

SKRIPSI

PERANCANGAN DAN PEMBUATAN SISTEM AQUISISI DATA BERBASIS WEB MENGGUNAKAN MIKROKONTROLLER ATmega16



Disusun Oleh :

ADITYA HENDRI MARSENO

NIM. 98.17.037

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2010**

1341112

MANAJEMEN SUMBER MANUSIA DAN MANAJEMEN
MANAJEMEN NEW ORGANISASI ATAU
ORGANISASI MELALUI KONGRES

1341112

MANAJEMEN SUMBER MANUSIA

1341112

1341112 SUMBER MANUSIA
MANAJEMEN SUMBER MANUSIA
MANAJEMEN SUMBER MANUSIA
MANAJEMEN SUMBER MANUSIA

1341112

LEMBAR PERSETUJUAN

PERANCANGAN DAN PEMBUATAN SISTEM AQUISISI DATA BERBASIS WEB MENGGUNAKAN MIKROKONTROLLER ATmega16

SKRIPSI

*Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Teknik Pada
Jurusan Teknik Elektro S-1 Konsentrasi Teknik Elektronika*

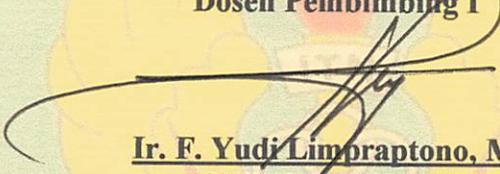
Disusun Oleh :

ADITYA HENDRI MARSENO

NIM : 98.17.037

Diperiksa dan Disetujui

Dosen Pembimbing I


Ir. F. Yudi Limpraptono, MT
NIP.Y.1039500274

Mengetahui

Ketua Jurusan Teknik Elektro S-1


Ir. F. Yudi Limpraptono, MT
NIP.Y.1039500274

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL
MALANG**

2010



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Raya Karanglo Km.2
Malang

BERITA ACARA UJIAN SKRIPSI

Nama : **ADITYA HENDRI MARSENO**
N.I.M : **98.17.037**
Jurusan : **TEKNIK ELEKTRO S-1**
Konsentrasi : **TEKNIK ELEKTRONIKA S-1**
Judul Skripsi : **PERANCANGAN DAN PEMBUATAN SISTEM AQUISISI
DATA BERBASIS WEB MENGGUNAKAN
MIKROKONTROLLER ATMega16**

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1)
pada :

Hari : **Selasa**
Tanggal : **6 Oktober 2009**
Dengan Nilai : **78,25 (B+)** *BY*

Panitia Majelis Penguji :



Ketua

Ir. Sidik Noertjahjono, MT
NIP.Y. 1028700163

Sekretaris

Ir. F. Yudi Limpraptono, MT
NIP.Y. 1039500274

Anggota Penguji :

Penguji I

Ir. TH. Mimien Mustikowati
NIP.P. 1030000352

Penguji II

Eko Nurcahyo
NIP.Y. 102800172



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Raya Karanglo Km.2
Malang

FOMULIR PERBAIKAN SKRIPSI

Nama : Aditya Hendri Marseno
NIM : 98.17.037
Masa Bimbingan : 4 Juli 2009 – 4 Januari 2010
Judul Skripsi : PERANCANGAN DAN PEMBUATAN SISTEM AQUISISI
DATA BERBASIS WEB MENGGUNAKAN
MIKROKONTROLLER ATMega16

NO.	TANGGAL	URAIAN	PARAF
1	6 Oktober 2009	ABSTRAKSI DIPERJELAS	
2	6 Oktober 2009	NO INDEKS DISESUIAIKAN	
3	6 Oktober 2009	GAMBAR RANGKAIN	
4	6 Oktober 2009	DATA SHEET MAXIM	

Disetujui,

Penguji I

Ir. TH. Mimien Mustikowati
NIP.P. 1030000352

Penguji II

Eko Nurcahyo
NIP.Y. 102800172

Mengetahui,

Dosen Pembimbing

Ir. F. Yudi Limpraptono, MT
NIP.Y. 1039500274

ABSTRAKSI

**Penulis : Aditya Hendri Marseno
NIM : 98.17.037**

PERENCANAAN DAN PEMBUATAN SISTEM AQUISISI DATA BERBASIS WEB MENGGUNAKAN MIKROKONTROLLER ATMega16

Teknologi informasi berkembang pesat dewasa ini, teknologi ini memerlukan perangkat yang dapat digunakan untuk berbagai keperluan. Salah satu perkembangan teknologi informasi adalah web browser kita dapat mengakses informasi secara langsung meskipun dari tempat yang jauh dan terpencil.

Bagaimana cara membuat sistem aqisisi data berbasis web browser menggunakan ATMega16. Rangkaian inputan berupa potensiometer yang akan mengeluarkan data analog yang akan diterima ADC, data analog akan diproses dan dijadikan data digital. Data dari ATMega16 akan dikirim ke PC melalui modul WIZnet EG-SR-7150MJ. Dan PC akan menampilkan kedalam bentuk web browser.

Dengan adanya sistem aqisisi data berbasis menggunakan ATMega16 dapat mempermudah kerja kita dalam mengaqisisi data dari tempat yang jauh dan terpencil serta dapat dilakukan secara berulang-ulang, hasil tersebut dapat kita simpan kedalam PC.

Hasil pengujian menunjukkan bahwa sistem dapat berjalan dengan baik dan sesuai yang diharapkan.

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat ALLAH SWT. karena berkat Rahmat dan Hidayah_Nyalah Skripsi dan Tugas Akhir ini dapat diselesaikan dengan baik.

Pada awal pelaksanaan sampai pada penyelesaian skripsi ini, penulis telah banyak memperoleh bimbingan, bantuan dan saran yang sangat bermanfaat, baik secara moril maupun materiil dari berbagai pihak. Untuk itu pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih kepada:

1. Prof. Dr. Ir. Abraham Lomi, MSEE, selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Sidik Noertjahjono, MT, selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Ir. F Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro Program Studi Teknik Elektronika Institut Teknologi Nasional Malang.
4. Bapak Ir. F Yudi Limpraptono MT, selaku Dosen Pembimbing, yang telah banyak meluangkan waktu, pikiran, arahan dan tuntunan yang progresif dalam penyusunan skripsi ini.
5. Bapak Ir. Yusuf Ismail Nahkoda MT, selaku Sekretaris Jurusan Teknik Elektro Program Studi Teknik Elektronika Institut Teknologi Nasional Malang.
6. Seluruh rekan-rekan yang telah membantu penulis dalam menyelesaikan skripsi ini.

Akhirnya penulis menyadari bahwa penulisan skripsi ini masih ada kekurangan-kekurangan, namun demikian penulis mengharapkan pandangan-pandangan dan saran-saran untuk menyempurnakan dalam pembuatan alat serta dalam penulisan skripsi ini, baik dari pihak mahasiswa Teknik Elektronika khususnya, maupun dari dari pihak lain yang menggunakan skripsi ini sebagai bahan acuan dan referensi.

Semoga ALLAH SWT. Selalu melimpahkan Taufiq dan Hidayah_Nya kepada semua pihak yang telah membantu dalam penyusunan skripsi ini.

Malang, Oktober 2009

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	i
BERITA ACARA SKRIPSI.....	ii
ABSTRAKSI.....	iii
KATA PENGANTAR.....	iv
BAB I PENDAHULUAN.....	1
1.1. LATAR BELAKANG.....	1
1.2. RUMUSAN MASALAH	2
1.3. TUJUAN.....	2
1.4. BATASAN MASALAH.....	2
1.5. METODOLOGI PENELITIAN.....	3
1.6. SISTEMATIKA PEMBAHASAN.....	4
BAB II DASAR TEORI.....	5
2.1. PENDAHULUAN.....	5
2.2. MIKROKONTROLER ATMega16.....	5
2.2.1. ARSITEKTUR ATMega16	6
2.2.2. FITUR ATMega16	8
2.2.3. KONFIGURASI PIN ATMEGA16	8
2.2.4. PETA MEMORI	10
2.2.5. STATUS REGISTER.....	12
2.3. ADC INTERNAL DALAM ATMega16.....	14
2.3.1. BLOK DIAGRAM ADC.....	15

2.3.2. FITUR ADC.....	16
2.4. RS 232	16
2.4.1. KOMUNIKASI SERIAL ASINKRON.....	17
2.4.2. KOMUNIKASI SERIAL SINKRON.....	18
2.4.3. TRANSMISI DATA SERIAL.....	19
2.4.4. TRANSMISI DATA SERIAL ASINKRON	20
2.5. WIZNET EG-SR-7150MJ	23
2.6. TCP/IP	27
2.7. HTTP.....	33
2.7.1. FORMAT HTTP	35
2.8. TOPOLOGI JARINGAN	37
2.9. OSI 7 LAYER.....	38
2.10. BASCOM.....	40

BAB III PERENCANAAN DAN PEMBUATAN ALAT 41

3.1. PERENCANAAN PERANGKAT KERAS (HARDWARE)	41
3.1.1. BLOK DIAGRAM.....	42
3.1.2. INPUT ANALOG	43
3.1.3. ATMega16.....	43
3.1.4. RS 232	44
3.1.5. WIZNET EG-SR-7150MJ	45
3.1.6. SWITCH.....	48
3.1.7.PC.....	49

3.2. PERANCANGAN PERANGKAT LUNAK (SOFTWARE).....	49
3.2.1. FLOW CHART UNTUK ATMega16.....	50
BAB IV PENGUKURAN DAN PENGUJIAN.....	52
4.1. PENGUJIAN MODUL WIZNET EG-SR-7150MJ.....	52
4.1.1. PENGUJIAN TCP/IP.....	52
4.1.2. HASIL PENGUJIAN TCP/IP.....	53
4.2. PENGUJIAN TAMPILAN PADA WEB BROWSER.....	54
BAB V KESIMPULAN DAN SARAN.....	55
5.1. KESIMPULAN.....	55
5.2. SARAN.....	55
DAFTAR PUSTAKA.....	XX
LAMPIRAN.....	XXX

DAFTAR GAMBAR

1. GAMBAR ARSITEKTUR ATMega16	6
2. GAMBAR KONFIGURASI PIN ATMega16.....	9
3. GAMBAR PETA MEMORI DATA AVR.....	11
4. GAMBAR MEMORY PROGRAM AVR	12
5. GAMBAR BLOK DIAGRAM ADC.....	15
6. GAMBAR SINKRONISASI AWAL AKHIR	17
7. GAMBAR ALIRAN DATA SINKRON	19
8. GAMBAR TRANSMISI DATA SERIAL	20
9. GAMBAR TRANSMISI DATA SERIAL ASINKRON.....	21
10. GAMBAR KOMUNIKASI HALF DUPLEX.....	22
11. GAMBAR KOMUNIKASI FULL DUPLEX.....	22
12. GAMBAR WIZNET EG-SR-7150MJ	23
13. GAMBAR OPERASI SERVER MODE.....	25
14. GAMBAR OPERASI CLIENT MODE	26
15. GAMBAR BLOK DIAGRAM SISTEM	42
16. GAMBAR WIZNET EG-SR-7150MJ	46
17. GAMBAR KONFIGURASI WIZNET EG-SR-7150	47
18. GAMBAR APLIKASI WIZNET EG-SR-7150	48
19. GAMBAR SISTEM JARINGAN	48
20. GAMBAR FLOW CHART.....	50
21. GAMBAR BLOK PENGUJIAN TCP/IP	52
22. GAMBAR HASIL PENGUJIAN TCP/IP	53

23. GAMBAR HASIL TAMPILAN PADA WEB BROWSER.....54

DAFTAR TABEL

1. TABEL SPESIFIKASI WIZNET EG-SR-7150MJ.....	24
2. TABEL KLASIFIKASI <i>IP</i>	29
3. TABEL CONTOH RESPON GET.....	36
4. TABEL METODE REQUEST.....	36
5. TABEL FUNGSI DARI OSI 7 LAYER	39

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi saat ini sangat cepat dan semakin berkembang pesat sesuai dengan kebutuhan manusia. Berdasarkan perkembangan teknologi ini kita bisa mengakses informasi dari jarak jauh, misalkan kita ingin mengetahui suatu kondisi ditempat yang jauh. Suatu sistem di internet yang memungkinkan siapapun agar bisa menyediakan informasi.

Disini kita memerlukan suatu sistem perekam data yang berfungsi untuk merekam suatu kejadian, misalnya gempa bumi disuatu daerah , polusi udara, suhu atau kelembaban suatu tempat, ketinggian air pada suatu bendungan dan masih banyak kejadian-kejadian lainnya yang dapat direkam dan ditampilkan secara langsung dari jarak jauh.

Dalam memperoleh informasi secara langsung diperlukan suatu jaringan yang terdiri dari PC yang dilengkapi *modem*, *internet account* dan *software internet* seperti *internet explorer*, *mozilla* yang berfungsi sebagai *search engine*. Jaringan ini dapat mengirim dan menerima data dengan sendirinya setelah diprogram sesuai dengan kebutuhan yang diperlukan.

Suatu komputer yang terkoneksi dengan jaringan menjadi satu node dari jaringan tersebut. Selain yang bukan komputer juga dapat menjadi *node* sepanjang mereka berkomunikasi menggunakan jaringan dengan mengirim dan menerima data terhadap *node-node* yang lain.

Istilahnya adalah *host* dimana komputer yang terkoneksi ke jaringan yang dapat memberikan layanan jaringan (*network service*). Jaringan mempunyai dua model yaitu *Peer To Peer* dan *Client/Server*.

Peer to Peer adalah jaringan dimana *host* dapat memberi data pada *peer* lain dan juga mengambil layanan dari *peer* lain.

Client/Server model ini memisahkan secara jelas mana yang dapat memberikan layanan jaringan dan mana pula yang hanya menerima layanan tersebut.

Dengan menggunakan teknologi tersebut informasi dapat diakses selama 24 jam.

Hal diatas menjadi dasar pada penulisan skripsi ini untuk menciptakan suatu sistem aquisisi data berbasis *web* menggunakan ATMega16.

1.2 Rumusan Masalah

Bagaimana merancang dan membuat suatu sistem aquisisi data berbasis *web* menggunakan ATMega16.

1.3 Tujuan

Merencanakan dan merealisasikan suatu sistem aquisisi data berbasis *web* menggunakan ATMega16.

1.4 Batasan Masalah.

Adapun batasan masalah dalam penulisan ini adalah :

1. Input analog 0 – 5Volt DC/AC.

2. ADC menggunakan ADC 8-bit internal yang terdapat pada ATmega16..
3. Menggunakan Mikrokontroler ATmega16.
4. Bahasa pemrograman menggunakan bahasa BAS COM
5. Converter serial ke Ethernet yang digunakan adalah EG-SR 7150MJ.

1.5 Metodologi Penulisan.

Metodologi yang digunakan dalam penelitian ini sebagai berikut :

1. Studi Literatur.

Mempelajari tentang teori-teori yang berhubungan dengan skripsi ini seperti jaringan *network*, *web browser*, dan mempelajari tentang komponen-komponen yang akan dipakai dalam rancangan ini.

2. Perencanaan Dan Pembuatan.

Disini akan mendesain rangkaian sistem aquisisi data berbasis web menggunakan ATmega16 (*hardware*) dan pembuatan program menggunakan bahasa BASCOM (*software*).

3. Implementasi Dan Pengujian Alat.

Melakukan beberapa pengujian dengan memasukkan inputan analog dan melihat hasilnya, sampai berjalan sesuai dengan yang diharapkan.

4. Kesimpulan.

Pengambilan kesimpulan melalui evaluasi tahap akhir serta kelebihan dan kekurangan alat.

1.6 Sistematika Pembahasan

Sistematika pembahasan dari skripsi ini terdiri dari pokok pembahasan yang saling berkaitan antara satu dengan yang lainnya, yaitu:

BAB I PENDAHULUAN

Pada bab ini akan dibahas tentang latar belakang, rumusan masalah, tujuan, batasan masalah dan sistematika penulisan.

BAB II DASAR TEORI

Pada bab ini akan membahas tentang teori-teori yang mendukung dalam perencanaan dan pembuatan sistem.

BAB III PERENCANAAN DAN PEMBUATAN ALAT

Pada bab ini akan dijelaskan tentang perencanaan dan pembuatan sistem aquisis data berbasis web menggunakan ATmega16 (*hardware* dan *software*).

BAB IV PENGUJIAN ALAT

Pada bab ini membahas tentang proses serta hasil dari pengujian alat, yang didasarkan oleh pengukuran-pengukuran.

BAB V PENUTUP

Berisikan kesimpulan yang didapat selama perancangan dan pembuatan alat serta saran-saran.

BAB II

DASAR TEORI

2.1. Pendahuluan.

Dalam Bab II ini akan membahas teori-teori yang akan dipergunakan dalam “Perencanaan Dan Pembuatan Sistem Aquisisi Data Berbasis Web Menggunakan Microkontroller ATmega16”, yang meliputi :

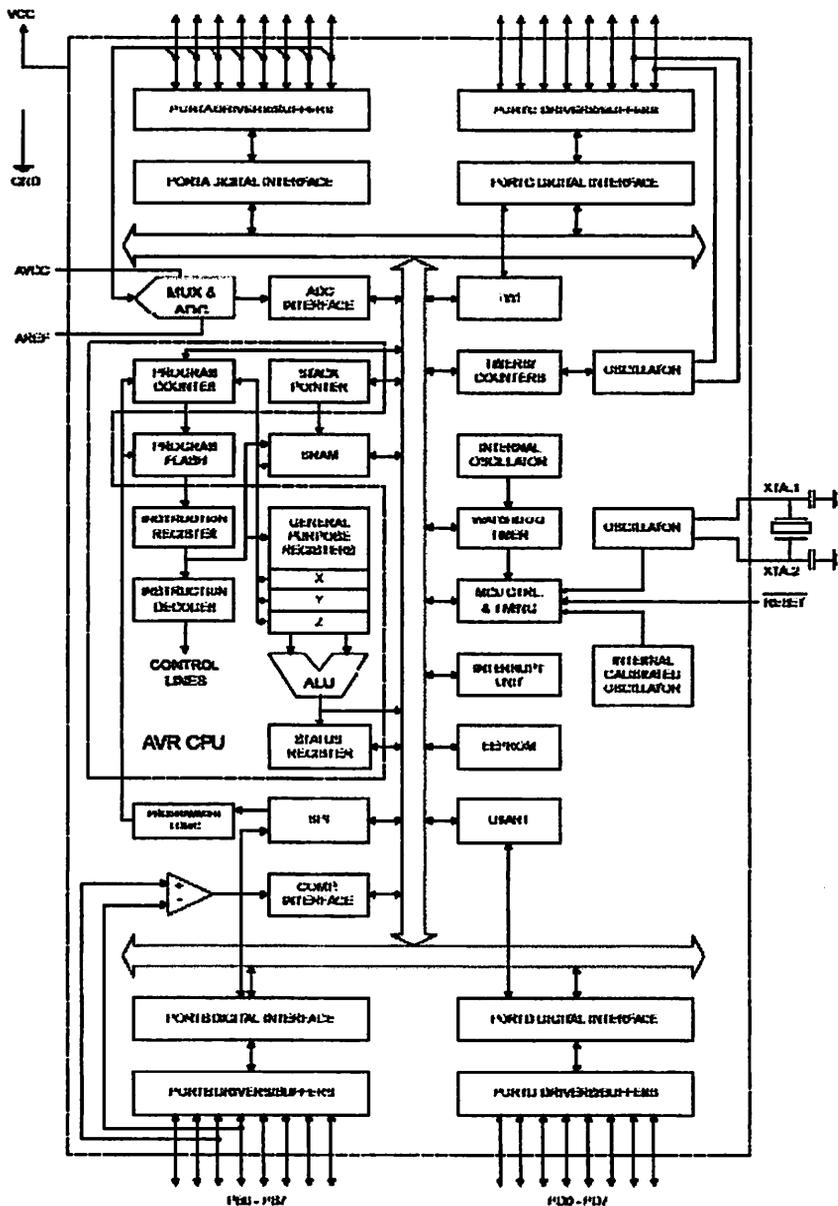
1. Mikrokontroller ATMEGA16
2. RS-232
3. Wiznet EG-SR-7150MJ
4. *TCP/IP(Transport Control Protocol/Internet Protocol).*
5. *HTTP(Hypertext Transfer Protocol).*
6. *OSI 7 Layer.*
7. Topologi Jaringan Komputer.
8. BASCOM sebagai bahasa pemrograman.

2.2. Mikrokontroler ATMEGA16

Mikrokontroller 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. Secara umum, AVR dapat dikelompokkan menjadi 4 kelas, yaitu ATtiny, keluarga AT90Sxx, keluarga ATmega dan AT86RFxx. Pada dasarnya yang membedakan masing-masing kelas adalah

memori, peripheral dan fungsinya. Dari segi arsitektur dan instruksi yang digunakan, mereka bias dikatakan hampir sama.

2.2.1. Arsitektur ATmega16



Gambar 2.1 Blok diagram Fungsional ATmega16 [1]

Dari gambar tersebut dapat dilihat bahwa ATmega16 memiliki bagian sebagai berikut

1. Saluran I/O sebanyak 32 buah, yaitu Port A, Port B, Port C dan Port D.
2. ADC 10 bit sebanyak 8 saluran.
3. Tiga buah *Timer/Counter* dengan kemampuan pembanding.
4. CPU yang terdiri atas 32 buah register.
5. *Watchdog Timer* dengan osilator internal.
6. SRAM sebesar 512byte.
7. Memori *Flash* sebesar 16Kbyte dengan kemampuan *Read While Write*.
8. Unit interupsi internal dan eksternal.
9. Port antarmuka SPI.
10. *EEPROM* sebesar 512 byte yang dapat diprogram saat operasi.
11. Antarmuka komparator analog.
12. Port *USART* untuk komunikasi serial.

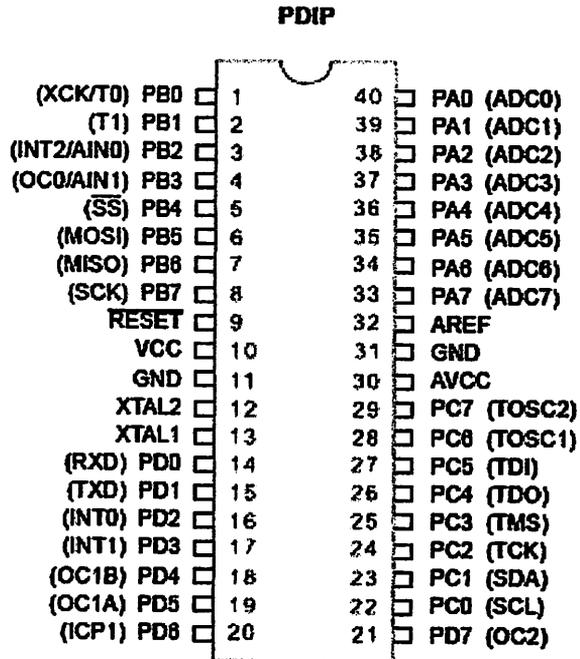
2.2.2. Fitur ATmega16

Kapabilitas detail dari ATmega16 adalah sebagai berikut :

1. Sistem mikroprosesor 8 bit berbasis RISC dengan kecepatan maksimal 16 MHz.
2. Kapabilitas memori *flash* 8 KB, SRAM sebesar 512 *byte* dan *EEPROM*(*Electrically Erasable Programmable Read Only Memory*) sebesar 512 *byte*.
3. ADC internal dengan fidelitas 10 bit sebanyak 8 *channel*.
4. Portal komunikasi serial (USART) dengan kecepatan maksimal 2,5 Mbps.
5. 6 pilihan *mode sleep* menghemat penggunaan daya listrik.

