

BAB II

LANDASAN TEORI

2.1 Restoran

Restoran adalah sebuah industri yang luar biasa dan hampir tidak pernah mati. Industri yang penuh potensi, prospek, berkembang dengan sangat cepat.

Usaha restoran membutuhkan ketekunan pengolahan dengan ketelitian, kecermatan, dan kerja keras, serta bukan sekedar cara memasak yang enak saja namun harus diperhatikan juga secara terus menerus semua bidang yang berkaitan. Pengadaan peralatan, perlengkapan, bahan mentah, pengendalian kualitas bahan, pengolahan dengan proses dan resep yang standar, merupakan faktor yang penting bagi keberhasilan usaha restoran. Faktor-faktor lain yang ikut mempengaruhi adalah selera dan keluhan pelanggan, faktor hukum, kegiatan promosi, masalah karyawan, pendanaan, penetapan harga, citra restoran, dan sebagainya. (Irene, 2013)

Keberadaan restoran, *cafe*, dan sejenisnya makin marak belakangan ini. Sehingga memunculkan persaingan bisnis antara restoran satu dengan yang lainnya. Salah satunya tentang pelayanan, kebanyakan restoran menggunakan pelayan untuk melayani pemesanan menu makanan. Jika pelayanan ingin maksimal dan lebih unggul dengan restoran lainnya maka pihak pemilik rumah makan harus menyediakan tenaga pelayan yang sebanding dengan jumlah meja pelanggan yang ada. Tetapi pada kenyataan jumlah pelayan yang ada tidak sebanding dengan jumlah meja pelanggan, sehingga menyebabkan pelayanan yang tidak maksimal.

2.2 TCP/IP

TCP merupakan protokol yang bersifat *connection oriented*. Artinya sebelum proses transmisi data terjadi, dua aplikasi TCP harus melakukan pertukaran kontrol informasi (*handshaking*). TCP juga bersifat *reliable* karena menerapkan fitur deteksi kesalahan dan retransmisi apabila ada data yang rusak. Sehingga keutuhan data dapat terjamin. Sedangkan *byte stream service* artinya paket akan dikirimkan ke tujuan secara berurutan (*sequencing*).

Protokol TCP bertanggung jawab untuk pengiriman data dari sumber ke tujuan dengan benar. TCP dapat mendeteksi kesalahan atau hilangnya data dan melakukan pengiriman kembali sampai data diterima dengan lengkap. TCP selalu meminta konfirmasi setiap kali data dikirim, untuk memastikan apakah data telah sampai di tempat tujuan. Kemudian TCP akan mengirimkan data berikutnya atau melakukan retransmisi (pengiriman ulang) apabila data sebelumnya tidak sampai atau rusak. Data yang dikirim dan diterima kemudian diatur berdasarkan nomor urut.

Tabel 2.1 Bagan Segmen TCP

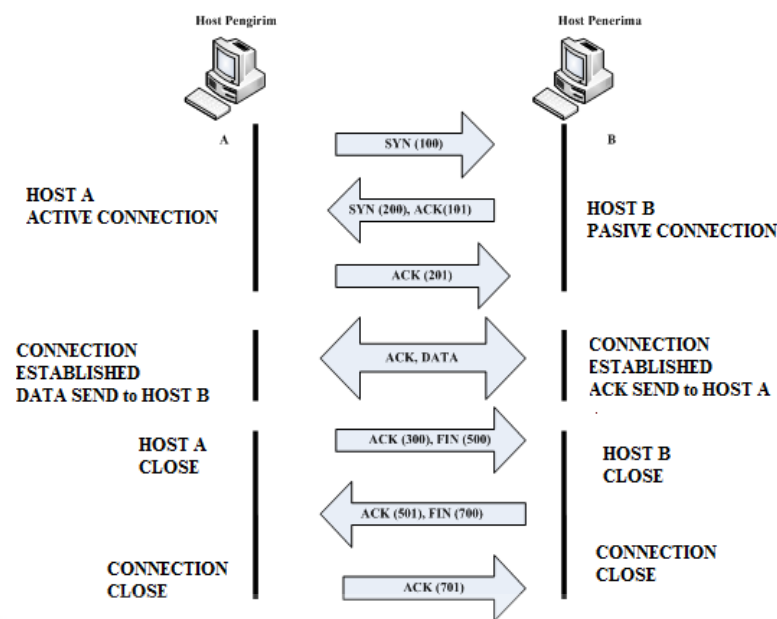
BIT OFFSET	0-3	4-7	8-15	16-31
0	Source Port			Destination Port
32	Sequence Number			
64	Acknowledgment Number			
96	Data Offset	Reserved	CWR ECE URG ACK PSH RST SYN FIN	Window Size
128	Checksum			Urgent Pointer
160	OPTIONAL			
160/192+	DATA			

