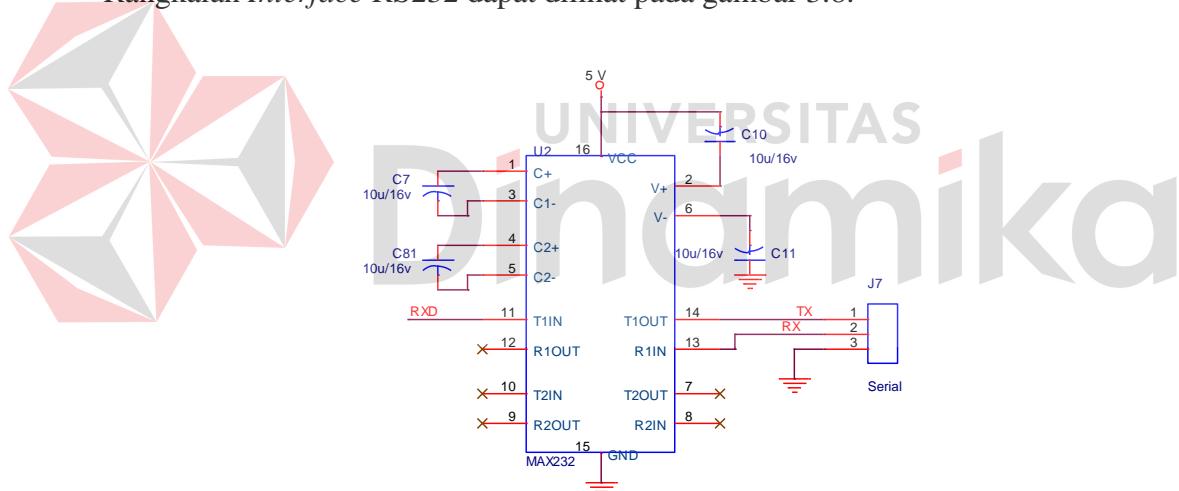


Gambar 3.7 Rangkian Demodulasi

3.1.4 Interface RS232

Interface RS232 merupakan suatu jembatan dalam metode komunikasi serial. Dalam perancangannya komponen yang digunakan adalah IC HIN232 dimana komponen pendukungnya lima buah kapasitor dengan nilai 1uF yang terhubung pada pin C1, C2, V+, V-. Penggunaan komponen ini dimaksudkan untuk komunikasi serial antara rangkaian demodulasi FSK dengan komputer. HIN232 ini akan mengubah level tegangan TTL data serial RX dan TX.

Rangkaian *Interface RS232* dapat dilihat pada gambar 3.8.

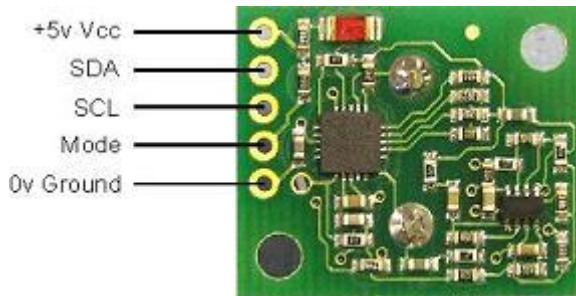


Gambar 3.8 Rangkian RS232

3.1.5 Sensor Ultrasonik

SRF02 adalah modul sensor jarak yang memiliki 2 mode komunikasi yaitu mode serial UART dan mode serial I2C. bagaimana prinsip kerja SRF02 secara detail sebagai berikut pada gambar 3.9.

- Mode Serial I2C



Gambar 3.9 Sensor Ultrasonic mode I2C

1. SDA, pin ini sebagai jalur data yang bersifat bi-directional atau dua arah yang digunakan untuk mengirimkan dan menerima data.

2. SCL, pin ini sebagai jalur clock yang digunakan untuk menghasilkan detak sinyal sinkronisasi.

Berikut ini adalah contoh kode program untuk mengakses SRF02 dalam mode serial I2C. Port C bit 1 terhubung ke pin SDA dan port D bit 0 terhubung ke pin SCL. Address devices = 0 pada mode I2C setara dengan 0xE0 :

```

while (1)
{
    // Place your code here
    tulis_SRF02(0,81);
    delay_ms(100);

    #asm("cli")
    jarak    = baca_SRF02(3);
    jarak_high = baca_SRF02(2);
    #asm("sei")
    jarak_high = jarak_high << 8;
    data_M    = jarak_high | jarak;
    data_M    = data_M/100;
    ftoa(data_M,2	lcd_out);
}

```

```

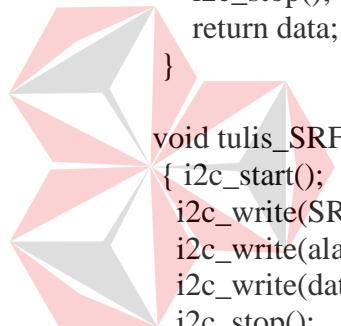
delay_ms(10);

if ((data_M>=0) && (data_M<=11))
{
    lcd_puts(lcd_out);
    puts(header);
    puts(lcd_out);
}
delay_ms(1000);
};

unsigned char baca_SRF02 (unsigned char alamat)
{   unsigned char data;
    i2c_start();
    i2c_write(SRF02_ADDRESS);
    i2c_write(alamat);
    i2c_start();
    i2c_write(SRF02_ADDRESS | 1);
    data=i2c_read(0);
    i2c_stop();
    return data;
}

void tulis_SRF02 (unsigned char alamat, unsigned char data)
{ i2c_start();
i2c_write(SRF02_ADDRESS);
i2c_write(alamat);
i2c_write(data);
i2c_stop();
}