2.2.3. Konfigurasi Pin ATmega16

Konfigurasi pin ATmega16 bisa dilihat pada gambar sebelumnya. Dari gambar tersebut dapat dijelaskan secara fungsional konfigurasi pin ATmega16 sebagai berikut :



Gambar 2.2 Konfigurasi PIN ATMega16 [1]

1. VCC berfungsi masukan catu daya.
2. GND merupakan *ground*.
3. Port A(PA0...PA7) merupakan pin I/O dua arah dan inputan ADC.
4. Port B(PB0...PB7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu Timer/Counter, komparator analog dan SPI.
5. Port C(PC0...PC7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu TWI, komparator analog dan Time Oscilator.

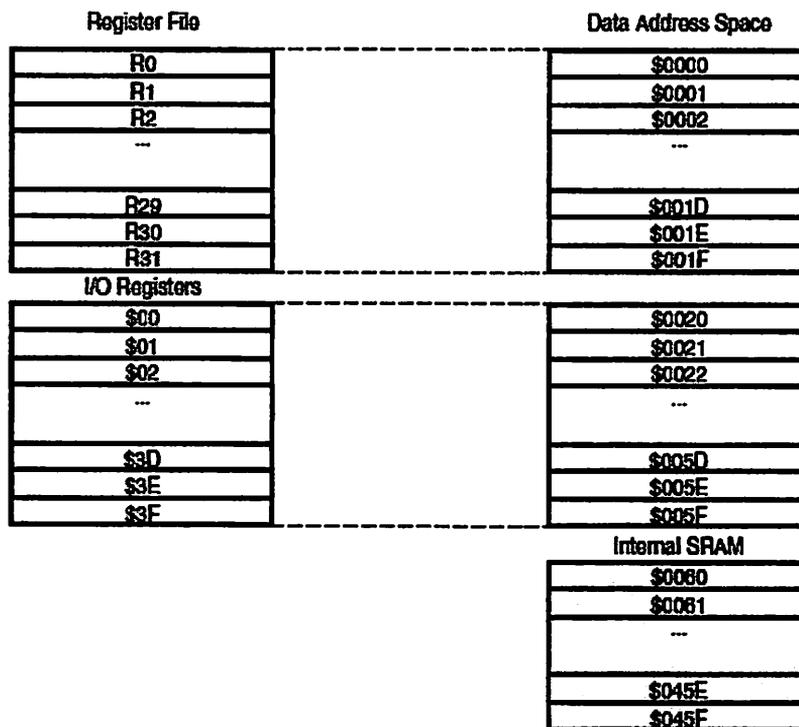
6. Port D(PD0...PD7) merupakan pin I/O dua arah dan pin fungsi khusus, yaitu komparator analog, interupsi eksternal dan komunikasi serial.
7. RESET merupakan pin yang digunakan untuk me-reset mikrokontroller.
8. XTAL1 dan XTAL 2 merupakan pin masukan *clock* eksternal.
9. AVCC merupakan pin masukan tegangan untuk ADC.
10. AREF merupakan pin masukan tegangan referensi ADC.

2.2.4. Peta Memori

AVR ATMega16 memiliki ruang pengamatan memori data dan memori program yang terpisah. Memori data terbagi menjadi 3 bagian, yaitu 32 buah register umum, 64 buah register I/O dan 512 *byte SRAM internal*. Register keperluan umum menempati space data pada alamat terbawah, yaitu \$00 sampai \$1F.

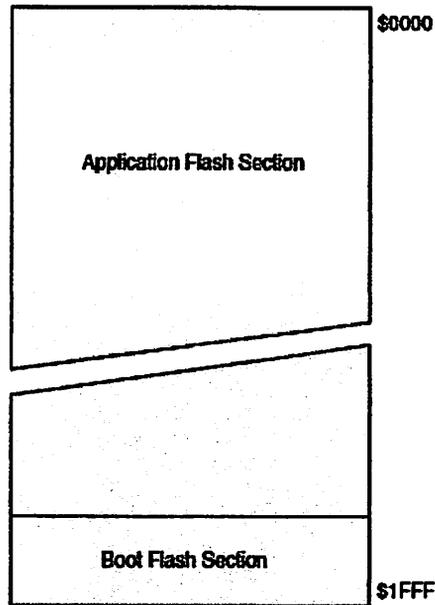
Sementara itu register khusus untuk menangani I/O dan control terhadap mikrokontroller menempati 64 alamat berikutnya, yaitu mulai dari \$20 hingga \$5F. Register tersebut merupakan register yang khusus digunakan untuk mengatur fungsi terhadap berbagai *peripheral* mikrokontroller, seperti *control register*, *timer/counter*, fungsi-fungsi I/O dan sebagainya.

Alamat memori berikutnya digunakan untuk SRAM 512 *byte*, yaitu pada lokasi \$60 sampai dengan \$25F. konfigurasi memori data ditunjukkan pada gambar di berikut ini.



Gambar 2.3 Peta Memori Data AVR ATmega16 [1]

Memori program yang terletak dalam *Flash PEROM* tersusun dalam *word* atau 2 *byte* karena setiap instruksi memiliki lebar 16-bit atau 32-bit. AVR ATmega 16 memiliki 4KbyteX16-bit *Flash PEROM* dengan alamat mulai dari \$000 sampai dengan \$FFF. AVR tersebut memiliki 12-bit *Program Counter(PC)* sehingga mampu mengamati isi *Flash*.



Gambar 2.4 Memory Program AVR ATmega16 [1]

2.2.5. Status Register (SREG)

Status register adalah register berisi status yang dihasilkan pada setiap operasi yang dilakukan ketika suatu instruksi dieksekusi. SREG merupakan bagian dari inti CPU mikrokontroler.

a. **Bit 7-1 : *Global Interrupt Enable.***

Bit harus di set meng-enable interupsi. Setelah itu, anda dapat mengaktifkan interupsi mana yang akan anda gunakan dengan cara meng-enable bit control register yang bersangkutan secara individu. Bit akan di-clear apabila terjadi suatu interupsi yang dipicu oleh hardware, dan bit tidak akan mengijinkan terjadinya interupsi, serta akan diset kembali oleh intruksi RETI.

b. **Bit 6-T : *Bit Copy Storage***

Instruksi BLD dan BST menggunakan bit-T sebagai sumber atau tujuan dalam operasi bit. Suatu bit dalam sebuah register GPR dapat disalin ke bit menggunakan instruksi BST, dan sebaliknya bit-T dapat disalin kembali ke suatu bit dalam register GPR menggunakan instruksi BLD.

c. **Bit 5-H : *Half Carry Flag***

d. **Bit 4-S : *Sign Bit***

Bit-S merupakan hasil operasi *EOR* antara *flag-N(-)* dan *flag V(komplemen dua overflow)*

e. **Bit 3- V : *Two's Complement Overflow Flag***

Bit berguna untuk mendukung operasi aritmatika.

f. **Bit 2-N : *Negative flag***

Apabila suatu operasi menghasilkan bilangan negative, maka *flag - N* akan diset.

g. **Bit 1-Z : *Zero Flag***

Bit akan diset bila hasil operasi yang diperoleh adalah nol.

h. **Bit 0-C : *Carry Flag***

Apabila suatu operasi menghasilkan *carry*, maka bit akan diset.

2.3. ADC Internal Didalam ATmega16

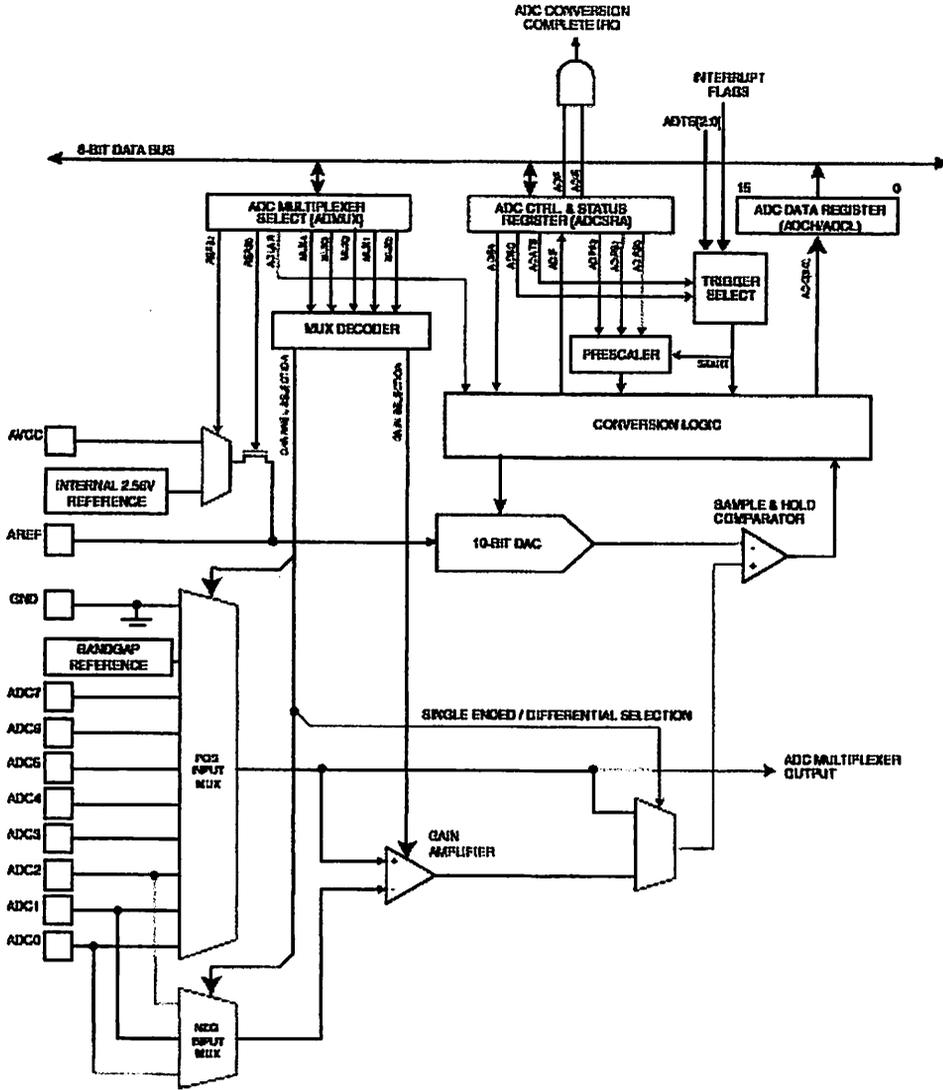
ADC digunakan untuk mengubah input analog (tegangan kontinyu) menjadi data digital (ex: 10110011) atau binary. Pada uC ATmega16 sudah terintegrasi ADC 8 input dengan resolusi sebesar 10 bit didalamnya sehingga kita bisa memanfaatkannya tanpa menggunakan IC ADC tambahan + menghemat pengeluaran. ADC pada uC ATmega16 berada di PORTA.

Pada skripsi ini input menggunakan potensiometer sebagai simulasi tegangan dari sensor atau bisa juga misalkan anda menggunakan sensor suhu LM35 sebagai input.

Sinyal input dari pin ADC akan dipilih oleh multiplexer untuk diproses oleh ADC, karena converter ADC dalam chip hanya satu buah sedangkan saluran inputnya ada delapan maka dibutuhkan multiplexer untuk memilih input pin ADC secara bergantian.

ADC mempunyai rangkaian untuk mengambil sample dan menahan (hold) tegangan input ADC sehingga dalam keadaan konstan selama proses konversi. ADC mempunyai catu daya yang terpisah yaitu pin AVCC-AGND. AVCC tidak boleh berbeda $\pm 0,3V$ dari Vcc.

2.3.1. Blok Diagram ADC Internal ATmega16



Gambar 2.5 Blok Diagram ADC Internal ATmega16 [2]

2.3.2. Fitur Yang Terdapat Pada ADC

- Resolusi mencapai 10-Bit
- 0.5 LSB Integral Non-linearity
- Akurasi mencapai ± 2 LSB
- Waktu konversi 13 - 260 μ s
- Resolusi maksimal sampai 15 kSPS
- 8 saluran ADC yang dapat digunakan secara bergantian
- 0 - VCC ADC Input Range
- Disediakan 2.56V tegangan referensi ADC
- Mode konversi kontinyu (free running)
- Interupsi ADC Complete
- Sleep Mode Noise Canceler

2.4. RS 232

Akses data dengan menggunakan port serial memiliki beberapa kelebihan dibandingkan dengan menggunakan port paralel. Kelemahan pada komunikasi paralel adalah banyaknya kabel yang harus terhubung. Hal ini menjadi tidak efisien bila kita menggabungkan alat ke komputer (*interfacing*) untuk jarak yang jauh. Oleh karena itu digunakan komunikasi serial untuk mengatasinya.

Pada komunikasi serial, data yang dikirim per bit (data tersebut antri), walaupun mempunyai kelemahan pengiriman data yang lebih lambat dibanding dengan komunikasi paralel, komunikasi serial bisa digunakan untuk jarak yang

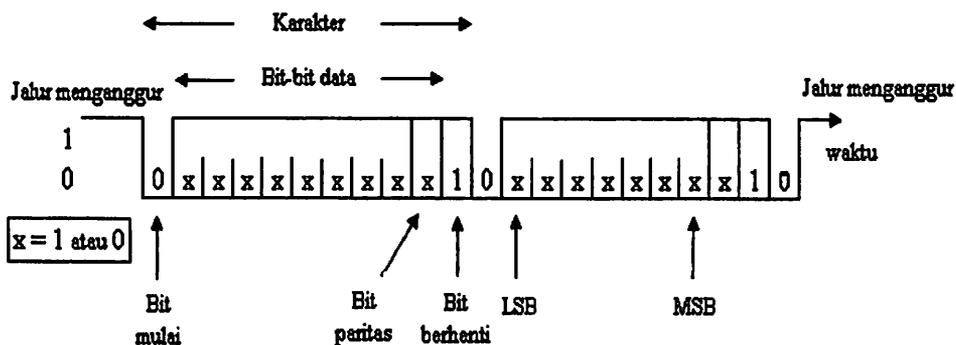
jauh. Berdasarkan formatnya, pengiriman data komunikasi serial dibedakan menjadi dua bentuk yaitu :

2.4.1. Komunikasi Serial Asinkron

Pada pengiriman data asinkron (tak sinkron), setiap karakter dikirimkan sebagai satu kesatuan (*entity*) bebas, yang berarti bahwa waktu antara pengiriman bit terakhir dari sebuah karakter dan bit pertama dari karakter berikutnya tidak tetap.

Pengiriman data tak sinkron lebih sederhana daripada pengiriman sinkron karena hanya isyarat data saja yang dikirimkan.

Detak penerima dibangkitkan secara lokal di dalam penerima dan tetap dijaga agar sesuai dengan detak pengirim yang menggunakan bit awal (*start bit*) dan bit akhir (*stop bit*) yang dikirimkan dengan setiap karakter. Penyesuaian detak pengirim dan penerima terjadi per karakter.



Gambar 2.6 : Sinkronisasi Awal Akhir [3]

Gambar menunjukkan sinkronisasi awal-akhir, bit awal dan akhir tidak membawa informasi, tetapi hanya menunjukkan awal dan akhir setiap karakter. Dari gambar dapat dilihat bahwa bit kedelapan disebut bit paritas, diikutsertakan dalam bentuk gelombang tersebut.

Bit ini akan dipasang pada 1 atau 0 untuk meyakinkan cacah bit pada setiap karakter adalah genap untuk paritas genap, atau ganjil untuk paritas ganjil.

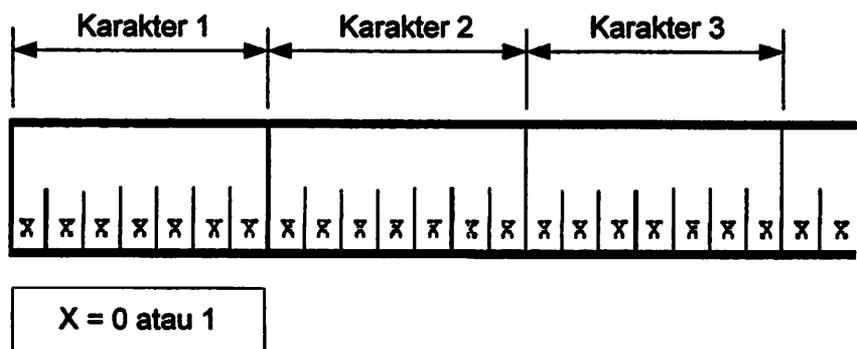
2.4.2. Komunikasi Serial Sinkron

Pada pengiriman data sinkron sejumlah blok data dikirimkan secara kontinyu tanpa bit awal atau akhir. Detak atau *clock* pada penerima dioperasikan secara kontinyu dan dikunci agar sesuai dengan detak pada pengirim.

Untuk mendapat keadaan yang sesuai, informasi pendetakan harus dikirimkan lewat jalur bersama-sama dengan data dan memanfaatkan metode penyandian tertentu sehingga informasi pendekatan dapat diikutsertakan.

Data secara kontinyu menunjukkan akan dikirimkan terus menerus tanpa adanya pembatas. Interval waktu antara bit terakhir dari suatu karakter dengan bit pertama dari karakter berikutnya adalah nol atau kelipatan bulat dari periode waktu yang diperlukan untuk mengirimkan sebuah karakter.

Jika pada data yang dikirimkan terdapat pembatas, pengirim akan menambah *byte* tambahan untuk mengganti pembatas tersebut. Sehingga tidak diperlukan adanya bit awal dan bit akhir.



Gambar 2.7 : Aliran Data Sinkron [3]

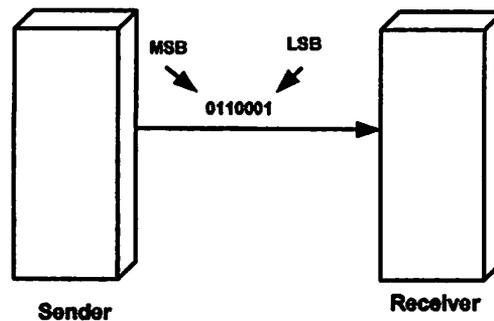
Gambar menunjukkan aliran bit sinkron. Penerima harus memulai pencacahan pada tengah-tengah bit pertama dari karakter pertama, jika tidak akan timbul kesalahan pada isyarat yang diterima. Setelah penyesuaian bit, penerima harus tahu pada kelompok mana bit tersebut akan membentuk karakter (penyesuaian karakter).

Penerima harus memantau data yang diterima setiap bit sampai penerima mengenali pola karakter sinkronisasi. Sehingga penerima dapat mengetahui himpunan bit mana yang membentuk karakter yang pertama dikirimkan. Karakter berikutnya dengan mudah dapat dikirimkan dan dikenali.

2.4.3. Transmisi Data Serial

Transmisi data serial adalah transmisi data yang pengiriman dan penerimaan datanya berurutan tiap bitnya. Jadi kita hanya membutuhkan satu saluran untuk mengirimkan data antar dua perangkat komunikasi.

Keuntungan transmisi serial yaitu biaya lebih murah karena hanya membutuhkan satu saluran saja sehingga banyak digunakan untuk komunikasi jarak jauh. Sedangkan kerugiannya adalah kecepatan pengiriman data lebih rendah dibandingkan dengan transmisi paralel.



Gambar 2.8: Transmisi Data Serial [3]

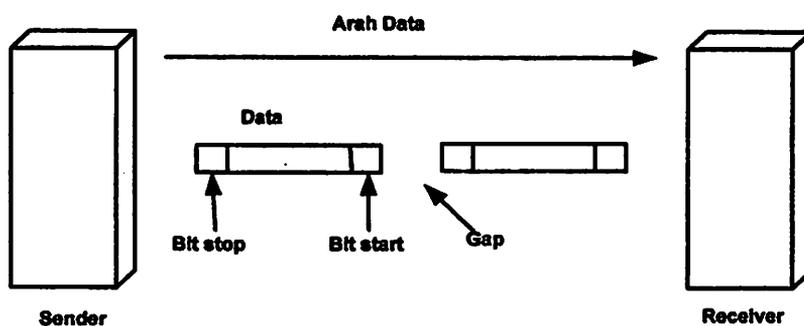
Pengiriman data akan dimulai dari LSB (*Least Significant Bit*) dan diakhiri dengan MSB (*Most Significant Bit*). Setiap karakter yang dikirimkan, disusun sesuai dengan suatu urutan dengan bit tertentu. Berdasarkan formatnya, salah satu jenis komunikasi serial yaitu :

2.4.4. Transmisi Data Serial Asinkron

Pada pengiriman data asinkron (tidak sinkron), setiap karakter dikirimkan sebagai satu kesatuan (*entity*) bebas yang berarti bahwa waktu antara pengiriman bit terakhir dari sebuah karakter dan bit pertama dari sebuah karakter berikutnya tidak tetap.

Pengiriman data asinkron lebih sederhana daripada pengiriman sinkron karena hanya isyarat data saja yang dikirimkan. Detak penerima dibangkitkan secara lokal di dalam penerima dan tetap dijaga agar sesuai dengan detak pengirim yang menggunakan bit awal (*start bit*) dan bit akhir (*stop bit*) yang dikirimkan setiap karakter dan data yang satu dengan data selanjutnya dipisahkan gap.

Penyesuaian detak pengiriman dan penerima terjadi karakter per karakter.



Gambar 2.9 : Transmisi Data Serial Asinkron [3]

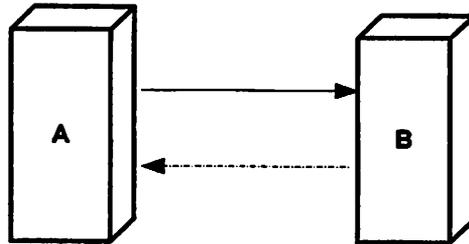
Gambar di atas menunjukkan sinkronisasi awal-akhir, bit awal dan akhir tidak membawa informasi, tetapi hanya menunjukkan awal dan akhir setiap karakter. Dari gambar dapat dilihat bahwa bit kedelapan disebut bit paritas, diikutsertakan dalam bentuk gelombang tersebut.