Sumber : Cisco CCNA & Jaringan Komputer

Keterangan :

1. *Source Port* (16 bits) : Berisi informasi port pengirim
2. *Destination Port* (16 bits) : Berisi informasi port penerima
3. *Sequence Number* (32 bits) : Berupa sequence number yang terdiri atas dua kondisi berikut:
 - a. Jika *flag SYN* di-set (yang ada di bagian *field Flags*), maka *field* ini berisi awal (inisial) dari *sequence number*.
 - b. Jika *flag SYN* tidak di-set, maka nilai pada *field* ini merupakan *sequence number*.
4. *Acknowledgement* atau ACK (32 bits) : jika *flag ACK* di-set, maka nilai pada *field* ini adalah nilai *sequence number* berikutnya yang di-“harapkan” oleh penerima.
5. *Data offset* (4 bits) : Menunjukkan ukuran TCP header. Total header sepanjang 32-bit words. Ukuran minimum header adalah 5 word. *Data offset* juga merupakan awal dari data.
6. *Reserved* (4 bits) : Untuk keperluan tertentu di masa akan datang. Nilai pada *field* ini semestinya adalah *zero* (nol).
7. *Flags* (8 bit) : *field* untuk kontrol bit (masing-masing 1 bit), yaitu :
 - a. CWR (*Congestion Windows Reduced*)
 - b. ECE (*ECN-Echo*)
 - c. URG (*URGent*)
 - d. ACK (*ACKnowledgement*)
 - e. PSH (*Push function*)
 - f. RST (*Reset*)

- g. SYN (*Synchronize*)
 - h. FIN (*Finish*)
8. *Window* (16 bits) : menunjukkan ukuran *window* penerima (*receive window*). Agar data dapat diterima dengan baik maka diperlukan pengaturan ukuran jumlah byte optimal yang ditentukan oleh field ini.
 9. *Checksum* (16 bits) : digunakan untuk *error-checking* dari header dan data
 10. *Urgent pointer* (16 bits) : digunakan untuk *sequence number* yang menandakan *urgent data* byte terakhir.
 11. *Option* (*Variable bits*) : Berisi berbagai opsi berupa angka sebagai berikut :
 - a. 0-End of option list
 - b. 1-No operation list
 - c. 2-Maximum segment size
 - d. 3-Window scale
 - e. 4-Selective Acknowledgement
 - f. 5,6,7
 - g. 8-Timestamp
 12. *Data* : Berisi data yang dikirim.



Sumber : Cisco CCNA & Jaringan Komputer

Gambar 2.1 Pengiriman data menggunakan TCP

Pada Gambar 2.1 terlihat komunikasi menggunakan TCP antara *host A* dan *host B*. Bahwa untuk memulai membuka suatu hubungan, *host A* harus terlebih dahulu mengirimkan paket SYN. Setelah *host B* menerima paket tersebut, lalu mengirim paket SYN serta meng-ACK paket SYN yang berasal dari *host A*. Saat *host A* menerima paket ini, dia akan meng-ACK serta mengirimkan data miliknya. Koneksi kemudian terbuka.

Setelah koneksi terbuka, *host A* mengirim paket data yang sudah di beri nomer. Setiap kali sebuah paket tiba di *host B*, maka *host B* akan mengeceknya dan mengirim ACK yang menandakan paket telah diterima dengan selamat. Proses ini berlangsung berulang-ulang hingga seluruh paket dikirim semuanya dengan selamat.

Setelah paket terakhir dikirim, *host A* mengirim ACK dan FIN yang meminta *host B* untuk memutuskan koneksi. *Host B* akan meng-ACK dan

mengirim paket FIN kepada *host* A. Selanjutnya *host* A meng-ACK paket FIN yang berasal dari *host* B, maka koneksi pun diputus.

Protokol TCP sangat cocok digunakan untuk koneksi yang membutuhkan kehandalan tinggi, seperti aplikasi telnet, ssh, ftp, http, dan beberapa layanan lainnya. (Sofana, 2009)

2.3 *Microcontroller*

Microcontroller adalah single chip komputer yang memiliki kemampuan untuk diprogram dan digunakan untuk tugas-tugas yang berorientasi kontrol. Karena kemampuannya yang tinggi, bentuknya yang kecil, konsumsi dayanya yang rendah dan harga yang murah maka *microcontroller* begitu banyak digunakan di dunia. *Microcontroller* digunakan mulai dari bidang elektronik, otomotif, industri, telekomunikasi, medis, sampai dengan pengendali robot serta persenjataan militer. AVR merupakan seri *microcontroller* 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*. Beberapa diantaranya mempunyai ADC dan PWM internal. AVR juga mempunyai *In-System Programmable Flash on-chip* yang mengijinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial SPI. (Purbo, 2010)

