

## **APLIKASI SERVER VIRTUAL IP UNTUK MIKROKONTROLER (IP Virtual Server Application for Microcontroller)**

**Ahmad Ashari**

IKE Department, Faculty of Mathematics and Natural Sciences  
Gadjah Mada University Yogyakarta

### **ABSTRAK**

Selama ini mikrokontroler yang terhubung ke satu komputer hanya dapat diakses melalui satu IP saja, padahal kebanyakan sistem operasi sekarang dapat memperjanjikan lebih dari satu IP untuk setiap komputer dalam bentuk virtual IP. Penelitian ini mengkaji pemanfaatan virtual IP dari IP aliasing pada sistem operasi Linux sebagai Server Virtual IP untuk mikrokontroler. Prinsip dasar Server Virtual IP adalah pembuatan Virtual Host pada masing-masing IP untuk memproses paket-paket data dan menerjemahkannya kedalam format paket data yang sesuai dengan mikrokontroler. Selain menerjemahkan paket data, virtual host juga menerjemahkan alamat IP ke dalam alamat mikrokontroler dengan format bilangan biner 4-bit. Hasil pengujian menunjukkan bahwa Server Virtual IP yang dibuat telah dapat melakukan transfer data dan penerjemahan IP ke mikrokontroler dengan benar.

**Kata-kata kunci:** *Virtual IP, Linux, Mikrokontroler*

### **ABSTRACT**

Until now the microcontroller that connected to a computer can only be accessed through a single IP, but most operating systems now can make more than one IP for each computer as virtual IP. This research will study the use of virtual IP from IP aliasing on a Linux operating system as an IP Virtual Server for the microcontrollers. The basic principle of IP Virtual Server is the creation of Virtual Host for each IP that processing data packets and translate them into the format of data packets which is match with the microcontroller. In addition to translating the data packets, the virtual host also translate IP addresses to the microcontroller address with 4-bit binary numbers format. The results show that the IP Virtual Server can perform the data transfer and translation of IP to the microcontroller correctly.

**Key words:** *Virtual IP, Linux, Microcontroller*

## 1. PENDAHULUAN

Pemanfaatan jaringan TCP/IP telah berkembang dengan begitu pesat, demikian juga pemanfaatannya dalam bidang elektronika dan instrumentasi. Fungsi kontrol dan monitor akan lebih mudah bila memanfaatkan jaringan TCP/IP, karena fungsi kontrol dan monitor ini dapat dihubungkan melalui jaringan TCP/IP yang dapat diakses secara luas.

Sistem operasi Linux memiliki kemampuan memiliki banyak alamat IP pada satu antarmuka (interface) jaringan yang disebut IP Aliasing (Pillay dan Yokley, 2001). Dengan memanfaatkan IP Aliasing pada Linux dan menjadikan mikrokontroler sebagai peralatan yang akan dikontrol dan dimonitor, maka mikrokontroler dapat memiliki Virtual IP, sehingga fungsi kontrol dan monitor dapat dilakukan secara terdistribusi.

Berdasar uraian di atas, pembuatan aplikasi server Virtual IP untuk mikrokontroler Berbasis TCP/IP sangat diperlukan. Aplikasi ini dimaksudkan untuk menerjemahkan alamat dan data dari protokol TCP/IP menjadi sebuah format data yang sesuai dengan mikrokontroler.

Jaringan TCP/IP adalah jaringan komputer yang menggunakan protokol standard yang populer dan digunakan secara luas baik untuk Internet maupun Intranet. Dengan menggunakan protokol TCP yang menerapkan komunikasi berbasis koneksi, validitas data yang ditransmisikan dapat terjamin karena bila ada paket data yang rusak maka paket data tersebut akan dikirim ulang. TCP/IP sangat cocok bila diaplikasikan pada suatu sistem kontrol maupun sistem monitoring yang mengutamakan validitas data. TCP/IP secara model tidak sepenuhnya mengacu pada model OSI tujuh lapis melainkan menggunakan model sendiri. Pada model TCP/IP setiap lapisan (layer) memiliki fungsi spesifik termasuk didalamnya fungsi

pengalamatan (Tanenbaum, 1996). Fungsi dari layer-layer TCP/IP sebagai berikut:

1. **Network Interface Layer.** Merupakan layer yang bersentuhan langsung dengan level hardware (Network Interface Card). Masih-masing NIC memiliki sebuah alamat fisik yang disebut dengan MAC (Media Access Control) address.
2. **Internetwork Layer.** Merupakan layer yang bertanggung jawab terhadap rotting dan juga enkapsulasi paket TCP/IP.
3. **Trasport Layer.** Merupakan layer yang bertanggung jawab dalam membuat komunikasi antar dua host dengan membuat sesi berbasis koneksi.
4. **Application Layer.** Lapisan ini bertanggung jawab menyediakan akses aplikasi terhadap jaringan TCP/IP.

IP aliasing adalah proses penambahan IP address pada sebuah Network Interface Card. Mengacu pada layer 1 dan 2 model protokol TCP/IP, IP aliasing memungkinkan sebuah MAC address memiliki banyak IP adress. Pada sistem operasi Linux IP aliasing diimplementasikan pada kernel mulai tahun 1995 (Pillay dan Yokley, 2001).