Bit ini akan dipasang pada 1 atau 0 untuk meyakinkan cacah bit pada setiap karakter adalah genap untuk paritas genap, atau ganjil untuk paritas ganjil.

Berdasarkan arah komunikasinya, pengiriman data serial dibedakan menjadi :

- **Half Duplex**

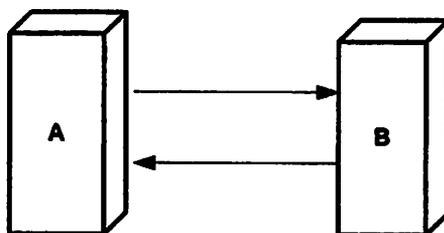
Merupakan sistem komunikasi yang mengirimkan data dalam satu arah. Sistem ini tidak dapat mengirimkan data secara bersamaan sehingga perlu saling menunggu secara bergantian untuk berkomunikasi. Pada gambar di bawah menunjukkan komunikasi *half duplex*.



Gambar 2.10 : Komunikasi Half Duplex [3]

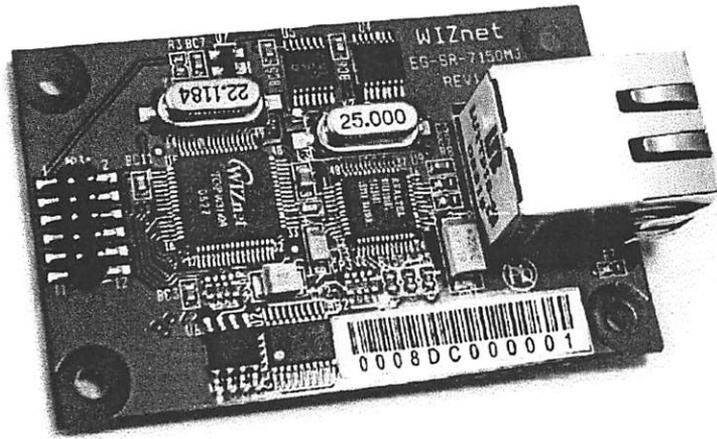
- **Full Duplex**

Merupakan sistem komunikasi yang mengirimkan data dalam dua arah. Sistem ini dapat mengirimkan data secara bersamaan sehingga tidak perlu saling menunggu secara bergantian untuk komunikasi. Gambar di bawah menunjukkan komunikasi *full duplex*.



Gambar 2.11 : Komunikasi Full Duplex [3]

2.5. Wiznet EG-SR-7150MJ



Gambar 2.12 Wiznet EG-SR-7150MJ [4]

Modul ini berfungsi untuk mengkonversi data serial menjadi tipe data *TCP/IP* dan sebaliknya. Penggunaan modul ini memungkinkan mikrokontroler *ATMega 16* dapat berkomunikasi dengan *Ethernet card*.

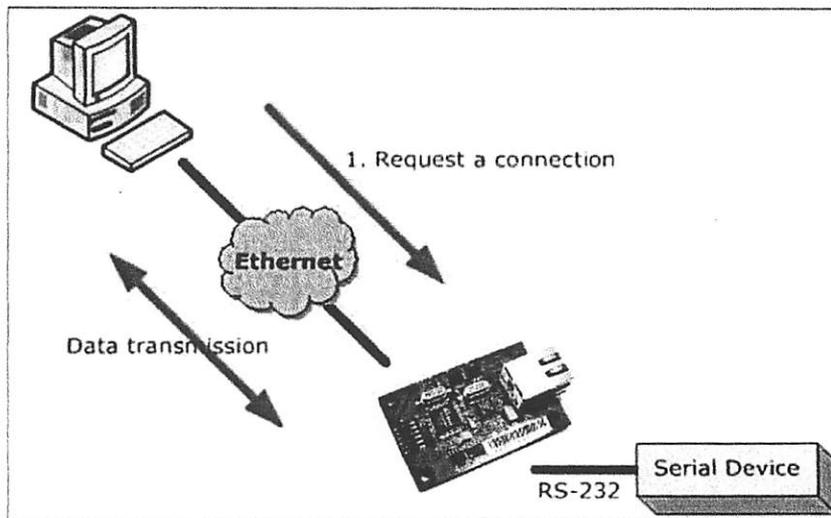
Fitur yang terdapat pada modul ini antara lain *RJ-45* konektor, konektor/pin antarmuka serial, kecepatan maksimal komunikasi serial ialah 230 kbps, dan kecepatan maksimal pada antarmuka *Ethernet* ialah 100Mbps. Protocol yang didukung pada modul ini adalah *TCP*, *UDP*, *IP*, *ARP*, *ICMP*, *MAC*, (*IGMP*, *PPPoE*).

Proses pengkonversian data *TCP/IP* ke data serial ditangani oleh program yang ada didalam IC *AT89C51RC2* yang terpasang pada modul *EG-SR-7150MJ*. Spesifikasi dari *Wiznet EG-SR-7150MJ* ada pada table berikut ini.

Table 2.1 Spseifikasi Wiznet EG-SR-7150MJ [4]

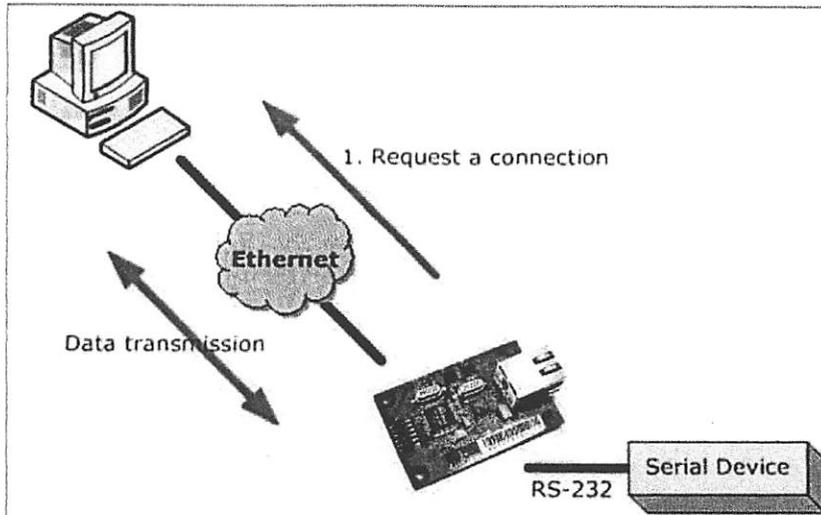
Category	Specifications
Form Factor	2mm Pitch 2x6 pins, 62x40 mm
LAN Interface	10/100 Mbps auto-sensing, RJ-45 connector
Protocol	TCP, UDP, IP, ARP, ICMP, MAC, (IGMP, PPPoE)
CPU	AT89C51RC2 (8bit MCU and 32K Flash)
Serial Interface	RS 232 (LVTTL)
Serial Signals	TXD, RXD, RTS, CTS, GND
Serial Parameters	Parity : None, Even, Odd Data bits : 7, 8 Flow control : RTS/CTS, XON/XOFF Speed : up to 230Kbps
Management	Configuration utility based on Windows
Temperature	0°C~70°C (Operating), -40°C ~85°C (Storage)
Humidity	10~90%
Power	150mA @ 3.3V (max)
Size	40mm x 62mm x 17mm

Mode operasi yang dapat ditetapkan pada modul ini ialah *server mode* dan *client mode*. Pada mode operasi *server mode* PC menjadi *client* dan modul Wiznet EG-SR-7150MJ menjadi *server* yang terhubung pada peralatan yang mempunyai komunikasi serial RS232. Untuk menjalankan aplikasinya, PC harus terlebih dahulu memberikan request dan setelah itu respon akan terjadi pada transmisi data. Berikut ialah gambar hubungan operasi *server mode*.



Gambar 2.13 Operasi *Server Mode* [4]

Untuk mode operasi *client mode* berlaku hubungan sebaliknya. PC menjadi *server* dan modul Wiznet EG-SR-7150MJ menjadi *client* dimana request akan dikirimkan ke PC dan PC akan merespon melalui transmisi data. Berikut ialah gambar hubungan operasi *client mode*.



Gambar 2.14 Operasi *Client Mode* [4]

Pada kedua mode operasi ini hal yang perlu diperhatikan ialah setting *ip address, subnet, gateway dan local port*.

Untuk melakukan setting pada modul Wiznet EG-SR-7150MJ dapat juga dilakukan melalui perintah *serial*. Metode perintah serial dapat dibagi menjadi 2 yaitu *hardware trigger* dan *software trigger*.

Pada *hardware trigger* pemberian perintah dilakukan dengan memberi logika *low* pada pin *H/W trigger* pada modul Wiznet EG-SR-7150MJ. Sedangkan pada *software trigger* pemberian perintah dilakukan dengan mengawali karakter yang dikirim dengan 3 karakter untuk *trigger* yang dapat diseting manual.

2.6. TCP/IP

TCP/IP mempunyai empat lapisan yang sesuai dengan model referensi *OSI* sehingga menjadi model komunikasi terbanyak dipakai. Konsep lapisan ini memudahkan programmer dalam membuat aplikasi jaringan. Tiga lapisan terbawah diatur oleh system operasi, tepatnya oleh *kernel*. Dan layer yang paling atas berupa program atau aplikasi yang dijalankan oleh *user* pada system operasi. Penjelasan fungsi masing-masing lapisan adalah sebagai berikut :

- ***Application Layer***

Disini aplikasi berjalan dengan memanfaatkan lapisan dibawahnya. Aplikasi jaringan akan memilih jenis protocol transport apa yang akan digunakan. Contoh beberapa jaringan yang berjalan pada protocol *TCP/IP*.

- *Telnet*, untuk *Remote Login*
- *FTP (File Transfer Protocol)* untuk mentransfer file
- *SMTP (Simple Mail Transfer Protocol)* untuk email
- *SNMP (Simple Network Management Protocol)* untuk manajemen jaringan.

- ***Transport Layer.***

Lapisan ini memecah data yang dikirim dari layer diatasnya menjadi unit-unit data yang lebih kecil. Tugas lain layer ini adalah membuka dan menutup komunikasi diantara dua computer. Pada

layer ini terdapat dua pilihan protocol yang dapat digunakan. *TCP(Transmission Control Protocol)* dan *UDP(User Datagram Protocol)*

- ***Network Layer***

Tugas utama layer ini adalah bertanggung jawab dalam proses pengiriman paket ke alamat yang benar. Seperti pemilihan rute pengiriman. Macam protokolnya, yaitu *IP(Internet Protocol)* ,*ARP(Address Resolution Protocol)* ,*ICMP(Internet Control Messages Protocol)* dan *IGMP(Internet Group Management Protocols)*.

- ***Link Layer***

Layer terbawah ini berfungsi mengirimkan dan menerima data dari dan ke media jaringan. Dengan kata lain merubah data digital menjadi sinyal listrik atau cahaya dan sebaliknya.

2.6.1. Klasifikasi *Internet Protocol (IP)*

TCP/IP melihat semua *IP address* sebagai dua bagian, jaringan *Network(Network Part)* dan bagian *Host(Host Part)* :

- *Network Part* dari *Address* mengidentifikasi jaringan dimana mesin terkoneksi.

- Bagian lain dari *address-host-part* mengidentifikasi mesin tertentu pada jaringan tersebut.

Berikut adalah table klasifikasi *IP* berdasarkan *range*-nya :

Table 2.2 Klasifikasi *IP* berdasarkan *range*-nya [5]

Class	Range
A	0.0.0.0 to 127.255.255.255
B	128.0.0.0 to 191.255.255.255
C	192.0.0.0 to 223.255.255.255
D	224.0.0.0 to 239.255.255.255
E	240.0.0.0 to 255.255.255.255

Ada dua fungsi dasar fungsi *IP* yaitu *addressing* (pengalamatan) dan *fragmentation* (pemecahan datagram menjadi unit-unit kecil). *IP* menggunakan *field* dalam *internet header* untuk melakukan pemecahan (*fragment*) dan merangkai kembali *datagram* tersebut.

Keterangan :

1. Level 1

- VER(sion) – 4 bit

Menunjukkan versi dari format header yang ada.

- **IHL (*Internet Header Length*)** – 4 bit

Adalah panjang dari internet header dalam 32 *bit word* (1 *byte* = 8 *bits*, 1 *word* = 32 *bits*), IHL ini merupakan tanda awal dari mulainya data dan minimum dari sebuah *header* adalah 5.

- **TOS (*Type Of Service*)** – 8 bits

Merupakan indikator dari parameter tentang kualitas layanan yang dikehendaki, spesifikasi melalui :

- *Precedence*
- *Delay*
- *Throughput*
- parameter-parameter reliabilitas.

- **Total Length** – 16 bits

Adalah panjang datagram yang diukur oleh *octet* termasuk *internet header* dan panjang data yang diijinkan dalam sebuah datagram. Panjang data maksimum adalah 65.535 *oktet*.

2. Level 2

- **Identification** – 16 bits

Ditentukan oleh pengirim data untuk menambah fungsi dalam “merakit”.

- *Flag* – 3 bits

Bit 0 berstatus *reserved* dan harus 0

Bit 1, DF 0 = *May Fragment* 1 = *Don't Fragment*

Bit 2, MF 1 = *Last Fragment* 1 = *More Fragment*

- *Fragment Offset* – 13 bits

Merupakan indikator dimana datagram-datagram ini menjadi anggota, ditentukan oleh octet (64 bits), *fragment* pertama mempunyai offset 0.

3. Level 3

- *TTL (Time To Live)* – 8 bits

Menentukan waktu maksimum dari suatu datagram dapat berada dalam system internet. Jika *field* ini berisi nilai 0, maka datagram harus 'dihancurkan'. Satuan waktunya adalah dalam detik, standar waktu maksimum yang dapat diberikan adalah 255 detik.

- *Protocol* – 8 bits

Menentukan level selanjutnya protocol yang digunakan dalam bagian data dari datagram, yang umum dikenal :

- 1 *ICMP(Internet Control Message Protocol)*
- 3 *Gateway To Gateway Protocol*

- 6 TCP(*Transmission Control Protocol*)
 - 8 RGP(*exterior Gateway Protocol*)
 - 17 UDP(*User Datagram Protocol*)
- *Header Cheksum – 16 bit*

Menentukan jumlah data pada header.

4. Level 4

- *Source IP Address 32 bits*

5. Level 5

- *Destination IP Address 32 bits*

6. Level 6

- *Option (Variable)*

Dapat digunakan dalam datagram. Jenis operasi yang digunakan :

- *Security*
- *Loose Source Routing*
- *Strict Route*

- *Record Route* (untuk tracing)
- *Stream ID* dan *Internet Time Stamp*
- *Padding (Variable)*

Digunakan untuk menentukan batas dari internet header sebanyak 32 bit.

2.7. *HTTP (Hypertext Transfer Protocol)*

HTTP, yang merupakan singkatan dari *Hypertext Transfer Protocol* adalah *protocol application level* (dalam OSI 7 layer) untuk system informasi yang terdistribusi, kolaboratif dan *hypermedia*.

Definisi *hypermedia* sendiri merujuk pada suatu media yang merupakan kombinasi dari berbagai media lainnya. *HTTP* telah digunakan dalam *World Wide Web(web)* sejak tahun 1990. Versi pertamanya, *HTTP/0.9* adalah protocol 'sederhana' yang memperbolehkan transfer data secara *raw* melalui internet.

Spesifikasi protocol ini didefinisikan oleh tim Bernerslee dalam RFC(*Request For Comment* -) 1945 dan digunakan di internet sejak tahun 1990. *HTTP/1.0*, yang didefinisikan di RFC 1945 melengkapinya dengan memperbolehkan pesan yang terkirim dalam format *MIME(Multipurpose Internet Mail Extensions)*, yang mengandung meta informasi tentang data yang ditransfer dan pengubahnya.

Sesuai dokumentasi *IEFT*, *HTTP/1.0* sendiri tidaklah cukup untuk menangani *proxy*(analogikan *proxy* dengan pintu keluar masuk koneksi dalam jaringan yang memiliki banyak komputer dengan koneksi internet tunggal) bertingkat, *caching*(tempat penyimpanan sementara, biasanya program akan mencari ke sini terlebih dahulu untuk informasi yang diminta), kebutuhan untuk koneksi persistem dan *virtual host*(host yang secara nyata tidak ada,biasanya hanya merupakan semacam alias).

Hal ini kemudian disempurnakan oleh versi berikutnya dari *HTTP*, yakni *HTTP/1.1* (RFC 2068), yang kita gunakan sekarang. *HTTP* sendiri juga digunakan sebagai protocol umum untuk komunikasi diantara berbagai *user agent*(*user agent* adalah istilah yang digunakan untuk program yang dijalankan disisi *client*, seperti *web browser* misalnya dalam konteks *HTTP*) dan *proxy/gateway* dengan system internet lain seperti *SMTP*(*simple mail transfer protocol* – RFC 821), *NNTP*(*network news transfer protocol* – RFC 977), *FTP*(*file transfer protocol* – RFC 959), *Gopher* (RFC 1636) dan *WAIS*(*wide area information servers*).

Dengan demikian,*HTTP* mengijinkan akses pada sumber daya yang tersedia pada aplikasi yang berbeda.

2.7.1. Format HTTP

Kita mengenal protocol *HTTP* menggunakan format URL (*Universal Resource Locator*) *HTTP* dalam bentuk *:http://host[:port][abs_path]* *host* adalah nama *domain* internet yang *legal port* adalah bilangan yang menunjukkan *port HTTP* di *host* jika *port* tidak disebutkan maka *port HTTP* diasumsikan sebagai 80 *abs_path* menyatakan lokasi *resource* di dalam *host*. Contoh :

http://www.itn.ac.id/home.html

Jika kita mengisikan URL tersebut ke *browser*, *browser* bertugas untuk mengartikan URL tersebut dan menterjemahkan dalam komunikasi protocol *HTTP*. Aturan dalam mengartikan format URL *HTTP* mengikuti aturan umum URL, yaitu *case sensitive*, kecuali nama dan skema URL *case insensitive*.

Komunikasi protocol *HTTP* terdiri atas pesan request yang diberikan oleh user agent dan response yang dikeluarkan oleh server.

Setiap *request* dan respon *HTTP* menggunakan format pesan generic yang didefinisikan oleh RFC 822.

Pesan *HTTP* terdiri atas baris mulai, header pesan dan isi pesan dipisahkan oleh sebuah baris kosong, yaitu hanya berisi karakter CRLF.

Table 2.3 Contoh Respon GET [6]

Status-lin	HTTP/1.0 ∞ OK
Respons Headers	Server Atmel AVR EWS
Entity Headers	Content-type: image/gif
	Content-length: 1340
CRLF	Carriage Return.Line Feed
Entity Body	<file contens Comes here>

Kata pertama pada *request line* ialah nama untuk metode yang akan direspon. Selama metode yang dijelaskan sesuai dengan aturan *HTTP*, maka akan dijalankan sesuai aplikasinya. Berikut ini table metode *request* pada umumnya.

Tabel 2.4 Tabel Metode Request Pada HTTP [6]

Method	Description
GET	Request to read a web page or whatever is identified by the request URL
POST	Append to a named resource ie.g. a web page,or provider a block of data to a data-handling process at the server.
HEAD	Request to read a web page's header

Table 2.3 Contoh Respon GET [6]

Status-lin	HTTP/1.0 ∞ OK
Respons Headers	Server Atmel AVR EWS
Entity Headers	Content-type: image/gif
	Content-length: 1340
CRLF	Carriage Return.Line Feed
Entity Body	<file contens Comes here>

Kata pertama pada *request line* ialah nama untuk metode yang akan direspon. Selama metode yang dijelaskan sesuai dengan aturan *HTTP*, maka akan dijalankan sesuai aplikasinya. Berikut ini table metode *request* pada umumnya.

Tabel 2.4 Tabel Metode *Request* Pada HTTP [6]

Method	Description
GET	Request to read a web page or whatever is identified by the request URL
POST	Append to a named resource ie.g. a web page,or provider a block of data to a data-handling process at the server.
HEAD	Request to read a web page's header

PUT	Request to store a web page
DELETE	Remove a web page
LINK	Connects two existing resources
UNLINK	Break an existing connection between two resource

2.8. Topologi Jaringan

Topologi jaringan adalah susunan fisik bagaimana node-node saling dihubungkan. Ada 3 jenis topologi :

1. Topologi Bus

Ethernet menggunakan satu kabel yang berfungsi sebagai penghubung untuk mentransmisikan data

2. Topologi Ring

Satu node yang dihubungkan dengan node-node yang lain sehingga membentuk ring.

3. Topologi Hub

Susunan dari topologi ini seperti sebuah bintang yang memiliki satu pusat dari mana data akan ditransmisikan ke seluruh node dalam jaringan. Topologi ini mempunyai kelebihan dibanding kan 2 topologi sebelumnya, yaitu bila terjadi kerusakan kabel maka

tidak akan membawa dampak pada node-node yang lain, tapi hanya pada node yang bersangkutan. Sehingga aktifitas jaringan tidak terganggu.

Banyak hal yang perlu diperhatikan dalam membangun suatu jaringan, diantaranya :

- Pertumbuhan node perlu dijaga dengan baik, karena akan menurunkan performa jika jaringan jika tidak dibarengi dengan pertumbuhan bandwidth yang sesuai.
- Nama dan alamat node harus unik sehingga setiap node dapat diidentifikasi.
- Lalulintas jaringan tidak melebihi kapasitas jaringan.
- Medium yang digunakan dalam jaringan.

2.9. OSI 7 Layer

Untuk menyelenggarakan komunikasi berbagai macam *vendor* computer diperlukan sebuah aturan baku yang standar dan disetujui berbagai pihak. Seperti halnya dua orang yang berlainan bangsa, maka untuk berkomunikasi memerlukan penterjemah/*interpreter* atau satu bahasa yang dimengerti oleh kedua belah pihak.

Dalam dunia computer dan telekomunikasi *interpreter* identik dengan protocol. Untuk itu maka badan dunia yang menangani masalah standarisasi ISO (*Internasional Standardization Organization*) membuat aturan baku yang dikenal

dengan nama model referensi OSI (*Open system Interconnection*). Dengan demikian diharapkan semua vendor perangkat telekomunikasi haruslah berpedoman dengan model referensi ini dalam mengembangkan protokolnya.

Model referensi OSI terdiri dari 7 lapisan, mulai dari lapisan fisik sampai dengan aplikasi. Model referensi ini tidak hanya berguna untuk produk-produk LAN saja, tetapi dalam membangun jaringan internet sekalipun sangat diperlukan.