2.3.1 *Microcontroller AVR*

Secara historis, mikrokontroler seri AVR pertama kali diperkenalkan ke pasaran sekitar tahun 1997 oleh perusahaan Atmel. Mikrokontroler AVR diklaim memiliki arsitektur dan set instruksi yang benar-benar baru dan berbeda. AVR merupakan mikrokontroler dengan arsitektur RISC (*Reduce Instruction Set Computer*) dengan lebar bus data 8 bit. Frekuensi kerja mikrokontroler AVR ini pada dasarnya sama dengan frekuensi osilator sehingga hal tersebut menyebabkan kecepatan kerja AVR untuk frekuensi osilator yang sama akan dua belas kali lebih cepat dibandingkan dengan mikrokontroler seri AT89S51/52. (Susilo, 2010).

2.3.2 *Fasilitas ATmega 16*

Microcontroller Advance Virtual RISC (AVR) merupakan *microcontroller* yang dibuat oleh perusahaan Atmel. Jenis *microcontroller* ini sangat banyak digunakan oleh para pengembang peralatan-peralatan elektronika. *Microcontroller* yang digunakan pada Tugas Akhir ini yaitu 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*

- a. *131 Powerful Instructions – Most Single Clock Cycle Execution*
- b. *32 x 8 General Purpose Fully Static Operation*
- c. *Up to 16 MIPS Throughput at 16 MHz*
- d. *On-chip 2-cycle Multiplier*

2. *Nonvolatile Program and Data Memories*

- a. *16K Bytes of In-System Self-Programmable Flash*

- b. Optional Boot Code Section with Independent Lock Bits*
- c. 512 Bytes EEPROM*
- d. 1K Bytes Internal SRAM*
- e. Programming Lock for Software Security*

3. Peripheral Features

- a. Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes*
- b. One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode*
- c. Real Time Counter with Separate Oscillator*
- d. Four PWM Channels*
- e. 8-channel, 10-bit ADC*
- f. Byte-oriented Two-wire Serial Interface*
- g. Programmable Serial USART*
- h. Master/Slave SPI Serial Interface*

4. Special Microcontroller Features

- a. Power-on Reset and Programmable Brown-out Detection*
- b. Internal Calibrated RC Oscillator*
- c. External and Internal Interrupt Sources*
- d. Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Powerdown, Standby and Extended Standby*

5. I/O and Package

- a. 32 Programmable I/O Lines*
- b. 40-pin PDIP, 44-lead TQFP, 44-lead PLCC, and 44-pad MLF*

6. Operating Voltages

a. 2.7 - 5.5V for Atmega16L

b. 4.5 - 5.5V for Atmega16

Pada Gambar 2.2 merupakan susunan kaki standar 40 pin *microcontroller*

AVR ATMega 16.

(XCK/T0) PB0	1	40	PA0 (ADC0)
(T1) PB1	2	39	PA1 (ADC1)
(INT2/AIN0) PB2	3	38	PA2 (ADC2)
(OC0/AIN1) PB3	4	37	PA3 (ADC3)
(SS) PB4	5	36	PA4 (ADC4)
(MOSI) PB5	6	35	PA5 (ADC5)
(MISO) PB6	7	34	PA6 (ADC6)
(SCK) PB7	8	33	PA7 (ADC7)
RESET	9	32	AREF
VCC	10	31	GND
GND	11	30	AVCC
XTAL2	12	29	PC7 (TOSC2)
XTAL1	13	28	PC6 (TOSC1)
(RXD) PD0	14	27	PC5 (TDI)
(TXD) PD1	15	26	PC4 (TDO)
(INT0) PD2	16	25	PC3 (TMS)
(INT1) PD3	17	24	PC2 (TCK)
(OC1B) PD4	18	23	PC1 (SDA)
(OC1A) PD5	19	22	PC0 (SCL)
(ICP) PD6	20	21	PD7 (OC2)

Sumber : *Datasheet* ATMega 16

Gambar 2.2. Konfigurasi Port I/O ATMega 16

2.4 Internet Protokol TCP/IP

Internet protocol suite atau *Transmission Control Protocol/Internet Protocol* (TCP/IP) adalah standar komunikasi data yang digunakan oleh komunitas internet dalam proses tukar-menukar data dari satu komputer ke komputer lain di dalam jaringan Internet. Protokol ini tidaklah dapat berdiri sendiri, karena memang protokol ini berupa kumpulan protokol (*protocol suite*). Protokol ini juga merupakan protokol yang paling banyak digunakan saat ini. Data tersebut diimplementasikan dalam bentuk perangkat lunak (*software*) di sistem operasi. Istilah yang diberikan kepada perangkat lunak ini adalah TCP/IP *stack*.