```

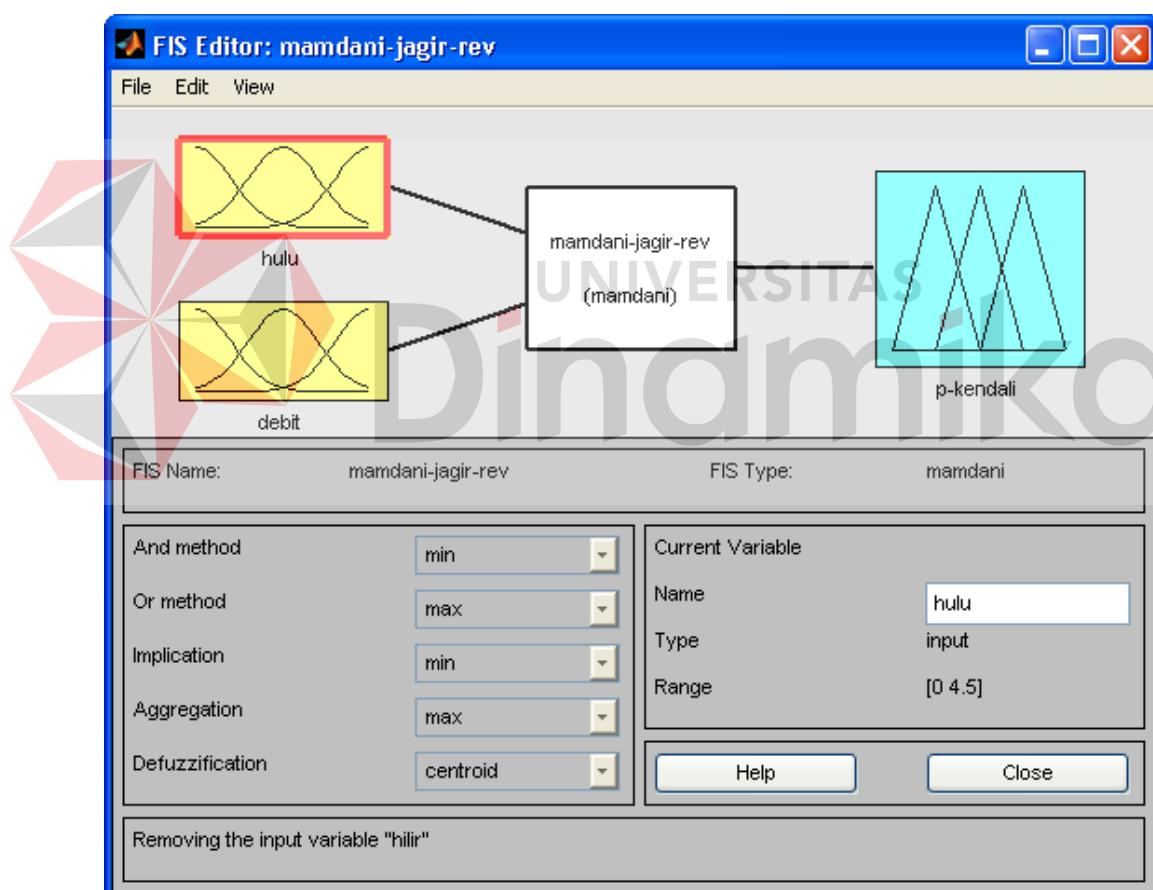


3.2 Perancangan Perangkat Lunak

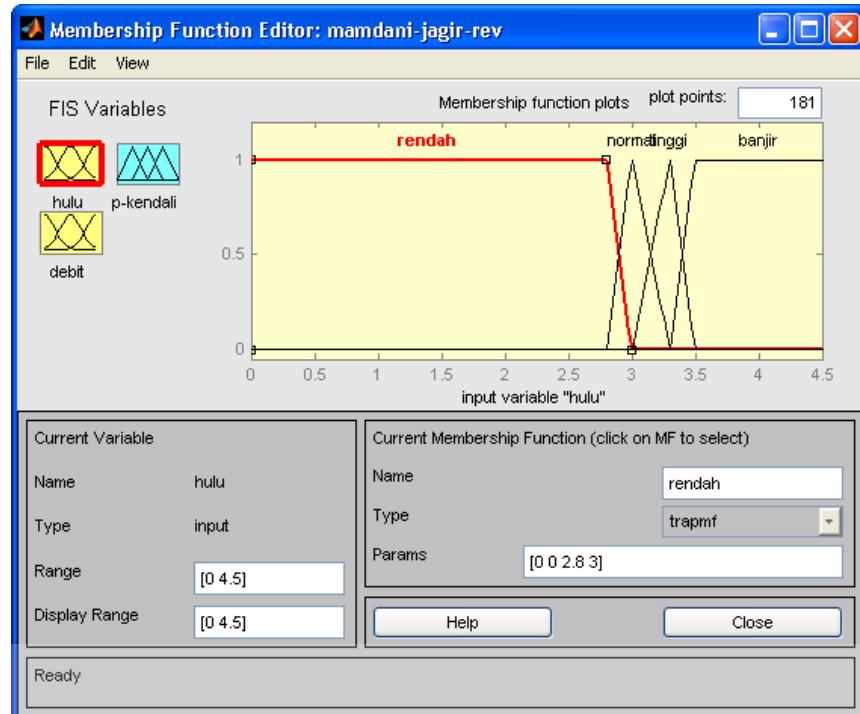
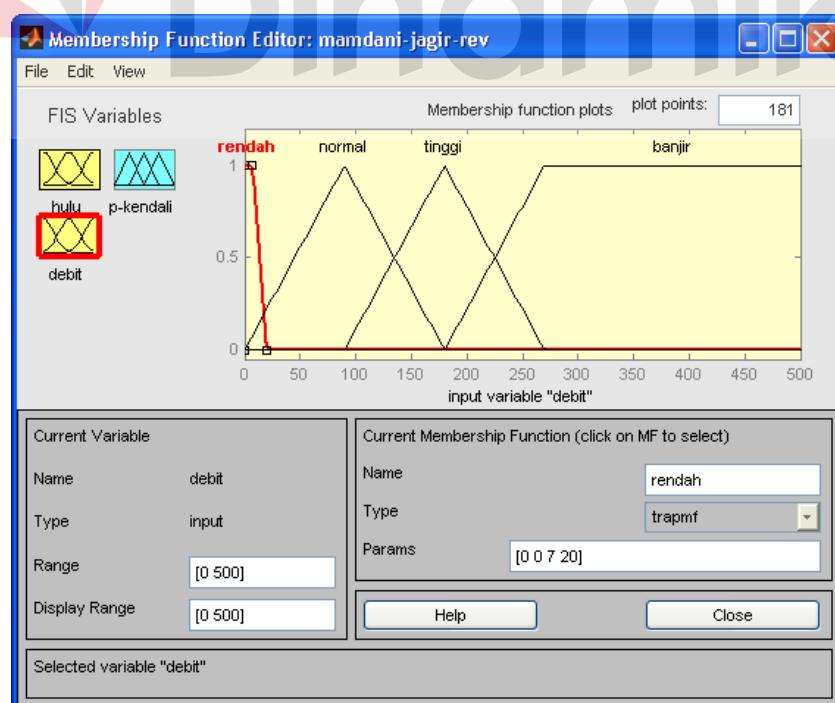
Selain *hardware* yang diperlukan pada perancangan dan pembuatan rancang bangun telemetri kualitas udara dengan memanfaatkan media *wireless* yaitu berupa radio frekuensi sebagai komunikasi data dengan sebuah komputer, juga diperlukan *software* atau program pada Mikrokontroler dan komputer untuk dapat bekerja sesuai dengan fungsinya. Perancangan perangkat lunak meliputi algoritma dan program baik pada mikrokontroler itu sendiri juga pada komputer.

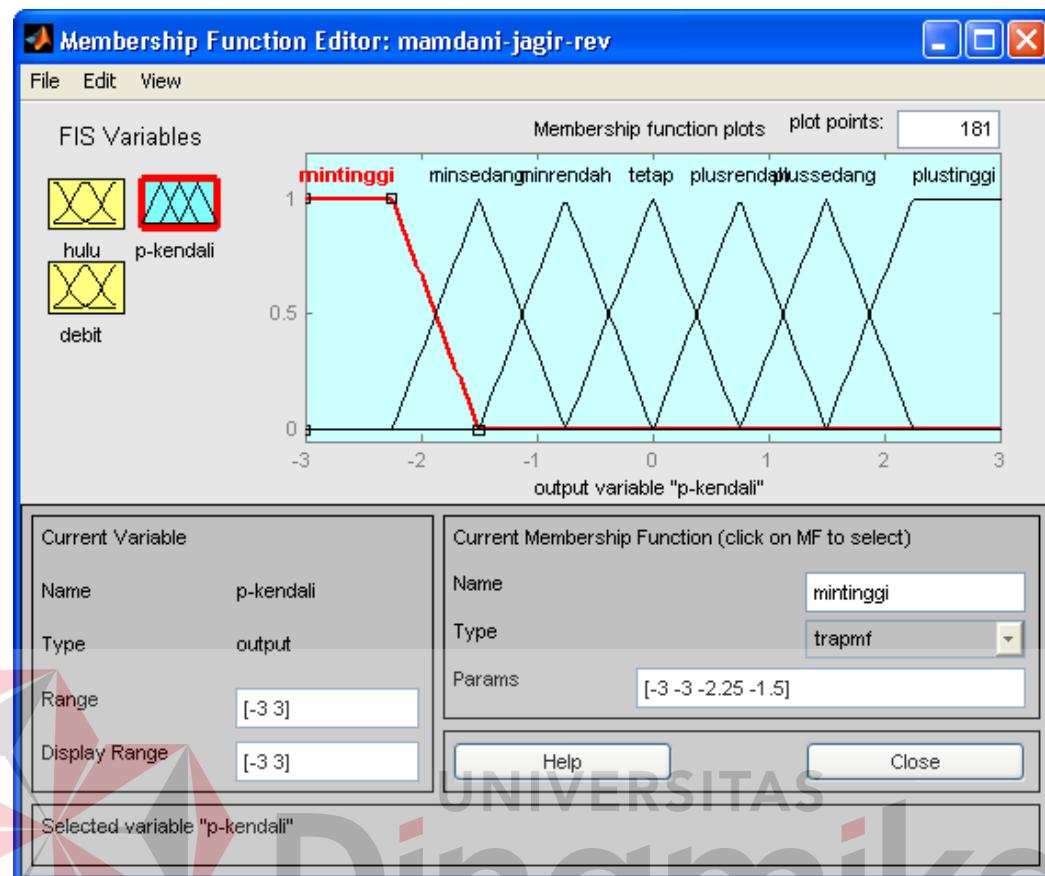
3.2.1 Desain fuzzy yang akan di terapkan dalam sistem

Sebelum membuat program fuzzy yang akan digunakan pada sistem, sebaiknya membuat desainnya terlebih dahulu pada Matlab untuk menentukan rule-rule yang akan di pakai agar perhitungan yang di pakai bisa lebih akurat dan bagus dalam pengambilan keputusanya