Berikut lapisan OSI 7 Layer :

Table Fungsi Dari OSI 7Layer [7]

APPLICATION LAYER
PRESENTATION LAYER
SESSION LAYER
TRANSPORT LAYER
NETWORK LAYER
DATA LINK LAYER
PHYSICAL LAYER

2.10. BASCOM

BASCOM adalah bahasa pemrograman pada ATMega, perangkat lunak tersebut merupakan bahasa tingkat tinggi yang khusus dirancang dan dibuat untuk memprogram mikrokontroler produk Atmel Corporation, dan salah satu jenis yang digunakan dalam sistem telemetri temperatur/kelembaban ini adalah ATMega yang kemudian dihubungkan dengan komponen transceiver.

Cara penulisan dan penyusunan program tersebut sama persis seperti layaknya dengan bahasa BASIC atau BASICA hanya saja semua perintahnya ditujukan pada operasi atau mengatur tingkah laku mikrokontroler. Dalam sistem yang dirancang ini pengukuran temperatur dan kelembaban mempunyai ketelitian sampai dua angka desimal dan proses pengirimannya dilakukan secara terpisah antara bagian integer atau bilangan bulat dengan bagian desimal atau bilangan pecahan yang kemudian setelah diterima kedua bagian dari data tersebut disatukan kembali, dan proses pelaksanaan ini dilakukan melalui suatu perangkat lunak BASCOM-AVR yang digunakan untuk memprogram mikrokontroler yang juga sekaligus sebagai komponen yang bertugas mengirim data kepada pihak penerima.

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

Dalam bab ini akan dibahas perancangan dan pembuatan alat. Pembahasan akan dilakukan pada setiap blok rangkaian, cara kerja masing-masing blok rangkaian, perhitungan dan fungsi masing-masing blok rangkaian tersebut. Secara garis besar terdapat dua bagian perangkat yang ada yaitu perancangan perangkat keras (*hardware*) dan perangkat lunak (*software*).

Pada perancangan perangkat keras akan meliputi seluruh peripheral yang digunakan pada sistem ini. Pada perancangan perangkat lunak akan meliputi diagram alir dan software secara umum. Akan tetapi kedua perangkat ini dalam kerjanya akan saling menunjang satu sama lain.

3.1. Perancangan Perangkat Keras (*hardware*)

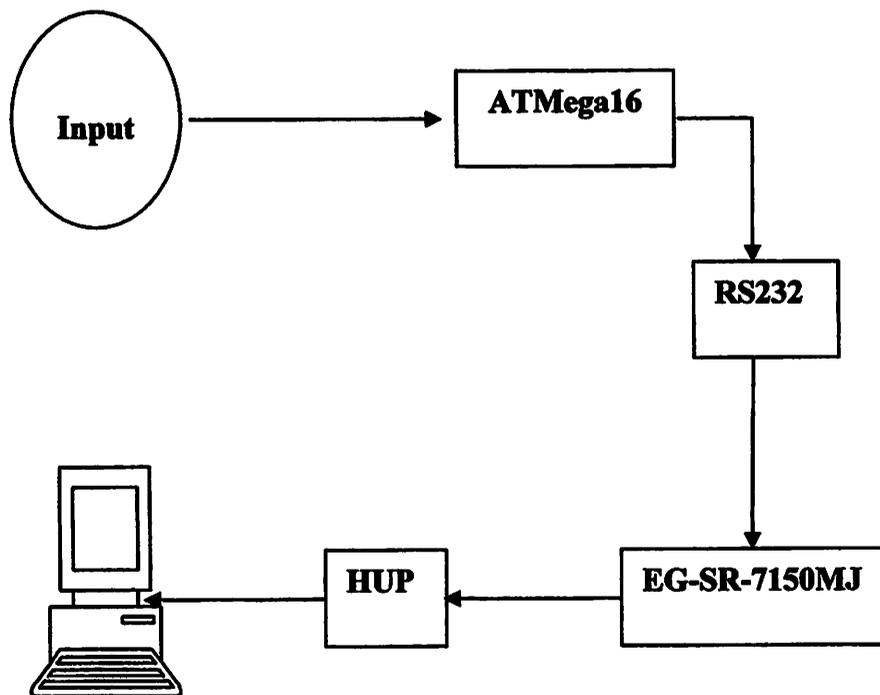
Perangkat keras sendiri terdiri dari rangkaian input yang berupa beberapa potensiometer dihubungkan ke ADC (*Analog Digital Converter*) yang terdapat di dalam minimum sistem ATmega 16. Dan kemudian dikoneksikan ke modul EG-SR-7150MJ, koneksi ini menggunakan rangkaian RS 232.

Output dari modul EG-SR-7150MJ bisa dikoneksikan ke PC menggunakan Switch. Kelebihan dari Switch ini bisa mengkoneksikan alat ke dalam jaringan

Lan. PC merupakan peralatan yang akan menjalankan aplikasi Web browser untuk mengakses sistem aquisisi data yang dirancang.

Spesifikasi rangkaian secara global ditetapkan terlebih dahulu untuk acuan dalam perencanaan selanjutnya. Spesifikasi yang direncanakan adalah sebagai berikut:

3.1.1. Blok Diagram



Gambar 3.1. Blok Diagram Sistem

➤ **Inputan Analog 0-5 Volt.**

Alat ini menggunakan potensiometer sebagai inputan atau simulasi tegangan dengan range 0-5 Volt.

➤ **ATMega16.**

ADC akan menerima inputan dari potensiometer yang akan dikonversi dalam bentuk digital. Lalu ADC akan meneruskan ke ATMega16. Mikrokontroler ATMega16 akan berkomunikasi dengan computer menggunakan TCP/IP, untuk itu diperlukan modul interface yang dapat memungkinkan berkomunikasi melalui Ethernet card.

Untuk dapat menampilkan informasi pada browser mengenai hasil aquisisi data dapat dilakukan dengan mengetik alamat/IP dari modul Ethernet to serial gateway. Mikrokontroler akan menerima data melalui port serial, dengan demikian mikrokontroler harus melakukan seleksi dari karakter yang diterima.

Pada hal ini browser akan mengirimkan request berupa karakter 'GET', yang berarti ada permintaan untuk membaca halaman web pada alamat yang kita akses. Kemudian mikrokontroler akan merespondengan mengirim HTTP1.0/200 OK yang berarti halaman web ada dan siap untuk dikirim lalu dilanjutkan dengan mengirim Content Type:TEXT/HTM yang berarti bahwa isi halaman web berupa text.

Terakhir akan dikirimkan body dari halaman web yang berisi kondisi suatu data naik mengalami perubahan atau tidak. Untuk dapat melakukan update pada tampilan web pada browser maka dapat digunakan program/script pada halaman web untuk melakukan auto refresh dengan periode waktu yang telah ditentukan. Mikrokontroler akan terlebih dahulu menyiapkan data yang diupdate dengan melakukan pembacaan pada port-port yang terhubung detector. Disini ATmega 16 menggunakan bahasa pemrograman BASCOM.

➤ **RS 232.**

Akses data dengan menggunakan port serial memiliki beberapa kelebihan dibandingkan dengan menggunakan port paralel. Kelemahan pada komunikasi paralel adalah banyaknya kabel yang harus terhubung. Hal ini menjadi tidak efisien bila kita menggabungkan alat ke komputer (*interfacing*) untuk jarak yang jauh. Oleh karena itu digunakan komunikasi serial untuk mengatasinya.

Pada komunikasi serial, data yang dikirim per bit (data tersebut antri), walaupun mempunyai kelemahan pengiriman data yang lebih lambat dibanding dengan komunikasi paralel, komunikasi serial bisa digunakan untuk jarak yang jauh. Berdasarkan formatnya, pengiriman data komunikasi serial dibedakan menjadi dua bentuk yaitu : komunikasi serial asinkron dan komunikasi serial sinkron.

Kita menggunakan RS232 untuk komunikasi dengan Ethernet secara serial maka dibutuhkan piranti tambahan berupa IC LVTTTL MAX3232CPE, untuk menyesuaikan level tegangan. Pada dasarnya IC ini hanya digunakan sebagai pengubah level tegangan ke level TTL, ialah pengkodean sinyal yang melewati RS232.

Transmisi data serial adalah transmisi data yang pengiriman dan penerimaan datanya berurutan tiap bitnya. Jadi kita hanya membutuhkan satu saluran untuk mengirimkan data antar dua perangkat komunikasi. Keuntungan transmisi serial yaitu biaya lebih murah karena hanya membutuhkan satu saluran saja sehingga banyak digunakan untuk komunikasi jarak jauh. Sedangkan kerugiannya adalah kecepatan pengiriman data lebih rendah dibandingkan dengan transmisi paralel.

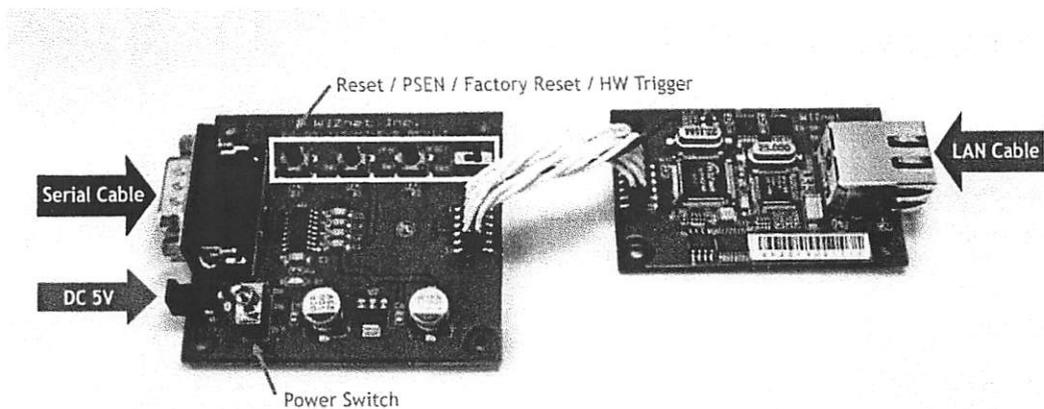
➤ **Wiznet EG-SR-7150MJ.**

Modul yang dipakai adalah Wiznet EG-SR-7150MJ, modul ini berfungsi untuk mengkonversi data serial menjadi tipe data TCP/IP dan sebaliknya. Dengan modul ini memungkinkan mikrokontroller ATMega16 dapat berkomunikasi dengan Ethernet Card, modul ini mempunyai pin untuk komunikasi serial dengan level tegangan LVTTTL(Low Voltage TTL) yaitu 3,3 V. Sedangkan pada mikrokontroller mempunyai level tegangan komunikasi serial TTL yaitu 5V. Adanya kondisi level tegangan yang berbeda maka diperlukan device untuk dapat menyamakan level tegangan serial, untuk itu pada modul EG-SR-7150MJ digunakan IC MAX3232

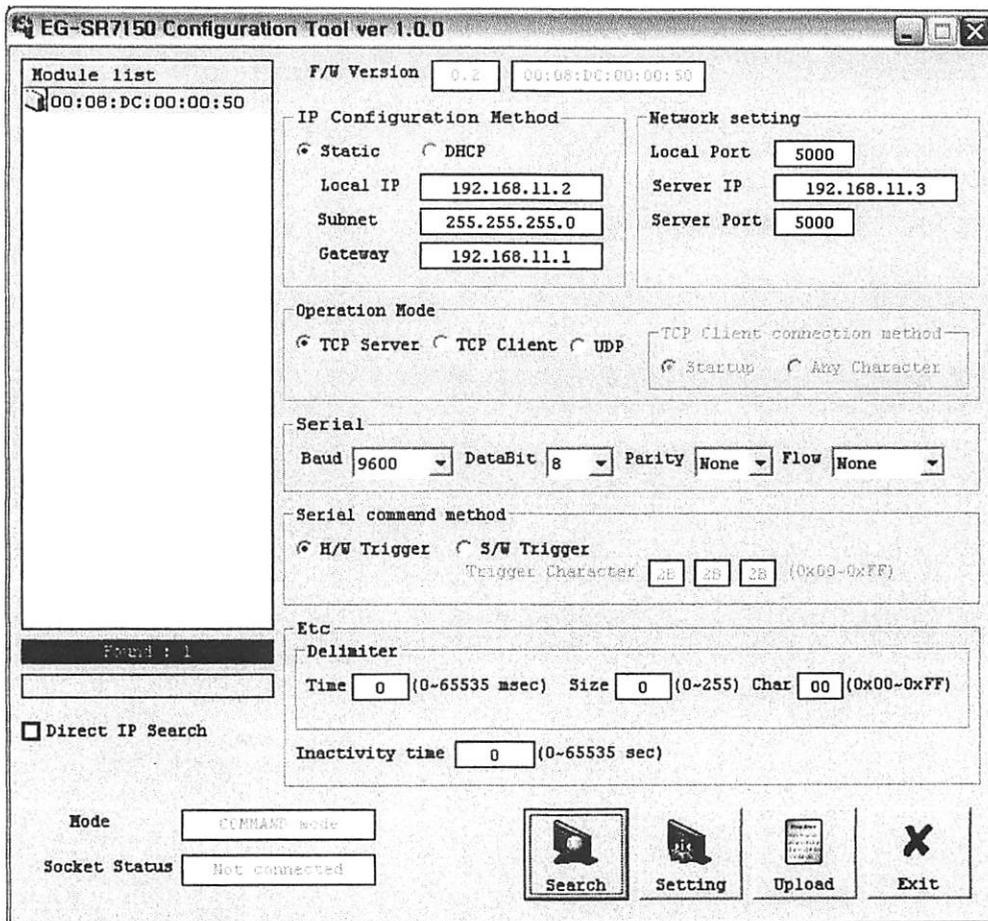
untuk dapat mengkonversi level tegangan serial LVTTTL 3,3V menjadi standart EIA232 dan pada mikrokontroller ATmega16 dipasang IC MAX232 untuk dapat mengkonversi level tegangan serial TTL 5V menjadi standart EIA232.

Bila level tegangan serial telah sama maka dapat dilakukan koneksi/sambungan agar mikrokontroller dapat berkomunikasi dengan PC melalui Ethernet Card.

Untuk dapat terkoneksi dengan Ethernet card maka diperlukan konfigurasi dari Wiznet EG-SR-7150MJ, seperti gambar berikut ini :



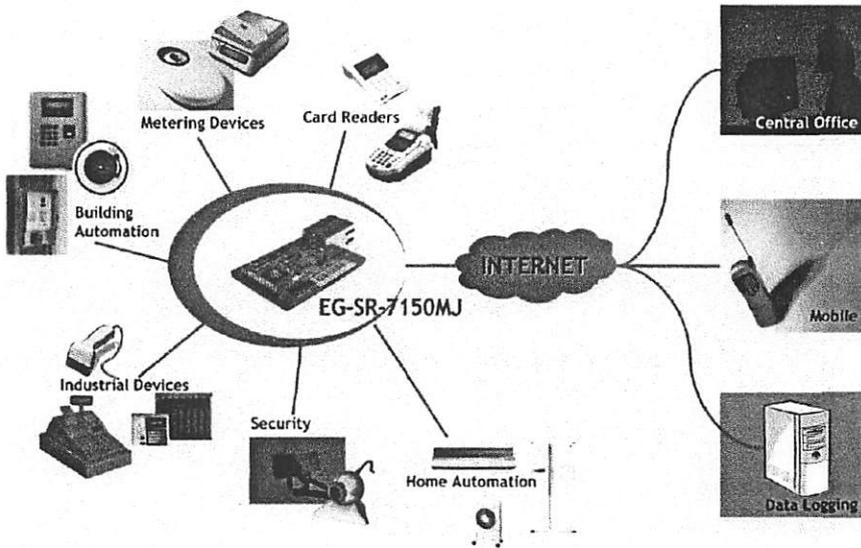
Gambar 3.2 Koneksi Wiznet EG-SR-7150MJ [4]



Gambar 3.3 Konfigurasi Wiznet EG-SR7150MJ [4]

Mode operasi yang dapat ditetapkan pada modul ini ialah server mode dan client mode.

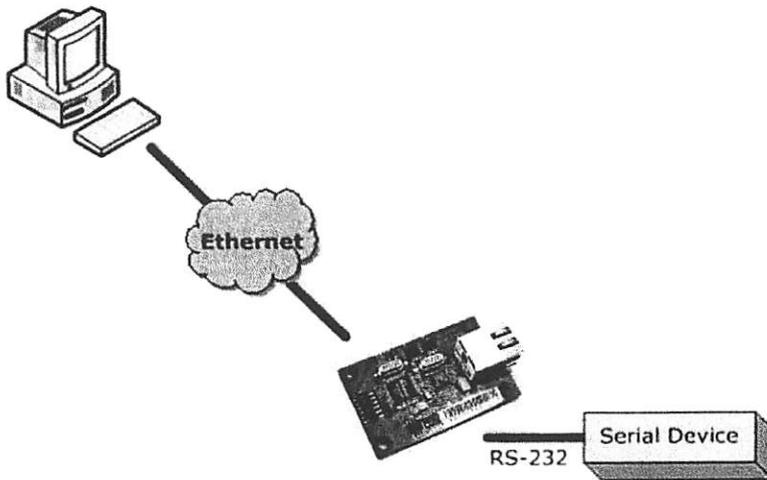
Wiznet EG-SR-7150MJ dapat diaplikasikan sebagai berikut



Gambar 3.4 Beberapa Aplikasi Wiznet EG-SR-7150MJ [4]

➤ **Switch.**

Switch disini merupakan alat untuk mengkoneksikan pada jaringan komputer. Seperti gambar sistem jaringan berikut ini :



Gambar 3.6 Sistem Jaringan [4]

➤ **Dan PC.**

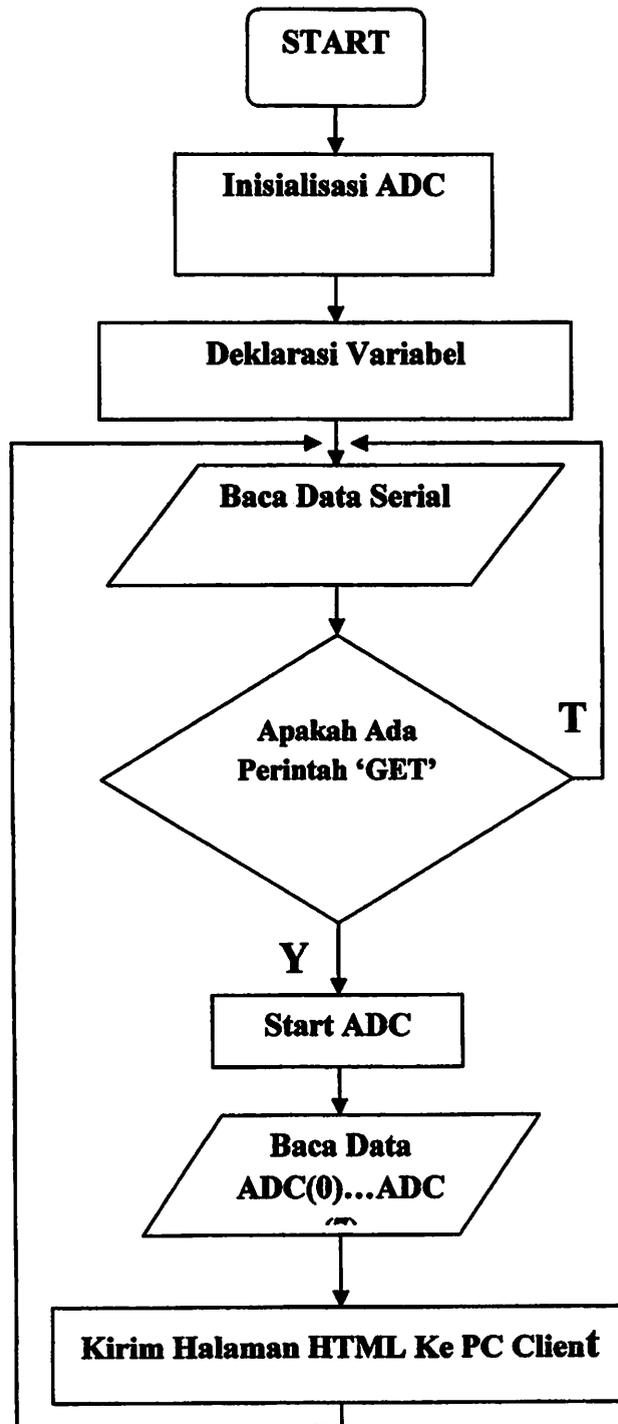
PC komputer client merupakan peralatan yang akan menjalankan aplikasi web browser untuk mengakses sistem akuisisi data yang dirancang.

3.2. Perancangan Perangkat Lunak (*software*)

Perangkat lunak ini berdasarkan pengendali utama yaitu mikrokontroler ATmega16. Pembuatan perangkat lunak sistem aplikasi berdasarkan pada semua kejadian yang harus dikerjakan perangkat keras.

Dalam perancangan alat ini perangkat lunak yang digunakan adalah bahasa pemrograman BASCOM. Pembuatan perangkat lunak harus melalui proses-proses uji coba secara *software* maupun secara *hardware*.

3.2.1. Flow Chart Untuk ATmega16



Gambar 3.7 Flow Chart

Penjelasan *flowchart* secara keseluruhan :

Pertama kali software akan melakukan inisialisasi ADC lalu dideklarasikan secara variable dan dikirimkan melalui data serial, apakah dari data serial mendapat perintah 'GET' jika tidak mendapat perintah 'GET' maka kembali baca data dari serial.

Jika software mendapat perintah 'GET' maka software akan memulai ADC dan membaca data dari ADC(0) sampai dengan ADC(7) setelah semua data terbaca maka software akan mengirim halaman HTML ke PC client. Dari PC client software akan kembali baca data serial.

BAB IV

PENGUKURAN DAN PENGUJIAN

Untuk memastikan alat dapat bekerja sesuai dengan spesifikasi perencanaan, diperlukan serangkaian pengujian dan pengukuran.

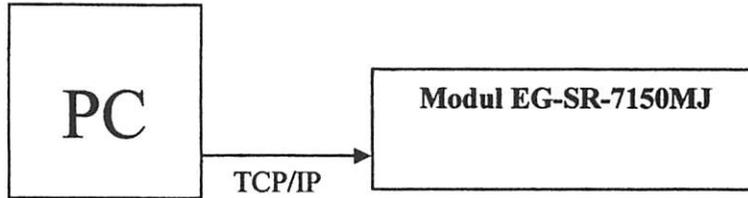
Bab pengujian dan pengukuran ini menguraikan tentang bagian alat yang diuji, tujuan pengujian, langkah-langkah pengujian dan hasil pengujian yang menunjukkan unjuk kerja dari tiap-tiap bagian alat. Pembahasan dalam bab ini dibagi menurut pembagian alat yang diuji untuk mengetahui unjuk kerja sistem secara keseluruhan. Untuk mengetahui kemampuan alat dan sistem kerja sesuai dengan program yang telah dibuat maka dilakukan pengujian pada alat dan sistem kerja alat.