Protokol TCP/IP dikembangkan pada akhir dekade 1970-an hingga awal 1980-an sebagai sebuah protokol standar untuk menghubungkan komputer-komputer dan jaringan untuk membentuk sebuah jaringan yang luas. TCP/IP merupakan sebuah standar jaringan terbuka yang bersifat independen terhadap mekanisme transport jaringan fisik yang digunakan, sehingga dapat digunakan di mana saja. Protokol ini menggunakan skema pengalamatan yang sederhana yang disebut sebagai alamat IP (*IP Address*) yang memungkinkan hingga beberapa ratus juta komputer untuk dapat saling berhubungan satu sama lainnya di Internet. Protokol ini juga bersifat *routable* yang berarti protokol ini cocok untuk menghubungkan sistem-sistem berbeda (seperti *Microsoft Windows* dan keluarga *UNIX*) untuk membentuk jaringan yang heterogen.

Protokol TCP/IP selalu berevolusi seiring dengan waktu, mengingat semakin banyaknya kebutuhan terhadap jaringan komputer dan Internet. Pengembangan ini dilakukan oleh beberapa badan, seperti halnya *Internet Society* (ISOC), *Internet Architecture Board* (IAB), dan *Internet Engineering Task Force* (IETF). Macam-macam protokol yang berjalan di atas TCP/IP, skema pengalamatan, dan konsep TCP/IP didefinisikan dalam dokumen yang disebut sebagai *Request for Comments* (RFC) yang dikeluarkan oleh IETF.

Secara umum lapisan protokol dalam jaringan komputer dapat dibagi atas tujuh lapisan. Dari lapisan terbawah hingga tertinggi dikenal *physical layer*, *link layer*, *network layer*, *transport layer*, *session layer*, *presentation layer* dan *application layer*. Masing-masing lapisan mempunyai fungsi masing-masing dan tidak tergantung antara satu dengan lainnya (Purbo, 2011).

2.5 Transmisi Data

Data biasanya dikirim dari komputer ke peralatan lain. (Muthusubramanian, 2000). *Mode* Transmisi data dapat digolongkan menjadi dua bagian berdasarkan cara pengiriman datanya yaitu :

1. Transmisi Serial

Data dikirimkan 1 bit demi 1 bit lewat kanal komunikasi yang telah dipilih.

2. Transmisi Paralel

Data dikirim sekaligus misalnya 8 bit bersamaan melalui 8 kanal komunikasi, sehingga kecepatan penyaluran data tinggi, tetapi karakteristik kanal harus baik dan mengatasi masalah “Skew” yaitu efek yang terjadi pada sejumlah pengiriman bit secara serempak dan tiba pada tempat yang dituju dalam waktu yang tidak bersamaan.

Untuk dapat melakukan pengiriman data maka *mode transmisi* dapat pula dibedakan berdasarkan cara sinkronisasinya yaitu sebagai berikut :

1. Asinkron.

Pengiriman data dilakukan 1 karakter setiap kali, sehingga penerima harus melakukan sinkronisasi agar bit data yang dikirim dapat diterima dengan benar.

Berikut ini adalah beberapa ciri dari sinkronisasi asinkron :

- a. Trasmisi kecepatan tinggi.
- b. Satu karakter dengan yang lainnya tidak ada waktu antara yang tetap.
- c. Bila terjadi kesalahan maka 1 blok data akan hilang.
- d. Membutuhkan *start pulse/start bit* (tanda mulai menerima *bit* data).
- e. *Idle transmitter* = ‘1’ terus menerus, sebaliknya ‘0’.

- f. Tiap karakter diakhiri dengan *stop pulse/stop bit*.
- g. Dikenal sebagai *start-stop transmission*.

2. Sinkron.

Pengiriman sinkron merupakan pengiriman data dimana penerima dan pemancar melakukan sinkronisasi terlebih dahulu dengan menggunakan sebuah *clock* dalam melakukan sinkronisasi. Berikut ini adalah beberapa ciri dari sinkronisasi sinkron :

- a. Pengiriman dilakukan per-blok data.
- b. Sinkronisasi dilakukan setiap sekian ribu bit data.
- c. Transmisi kecepatan tinggi.
- d. Tiap karakter tidak memerlukan bit awal/akhir.
- e. Dibutuhkan 16-32 bit untuk sinkronisasi.
- f. Bila terjadi kesalahan, 1 blok data akan hilang.
- g. Pemakaian saluran komunikasi akan efektif, karena transmisi hanya dilakukan bila dimiliki sejumlah blok data.
- h. Pengirim dan penerima bekerja sama, karena sinkronisasi dilakukan dengan mengirimkan pola data tertentu (karakter sinkronisasi) antara pengirim dan penerima.