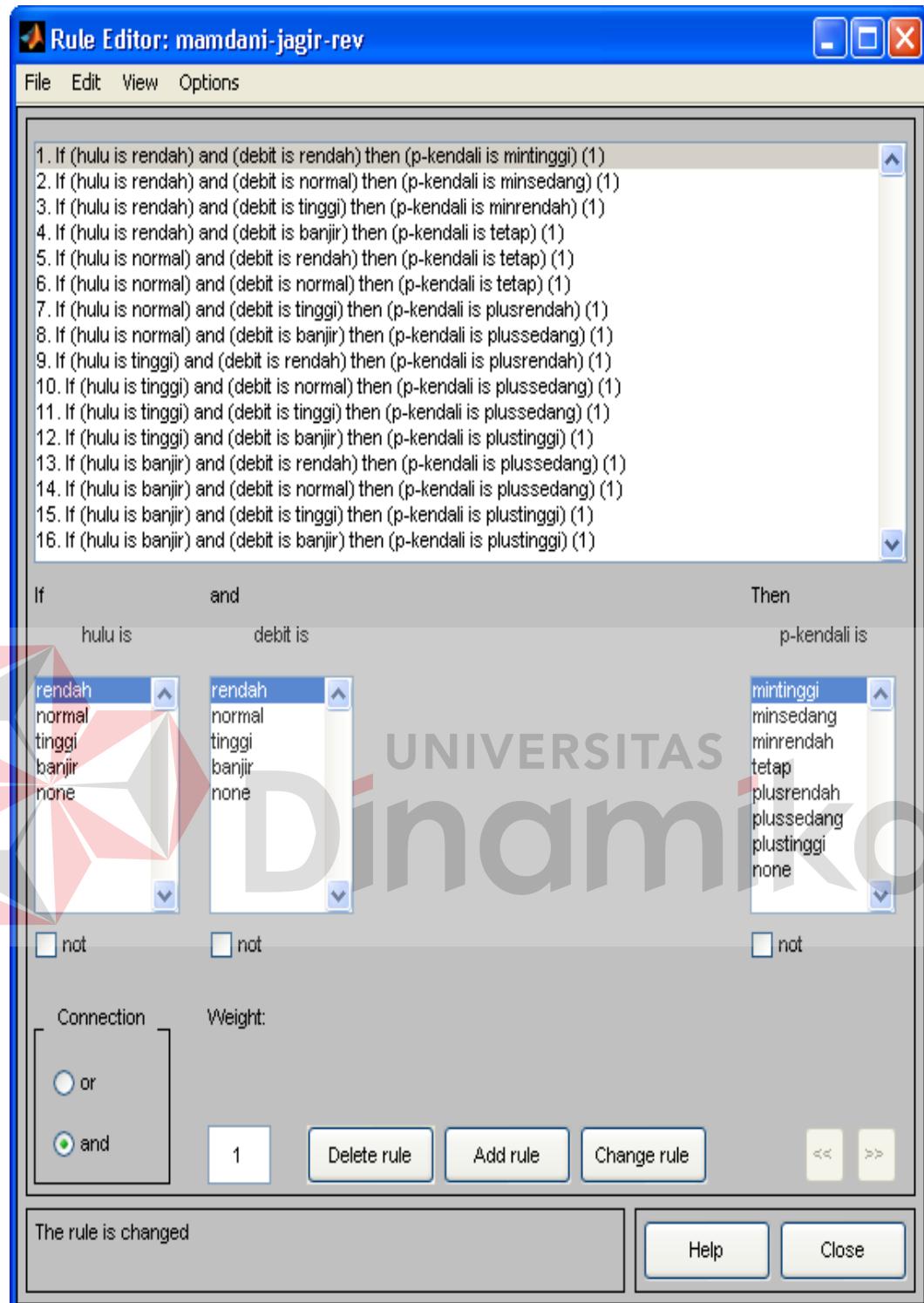


Gambar 3.10 Desain Sistem Fuzzy

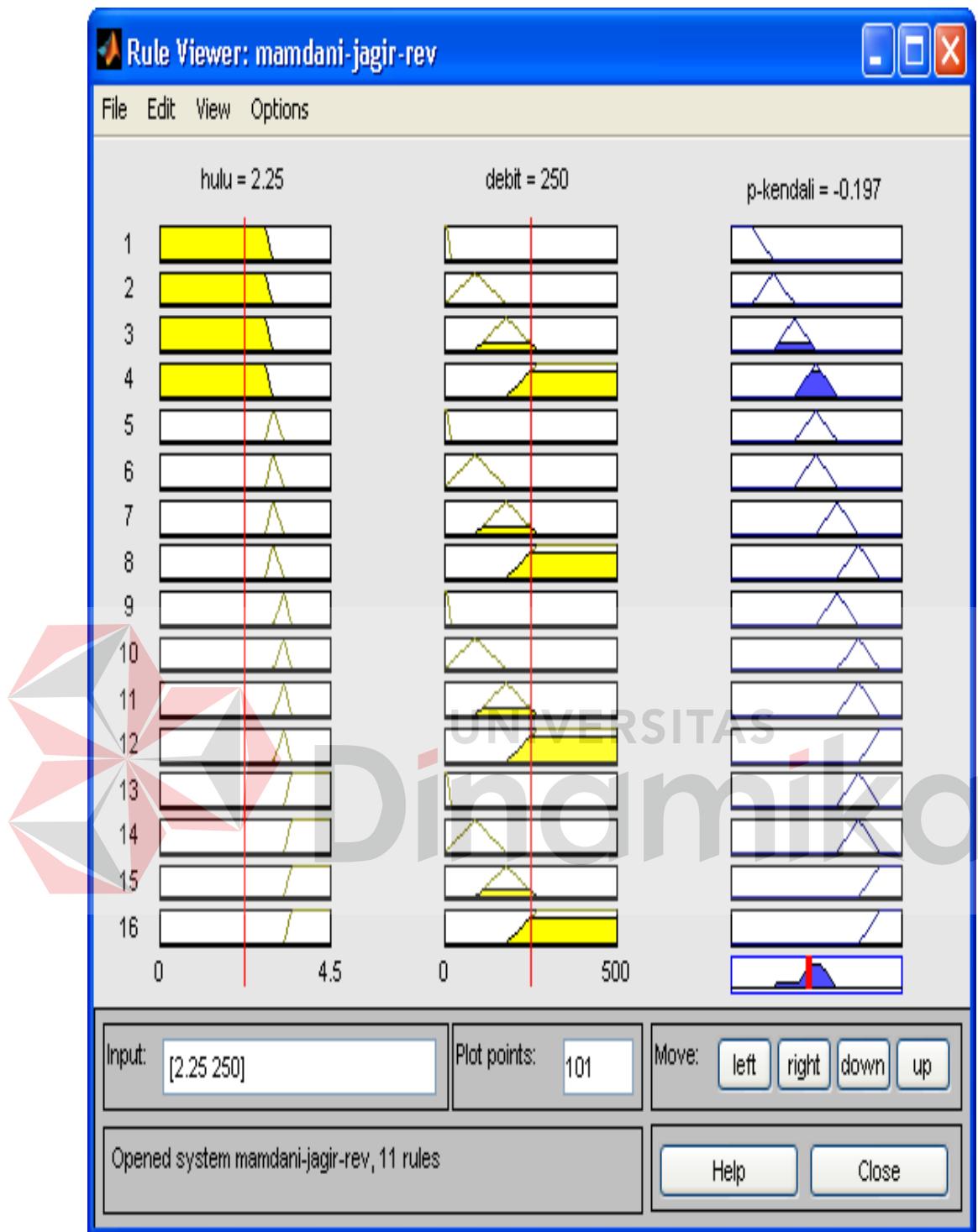
Gambar 3.11 *Membership Function Hulu*Gambar 3.12 *Membership Function Debit*



Gambar 3.13 *Membership Function* Pintu kendali



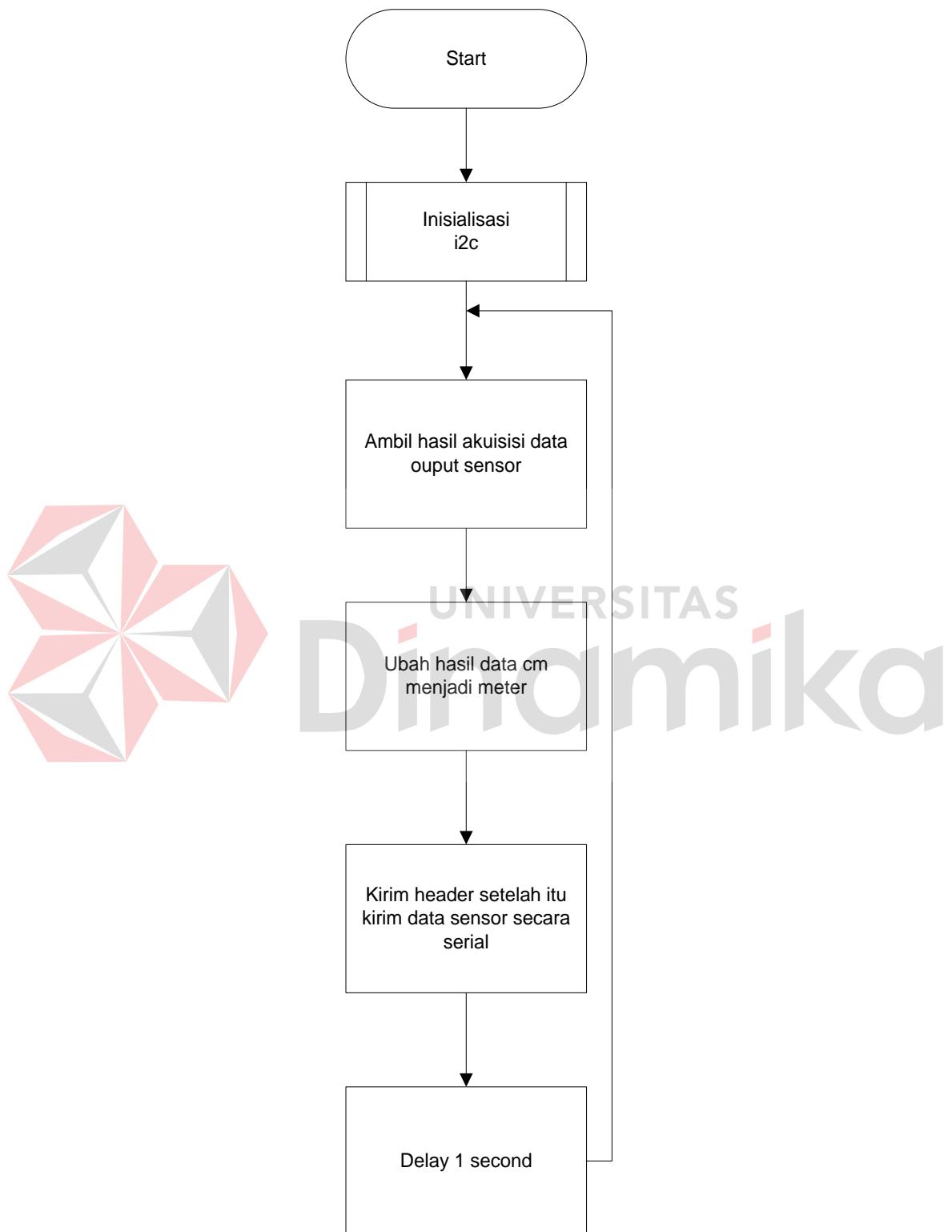
Gambar 3.14 Rule editor



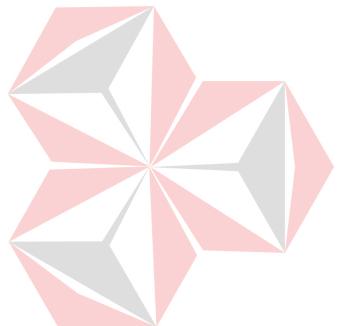
Gambar 3.15 *Rule Viewer*

Setelah mendapatkan rule seperti gambar 3.14 dan 3.15 maka rule bisa dapat dimasukkan dalam program utama untuk di terapkan pada sistem.

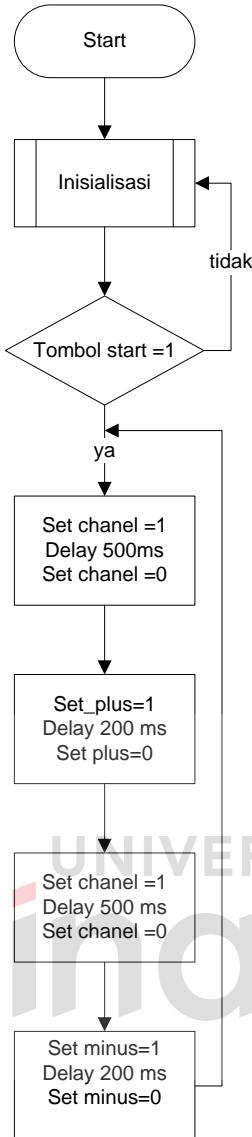
3.2.2 Flowchart Sistem Keseluruhan



Gambar 3.16 Flowchart *transmitter*



UNIVERSITAS
Dinamika



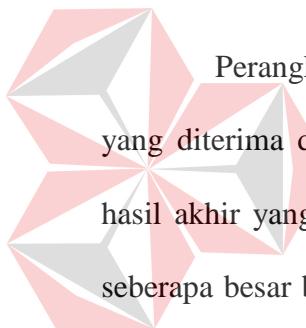
Gambar 3.17 Flowchart *reciever*

Dalam perancangan sistem ini, komunikasi serial yang terjadi pada mikrokontroler ini terbagi menjadi dua bagian pertama mikrokontroler dengan sensor yaitu bagian *transmitter* dapat dilihat pada gambar 3.16, dan mikrokontroler dengan IC modem yang terhubung dengan komputer yaitu bagian *receiver* dapat dilihat pada gambar 3.17.