Pengujian dilakukan pada blok sistem adapun yang di uji adalah :
Pengujian Wiznet EG-SR-7150MJ dan Tampilan pada web browser

4.1. Pengujian Modul Wiznet EG-SR-7150MJ

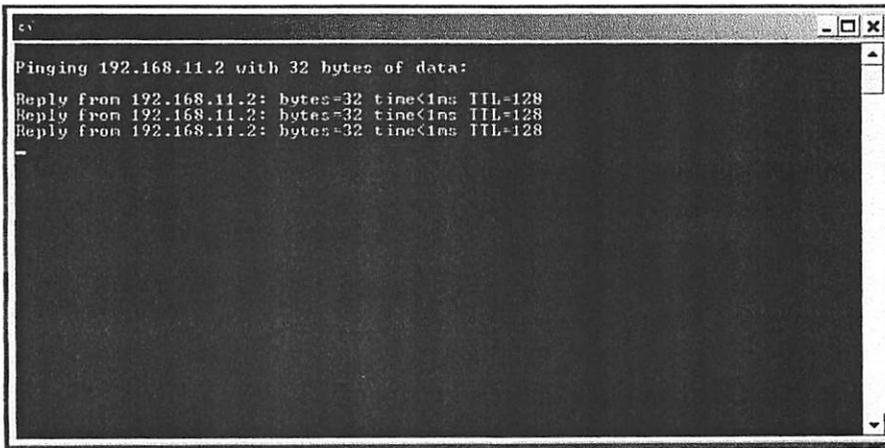
Pengujian modul Wiznet EG-SR-7150MJ ini dapat didasarkan pada 2 pengujian yaitu :

4.1.1. Pengujian TCP/IP



Gambar 4.1 Blok Pengujian TCP/IP

4.1.2. Hasil Pengujian TCP/IP



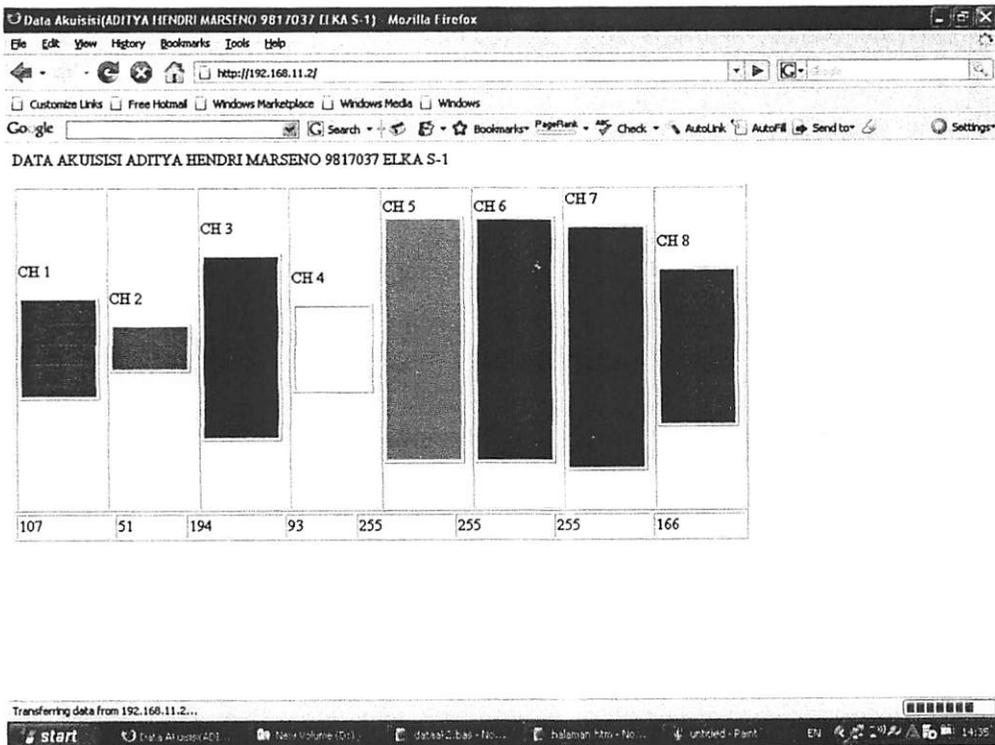
Gambar 4.2 Pengujian TCP/IP

Pengujian dilakukan dengan cara melakukan ping 192.168.11.2 yang merupakan alamat ip dari modul Modul EG-SR-7150MJ. Pengujian dilakukan untuk mengetahui apakah protocol TCP/IP yang terpasang telah bekerja dengan baik atau tidak, TCP/IP telah terpasang pada server dengan konfigurasi seperti yang terlihat pada gambar diatas. Hasil dari pengujian Modul EG-SR-7150MJ akan didapat karakter yang dikirim oleh browser pada saat ada request <http://192.168.11.2>

Setelah mengetahui data yang dikirim oleh browser maka dapat diketahui browser akan mengirim perintah 'GET' pada awal kata data bila kita melakukan request <http://192.168.11.2>. Karakter 'GET' ini akan dijadikan isyarat pada program mikrokontroler ATmega16 untuk melakukan respon bila terjadi pengaksesan halaman web.

4.2. Pengujian Pada Tampilan Web Browser.

Untuk mengetahui apakah tampilan browser sesuai dengan yang diharapkan.



Gambar 4.3 Hasil Tampilan Pada Web Browser

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari pengujian yang dilakukan, maka dapat diambil kesimpulan bahwa :

- Dari koneksi web server telah diuji dengan aplikasi ping dan didapatkan delay time yang pendek.
- Pengujian web server dilakukan dengan memanggil IP webserver dapat berjalan dengan baik dan dapat mengirimkan halaman web berisi informasi dari sistem akuisisi data.
- Setelah pengujian ADC didapatkan hasil dengan baik bahwa masing-masing kanal ADC dapat memberikan informasi data dengan baik sesuai inputan.
- Dan hasil akhirnya menyatakan alat ini bekerja dengan baik sesuai yang diharapkan. Kontribusi alat ini bisa digunakan dalam berbagai bidang seperti BMG, BPS, untuk mengetahui tingkat polusi udara suatu wilayah, ketinggian air suatu sungai atau bendungan dan masih banyak lagi.

5.2. Saran

- Untuk kesempurnaan alat ini diperlukan tingkat ketelitian dan komponen-komponen yang akurat.

DAFTAR PUSTAKA

1. www.atmel.com/datasheet/atmega16
2. www.atmel.com/datasheet/atmega16/adc
3. www.rs232.com
4. www.wiznet.co.krEG-SR-7150MJ
5. Suryadi, MT *TCP/IP dan Internet Terbitan Elex Media Komputindo*
6. www.ilmujaringankomputer.com/http
7. www.ilmujaringankomputer.com/osi7layer

LAMPIRAN



INSTITUT TEKNOLOGI NASIONAL
JL. Raya Karanglo, Km 2
MALANG

FORMULIR BIMBINGAN SKRIPSI

Nama : ADITYA HENDRI MARSENO
Nim : 98.17.037
Masa Bimbingan : 4 JULI 2009 – 4 JANUARI 2010
Judul : PERANCANGAN DAN PEMBUATAN SISTEM AQUISISI DATA
BERBASIS WEB MENGGUNAKAN ATMega16

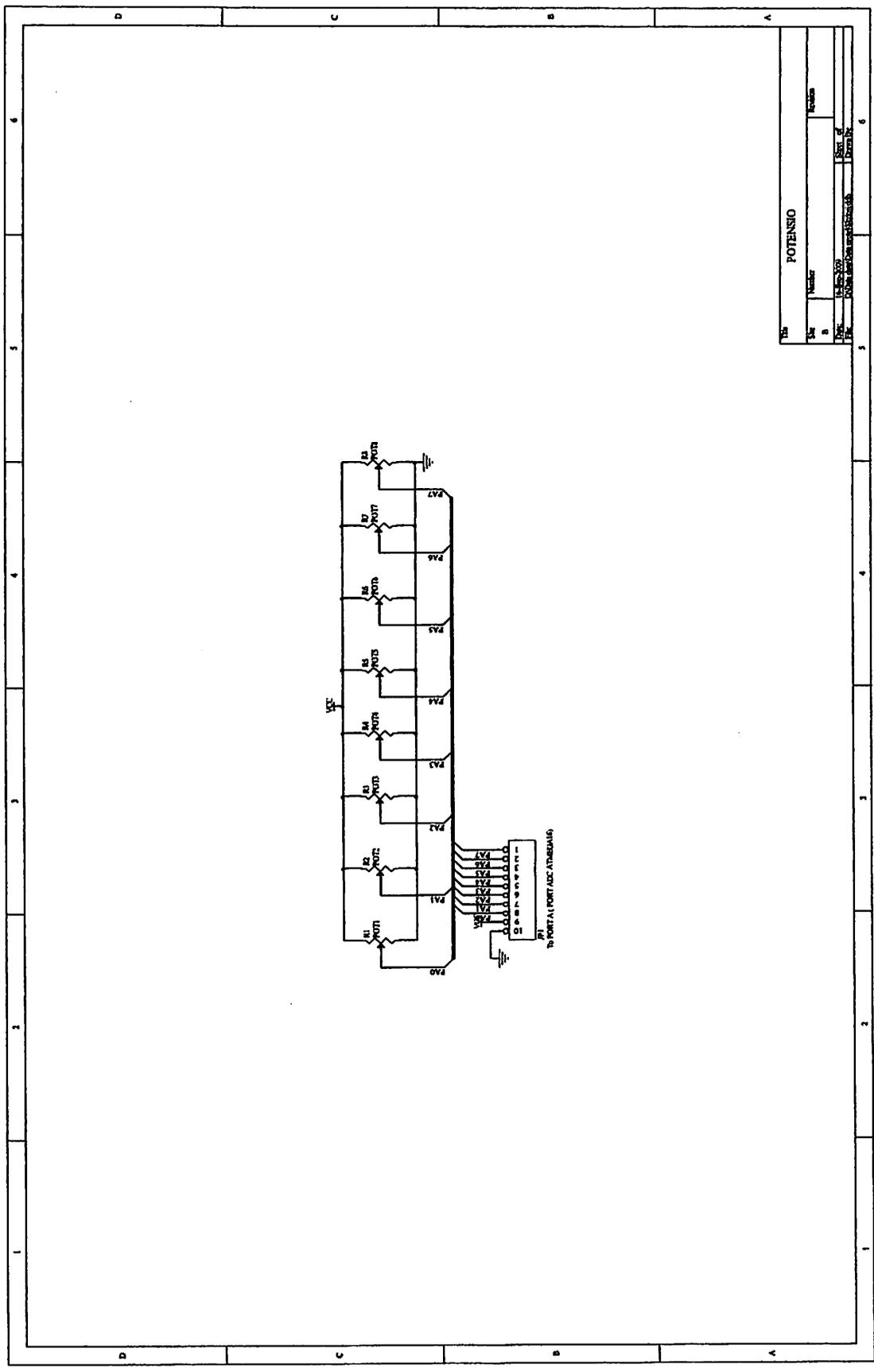
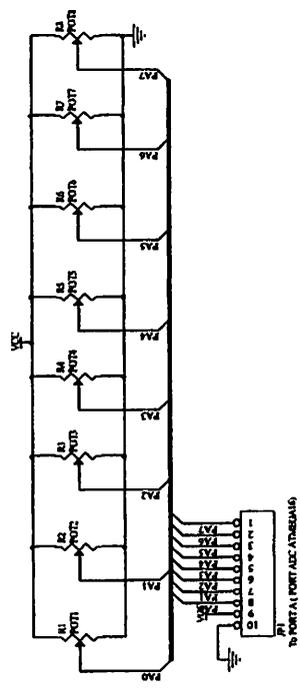
NO	TANGGAL	KETERANGAN	PARAF
1		Bab I & 2 <u>ii</u> <u>iii</u>	
2		Bab <u>iii</u>	
3		Bab <u>iii</u>	
4		Bab <u>iii</u>	
5		Demmo	
6		Selesai	
7		Laporan <u>akhir</u>	
8			
9			
10			

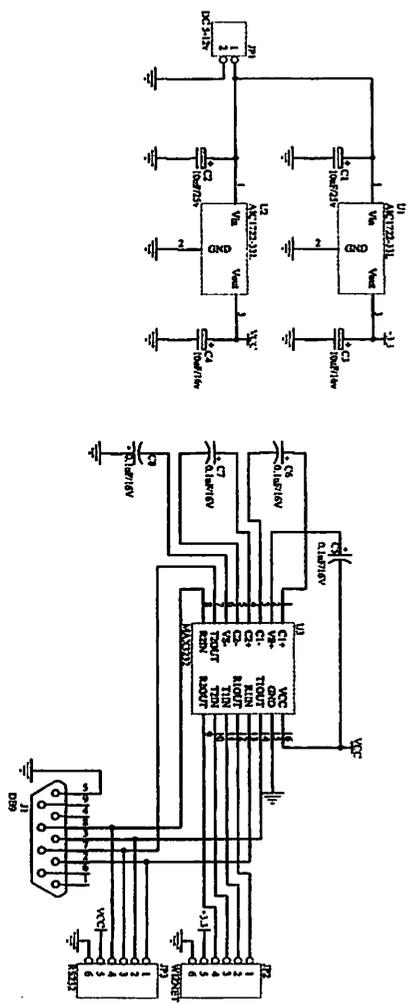
Malang, 2009
Dosen Pembimbing

Ir.F. Yudi Lampraptono, MT
NIP Y. 1039500274

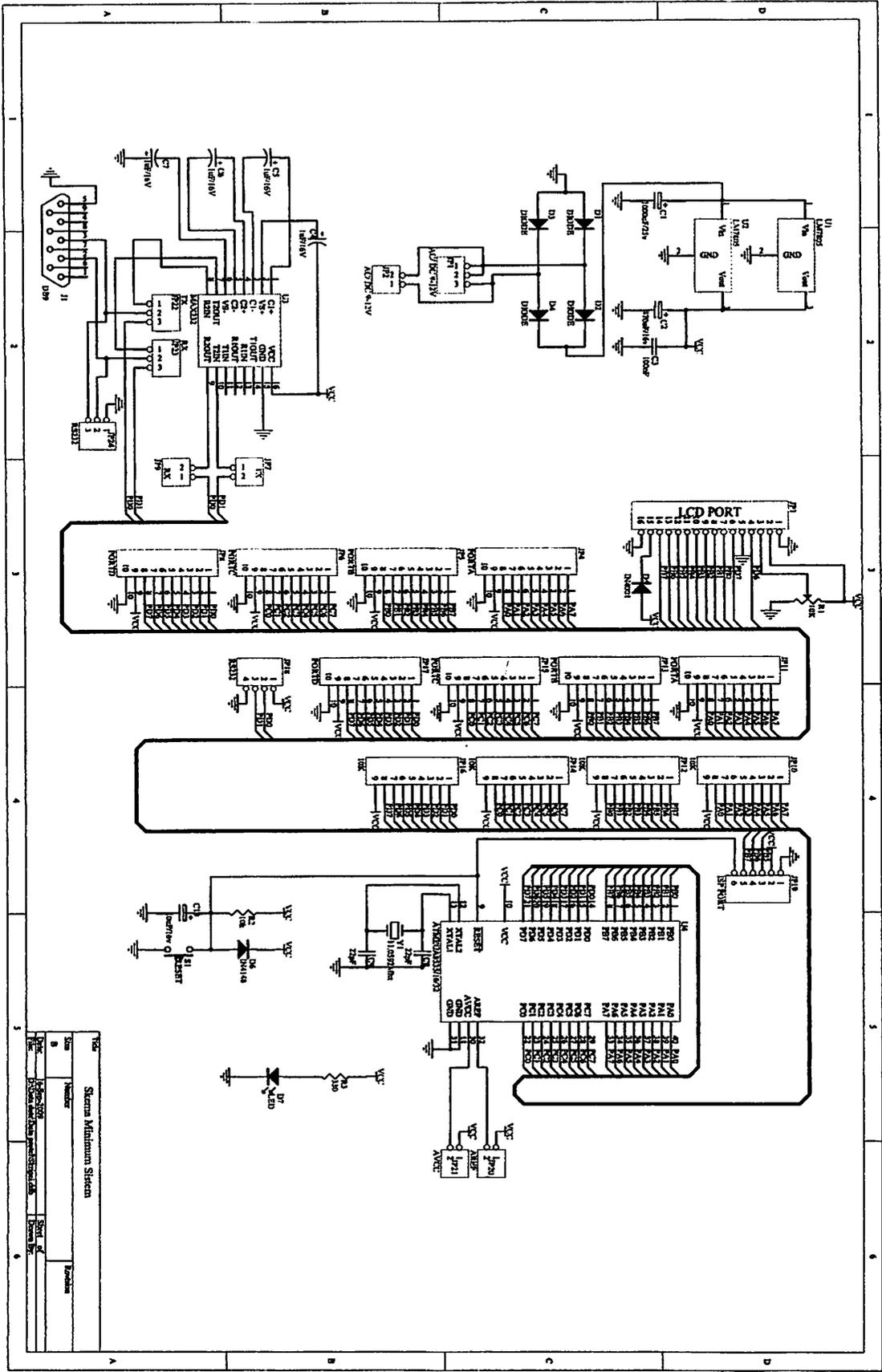
FORM S-4B

Title		POTENSIO	
Scale	Number	Revision	
Proj.	14-2-200	Sheet of	
PC	PLAN DE PORTA DE ATOMEGAS	From No.	





Title		Schemi MAX3232	
Doc	Number	Revision	
Proj	Number	Sheet of	
Proj	Number	Sheet of	



Tipe		Skema Minimum Sistem	
Size	Number	Serial No	Revisi
0			
Date: 14-05-2024		Sheet of: 1	
By: [Signature]		Drawn by: [Signature]	


```
Print #1 , "HTTP/1.0 200 OK{013}{010}"
Print #1 , " "
Print #1 , "<!DOCTYPE HTML PUBLIC '-//W3C//DTD HTML 4.01 Transitional//EN'"
Print #1 , "<script language='JavaScript'>>window.setTimeout('location.reload();,5000);</script'"
Print #1 , "<Html><Head><Title> Data Akuisisi </Title>"
Print #1 , "<meta http-equiv='Content-Type' content='text/html; charset=iso-8859-1'"
Print #1 , "</head><body><p>DATA AKUISISI</p>"
Print #1 , "<table width='75%' height='255' border='1'"
Print #1 , "<tr><td bgcolor='#FFFFFF'><p>CH 1</p><table width='95%' height="
Print #1 , Ch1
Print #1 , "" border='1'"
Print #1 , "<tr><td bgcolor='#FF0000'>&nbsp;</td></tr></table><p>&nbsp;</p></td><td><p>CH 2</p>"
Print #1 , "<table width='95%' height="
Print #1 , Ch2
Print #1 , "" border='1'"<tr><td bgcolor='#00FF00'>&nbsp;</td>"
Print #1 , "</tr></table><p>&nbsp;</p></td><td><p>CH 3</p><table width='95%' height="
Print #1 , Ch3
Print #1 , "" border='1'"
Print #1 , "<tr><td bgcolor='#0000FF'>&nbsp;</td></tr></table><p>&nbsp;</p></td><td><p>CH 4</p>"
Print #1 , "<table width='95%' height="
Print #1 , Ch4
Print #1 , "" border='1'"<tr><td bgcolor='#FFFF00'>&nbsp;</td>"
Print #1 , "</tr></table><p>&nbsp;</p></td><td><p>CH 5</p><table width='95%' height="
Print #1 , Ch5
Print #1 , "" border='1'"
Print #1 , "<tr><td bgcolor='#00FFFF'>&nbsp;</td></tr></table><p>&nbsp;</p></td><td><p>CH 6</p>"
Print #1 , "<table width='95%' height="
Print #1 , Ch6
Print #1 , "" border='1'"<tr><td bgcolor='#FF00FF'>&nbsp;</td>"
Print #1 , "</tr></table><p>&nbsp;</p></td><td><p>CH 7</p><table width='95%' height="
Print #1 , Ch7
Print #1 , "" border='1'"
Print #1 , "<tr><td bgcolor='#000000'>&nbsp;</td></tr></table><p>&nbsp;</p></td><td><p>CH 8</p>"
Print #1 , "<table width='95%' height="
Print #1 , Ch8
Print #1 , "" border='1'"<tr><td bgcolor='#990000'>&nbsp;</td>"
Print #1 , "</tr></table><p>&nbsp;</p></td></tr></table><table width='75%' border='1'"
Print #1 , "<tr><td>"
Print #1 , Ch1
Print #1 , "</td><td>"
Print #1 , Ch2
Print #1 , "</td><td>"
Print #1 , Ch3
Print #1 , "</td><td>"
Print #1 , Ch4
Print #1 , "</td><td>"
Print #1 , Ch5
Print #1 , "</td><td>"
Print #1 , Ch6
```

```
Print #1 , "</td>"
Print #1 , "<td>"
Print #1 , Ch7
Print #1 , "</td><td>"
Print #1 , ch8
Print #1 , "</td></tr></table><p>&nbsp;</p>"
Print #1 , "</body>"
Print #1 , "</Html>"
Print #1 , " "
Nm = " "
End If
Loop Until Nm = " "
Close #1
```

DATA SHEET

MENGENAL MIKROKONTROLER AVR ATMega16

Mokh. Sholihul Hadi

m_sholihul_hadi@yahoo.com

Lisensi Dokumen:

Copyright © 2003-2008 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

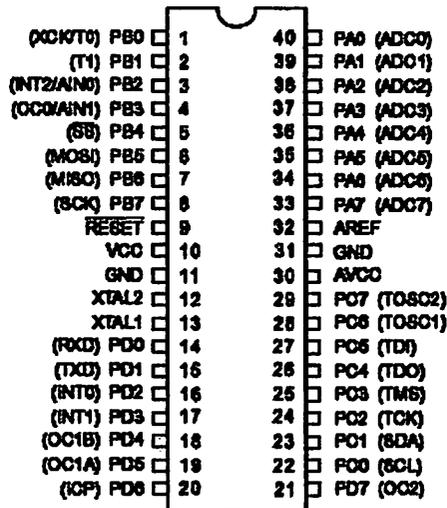
AVR merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur *RISC (Reduced Instruction Set Computer)*. Hampir semua instruksi dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 register general-purpose, timer/counter fleksibel dengan mode *compare*, *interrupt internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, dan *mode power saving*, ADC dan PWM internal. AVR juga mempunyai *In-System Programmable Flash on-chip* yang memungkinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial SPI. ATMega16.

ATMega16 mempunyai *throughput* mendekati 1 MIPS per MHz membuat disainer sistem untuk mengoptimasi konsumsi daya versus kecepatan proses.