3. Isokron

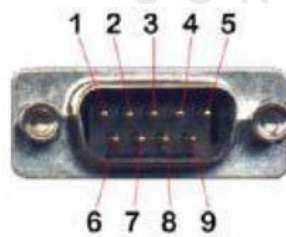
Merupakan kombinasi antara asinkron dan sinkron. Tiap karakter diawali dengan *start bit* dan diakhir data ditutup dengan *stop bit*, tetapi pengirim dan penerima disinkronisasikan.

2.5.1 Transmisi Data Serial

Transmisi data secara serial adalah transmisi data di mana data tersebut akan dikirimkan tiap bit dalam satuan waktu.

Terdapat dua cara dalam mentransmisikan data secara serial, yaitu secara *synchronous* dan *asynchronous*. Transmisi secara *synchronous* yaitu pengiriman data serial bersamaan dengan sinyal *clock*, sedangkan *asynchronous* yaitu pengiriman data serial tidak bersamaan dengan sinyal *clock* sehingga *receiver* harus membangkitkan sinyal *clock* sendiri (tidak perlu sinkronisasi). Berdasarkan arah proses komunikasi serial terdapat tiga metode, yaitu *Simplex*, *Half-Duplex*, dan *Full-duplex*.

Satuan kecepatan transfer data (*baudrate*) pada komunikasi serial adalah *bits per second* (bps). Untuk menjaga kompatibilitas dari beberapa peralatan komunikasi data yang dibuat oleh beberapa pabrik, pada tahun 1960 *Electronics Industries Association* (EIA) menstandarkan antarmuka serial dengan nama RS232. Gambar 2.3 adalah pin serial dan Tabel 2.2 adalah penjelasan dari Pin serial tersebut (Wahyuni, 2013).



Gambar 2.3. Pin Serial

Tabel 2.2. Penjelasan Pin Serial

PIN	NAMA	DESKRIPSI
1	CD	<i>Carrier Detect</i>
2	RXD	<i>Receive Data</i>
3	TXD	<i>Transmit Data</i>
4	DTR	<i>Data Terminal Ready</i>
5	GND	<i>System Ground</i>

6	DSR	<i>Data Set Ready</i>
7	RTS	<i>Request To Send</i>
8	CTS	<i>Clear To Send</i>
9	RI	<i>Ring Indicator</i>

2.6. WIZ110SR

WIZ110SR merupakan modul *serial to ethernet gateway* yang beredar dipasaran. Modul ini digunakan untuk menghubungkan antara komputer *server* dengan *microcontroller* agar dapat berkomunikasi. Modul yang digunakan pada Tugas Akhir yaitu WIZ110SR. WIZ110SR digunakan untuk mengubah data serial ke format data TCP/IP (*Ethernet*). WIZ110SR mempunyai karakteristik sebagai berikut:

1. Berbasis W5100 & GC89L591A0.
2. Protokol TCP, UDP, IP, ARP, ICMP, MAC, PPPoE.
3. Antarmuka *ethernet* 10/100 Mbps (auto).
4. Antarmuka UART RS232, hingga 230 Kbps.
5. Catu daya 5V DC. [Wiznet, 2007].

Modul WIZ110SR tersebut dapat dilihat pada Gambar 2.4.



Sumber : WIZ110SR *User's Manual*

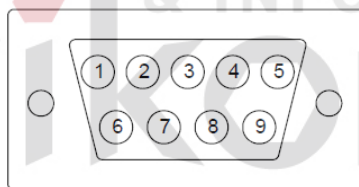
Gambar 2.4 Modul WIZ110SR

Spesifikasi dari modul WIZ110SR dapat dilihat pada Tabel 2.3 serta pinout port serial dapat dilihat pada Gambar 2.5.