Untuk sistem bagian *transmitter* berfungsi untuk mengakses sensor ultrasonik dengan komunikasi I2C, dan selanjutnya mengolah data sensor yang

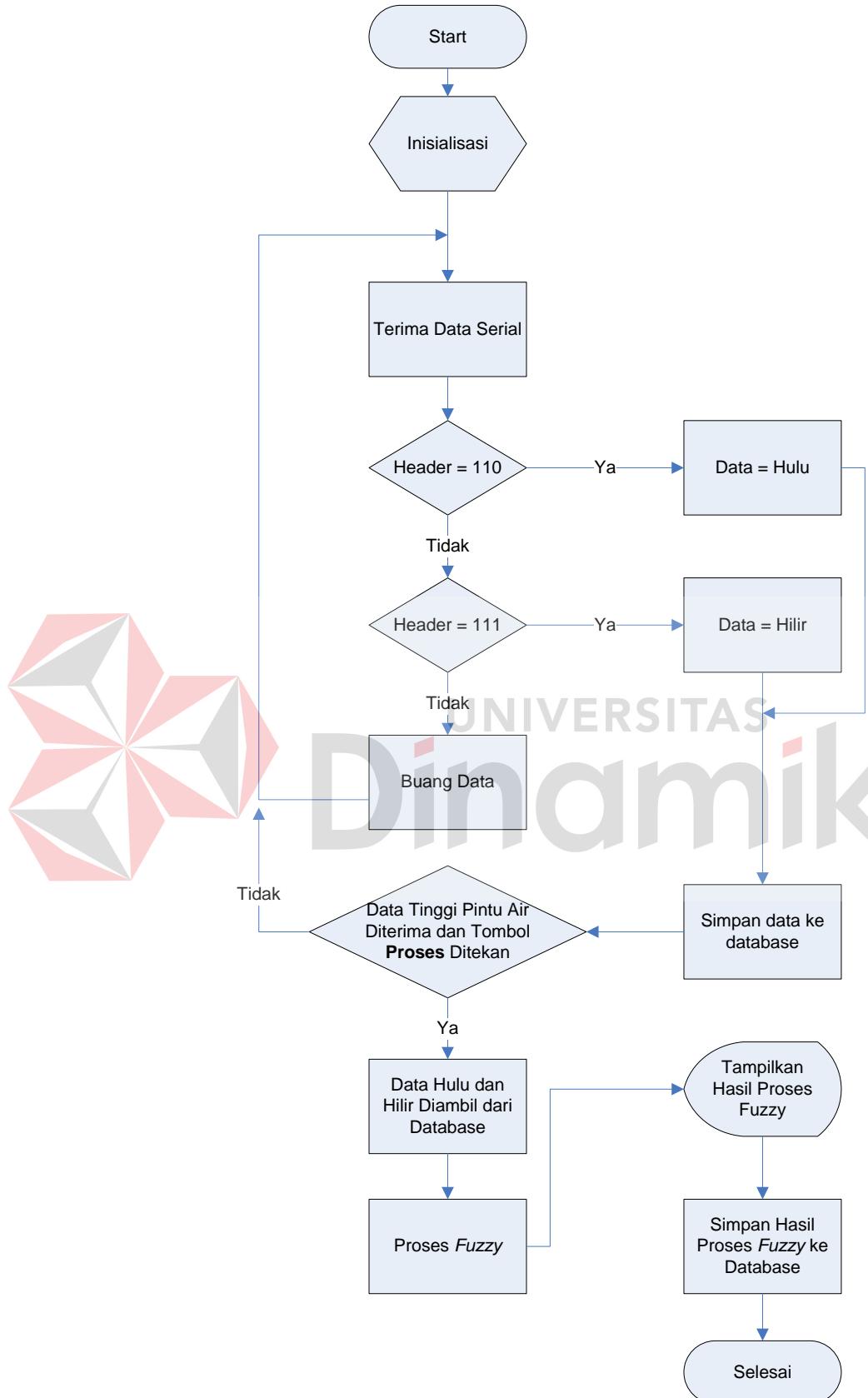
berupa data dalam satuan “cm” dan kemudian di konversi menjadi satuan “meter”, selanjutnya mikrokontroler akan mengirimkan data secara *serial* ke IC TCM untuk memodulasi data menjadi data analog supaya data bisa transmisikan lewat frekuensi radio.

Dan dalam sistem *receiver* mikrokontroler hanya berfungsi untuk mengubah *chanel* pada *walkie talkie* agar chanel dapat berganti, sehingga data dari kedua *transmitter* dapat di terima secara bergantian, jadi saat chanel 1 maka data yang diterima adalah dari *walkie talkie* transmitter yang punya settingan chanel 1 dan jika *chanel walkie talkie receiver* 22, maka data dari *walkie talkie transmitter* yang settingan chanel 22 yang diterima dan seterusnya bergantian.



Perangkat lunak pada komputer dibuat untuk memproses data sensor yang diterima dari mikrokontroler, untuk diproses lebih lanjut untuk mencapai hasil akhir yang diinginkan penulis pada tugas akhir ini, yaitu sebuah masukan seberapa besar buka-tutup pintu air yang bertujuan untuk mengatur aliran sungai sehingga bisa meminimalisasi terjadinya banjir.

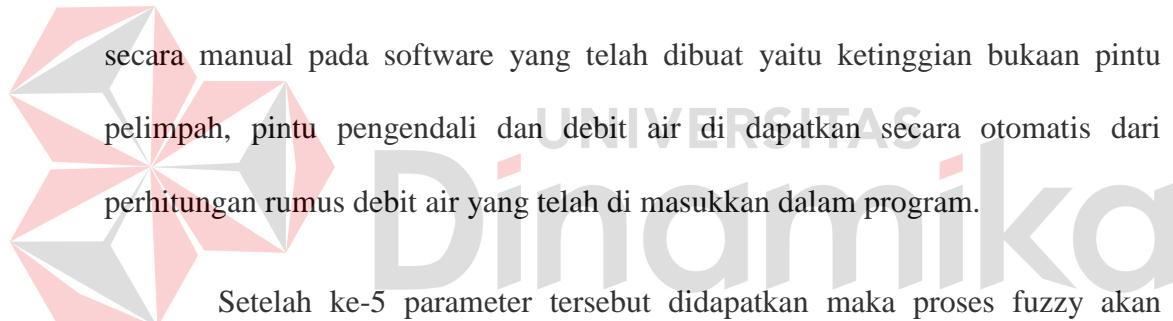
Untuk perhitungan itu penulis menggunakan metode fuzzy mandani, dan data masukan didapat dari ke dua sensor ultrasonik yang diletakkan pada sisi-sisi hulu dan hilir, setelah elavasi hulu dan hilir di dapatkan selanjutnya memasukkan data bukaan ketiga pintu air. Dengan data –data tersebut maka dapat di ketahui debit air sungai dan selanjutnya dengan melihat kapasitas sungai setelah pintu air dan ketinggian elevasi hilir dapat diperhitungakan bukaan tinggi-rendahnya bukaan pintu air dengan metode fuzzy, sehingga aliran sungai dapat dijaga kestabilannya.



Gambar 3.18 Flowchart pada program komputer

Flowchart pada gambar 3.18 merupakan keseluruhan sistem yang digunakan, dimana sistem menggunakan metode fuzzy untuk menditeksi banjir fuzzy yang digunakan menggunakan fuzzy mamdani.

Prosesnya dapat di lihat pada alur flowchart pada gambar 3.18 dimana data yang diterima dari sensor hulu dan hilir langsung di filter supaya data yang benar saja yang masuk dan kemudian di simpan ke database, setelah itu proses data yang sudah masuk dalam database akan di masukkan ke dalam proses fuzzy sebagai inputan, dalam proses fuzzy yang penulis buat menggunakan 5 parameter yaitu ketinggian elevasi sungai pada hilir dan hulu yang data nya di dapatkan secara otomatis dari alat pendeteksi ketinggian elevasi dan kemudia inputan secara manual pada software yang telah dibuat yaitu ketinggian bukaan pintu pelimpah, pintu pengendali dan debit air di dapatkan secara otomatis dari perhitungan rumus debit air yang telah di masukkan dalam program.



Setelah ke-5 parameter tersebut didapatkan maka proses fuzzy akan mulai berjalan dan akan mengeluarkan output saran pengaturan tinggi rendahnya pintu air pada petugas penjaga pintu air agar dapat meminimalisir terjadinya banjir.

BAB IV

PENGUJIAN SISTEM

Pada pengujian sistem ini penulis menguraikan tentang prosedur pengujian serta hasil yang didapat dan analisa dari hasil penelitian tugas akhir ini. Pengujian yang dilakukan meliputi perangkat keras, perangkat lunak dan sistem secara keseluruhan.

4.1 Pengujian minimum system ATmega 16

4.1.1 Prosedur Pengujian

Tujuan : Untuk mengetahui apakah seluruh port minimum system berfungsi dengan baik.

Alat yang digunakan :

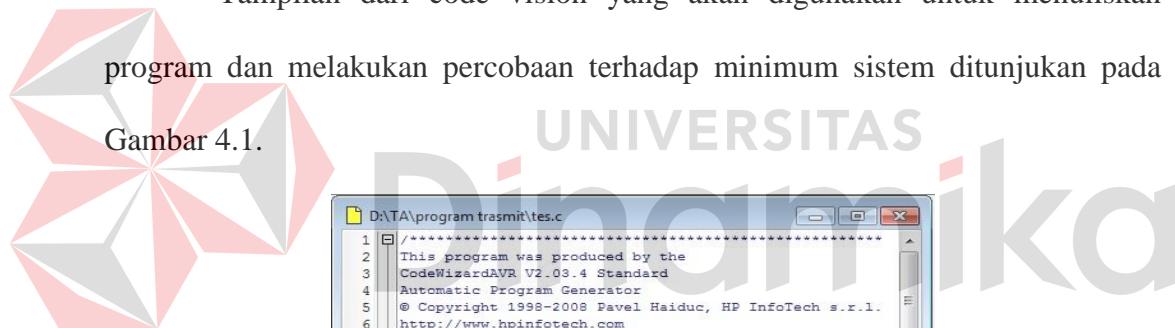
- a. Rangkaian *minimum system* ATmega 16/8
- b. PC Komputer
- c. Kabel *downloader*
- d. Program *codevision AVR*
- e. *Power supply*
- f. *Digital multimeter*

Prosedur pengujian alat :

- a. Aktifkan *power supply* dan hubungkan dengan *minimum system* selanjutnya nyalakan komputer. Kemudian sambungkan *minimum system* dengan komputer, dengan kabel downloader pada *port parallel*.

- b. Untuk mendownload program yang telah dibuat kedalam *minimum system* maka harus yang harus dilakukan adalah menjalankan menu *signature* pada *code vision* untuk mengetahui apakah chip bisa di download program atau tidak. Setelah itu membuka *file* dengan menu *open file*, *file* program yang akan di tuliskan ke *minimum system* setelah itu di *compile* dan jika sudah benar selanjutnya di *read the chip*
- c. Setelah itu dilakukan pengukuran output tegangan pada masing-masing port yang terdapat pada *minimum system* dengan menggunakan *digital multimeter*.