Beberapa keistimewaan dari AVR ATMega16 antara lain:

1. *Advanced RISC Architecture*
 - *130 Powerful Instructions – Most Single Clock Cycle Execution*
 - *32 x 8 General Purpose Fully Static Operation*
 - *Up to 16 MIPS Throughput at 16 MHz*
 - *On-chip 2-cycle Multiplier*
2. *Nonvolatile Program and Data Memories*
 - *8K Bytes of In-System Self-Programmable Flash*
 - *Optional Boot Code Section with Independent Lock Bits*
 - *512 Bytes EEPROM*
 - *512 Bytes Internal SRAM*
 - *Programming Lock for Software Security*
3. *Peripheral Features*
 - *Two 8-bit Timer/Counters with Separate Prescalers and Compare Mode*
 - *Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes*

- One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four PWM Channels
 - 8-channel, 10-bit ADC
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
4. Special Microcontroller Features
- Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
5. I/O and Package
- 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, 44-lead PLCC, and 44-pad MLF
6. Operating Voltages
- 2.7 - 5.5V for Atmega16L
 - 4.5 - 5.5V for Atmega16



Gambar 1 Pin-pin ATMega16 kemasan 40-pin

Pin-pin pada ATMega16 dengan kemasan 40-pin DIP (*dual in-line package*) ditunjukkan oleh gambar 1. Guna memaksimalkan performa, AVR menggunakan arsitektur *Harvard* (dengan memori dan bus terpisah untuk program dan data).

Port sebagai input/output digital

ATMega16 mempunyai empat buah port yang bernama *PortA*, *PortB*, *PortC*, dan *PortD*. Keempat port tersebut merupakan jalur *bi-directional* 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 *DDRx*, bit *PORTxn* terdapat pada

I/O address PORTx, dan bit PINxn terdapat pada I/O address PINx. Bit DDxn dalam register DDRx (*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*. Jika ini bukan suatu masalah, maka bit PUD pada register SFIOR dapat diset 1 untuk mematikan semua *pull-up* dalam semua port. Peralihan dari kondisi *input dengan pull-up* ke kondisi *output low* juga menimbulkan masalah yang sama. Kita harus menggunakan kondisi *tri-state* (DDxn=0, PORTxn=0) atau kondisi *output high* (DDxn=1, PORTxn=0) sebagai kondisi transisi.

Tabel 1 Konfigurasi pin port

DDxn	PORTxn	PUD (In SFIOR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pin will source current if ext. pulled low.
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

Bit 2 – PUD : *Pull-up Disable*

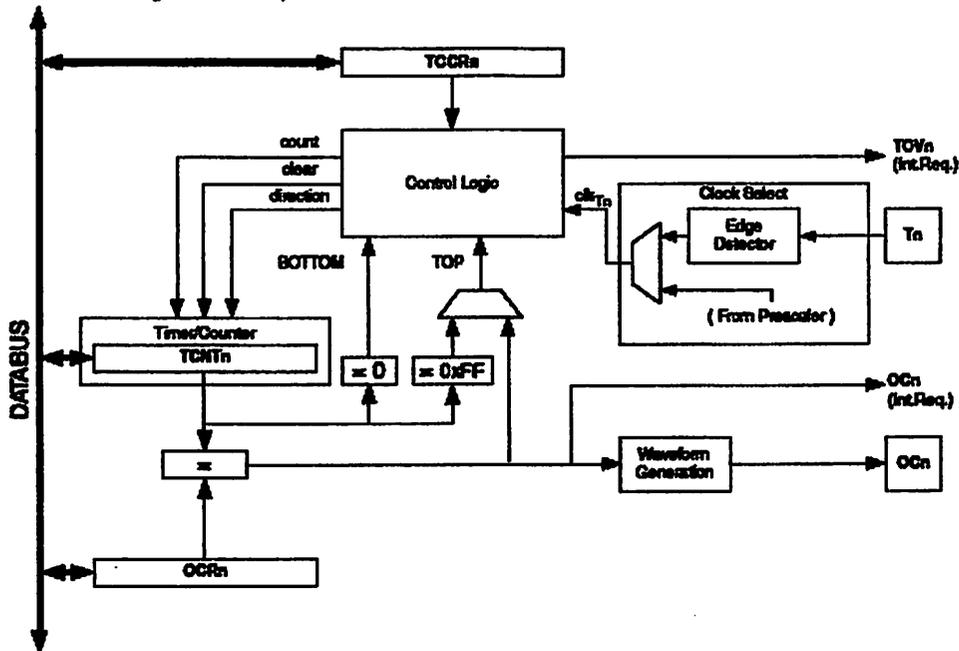
Bila bit diset bernilai 1 maka *pull-up* pada port I/O akan dimatikan walaupun register DDxn dan PORTxn dikonfigurasi untuk menyalakan *pull-up* (DDxn=0, PORTxn=1).

Timer

Timer/counter adalah fasilitas dari ATmega16 yang digunakan untuk perhitungan pewaktuan. Beberapa fasilitas *channel* dari timer counter antara lain: *counter channel* tunggal, pengosongan data timer sesuai dengan data pembanding, bebas *-glitch*, tahap yang tepat *Pulse Width Modulation (PWM)*, pembangkit frekuensi, *event counter external*.

Gambaran Umum

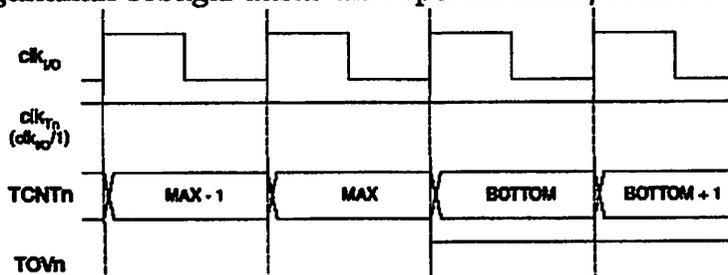
Gambar diagram *block* timer/counter 8 bit ditunjukkan pada gambar 2. Untuk penempatan pin I/O telah di jelaskan pada bagian I/O di atas. CPU dapat diakses register I/O, termasuk dalam pin-pin I/O dan bit I/O. *Device* khusus register I/O dan lokasi bit terdaftar pada deskripsi timer/counter 8 bit.



Gambar 2 Blok diagram timer/counter

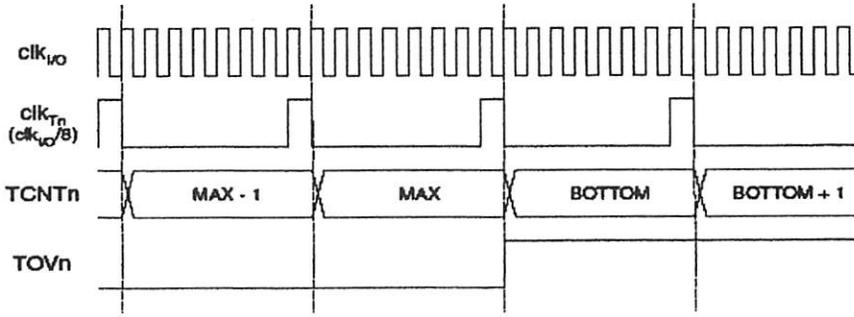
Timing Diagram Timer/Counter

Timer/counter didesain sinkron *clock* timer (clk_{T0}) oleh karena itu ditunjukkan sebagai sinyal *enable* clock pada gambar 3. Gambar ini termasuk informasi ketika *flag interrupt* dalam kondisi set. Data timing digunakan sebagai dasar dari operasi timer/counter.



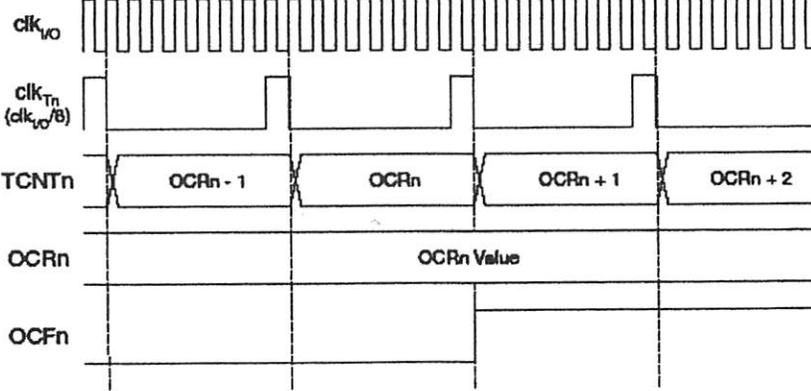
Gambar 3 Timing diagram timer/counter, tanpa prescaling

Sesuai dengan gambar 4 timing diagram timer/counter dengan *prescaling* maksudnya adalah counter akan menambahkan data counter ($TCNTn$) ketika terjadi pulsa *clock* telah mencapai 8 kali pulsa dan sinyal *clock* pembagi aktif *clock* dan ketika telah mencapai nilai maksimal maka nilai $TCNTn$ akan kembali ke nol. Dan kondisi *flag timer* akan aktif ketika $TCNTn$ maksimal.



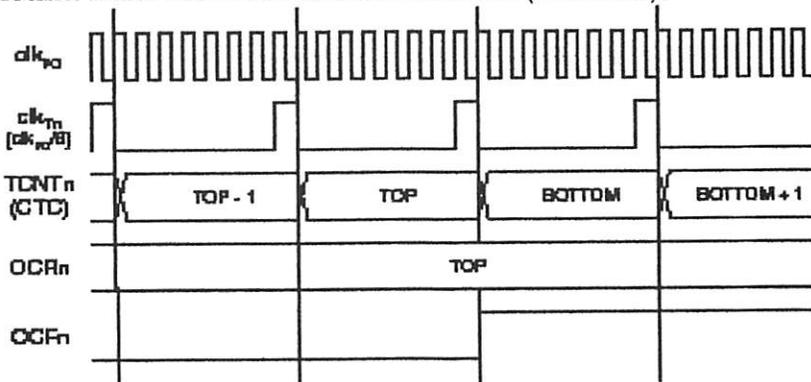
Gambar 4 Timing diagram timer/counter, dengan prescaling

Sama halnya timing timer diatas, timing timer/counter dengan seting OCFO timer mode ini memasukan data ORCn sebagai data input timer. Ketika nilai ORCn sama dengan nilai TCNTn maka pulsa *flag timer* akan aktif. TCNTn akan bertambah nilainya ketika pulsa *clock timer* telah mencapai 8 pulsa. Dan kondisi *flag* akan berbalik (komplemen) kondisi ketika nilai TCNTn kembali ke nilai 0 (*overflow*).



Gambar 5 Timing diagram timer/counter, menyeting OCFO, dengan pescaler (fclk_I/O/8)

Ketika nilai ORCn sama dengan nilai TCNTn maka pulsa *flag timer* akan aktif. TCNTn akan bertambah nilainya ketika pulsa *clock* telah mencapai 8 pulsa. Dan kondisi *flag* akan berbalik (komplemen) kondisi ketika nilai TCNTn kembali ke nilai 0 (*overflow*).



Gambar 6 Timing diagram timer/counter, menyeting OCFO, pengosongan data timer sesuai dengan data pembandingan, dengan pescaler (fclk_I/O/8)

Deskripsi Register Timer/Counter 8 bit

Bit	7	6	5	4	3	2	1	0	TCCR0
	FOCO	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Gambar 7 Register timer counter 8 bit

Bit 7 – FOCO : perbandingan kemampuan output

FOCO hanya akan aktif ketika spesifik-spesifik bit WGM00 tanpa PWM mode. Adapun untuk meyakinkan terhadap kesesuaian dengan *device-device* yang akan digunakan, bit ini harus diset nol ketika TCCR0 ditulisi saat mengoperasikan mode PWM. Ketika menulisi logika satu ke bit FOCO, dengan segera dipaksakan untuk disesuaikan pada unit pembangkit bentuk gelombang. Output OCO diubah disesuaikan pada COM01: bit 0 menentukan pengaruh daya perbandingan.

Bit 6,3 – WGM01:0: Waveform Generation Mode

Bit ini mengontrol penghitungan yang teratur pada counter, sumber untuk harga counter maksimal (TOP), dan tipe apa dari pembangkit bentuk gelombang yang digunakan. Mode-mode operasi didukung oleh unit timer/counter sebagai berikut : mode normal, pembersih timer pada mode penyesuaian dengan perbandingan (CTC), dan dua tipe mode *Pulse Width Modulation (PWM)*.

Tabel 2 Deskripsi Bit Mode Pembangkit Bentuk Gelombang

Mode	WGM01 (CTC0)	WGM00 (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0	TOV0 Flag Set-on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	TOP	MAX

catatan: definisi nama-nama bit CTC0 dan PWM0 sekarang tidak digunakan lagi. Gunakan WGM 01: 0 definisi. Bagaimanapun lokasi dan fungsional dan lokasi dari masing-masing bit sesuai dengan versi timer sebelumnya.

Bit 5:4 – COM01:0 Penyesuaian Perbandingan Mode Output

Bit ini mengontrol pin output *compare (OCO)*, jika satu atau kedua bit COM01:0 diset, output OCO melebihi fungsional port normal I/O dan keduanya terhubung juga. Bagaimanapun, catatan bahwa bit *Direksi Data Register (DDR)* mencocokkan ke pin OCO yang mana harus diset dengan tujuan mengaktifkan. Ketika OCO dihubungkan ke pin, fungsi dari bit COM01:0 tergantung dari pengesetan bit WGM01:0. Tabel di bawah menunjukkan COM fungsional ketika bit-bit WGM01:0 diset ke normal atau mode CTC (non PWM).

Tabel 3 Mode Output Pembanding, tanpa PWM

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Toggle OC0 on compare match
1	0	Clear OC0 on compare match
1	1	Set OC0 on compare match

Tabel 4 menunjukkan bit COM01:0 fungsional ketika bit WGM01:0 diset ke mode fast PWM.

Tabel 4 Mode Output Pembanding, Mode fast PWM

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match, set OC0 at TOP
1	1	Set OC0 on compare match, clear OC0 at TOP

Tabel 5 menunjukkan bit COM01:0 fungsional ketika bit WGM01:0 diset ke *mode phase correct* PWM.

Tabel 5 Mode Output Pembanding, Mode phase correct PWM

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match when up-counting. Set OC0 on compare match when downcounting.
1	1	Set OC0 on compare match when up-counting. Clear OC0 on compare match when downcounting.

Bit 2:0 – CS02:0 : Clock Select

Tiga bit *clock select* sumber *clock* digunakan dengan timer/counter. Jika mode pin *eksternal* digunakan untuk timer counter0, perpindahan dari pin T0 akan memberi clock counter.

Tabel 6 Deskripsi bit clock select

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	$clk_{IO}/(No\ prescaling)$
0	1	0	$clk_{IO}/8$ (From prescaler)
0	1	1	$clk_{IO}/32$ (From prescaler)
1	0	0	$clk_{IO}/256$ (From prescaler)
1	0	1	$clk_{IO}/1024$ (From prescaler)
1	1	0	External clock source on T0 pin. Clock on falling edge.
1	1	1	External clock source on T0 pin. Clock on rising edge.

Sesuai dengan tabel diatas maka sumber *clock* dapat dibagi sehingga timer/counter dapat disesuaikan dengan banyak data yang dihitung.

Register Timer/Counter TCNT0

Bit	7	6	5	4	3	2	1	0	
	TCNT0[7:0]								TCNT0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Gambar 8 Register timer TCNT0

Register timer/counter memberikan akses secara langsung, keduanya digunakan untuk membaca dan menulis operasi, untuk penghitung unit 8-bit timer/counter. Menulis ke blok-blok register TCNT0 (*removes*) disesuaikan dengan clock timer berikutnya. Memodifikasi counter (TCNT0) ketika perhitungan berjalan, memperkenalkan resiko kehilangan perbandingan antara TCNT0 dengan register OCR0.

Register Timer/Counter OCR0

Bit	7	6	5	4	3	2	1	0	
	OCR0[7:0]								OCR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Gambar 9. Register timer OCR0

Register output pembanding berisi sebuah harga 8 bit yang mana secara terus-menerus dibandingkan dengan harga counter (TCNT0). Sebuah penyesuaian dapat digunakan untuk membangkitkan output *interrupt* pembanding, atau untuk membangkitkan sebuah output bentuk gelombang pada pin OC0.

Register Timer/Counter *Interrupt Mask*

Bit 1-OCIE0: output timer counter menyesuaikan dengan kesesuaian *interrupt* yang aktif.

Ketika bit OCIE0 ditulis satu, dan 1-bit pada register status dalam kondisi set (satu), membandingkan timer/counter pada *interrupt* yang sesuai diaktifkan. Mencocokkan *interrupt* yang dijalankan kesesuaian pembanding pada timer/counter0 terjadi, ketika bit OCF0 diset pada *register* penanda timer/counter-TIFR.

Bit 0 - TOIE0: Timer/Counter 0 Overflow Interrupt Enable

Ketika bit TOIE0 ditulis satu, dan 1-bit pada *register* status dalam kondisi set (satu), timer/counter melebihi *interrupt* diaktifkan. Mencocokkan *interrupt* dijalankan jika kelebihan pada timer/counter0 terjadi, ketika bit TOV0 diset pada *register* penanda timer/counter-TIFR

Register Timer/Counter Register - TIFR

Bit	7	6	5	4	3	2	1	0	
	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	TIFR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Gambar 10 Register timer TIFR

Bit 1 – OCF0: Output Compare Flag 0

OCF0 dalam kondisi set (satu) kesesuaian pembandingan terjadi antara timer/counter dan data pada OCRO – Register 0 keluaran pembandingan. OCF0 diclear oleh *hardware* ketika eksekusi pencocokan penanganan *vector interrupt*. Dengan alternatif mengclearkan OCF0 dengan menuliskan logika satu pada flag. Ketika I-bit pada SREG, OCIE0 (Timer/Counter0 penyesuaian pembandingan *interrupt enable*), dan OCF0 diset (satu), timer/counter pembandingan kesesuaian *interrupt* dijalankan.

Bit 0 – TOV0: Timer/Counter Overflow Flag

Bit TOV0 diset (satu) ketika kelebihan terjadi pada timer/counter0. TOV0 diclearkan dengan *hardware* ketika penjalanan pencocokan penanganan *vector interrupt*. Dengan alternatif, TOV0 diclearkan dengan jalan memberikan logika satu pada flag. Ketika I-bit pada SREG, TOIE0 (Timer/Counter0 *overflow interrupt enable*), dan TOV0 diset (satu), timer/counter *overflow* interrupt dijalankan. Pada tahap mode PWM yang tepat, bit ini di set ketika timer/counter merubah bagian perhitungan pada \$00.

Serial pada ATmega16

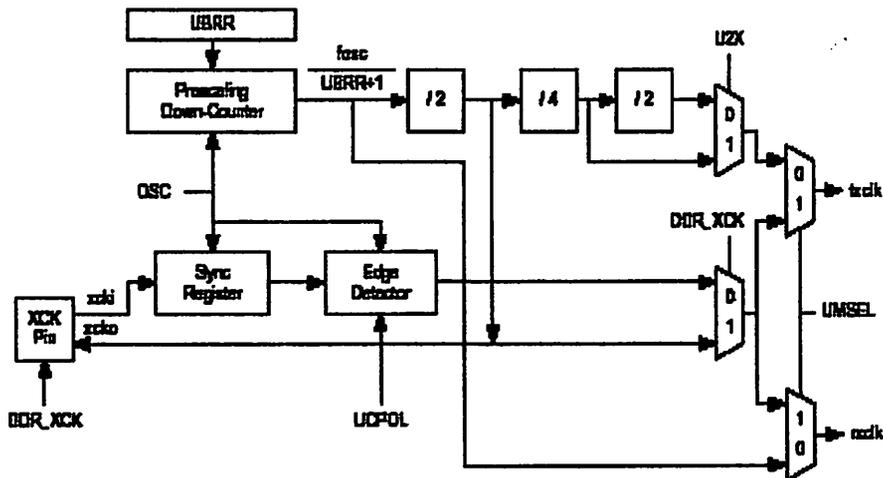
Universal synchronous dan *asynchronous* pemancar dan penerima serial adalah suatu alat komunikasi serial sangat fleksibel.

Jenis yang utama adalah :

- Operasi *full duplex* (register penerima dan pengirim serial dapat berdiri sendiri)
- Operasi *Asynchronous* atau *synchronous*
- Master* atau *slave* mendapat clock dengan operasi *synchronous*
- Pembangkit *baud rate* dengan resolusi tinggi
- Dukung *frames serial* dengan 5, 6, 7, 8 atau 9 Data bit dan 1 atau 2 Stop bit
- Tahap *odd* atau *even parity* dan *parity check* didukung oleh *hardware*
- Pendeteksian data *overrun*
- Pendeteksi *framing error*
- Pemfilteran gangguan (noise) meliputi pendeteksian bit *false start* dan pendeteksian *low pass filter* digital
- Tiga *interrupt* terdiri dari TX complete, TX data register empty dan RX complete.
- Mode komunikasi multi-processor
- Mode komunikasi *double speed asynchronous*

Generator Clock

Logic generator clock menghasilkan dasar clock untuk pengirim dan penerima. USART mendukung empat mode operasi clock: Normal *Asynchronous*, *Double Speed Asynchronous* mode *Master Synchronous* dan *Slave Synchronous*. Bit UMSEL pada USART control dan status register C (UCSRC) memilih antara operasi *Asynchronous* dan *Synchronous*. *Double speed* (hanya pada mode *Asynchronous*) dikontrol oleh U2X yang mana terdapat pada register UCSRA. Ketika menggunakan mode operasi *synchronous* (UMSEL = 1) dan data direction register untuk pin Xck (DDR_XCK) mengendalikan apakah sumber clock tersebut adalah *internal (master mode)* atau *eksternal (slave mode)* pin-pin XCK hanya akan aktif ketika menggunakan mode *Synchronous*.



Gambar 11 Blok diagram clock generator logic

Keterangan sinyal :

txclk : clock pengirim (*internal clock*)

rxclk : clock dasar penerima (*internal clock*)

xcki : input dari pin XCK (sinyal *internal*). Digunakan untuk operasi *slave synchronous*.

xcko : clock output ke pin XCK (sinyal *internal*). Digunakan untuk operasi *master synchronous*

fosc : frekuensi pin XTAL (system clock)

Generator Internal Clock – Pembangkit Baud rate

Generasi *internal clock* digunakan untuk mode – mode operasi master *asynchronous* dan *synchronous*. Register USART baud rate (UBRR) dan *down-counter* dikoneksikan kepada fungsinya sebagai *programmable prescaler* atau pembangkit *baud rate*. *Down-counter*, dijalankan pada *system clock* (f_{osc}), dibebani dengan nilai UBRR setiap counter telah dihitung mundur ke nol atau ketika register UBRR ditulisi. Clock dibangkitkan setiap counter mencapai nol. Clock ini adalah pembangkit *baud rate clock output* ($f_{osc}/(UBRR+1)$). Pemancar membagi baud rete generator clock output dengan 2, 8,

atau 16 cara tergantung pada mode. Pembangkit output *baud rate* digunakan secara langsung oleh penerima clock dan unit-unit pelindung data. Unit-unit *recovery* menggunakan suatu mesin status yang menggunakan 2, 8, atau 16 cara yang tergantung pada cara menyimpan status dari UMSEL, bit-bit U2X dan DDR_XCK.