Tabel 2.3 Spesifikasi WIZ110SR

ITEM	DESCRIPTION
MCU	8051 Compliant (having internal 26K Flash, 16K SRAM, 2K EEPROM)
TCP/IP	W5100 (Ethernet PHY Embedded)
Network Interface	10/100 Mbos auto-sensing RJ-45 Connector
Serial Interface	RS232
Serial Signal	TXD, RXD, RTS, CTS, GND
Serial Parameters	Parity : None, Even, Odd
	Data Bits : 7,8
	Flow Control : None, RTS/CTS, XON/XOFF
	Speed : up to 230Kbps
Input Voltage	DC 5V
Power Consumption	Under 180mA
Temperature	0°C~ 80°C (operation), -40°C ~ 85°C (storage)
Humidity	10~90%

Sumber : WIZ110SR User's Manual



Gambar 2.5. Pinout Port Serial Modul WIZ110SR

Penjelasan dari port serial dari modul WIZ110SR dapat dilihat pada

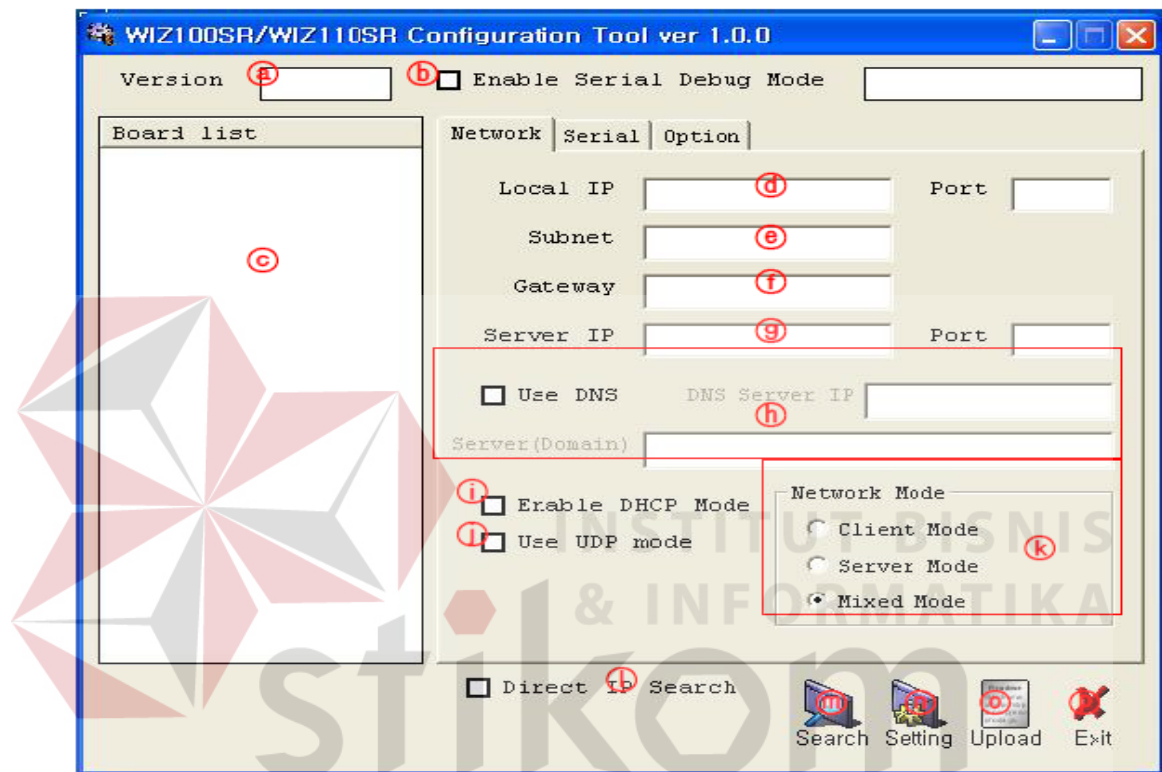
Tabel 2.4.

Tabel 2.4 Konfigurasi Pin WIZ110SR

Pin Number	Signal	Description
1	NC	Not Connected
2	RxD	Receive Data
3	TxD	Transmit Data
4	DTR	Data Terminal Ready
5	GND	Ground

6	DSR	Data Set Ready
7	RTS	Request To Send
8	CTS	Clear To Send
9	NC	Not Connected

2.7.1 Konfigurasi Network



Sumber : WIZ110SR User's Manual

Gambar 2.6 Layar Editor Konfigurasi Network

Pada Gambar 2.6 dapat diberikan penjelasan sebagai berikut :

- Menunjukkan versi *firmware* dari modul WIZ110SR.
- Memonitor status dan pesan dari koneksi serial melalui terminal.
- Menampilkan MAC *address* dari setiap modul WIZ110SR yang terhubung dalam satu jaringan.
- Mengisi alamat IP dan *port* yang diinginkan pada modul WIZ110SR.

- e. Mengisi *subnet mask* dari modul WIZ110SR.
- f. Mengisi alamat *gateway* dari modul WIZ110SR.
- g. Mengisi alamat IP dari *server* ketika modul dalam *mode client*.
- h. Mengisi alamat DNS *server* yang digunakan modul WIZ110SR.
- i. Mengaktifkan DHCP *server* pada modul WIZ110SR.
- j. Menggunakan *mode* UDP.
- k. Memilih *mode network* dari modul WIZ110SR yang tersedia dalam tiga *mode* yaitu *server mode*, *client mode*, dan *mixed mode*.
- l. Pencarian langsung melalui IP.