4.1.2 Hasil Pengujian



Tampilan dari code vision yang akan digunakan untuk menuliskan program dan melakukan percobaan terhadap minimum sistem ditunjukan pada

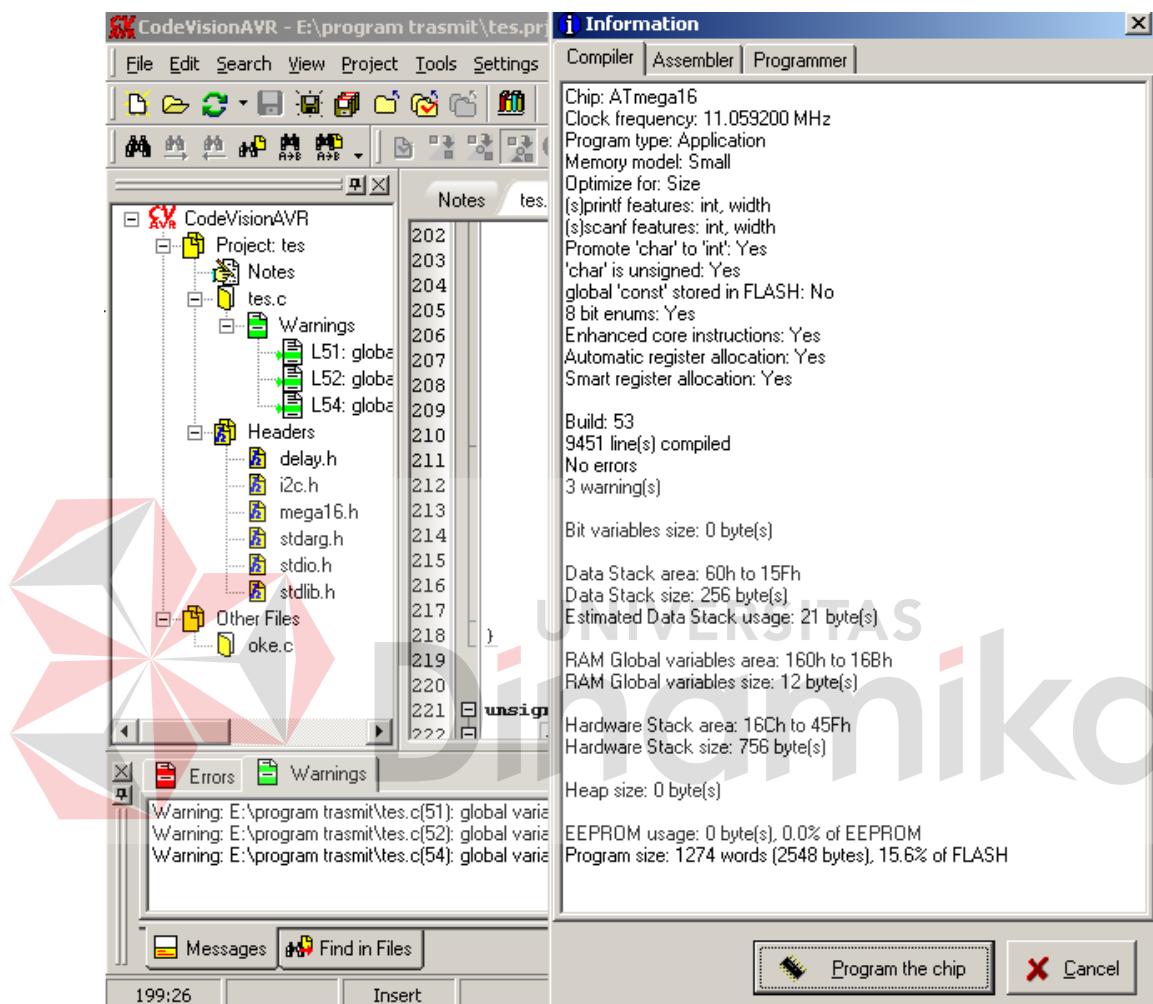
```

1  /*********************************************************************
2   * This program was produced by the
3   * CodeWizardAVR V2.03.4 Standard
4   * Automatic Program Generator
5   * Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.
6   * http://www.hpinfotech.com
7
8   Project :
9   Version :
10  Date : 7/8/2010
11  Author :
12  Company :
13  Comments:
14
15
16  Chip type          : ATmega16
17  Program type       : Application
18  Clock frequency   : 11.059200 MHz
19  Memory model      : Small
20  External RAM size : 0
21  Data Stack size   : 256
22 *****/
23
24  #include <mega16.h>
25  #include <delay.h>
26  #include <stdlib.h>
27  #include <stdio.h>
28
29
30 // I2C Bus functions
31 #asm
32     .equ __i2c_port=0x15 ;PORTC
33     .equ __sda_bit=1
34     .equ __scl_bit=0
35 #endasm
36 #include <i2c.h>
37
38 // Alphanumeric LCD Module functions
39 #asm

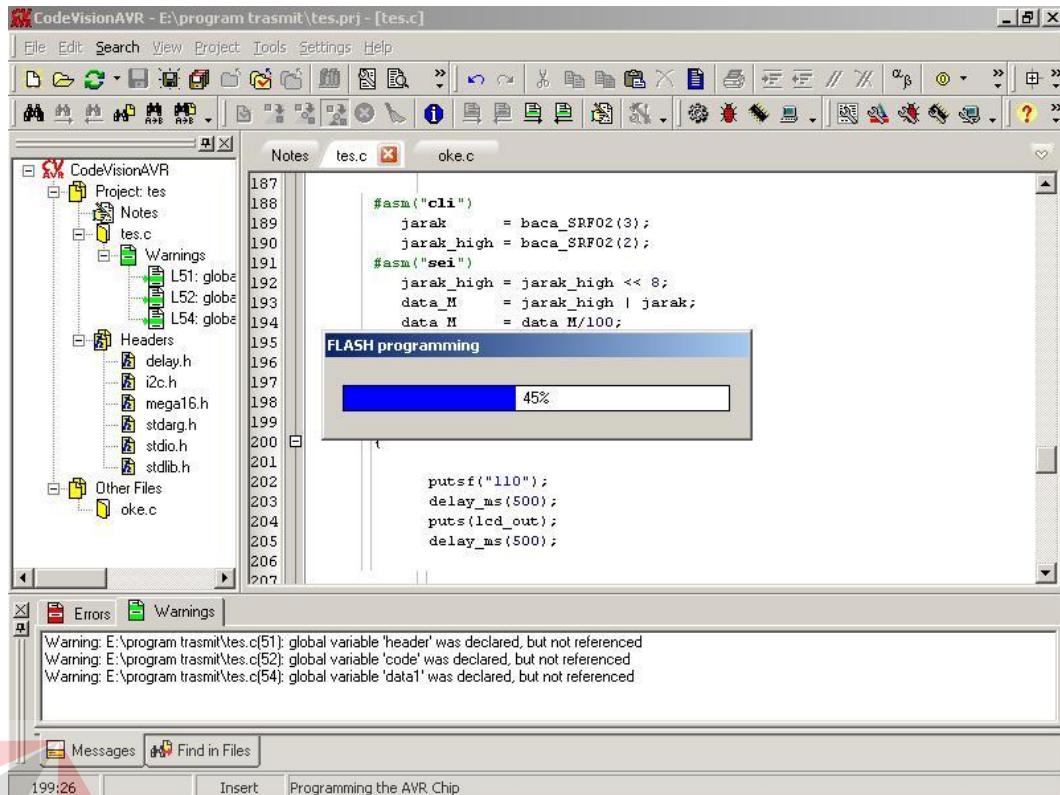
```

Gambar 4.1 Tampilan kode program

Pada gambar 4.2 menunjukkan bahwa minimum sistem telah terhubung dengan computer dan program sudah tidak mengalami error, menandakan program siap untuk di *load* ke dalam mikrokontroler.



Gambar 4.2 Tampilan hasil program berhasil di compile yaitu program siap di downloadkan ke mikrokontroler



Gambar 4.3 Tampilan proses penulisan program ke mikrokontroler

4.1.3 Analisa

Dari hasil percobaan diatas apabila menu *signature,open file* dan *write*

dapat berhasil di kerjakan dan setiap tegangan keluaran yang di hasilkan tiap pin pada *port-port* berkisar 0 V sampai dengan 5 V, maka minimum sistem dapat dikatakan bekerja dengan baik.

4.2 Pengujian IC MAX RS232 dan IC modem TCM3105

4.2.1 Prosedur Pengujian

Tujuan : Untuk mengetahui apakah *IC Modem TCM 1305* yang terhubung dengan *minimum system* berfungsi dengan baik.

Alat yang digunakan :

- Pada system penerimaan data

- b. HT (Handy Talky)
- c. Sebuah rangkaian IC modem TCM3105
- d. Sebuah rangkaian RS232
- e. Kabel dengan konektor DB9
- f. Komputer
- g. Program SerialNG

Prosedur pengujian alat pada sistem pengiriman data :