Tabel di bawah menunjukkan penyamaan perhitungan *baud rate* dan nilai UBRR tiap mode operasi menggunakan sumber pembangkit clock *internal*.

Tabel 7 Persamaan untuk menyeting perhitungan register baud rate

Operating Mode	Equation for Calculating Baud Rate ⁽¹⁾	Equation for Calculating UBRR Value
Asynchronous Normal Mode (U2X = 0)	$BAUD = \frac{f_{osc}}{16(UBRR + 1)}$	$UBRR = \frac{f_{osc}}{16BAUD} - 1$
Asynchronous Double Speed Mode (U2X = 1)	$BAUD = \frac{f_{osc}}{8(UBRR + 1)}$	$UBRR = \frac{f_{osc}}{8BAUD} - 1$
Synchronous Master Mode	$BAUD = \frac{f_{osc}}{2(UBRR + 1)}$	$UBRR = \frac{f_{osc}}{2BAUD} - 1$

note: *baud rate* menunjukkan pengiriman rate bit tiap detik (bps)

BAUD : *baud rate* (pada bit-bit per detik, bps) fosc frekuensi sistem clock osilator

UBRR: terdiri dari UBRRH dan UBBRL, (0-4095)

Eksternal Clock

Eksternal clock digunakan untuk operasi mode *slave synchronous*. *Eksternal* clock masuk dari pin XCK dicontohkan oleh suatu daftar sinkronisasi register untuk memperkecil kesempatan meta-stabilitas. Keluaran dari sinkronisasi register kemudian harus menerobos detector tepi sebelum digunakan oleh pengirim dan penerima.

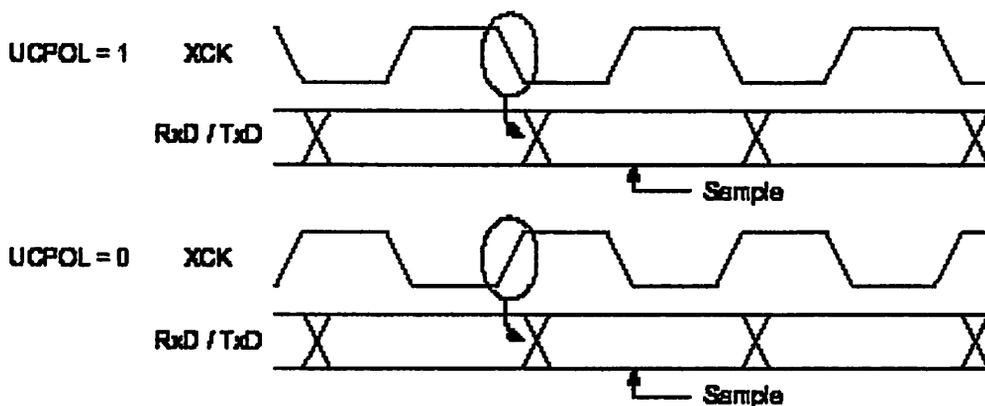
Proses ini mengenalkan dua *period delay clock* CPU dan oleh karena itu maksimal frekuensi clock XCK *eksternal* dibatasi oleh persamaan sebagai berikut

$$F_{xck} < f_{osc} / 4$$

Keterangan: fosc tergantung pada stabilitas sistem sumber clock.

Operasi Synchronous Clock

Ketika mode sinkron digunakan (UMSEL=1), pin XCK akan digunakan sama seperti clock input (*slave*) atau clock output (*master*). Dengan ketergantungan antara tepi clock dan data sampling atau perubahan data menjadi sama. Prinsip dasarnya adalah data input (on RxD) dicontohkan pada clock XCK berlawanan dari tepi data output (TxD) sehingga mengalami perubahan.



Gambar 12 Operasi synchronous Clock

UCPOL bit UCRSC memilih tepi yang mana clock XCK digunakan untuk data sampling dan yang mana digunakan untuk perubahan data. Seperti yang ditunjukkan pada gambar di atas, ketika UCPOL nol data akan diubah pada tepi kenaikan XCK dan dicontohkan pada tepi XCK saat jatuh. Jika UCPOL dalam kondisi set, data akan mengalami perubahan pada saat tepi XCK jatuh dan data akan dicontohkan pada saat tepi XCK naik.

Inisialisasi USART

USART harus diinisialisasi sebelum komunikasi manapun dapat berlangsung. Proses inisialisasi normalnya terdiri dari pengesetan *baud rate*, penyetingan *frame format* dan pengaktifan pengirim atau penerima tergantung pada pemakaian. Untuk interrupt menjalankan operasi USART, global *interrupt flag* (penanda) sebaiknya dibersihkan (dan *interrupt global disable*) ketika inisialisasi dilakukan.

Sebelum melakukan inisialisasi ulang dengan mengubah *baud rate* atau *frame format*, untuk meyakinkan bahwa tidak ada transmisi berkelanjutan sepanjang periode *register* yang diubah. Flag TXC dapat digunakan untuk mengecek bahwa pemancar telah melengkapi semua pengiriman, dan flag RXC dapat digunakan untuk mengecek bahwa tidak ada data yang tidak terbaca pada buffer penerima. Tercatat bahwa flag TXC harus dibersihkan sebelum tiap transmisi (sebelum UDR ditulisi) jika itu semua digunakan untuk tujuan tersebut.

REFERENSI

www.atmel.com.Datasheet AVR ATmega16

Biografi Penulis



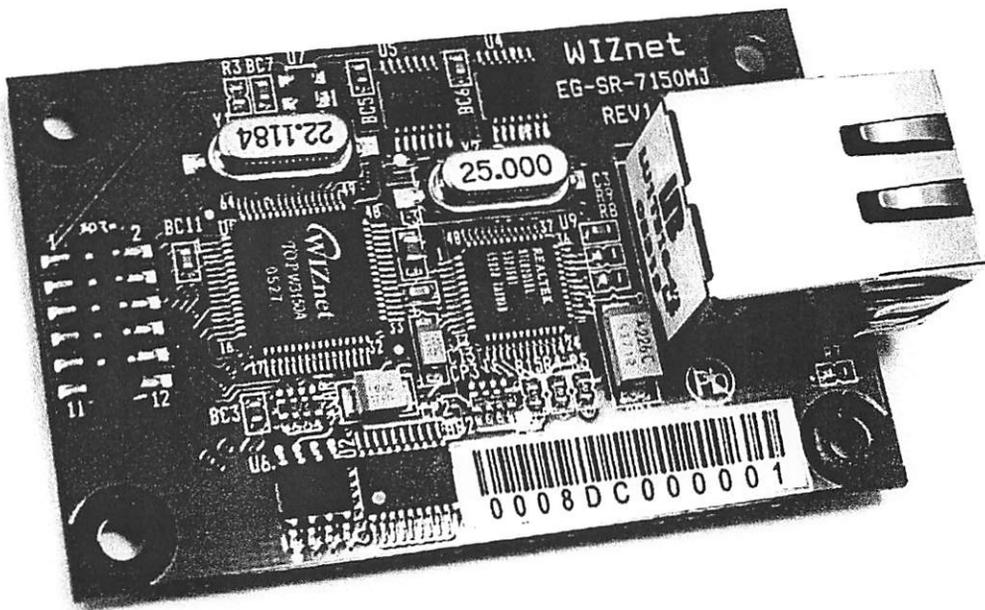
Mokh. Sholihul Hadi, lahir di Jombang 25 Mei 1982. Menyelesaikan pendidikan Sarjana di Jurusan Teknik Elektro Universitas Brawijaya Malang tahun 2004. Terhitung sejak tahun yang sama mengabdikan diri menjadi PNS sebagai dosen di Jurusan Teknik Elektro Universitas Negeri Malang. Beberapa penghargaan yang pernah diperoleh antara lain: Nominator Peneliti Muda terbaik tingkat Nasional bidang Ilmu Pengetahuan Teknik dan Rekayasa oleh LIPI tahun 2006, Medali Emas Pekan Ilmiah Mahasiswa Nasional tahun 2004, Medali Emas sebagai Peneliti Remaja terbaik tingkat Nasional bidang Ilmu

Pengetahuan Teknik dan Rekayasa oleh LIPI tahun 2004, Medali Perak Lomba Karya Ilmiah Olahraga Mahasiswa Indonesia Direktorat Jenderal Olahraga Republik Indonesia tahun 2004, Mahasiswa Teladan Universitas Brawijaya Malang tahun 2004 dll. Bidang penelitian yang sedang ditekuni antara lain: Robotika, Elektronika Medis, *Artificial Intelligent*, dan *Nano Technology*.

EG-SR-7150MJ

User's Manual

(V 1.1)



WIZnet

©2006 WIZnet Inc. All Rights Reserved.

For more information, visit our website at <http://www.wiznet.co.kr>

If you have any question about WIZnet Products, write them down onto our [Q&A Board](#) on our website at www.wiznet.co.kr. A WIZnet engineer will promptly provide you with an answer.



EG-SR-7150MJ User's Manual

1.	Introduction	3
1.1.	Key Features	3
1.2.	Products Contents (EG-SR-7150MJ-EVB)	3
1.3.	Specifications of the EG-SR-7150MJ.....	6
2.	Getting Started.....	7
2.1.	Hardware Installation procedure.....	7
2.2.	Configuration Tool	8
2.2.1.	Configuration tool features	8
2.3.	Serial Communication Specification	12
2.3.1.	Frame Format.....	12
2.3.2.	STX & ETX.....	12
2.3.3.	Reply Code.....	13
2.3.4.	Command Code	13
2.4.	Operation Flow	15
2.5.	Factory Default	15
3.	Demonstration and Test.....	16
3.1.	Case 1: Getting IP address using H/W trigger	16
3.2.	Case 2: Changing IP address using H/W trigger.....	16
3.3.	Case 3: Changing IP address using S/W trigger.....	17
4.	PIN Assignment and Dimensions	18
5.	Reference Schematic	20
6.	ETC.....	20
6.1.	Firmware Uploading through the FLIP software.....	20
6.2.	Warranty	21

1. Introduction

The EG-SR-7150MJ is a gateway module that converts serial data into TCP/IP data type. It transmits the data sent by a serial equipment to the Internet and TCP/IP data to the equipment.

With the EG-SR-7150MJ mounted with an RJ-45 connector, users can have an easier and quicker interface with the Ethernet.

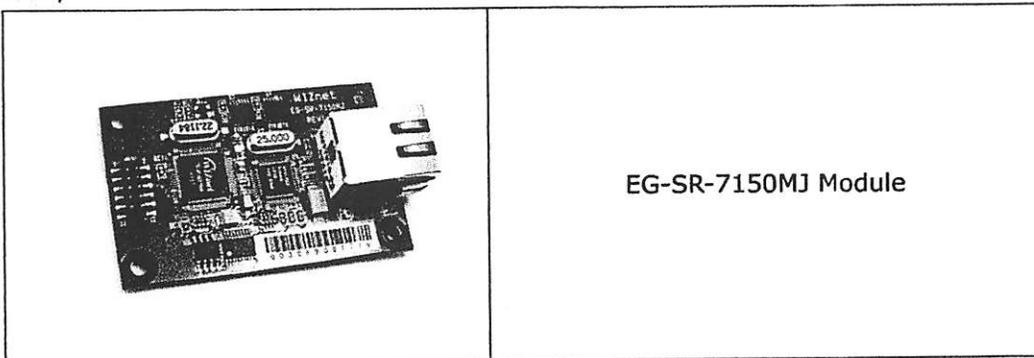
The EG-SR-7150MJ provides serial commands, with which the developers of any serial device can add local configuration capability to their products. For example, a card reader developer can program the keypad on a card reader to configure serial or network on-site without the use of a laptop or PC.

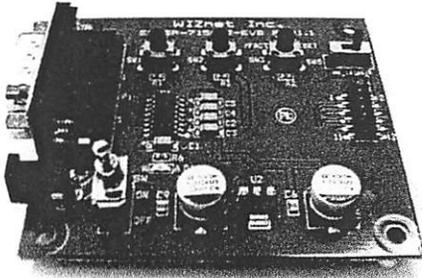
1.1. Key Features

- Ready-to-go serial to Ethernet gateway module mounted with an RJ-45 connector
- Serial Command Support
 - Simple command frame format
 - Comprehensive & readable command set for network and serial settings
 - On-site configuration without PC
- High stability & reliability by using a W3150A WIZnet Chip, a fully-hardwired TCP/IP stack
- Easy and powerful configuration program
- 10/100Mbps Ethernet interface, Max. 230Kbps Serial interface
- RoHS compliant

1.2. Products Contents (EG-SR-7150MJ-EVB)

The EG-SR-7150MJ-EVB, the evaluation kit for the EG-SR-7150MJ contains the following items;

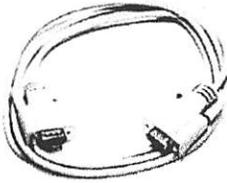




Test Board for EG-SR-7150MJ



12pin Cable
(Connecting EG-SR-7150MJ to Test Board)



Serial Cable
(Connecting Serial Device to Test Board)



LAN Cable
(Connecting EG-SR-7150MJ to Host)



Power
(DC 5V Adaptor)



CD
(Containing Manual, H/W & S/W
Materials)

☞ Please immediately notify your sales representative if any of the items above is missing or damaged.

1.3. Specifications of the EG-SR-7150MJ

Category	Specifications
Form Factor	2mm Pitch 2x6 pins, 62x40 mm
LAN Interface	10/100 Mbps auto-sensing, RJ-45 connector
Protocol	TCP, UDP, IP, ARP, ICMP, MAC, (IGMP, PPPoE)
CPU	AT89C51RC2 (8bit MCU and 32K Flash)
Serial Interface	RS 232 (LVTTTL)
Serial Signals	TXD, RXD, RTS, CTS, GND
Serial Parameters	Parity : None, Even, Odd Data bits : 7, 8 Flow control : RTS/CTS, XON/XOFF Speed : up to 230Kbps
Management	Configuration utility based on Windows
Temperature	0℃~70℃ (Operating), -40℃~85℃ (Storage)
Humidity	10~90%
Power	150mA @ 3.3V (max)
Size	40mm x 62mm x 17mm

2. Getting Started

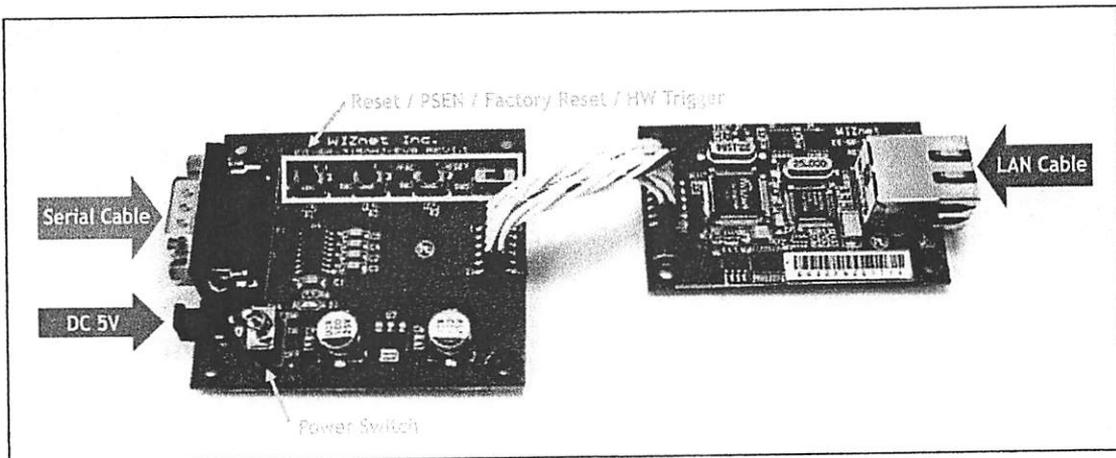
This Chapter describes how to set up and configure the EG-SR-7150MJ.

The following items are required to get started.

- Power Cable (included in the EG-SR-7150MJ-EVB package)
- Serial and Ethernet Cables (included in the of EG-SR-7150MJ-EVB package)
- PC or Laptop with Network Interface Card (hereafter, NIC) and/or one RS232 serial port

2.1. Hardware Installation procedure

Follow steps below to prepare the module and evaluation board for testing.



STEP 1: Connect the EG-SR-7150MJ module to the test board by using the 12pin cable.

STEP 2: Connect the 5V DC power line to the power jack of the test board.

STEP 3: Use the RJ45 Ethernet cable in order to connect the module to an Ethernet network.

STEP 4: Use the serial data cable to connect the test board to a serial device.

2.2. Configuration Tool

2.2.1. Configuration tool features

① Search

The Search function is used to search all modules existing on the same Subnet. The UDP broadcast is used for searching modules on a LAN.

The MAC address for a searched module will be listed in the "Module list".

If **Direct IP Search** is checked, TCP will be used for searching instead of UDP. This mode is used more for searching the EG-SR-7150MJ modules on remote networks than local networks with the same subnet. An IP address assigned to the module will be required.

② **Setting**

If you select one of the MAC addresses listed in the "Module list", the configuration value of the selected module will be displayed. After changing each value in the configuration program, click the "Setting" button to complete the configuration.

The module will be initialized with the new configurations.

③ **IP Configuration method: Static, DHCP**

Static: The IP address can be manually assigned by users.

DHCP: The module assigns IP, subnet and gateway addresses by acquiring them from the DHCP server

☞ Other configurations should be set manually except for the IP configuration of DHCP.

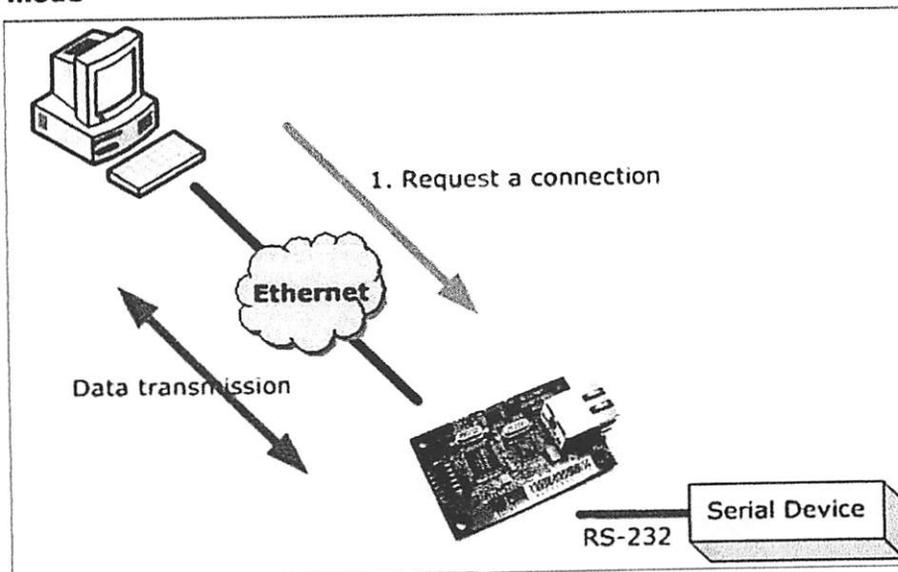
④ **Operation mode: TCP server, TCP client, UDP**

Three different operation modes are supported — TCP Server, TCP Client, and UDP.

The main difference between the TCP and UDP protocols is that TCP guarantees the delivery of data by requesting the recipient to send an acknowledgement to the sender. On the other hand, UDP does not require this type of verification, so data can be delivered quicker, but its delivery can not be guaranteed.

The TCP Server and TCP Client mode are related to the first step of connection establishment. Once the connection is established, data will be transparently transmitted in both directions (from Server to Client or from Client to Server).

TCP server mode

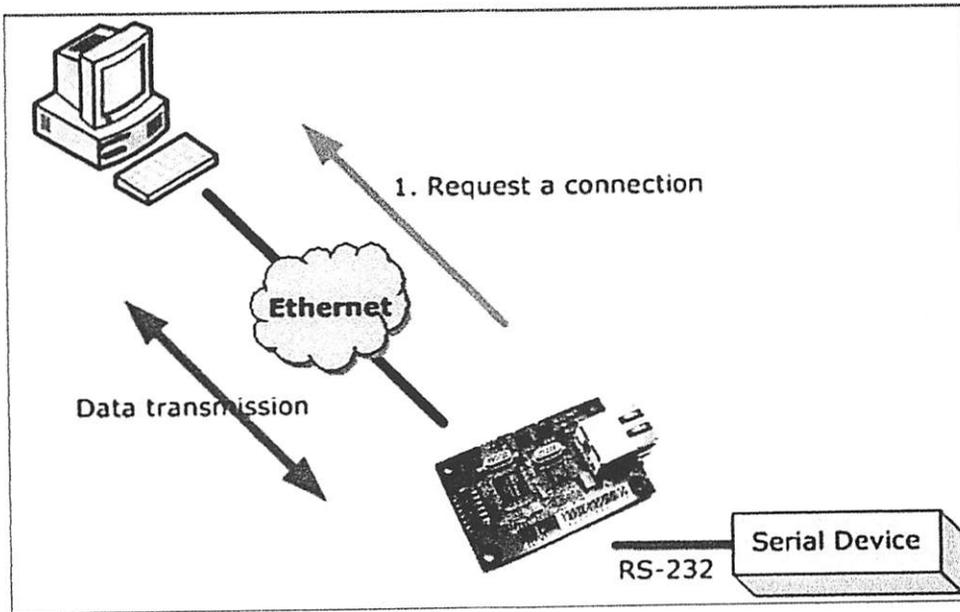


To operate this mode, the **Local IP, Subnet, gateway address and local port number** should be configured. The EG-SR-7150MJ waits to be connected by the host computer, allowing the host computer to establish a connection and get data from the serial device.

As illustrated in the figure above, the data transmission is as follows:

1. The host connects to the EG-SR-7150MJ which is configured as TCP Server Mode.
2. Once the connection is established, data can be transmitted in both directions - from the host to the EG-SR-7150MJ, and from the EG-SR-7150MJ to the host.