2.7.2. Network Mode

1. TCP Server Mode

Mode ini, WIZ110SR menunggu koneksi dari *client*. *Mode* ini sangat berguna untuk memonitoring perangkat yang ingin terhubung dengan perangkat dimana modul ini dipasang. Jika menjalankan *mode* ini *ip address*, *subnet*, *gateway* dan *local port* harus diisi supaya *setting network* dari *server* yang harus diketahui *client* agar terhubung ke *server*. Pada *mode* ini *serial device* dapat berkomunikasi dengan beberapa *ethernet device* sekaligus.

2. TCP Client Mode

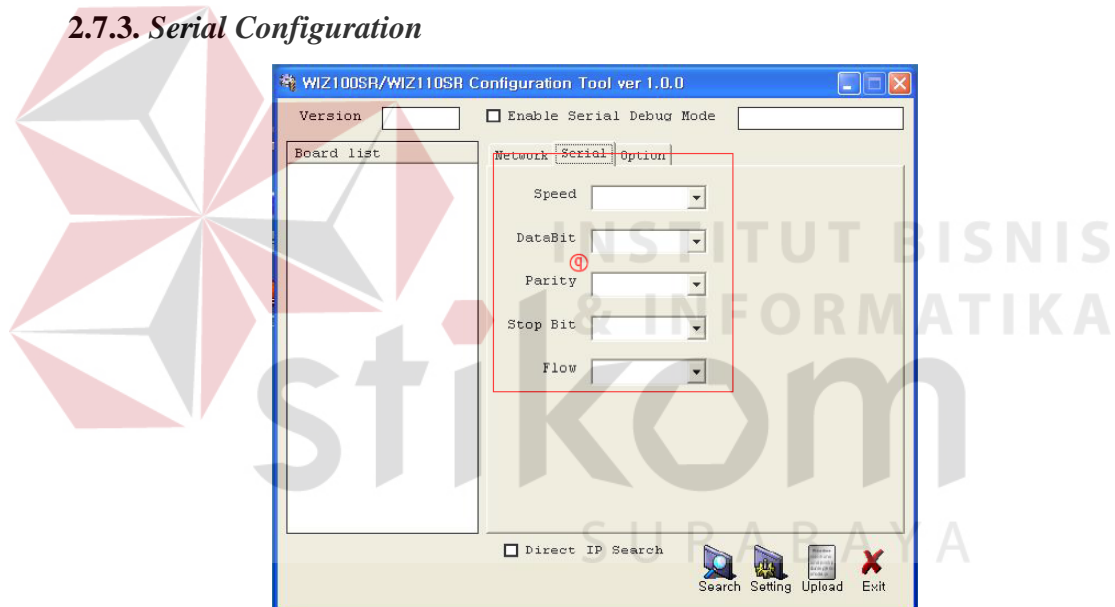
Mode ini modul akan mencari *server* dan membuat koneksi ke *server*. Jika menjalankan *mode* ini *ip address*, *subnet*, *gateway*, *dns* dan *local port* harus diisi. Perbedaannya dengan *setting server* adalah ketika modul menjadi *client*, maka kita harus mengetahui *setting network* dari *server* seperti IP *address*, *subnet*, dan *port*, sedangkan untuk *server* tidak perlu mengetahui *setting network* dari *client*. Pada

mode ini *serial device* hanya dapat berkomunikasi dengan satu *ethernet device* yaitu *server* yang terhubung dengan modul ini.

3. *Mixed Mode*

Mode ini modul awalnya akan *standby* beroperasi seperti *mode server* dan menunggu adanya koneksi dari *client*. Ketika ada *client* yang terhubung, maka modul ini akan menjadi *mode server* biasa. Tetapi apabila ada data serial yang masuk melalui port serial sebelum ada *client* yang terhubung, maka modul ini akan berubah menjadi *mode client* dan mencari *server* untuk menghubungkan koneksi.