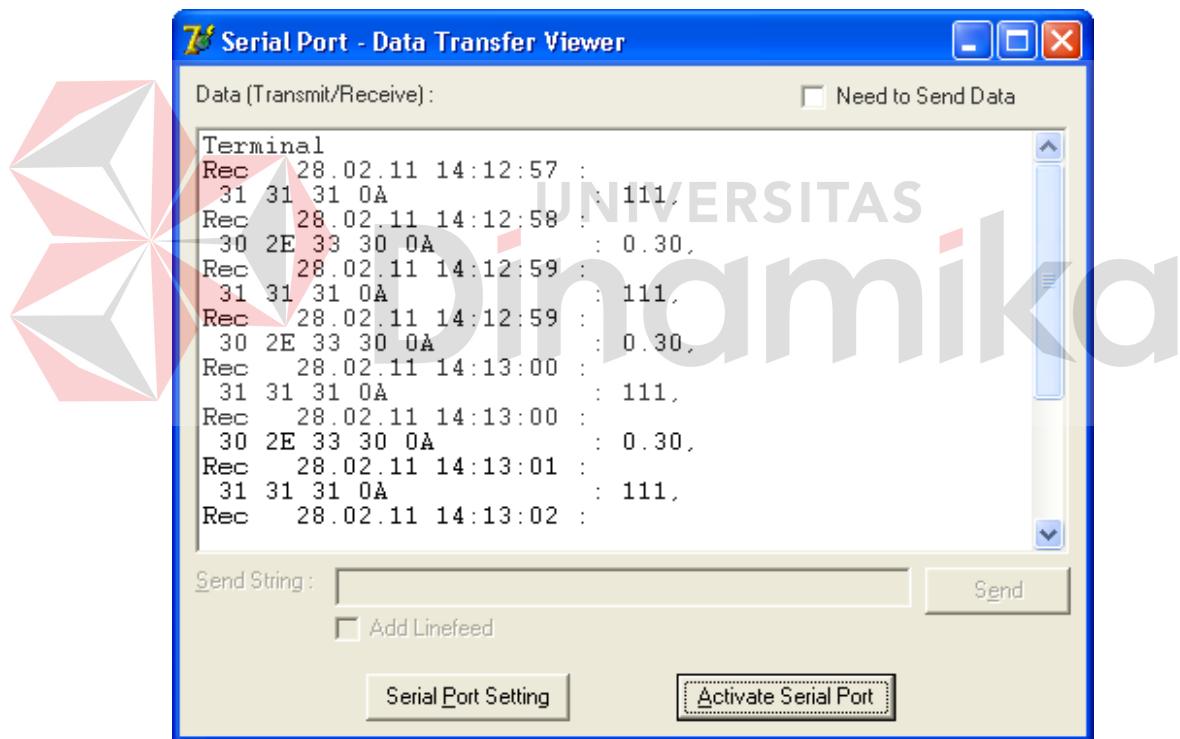
- a. Berikan catu daya 5 volt pada rangkaian.
- b. Setting konfigurasi pin mikrokontroler hubungkan pin no. 15 (TX) dengan pin no. 14 (TXD) dari IC modem.
- c. Hubungkan jalur mic pada HT ke IC modem, yang terletak pada pin no. 11 (TXA) dan melalui beberapa komponen 2 kapasitor *non-polar* dan 1 *variable resistor*.
- d. Membuat program uji coba pengiriman data pada mikrokontroler, berikut adalah program uji cobanya:
- e. Set vox pada HT untuk proses pengiriman data.

Prosedur pengujian alat Pada sistem penerimaan data :

- a. Berikan catu daya 5 volt pada rangkaian.
- b. Hubungkan jalur *speaker* pada HT ke IC modem, yang terletak pada pin no. 7 (RXA) dan melalui beberapa komponen 2 buah *diode zener*.

- c. Hubungkan jalur RXD (pin no. 8) pada IC modem ke IC RS232 pada pin no. 11 (T1IN)
- d. Hubungkan jalur T1OUT dan R1IN pada IC RS232 ke komputer melalui kabel DB9.
- e. Jalankan program serialNG untuk melihat hasil pengiriman data dengan metode *wireless*. Hasil pengujian yang tampak pada SerialNG seperti yang ditunjukkan gambar 4.4

4.2.3 Pengujian IC modem TCM 1305



Gambar 4.4 Tampilan hasil data sensor yang telah terkirim

4.3.1 Analisa

Setelah dilakukan pengujian dalam penerimaan data oleh komputer sesuai dengan gambar 4.3, maka dapat diambil kesimpulan bahwa IC modem telah bekerja dengan baik dalam memodulasi dan demodulasi. Dan begitu juga

IC MAX232 dapat mengirimkan data ke komputer secara serial dengan baik

4.3 Pengujian Pengiriman Data dan Proses Switcing Chanel

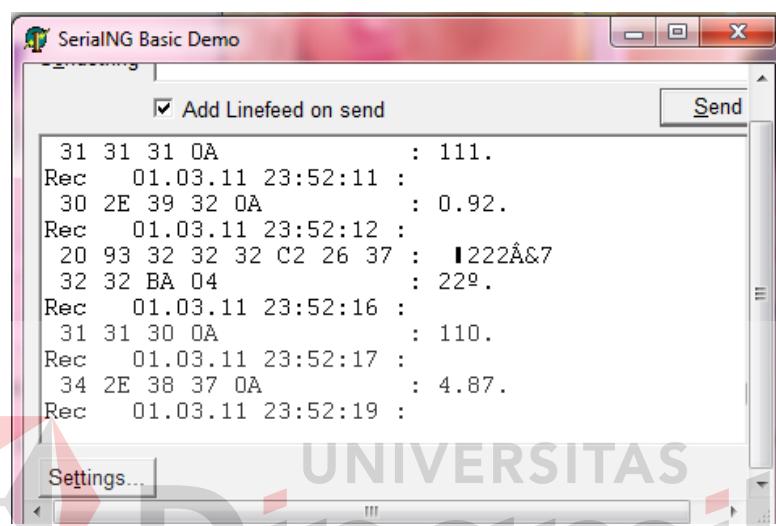
4.3.1 Prosedur Pengujian

Tujuan : Untuk mengetahui apakah komunikasi nirkabel sudah berjalan dengan baik dan proses swicting chanel juga telah berfungsi sebagai pemindah saluran data, dari sensor hulu ke hilir dan seterusnya

Alat yang digunakan :

- a. 2 modul transmitter dan 1 modul receiver
- b. Komputer
- c. Program SerialNG
- d. Prosedur pengujian alat pada sistem pengiriman data :
 - a. Berikan catu daya 5 volt pada rangkaian.
 - b. Nyalakan masing-masing modul
 - c. Set walkie talkie yang terdapat pada modul transmitter dengan volume maksimal dan set vox sampai terdapat tulisan Tx pada layar lcd walkie talkie
 - d. Set volume pada walkie talkie yang terdapat pada modul reciever dengan volume paling rendah yaitu 1, dan set vox sampai muncul tulisan Rx pada layar lcd walkie talkie
 - e. Jalankan serialNG pada komputer dan setting COM sesuai Port serial yang digunakan yang digunakan dan setting baudrate 1200

4.3.3 Pengujian Pengiriman Data dan Proses Switcing Chanel



Gambar 4.5 Tampilan hasil data sensor yang telah terkirim

4.3.4 Analisa

Setelah dilakukan pengujian dalam penerimaan data oleh komputer sesuai dengan gambar 4.4, maka dapat diambil kesimpulan bahwa hasil pengiriman data serial secara *wireless* dan proses swicting saluran data sensor yang berada di hilir dan yang berada di hulu dapat berjalan dengan baik.

4.4 Pengujian Sensor Ultrasound

4.4.1 Prosedur Pengujian

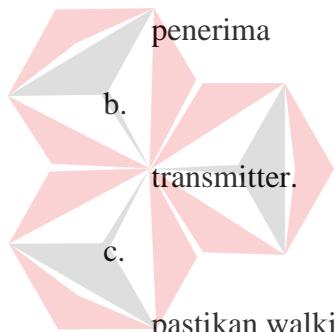
Tujuan : Untuk mengetahui apakah Sensor Ultrasound yang terhubung minimum sistem berfungsi dengan baik.

Alat yang digunakan :

- a. modul receiver
- b. modul transmitter
- c. Kabel serial to usb
- d. 2 walkie talkie
- e. PC Komputer
- f. Kabel serial
- g. *Power supply*
- h. Program SerialNG

Prosedur pengujian alat :

- a. Berikan catu daya 5 volt pada minimum system pengirim dan



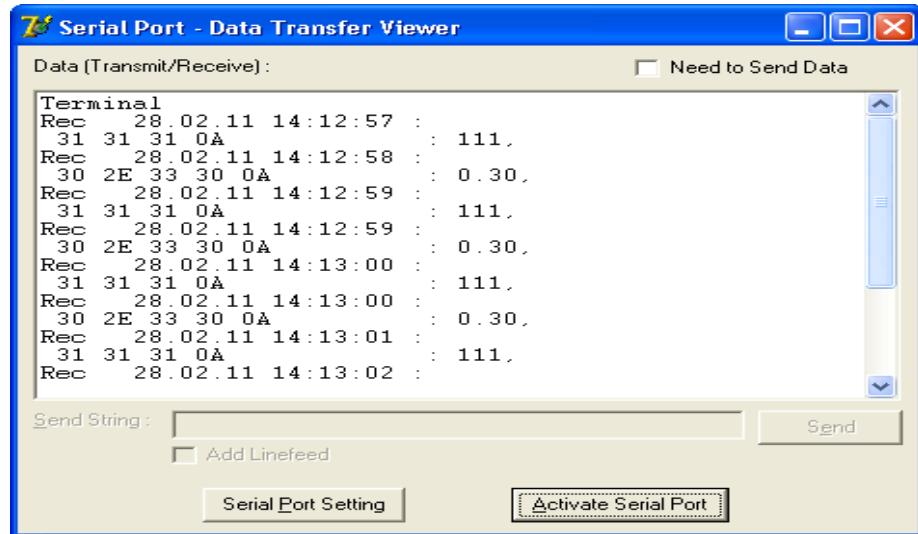
Download program seperti pada lampiran listing program

Hubungkan sensor ultrasound ke minimum system pengirim

pastikan walkie talkie sudah siap mengirim data.

- d. Hidupkan minimum system dan walkie talkie di sisi penerima
dan sambungkan ke computer dengan kabel serial
- e. Pastikan walkie talkie sudah siap untuk menerima data
- f. Jalankan program serialNG
- g. Hidupkan minimum sistem di sisi pengirim dan perhatikan
data yang masuk di program serialNG seperti gambar 4.6

4.4.2 Pengujian Sensor Ultrasonik



Gambar 4.6 Tampilan hasil data sensor

Tabel 4.1 hasil pembacaan sensor ultrasonik

Jarak (cm)	Hasil pembacaan sensor (I)	Hasil pembacaan sensor (II)	Hasil pembacaan sensor (III)	Toleransi
50	48	46	47	6 %
100	96	95	100	3 %
150	146	148	144	2,6 %
200	196	194	195	2,5 %
250	246	236	243	3,6 %
300	285	288	297	3,3 %
350	349	330	347	2,3 %
400	390	380	374	7 %
450	449	421	433	3,5 %
500	479	470	487	4,4 %
550	525	530	535	3,6 %

600	603	600	585	0,6 %
-----	-----	-----	-----	-------

4.4.3 Analisa

Setelah dilakukan pengujian terhadap sensor ultrasonik dan di dapatkan data seperti Tabel 4.1, dan data sudah sesuai dengan ukuran aslinya dengan toleransi yang sudah ditentukan jadi sensor ultrasonik telah berfungsi dengan baik.