TCP client mode

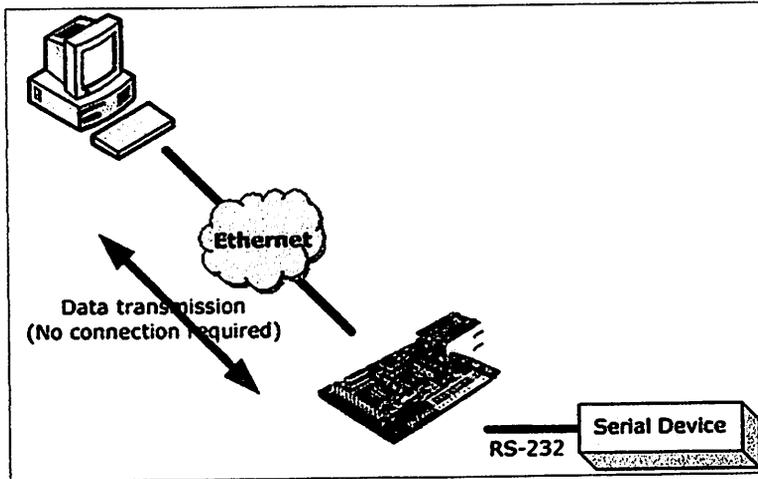


To operate this mode, the **Local IP, Subnet, gateway address, server IP, server port number** should be set. In the **TCP Client mode**, the EG-SR-7150MJ proceeds active open for establishing a TCP connection to a host computer.

As illustrated in the figure above, data transmission is as follows:

1. The EG-SR-7150MJ operating as TCP Client Mode establishes a connection based on the condition set in the **TCP client connection method (Startup, Any character)**. i.e. the EG-SR-7150MJ can try to connect as soon as one starts up(**Startup**), or later when data from serial device arrives. (**Any character**).
2. After the connection is established, data can be transmitted in both directions - from the host to the EG-SR-7150-MJ, and from the EG-SR-7150-MJ to the host.

UDP mode

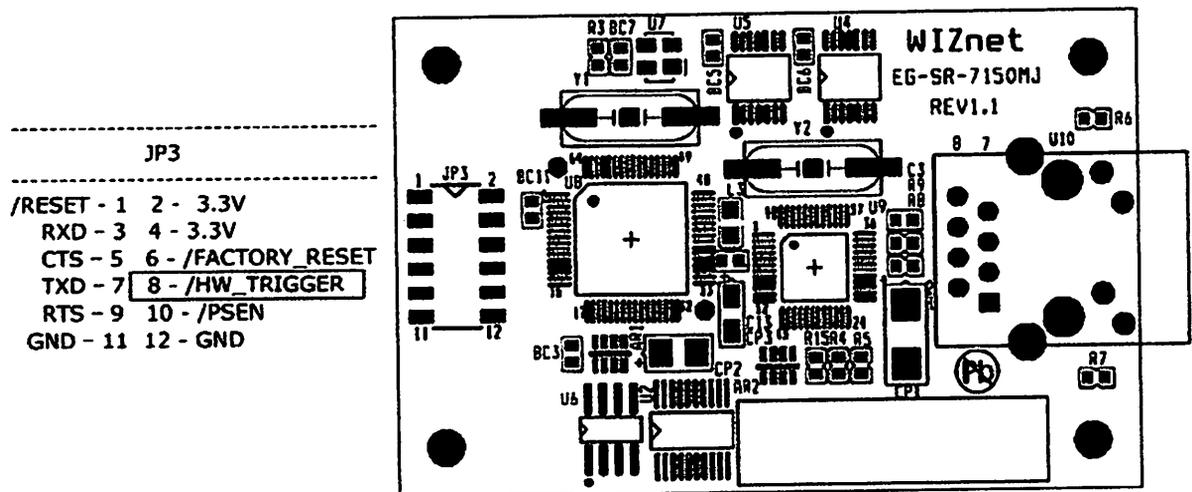


In UDP mode, any TCP/IP connection procedure is not required.

⑤ Serial command method: H/W trigger, S/W trigger

With this menu, you can designate how the Serial command mode can be entered. Two types are supported - H/W Trigger and S/W Trigger.

H/W trigger: Serial command mode can be triggered by pulling H/W trigger pin to low. It can be exited by pulling it to high.



S/W trigger: Serial command mode can be triggered when 3 user-defined characters are detected. It can be exited by using the WR command.

⑥ Delimiter: Time, Size, Character

You can designate how the serial data can be packed and sent to the Ethernet. There are 3 delimiters - Time, Size and Character. If all of them are set as '0', whenever the serial data arrives, they will be sent to the Ethernet without any condition. When any of the three delimiters is satisfied, data can be sent to the Ethernet.

Ex) Delimiter: Size=10, Char=0x0D

Serial data: 0123456789abc

Ethernet data: 0123456789

⇒ "abc" data remains in the serial buffer of module

⑦ Inactivity time

After the connection is established, if there is not any data transmission within the time defined in "Inactivity time", the connection will be automatically closed.

⑧ Upload

Upload the firmware through the network.

⇒ After uploading the firmware, 10~20 seconds are required for initialization.

2.3. Serial Communication Specification

In this chapter, we describe the structure of the data frames used in issuing commands and receiving responses to and from the device.

2.3.1. Frame Format

Command Frame format

Descriptor	STX	Command code	Parameter	ETX
Length(bytes)	1	2	Variable	1

Reply Frame format

Descriptor	STX	Reply code	Parameter	ETX
Length(bytes)	1	1	Variable	1

2.3.2. STX & ETX

Setting	Comments
STX	'<' : Hex = 3Ch
ETX	'>' : Hex = 3Eh

2.3.3. Reply Code

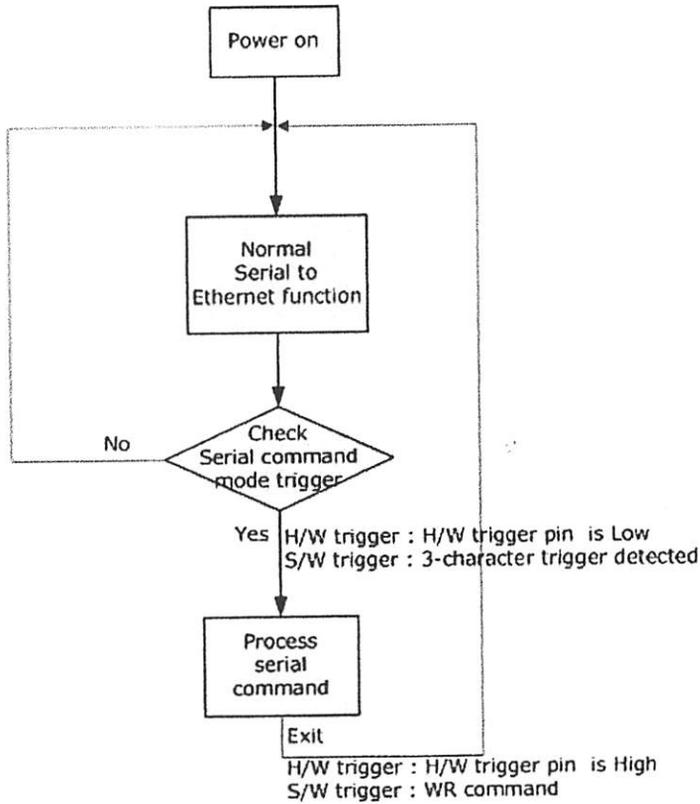
Reply	Comments
S	Command was successful
F	Command failed
1	Invalid command
2	Invalid parameter
E	Enter serial command mode

2.3.4. Command Code

Command	Parameter	Comments
WI	xxx.xxx.xxx.xxx (eg. 192.168.11.133)	Set Local IP
WS	xxx.xxx.xxx.xxx (eg. 255.255.255.0)	Set Subnet mask
WG	xxx.xxx.xxx.xxx (eg. 192.168.11.1)	Set Gateway
WP	0~65535	Set Local IP's port number
WD	0 : Static 1 : DHCP	Set the IP configuration method
WM	0 : TCP server 1 : TCP client 2 : UDP	Set the operation mode
WC	0 : startup 1 : any character	TCP client method
WB	XXXX eg. [Baudrate]1: 115200, 2: 57600, 3: 38400, 4: 19200, 5: 9600, 6: 4800, 7: 2400, 8: 1200 [data byte] 7 : 7bit, 8bit [parity] 0 : no parity, 1 : Odd, 2 : Even [Flow] 0 : no, 1 : Xon/Xoff, 2 : RTS/CTS	Set the serial baud rate, data, parity and flow control. 4bytes:[Baud][data byte][parity][flow]
WT	0 : Disable 1 : H/W trigger 2 : S/W trigger	Set the serial command method
WE	xxxxxx (eg. In hex format : 2B 2B 2B)	Set the command mode character

WX	xxx.xxx.xxx.xxx (eg. 192.168.11.144)	Set server IP address
WN	0~65535	Set server port number
WR		Restart
OC	XX	Set delimiter character in hex
OS	0~255	Set delimiter size
OT	0~65535	Set delimiter time
OI	0~65535	Set Inactivity timer value
Command	Parameter	Comments
RI		Get Local IP
RS		Get Subnet mask
RG		Get Gateway
RP		Get Local IP's port number
RD		Get the IP configuration method
RM		Get the operation mode
RC		Get the TCP client method
RB		Get the serial baud rate
RT		Get the serial command method
RE		Get the command mode character
RF		Get the firmware version
RX		Get the server IP address
RN		Get the server port number
QC		Get delimiter character in hex
QS		Get delimiter size
QT		Get delimiter time
QI		Get Inactivity timer value

2.4. Operation Flow



2.5. Factory Default

While the Factory Reset is low and the /Reset is applied, the module is initialized with the factory default value.

IP configuration	Static
Local IP address	192.168.11.2
Subnet mask	255.255.255.0
Gateway address	192.168.11.1
Local port number	5000
Server IP address	192.168.11.3
Server port number	5000
Operation mode	TCP server mode
Serial port	9600 bps 8-N-1
Serial command method	H/W trigger

3. Demonstration and Test

In this chapter, three examples are given to show how functions of the EG-SR-7150MJ can be tested. The testing environment is as follows:

Hardware

- ◆ PC having RS-232 serial port.
- ◆ EG-SR-7150MJ & Test board

Software

- ◆ Windows operating system installed on testing PC.
- ◆ EG-SR-7150MJ Configuration tool
- ◆ Hyper Terminal Program

Testing Structure

- ◆ Ethernet cross cable to connect the LAN ports of PC and EG-SR-7150MJ.
- ◆ RS-232 cable to connect the COM port of PC (usually COM1 or COM2) and serial port of EG-SR-7150MJ-EVB.

3.1. Case 1: Getting IP address using H/W trigger

STEP1: Configure the trigger mode as "H/W trigger" in the Configuration Tool.

STEP2: Check the serial port setting such as baud rate of the module.

STEP3: Start HyperTerminal program and set the serial port of PC to the setting of module checked in STEP2.

STEP4: Pull H/W trigger pin to low to enter the serial command mode.

STEP5: Use HyperTerminal program to send "<RI>" (command to request IP address)

STEP6: HyperTerminal program displays "<S192.168.11.2>"

(It indicates that the command was successfully executed and IP address is 192.168.11.2)

STEP7: Pull H/W trigger pin to high to exit the serial command mode

3.2. Case 2: Changing IP address using H/W trigger

STEP1: Configure the trigger mode as "H/W trigger" in the Configuration Tool.

STEP2: Check the serial port setting such as baud rate of the module.

STEP3: Start HyperTerminal program and set the serial port of PC to the setting of module checked in STEP2.

STEP4: Pull H/W trigger pin to low to enter serial command mode.

STEP5: Use HyperTerminal program to send "<WI192.168.11.10>"

(command to change the IP address as 192.168.11.10)

STEP6: HyperTerminal program displays "<S>"

(Indicates the command was successfully executed)

STEP7: Use HyperTerminal program to send "<RI>" (command to request IP address)

STEP8: HyperTerminal program displays "<S192.168.11.10>"

(Indicates the command was executed successfully and IP address is 192.168.11.10)

STEP9: Pull H/W trigger pin to high to exit serial command mode

☛ All changes are applied after exit the serial command mode

3.3. Case 3: Changing IP address using S/W trigger

STEP1: Configure the trigger mode as S/W trigger at the Configuration program, and check the three trigger characters. For example, assume the trigger is "25 25 25"

STEP2: Check the serial port setting such as baud rate of the module.

STEP3: Start HyperTerminal program and set the serial port of the PC to the serial setting of the module checked in STEP2.

STEP4: Use HyperTerminal program to send three trigger characters to enter the serial command mode; %%% (in hex :0x25 0x25 0x25) in this case.

STEP5: Use HyperTerminal program to send "<WI192.168.11.10>"

(command to change the IP address as 192.168.11.10)

STEP6: HyperTerminal program displays "<S>"

(Indicate the command was executed successfully)

STEP7: Use HyperTerminal program to send "<RI>" (command to request IP address)

STEP8: HyperTerminal program displays "<S192.168.11.10>"

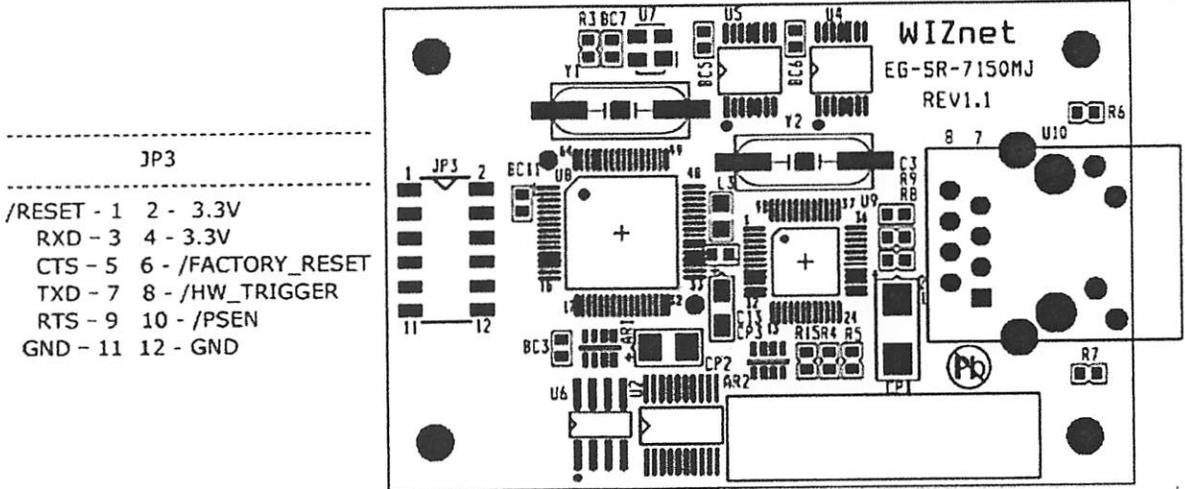
(Indicate the command was executed successfully and IP address is 192.168.11.10)

STEP9: Use HyperTerminal program to send "<WR>"

(command to exit serial command mode)

☛ All changes are applied after exiting the serial command mode.

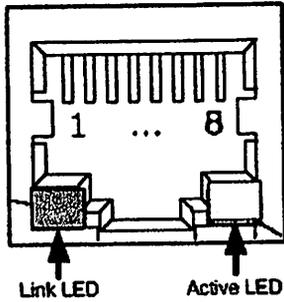
4. PIN Assignment and Dimensions



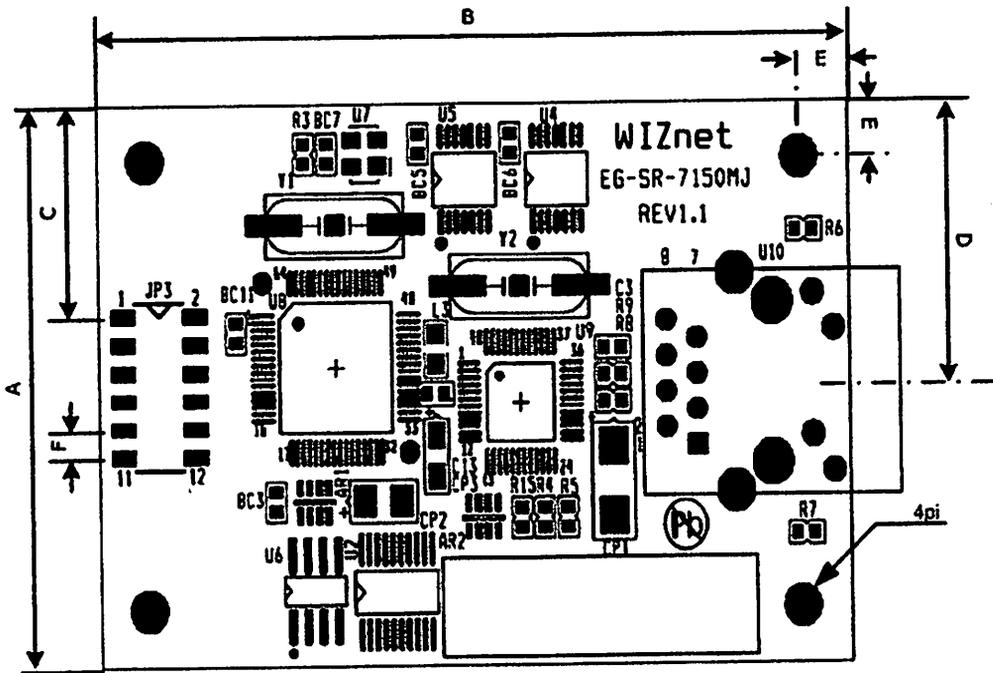
Name	Functions	I/O	
3.3V	Power		
/RESET	Low active reset Minimum 1.2 usec is required.	Input	
RXD	RS-232 Data Input	Input	
CTS	RS-232 Clear To Send	Input	Optional
TXD	RS-232 Data Output	Output	
RTS	RS-232 Request To Send	Output	Optional
Factory Reset	Pull Factory Reset to low and if /RESET is activated, the configuration is changed to factory default.	Input	
H/W Trigger	Pull H/W Trigger to low, enter the serial command mode	Input	
/PSEN	Pull /PSEN to low and if /RESET is activated, the module enter the bootloader for FLIP connection	Input	

➤ All signal levels are 3.3V LVTTTL.

Ethernet port Pinouts

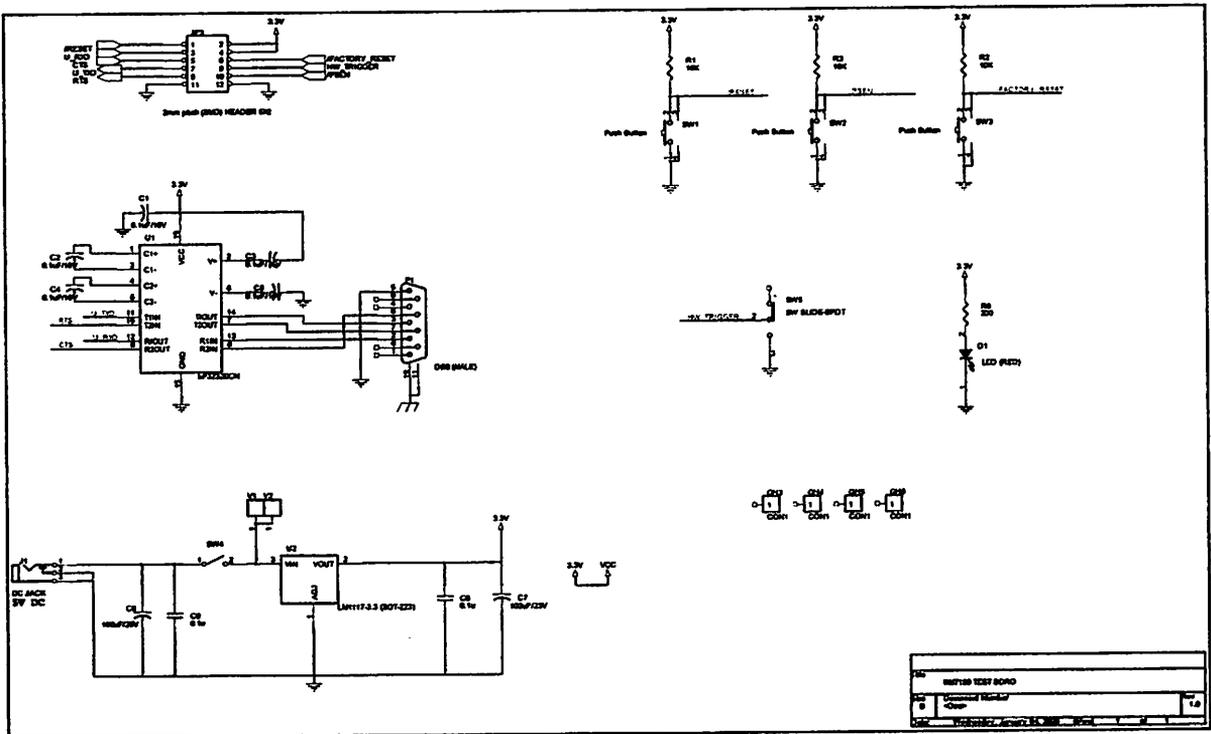


Pin	Signal
1	TX+
2	TX-
3	RX+
6	RX-



Symbol	Dimension(mm)
A	40
B	62
C	15
D	20
E	4
F	2

5. Reference Schematic



6. ETC

6.1. Firmware Uploading through the FLIP software

The following items are required to get started. :

- EG-SR-7150-MJ test board
- UART cross cable
- Program file in HEX file format
- FLIP utility installed on your PC

On a PC, one must have a file in HEX format to program the EG-SR-7150MJ. For example this file could be named "app.hex".

Step 1

Connect the EG-SR-7150MJ test board to a PC with the UART cable supplied.

Important : If you have any program running on your PC with which COM port is used such as "Hyperterminal", be sure to connect the cable to the COM port not used.

Step 2

Power on the test board.

While pressing the /PSEN button, click the /RESET button. Then release the /PSEN button

Step 3

Run the ISP software named FLIP by ATMEL.

Step 4

Select the device by pushing the F2 button. Here you must choose AT89C51RC2.

Step 5

Set up the communication port by pushing the F3 button. Make sure to select the same port as the one you have plug in the UART cable of the EG-SR-7150MJ test board.

Step 6

Now, you should be connected to the board and able to program.

Now you will have to browse your PC to load your file in hex format.

Step 7

After programming, check if the BSB, SBV and SSB are set as FF, 00 and FF respectively.

6.2. Warranty

WIZnet Co., Ltd offers the following limited warranties applicable only to the original purchaser. This offer is non-transferable.

WIZnet warrants our products and its parts against defects in materials and workmanship under normal use for period of standard ONE(1)YEAR for the EG-SR-7150MJ Module, Evaluation Board and labor warranty after the date of original retail purchase. During this period, WIZnet will repair or replace a defective products or part free of charge.

Warranty Conditions:

1. The warranty applies only to products distributed by WIZnet or our official distributors.
2. The warranty applies only to defects in material or workmanship as mentioned above in 6.2 Warranty. The warranty applies only to defects which occur during normal use