2.7.3. Serial Configuration



Sumber : WIZ110SR User's Manual

Gambar 2.7 Layar Editor Konfigurasi Serial

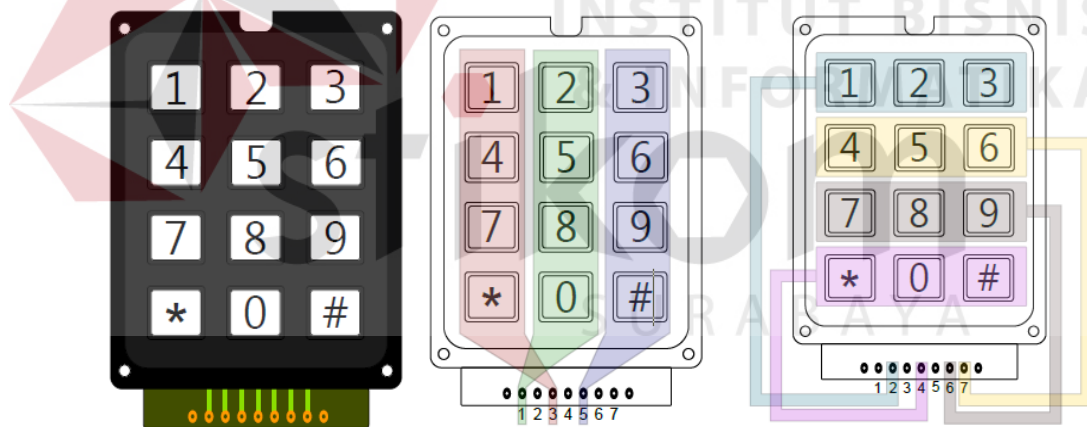
Pada Gambar 2.7 dapat diberikan penjelasan sebagai berikut :

- Baudrate* adalah kecepatan atau jumlah data yang dapat ditransfer dalam satuan detik.
- Databit* adalah panjang data yang dapat dikirim dalam satu kali transmisi.

- c. *Parity* adalah bit tambahan yang digunakan untuk mengecek data yang dikirim valid atau tidak.
- d. *Stop* bit adalah bit penanda untuk tranmisi data apabila sudah selesai.
- e. *Flow* adalah pengaturan untuk mengatur aliran data baik melalui *hardware* atau *software* (Wiznet, 2007).

2.7. Keypad

Keypad yang digunakan adalah *keypad* matrik 3x4 yang terdiri dari tombol '0' sampai '9' dan tanda '*' dan pagar. *Keypad* ini mempunyai delapan pin, yang terdiri dari 4 pin baris, 3 pin kolom, dan sebuah *common*. Jika kita menekan tombol '1' maka baris 1 dan kolom 1 akan terhubung dengan *common*. Gambar *keypad* bisa dilihat pada Gambar 2.8.



Sumber : COM-08653 datasheet

Gambar 2.8 Keypad

Sehingga bila tidak terhubung dengan *common* maka di kaki tersebut kita akan mendapatkan kondisi *high* (1) (robotics, 2011). Kondisi pin *keypad* dapat dilihat pada Tabel 2.5.

Tabel 2.5 Kondisi pin *keypad*

Tombol	K1	K2	K3	B1	B2	B3	B4
1	0	1	1	0	1	1	1
2	1	0	1	0	1	1	1
3	1	1	0	0	1	1	1
4	0	1	1	1	0	1	1
5	1	0	1	1	0	1	1
6	1	1	0	1	0	1	1
7	0	1	1	1	1	0	1
8	1	0	1	1	1	0	1
9	1	1	0	1	1	0	1
*	0	1	1	1	1	1	0
0	1	0	1	1	1	1	0
#	1	1	0	1	1	1	0

Keterangan :

K = kolom

B = baris.

2.8. *Moving Sign*

Moving sign adalah *display* lampu LED bergerak yaitu salah satu media elektrik yang sangat berguna untuk sarana iklan, informasi dan dekorasi kantor dan perkotaan. Sangat beda dengan spanduk yang sifatnya hanya nonpermanen (karena cepat rusak). *Moving sign* bisa diubah tampilannya sewaktu-waktu menggunakan komputer sehingga sangat praktis (Teguh, 2011).

Moving sign yang dapat diprogram untuk menampilkan karakter beserta gerakan animasinya. Pada *moving sign* ini terdapat jam dan kalender. Dilengkapi dengan *software* editor dan simulator, dimana terdapat editor huruf/ font sehingga user dapat membuat hurufnya sendiri (PT. Arion Indonesia, 2013).

Kemampuan dan Fungsi dari alat ini dapat beraneka sesuai dengan inovatif dan pemikiran orang-orang. Akan tetapi pada dasarnya alat ini memiliki beberapa fungsi yaitu menampilkan jam sebagai saran umum, menampilkan tulisan yang berguna untuk penyedia promosi dan iklan bagi empunya *moving sign*, Sebagai

penambah daya tarik di berbagai prasarana (Toko, Hotel, papan nama, counter, penulisan label harga, dll)

Gambar *moving sign display* dapat dilihat pada Gambar 2.9.



Gambar 2.9 *Moving Sign*