4.5 Pengujian Data base

4.5.1 Prosedur Pengujian

Tujuan : Untuk mengetahui apakah data base yang telah dibuat telah berjalan dengan baik

- a. PC Komputer
- b. Aplikasi visual basic untuk testing data base
- c. Data base my sql

Prosedur pengujian sistem :

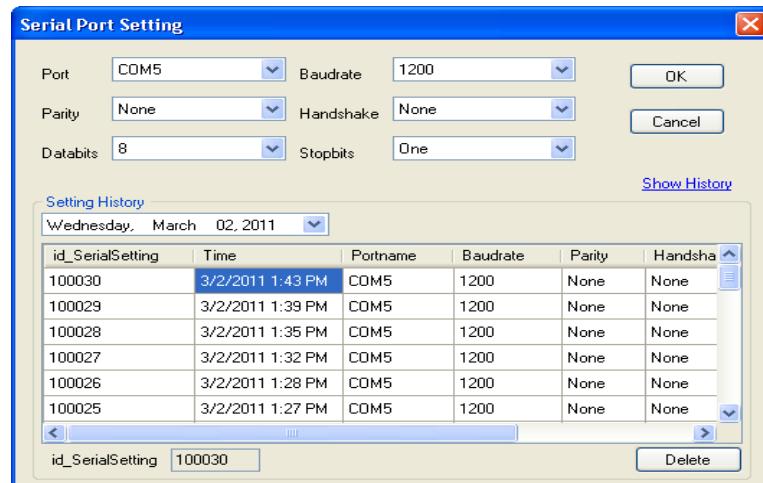
Pengujian pada program data base ini dilakukan dengan cara mensinkronisasi aplikasi VB yang telah dibuat dengan data base

Prosedur pengujian program GUI ini adalah :

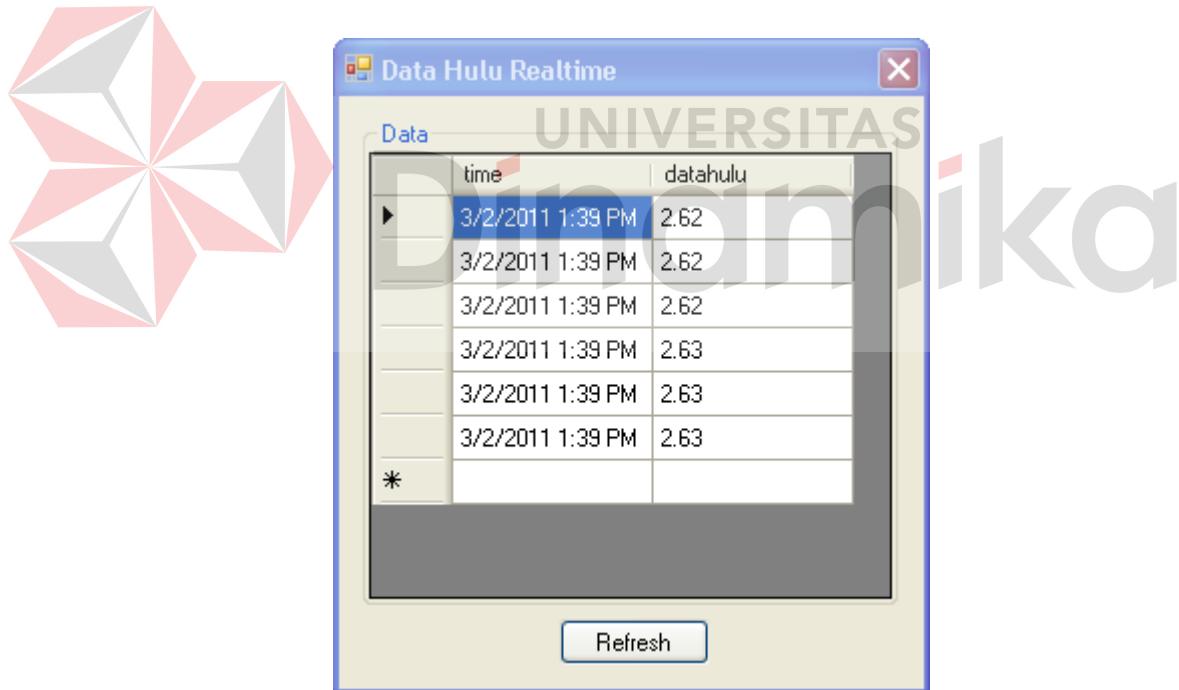
- a. Siapkan komputer yang telah terinstal visual basic dan data base my sql
- b. Jalankan aplikasi vb yang telah dibuat dan my sql
- c. Setting serial port

- d. Nyalakan modul transmitter dan modul receiver

4.5.2 Pengujian Data base



Gambar 4.7 Tampilan setting serial port



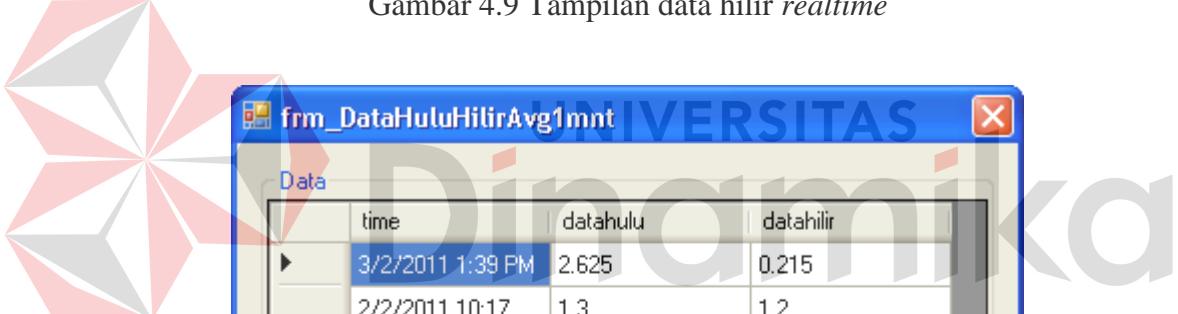
Gambar 4.8 Tampilan data hulu *realtime*

Data Hilir Realtime

Data

	time	datalhilir
▶	3/2/2011 1:45 PM	0.23
	3/2/2011 1:45 PM	0.26
	3/2/2011 1:45 PM	0.24
	3/2/2011 1:45 PM	0.23
	3/2/2011 1:45 PM	0.24
	3/2/2011 1:45 PM	0.24
	3/2/2011 1:45 PM	0.23
	3/2/2011 1:45 PM	0.24
	3/2/2011 1:44 PM	0

Refresh

Gambar 4.9 Tampilan data hilir *realtime*


frm_DataHuluHilirAvg1mnt

Data

	time	datahulu	datalhilir
▶	3/2/2011 1:39 PM	2.625	0.215
	2/2/2011 10:17 ...	1.3	1.2
	2/2/2011 10:15 ...	1.1	1.1
	2/2/2011 10:13 ...	2.16	2.16
	2/2/2011 10:11 ...	2.2	2.2
*			

Refresh Shrink

Gambar 4.10 Tampilan data hulu dan hilir *average 1 minute*

4.5.3 Analisa

Dari pengujian ini disimpulkan bahwa data base sudah bisa berjalan

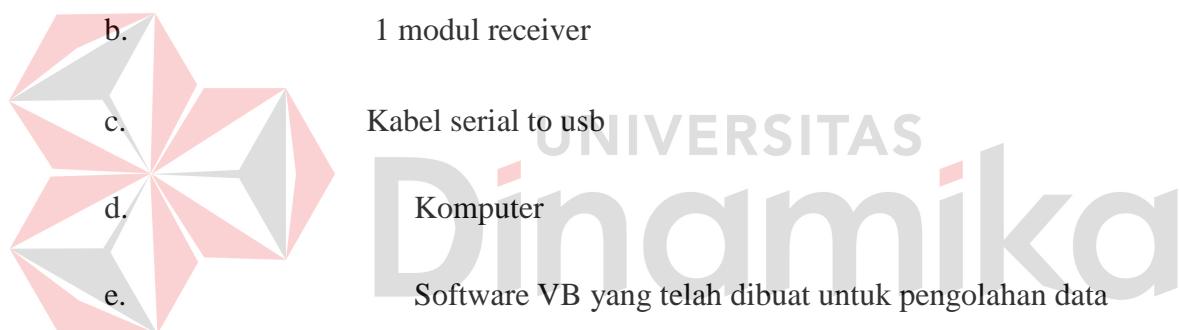
dengan baik untuk penyimpanan data dan siap untuk di gabungkan pada program utama

4.6 Pengujian sistem keseluruhan

4.6.1 Prosedur Pengujian

Tujuan : Untuk mengetahui apakah sistem keseluruhan yang telah dibuat sudah bekerja dengan baik atau belum mulai dari hardware, software dan pengiriman datanya yang melalui media nirkabel

Alat yang digunakan :



Prosedur pengujian sistem keseluruhan :

- a. Jalankan 2 modul transmitter
- b. Jalan modul reiver
- c. Sambungkan kabel serial dari modul receiver pada komputer
- d. Jalankan software sistem yang telah dibuat
- e. Analisa hasil inputan dari sensor dan outpu dari sistem

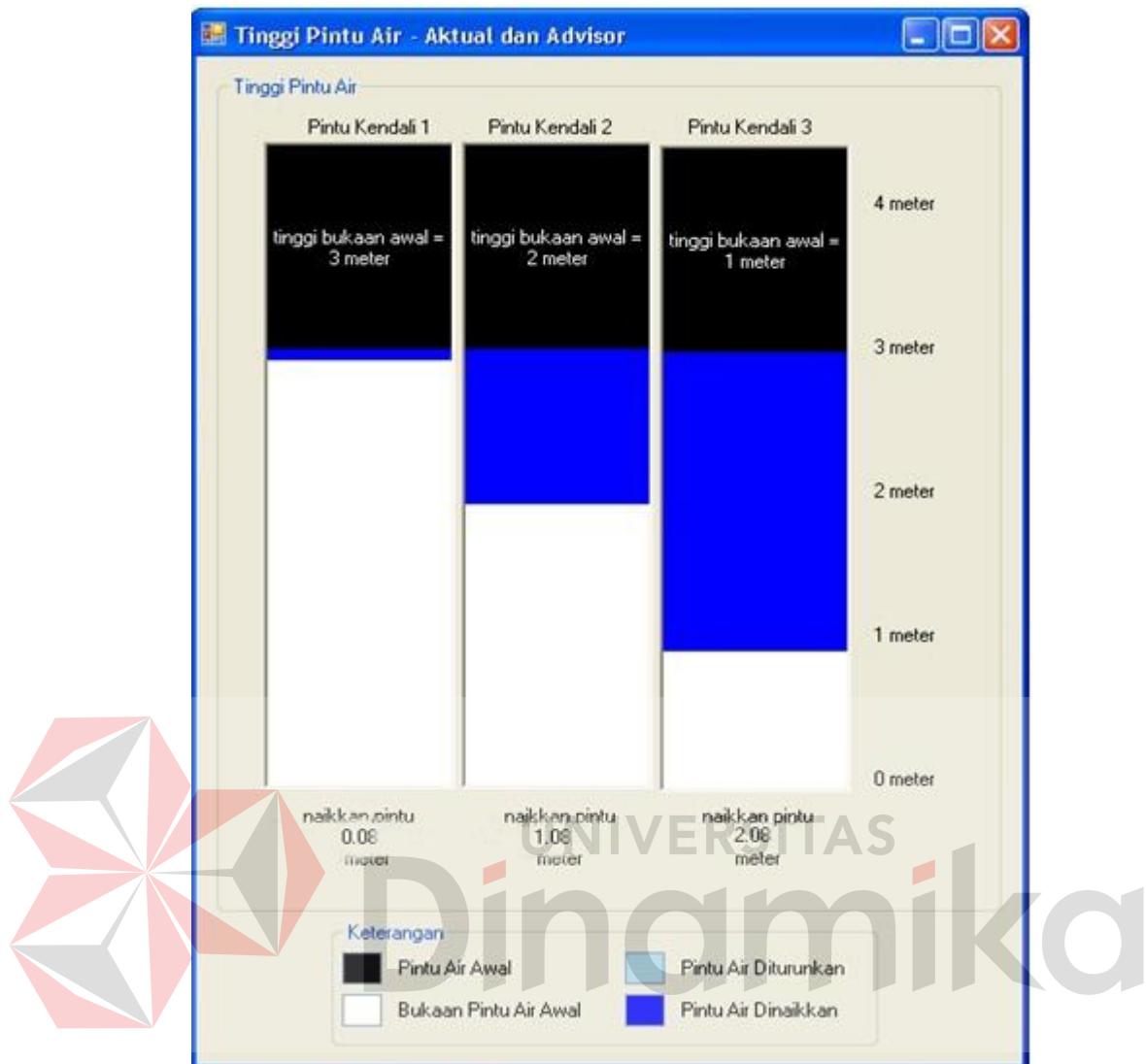
4.6.2 Pengujian sistem keseluruhan

Jagir River - Flood Preventing Advisor

Status Awal		Advisor	
Tinggi Muka Air Awal Tinggi Muka Air Hulu Pintu <input type="text" value="3.5"/> Optimum Range : Min 2.8 m - Max 3.3 m Tinggi Muka Air Hilir Pintu <input type="text" value="0"/> Optimum Range : Min 0 m - Max 2 m		Tinggi Pintu Air Rekomendasi Tinggi Bukaan Pintu Kendali 1 <input type="text" value="3.0833333"/> Tinggi Bukaan Pintu Kendali 2 <input type="text" value="3.0833333"/> Tinggi Bukaan Pintu Kendali 3 <input type="text" value="3.0833333"/> Tinggi Bukaan Pintu Pelimpah 1 <input type="text" value="0"/> Tinggi Bukaan Pintu Pelimpah 2 <input type="text" value="0"/> Tinggi Bukaan Pintu Pelimpah 3 <input type="text" value="0"/> Threshold <input type="text" value="2.4166666"/>	
Tinggi Pintu Air Awal (Input Manual) Tinggi Bukaan Pintu Kendali 1 <input type="text" value="3"/> Tinggi Bukaan Pintu Kendali 2 <input type="text" value="2"/> Tinggi Bukaan Pintu Kendali 3 <input type="text" value="1"/> Tinggi Bukaan Pintu Pelimpah 1 <input type="text" value="0"/> Tinggi Bukaan Pintu Pelimpah 2 <input type="text" value="0"/> Tinggi Bukaan Pintu Pelimpah 3 <input type="text" value="0"/>		Tinggi Muka Air Optimal Tinggi Muka Air Hulu Pintu <input type="text" value="3"/> Tinggi Muka Air Hilir Pintu <input type="text" value="0"/>	
<input checked="" type="checkbox"/> Inputkan Tinggi Muka Air secara Manual <input type="button" value="Proses"/>		Debit Debit Awal <input type="text" value="247.30111"/> Debit Advisor <input type="text" value="348.32940"/> <input type="button" value="Tampilkan Debit Per Pintu"/>	

Gambar 4.11 Tampilan Software Preventing Advisor Detail





Gambar 4.12 Tampilan kondisi pintu air



Gambar 4.13 Tampilan Warning system pada saat terjadi kondisi siaga merah

4.6.3 Analisa

Melihat dari hasil yang telah di uji dapat di simpulkan software sistem yang telah di buat sudah bisa berjalan dengan baik dan modul modul yang telah dibuat juga sudah bisa bekerja dengan baik dalam hal mengirimin dan menerima data



BAB V

KESIMPULAN

Berdasarkan pengujian pada perangkat keras dan perangkat lunak yang di pergunakan dalam tugas akhir ini, maka dapat diambil kesimpulan dan saran dari hasil yang diperoleh.

5.1 Kesimpulan

1. ke akuratan data yang dikirimkan melalui walkie talkie terkadang masih sering kacau, di karena noisenya lumayan besar jadi masih gampang terintervensi dengan frekwensi-frekvensi lainnya
2. Pengukuran jarak yang dihasilkan dari pembacaan sensor ultrasonik tidak begitu akurat dikarenakan karenan karakteristik gelombang ultrasonik yang sedikit menyebar jadi tidak lurus, sehingga sangat rentan noise di saat pengukuran jaraknya, hasil keakuratanya bisa di lihat pada Tabel 4.1
3. Jarak jangkau pengiriman data secara nirkabelnya masih dalam batasan range maksimal kurang lebih 1 km
4. Perhitungan fuzzynya masih dalam tahap riset atau dalam pengembangan jadi untuk hasil yang benar benar sesuai atau sama persis dengan aturan yang ada buku pedoman operasi DAM JAGIR

5.2 Saran

Untuk perbaikan dan pengembangan sistem yang telah di rangang dapat dilakukan dengan :

1. Penggantian sensor pengukur jaraknya yang akurasinya lebih tinggi dan tidak rentan oleh intervensi dari luar dan tidak menyebar seperti ultrasonik
2. Penggunaan media nirkabelnya menggunakan modem gsm yang tidak rentan oleh noise dan intervensi gelombang dari luar dan jarak jangkaunya lebih luas
3. Menambah sensor pengukur penditeksi bukaan pintu pada setiap pintu air sehingga sistem berjalan secara otomatis tanpa perlu lagi memasukan seberapa tinggi bukaan pintu secara manual pada software.
4. Penyempurnaan sistem pada sistem fuzzy nya sehingga perhitungan pengaturan pintu air untuk mencegah terjadinya banjir pada sungai lebih akurat lagi.

DAFTAR PUSTAKA

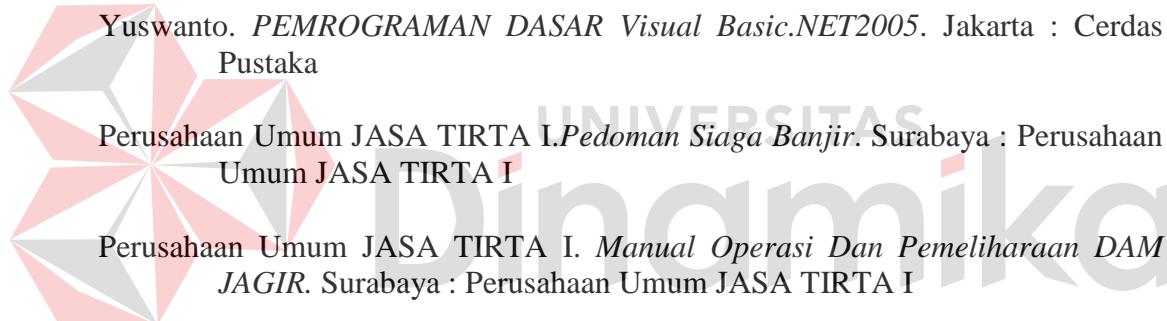
Adrianto,H.2008.*Pemrograman Mikrokontroler AVR ATMEGA16.* Bandung:Informatika

Nalwan, P. A. 2003. *Panduan Praktis Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51.* Jakarta:PT Elex Media Komputindo

Iswanto. 2008. *Design dan Implementasi Sistem Embedded Mikrokontroller ATMega8535 dengan Bahasa Basic.* Yogyakarta:Gava Media.

Winoto, Ardi. 2008. *Mikrokontroler AVR ATMega8/32/16/8535 dan Pemrogramannya dengan Bahasa C pada WinAVR.* Bandung:Informatika

Kusumadewi, Sri, Haripurnomo. *APLIKASI LOGIKA FUZZY (untuk pendukung keputusan).* Jogjakarta : Graha Ilmu



Yuswanto. *PEMROGRAMAN DASAR Visual Basic.NET2005.* Jakarta : Cerdas Pustaka

Perusahaan Umum JASA TIRTA I.*Pedoman Siaga Banjir.* Surabaya : Perusahaan Umum JASA TIRTA I

Perusahaan Umum JASA TIRTA I. *Manual Operasi Dan Pemeliharaan DAM JAGIR.* Surabaya : Perusahaan Umum JASA TIRTA I

Nugraha.2010. *blog.ub.ac.id*

(<http://blog.ub.ac.id/sukmanugraha/2010/03/22/real-time-clock-menggunakan-i2c-bus-pada-modul-dst-52/>, diakses tanggal 15 Juni 2010 pukul 20:19)

(http://id.wikipedia.org/wiki/frequency-shift_keying/, diakses tanggal 5 februari 2011 pukul 20:17)

(http://www.electroniclab.com/sensor_ultrasonic, diakses tanggal 5 februari 2011 pukul 20:25)

Datasheet SRF-02, (http://www.sensirion.com/en/pdf/product_information/Datasheet-humidity-sensor-SRF-02x.pdf, diakses tanggal 11 Desember 2010 pukul 09:44)

Datasheet TCM3105, (<http://datasheets.maxim-ic.com/en/ds/TCM3015.pdf>, diakses tanggal 4 Desember 2010 pukul 15:00)