Mikrokontroler Atmel AVR telah banyak di gunakan untuk aplikasi-aplikasi yang menarik (Korbel dan Janes, 2004). Penggunaan Linux untuk mengendalikan mikrokontroler Atmel melalui antarmuka port serial ataupun paralel juga dapat dilakukan (Deegan, 2005). Integrasi beberapa jaringan sensor dalam satu virtual sensor network telah dilakukan oleh Lei dkk. (2006).

Pemrograman jaringan menggunakan bahasa Java memungkinkan membuat program aplikasi berbasis client-server (Hallot, 2004), dimana program ini dapat berjalan di semua sistem operasi tak terkecuali di Linux.

## 2. METODE PENELITIAN

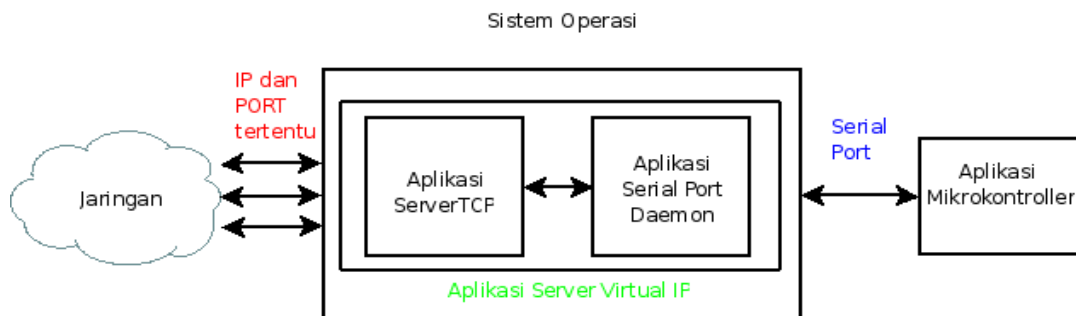
Sistem aplikasi server virtual IP untuk mikrokontroler, secara garis besar bekerja dengan prinsip menerjemahkan alamat IP menjadi sebuah alamat dengan format *unsigned char* yang di mengerti oleh mikrokontroler dan juga sebaliknya. Secara global arsitektur dari sistem ini terlihat pada Gambar 1.

Aplikasi Server Virtual IP (Gambar 1) merupakan aplikasi yang berfungsi untuk melakukan penerjemahan dari alamat IP ke alamat *unsigned char* serta melakukan proses antar muka antara PC dengan mikrokontroler. Aplikasi ini ditulis dengan menggunakan Java sebagai bahasa pemrogramannya. Aplikasi ini tersusun atas tiga kelas yaitu:

1. Main, kelas utama yang menjalankan aplikasi.

2. Server, kelas yang menangani koneksi TCP dengan client serta melakukan translasi alamat ke format *unsigned char*.
3. SerialPortDaemon, kelas yang mengani antar muka antara PC dengan mikrokontroler melalui serial port.

Aplikasi Mikrokontroler pada Gambar 1 berfungsi sebagai penerjemah header alamat untuk membuat jalur data ke mikrokontroler yang dituju serta mengirim data. Aplikasi ini terbagi menjadi dua bagian, yaitu perangkat keras dan perangkat lunak. Perangkat keras terdiri atas konfigurasi sistem minimum ATMEGA 16 dengan rangkaian antar muka serial RS 232. Perangkat lunak merupakan aplikasi yang menerjemahkan header alamat kemudian menjalankan prosedur kirim data melalui port tertentu.

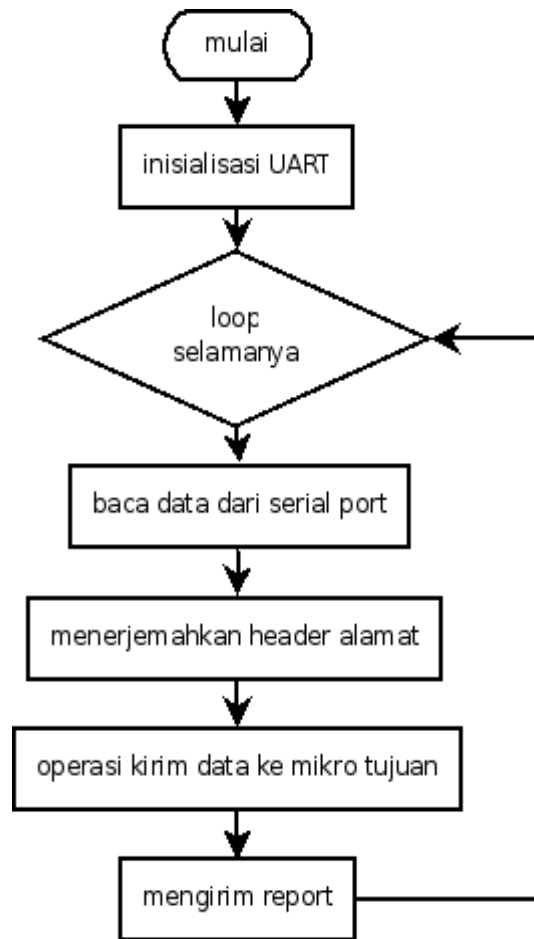


Gambar 1 Arsitektur sistem aplikasi server virtual IP global

Program ditulis dengan menggunakan bahasa C dan di compile dengan AVR-GCC. Program terdiri atas dua bagian yaitu main.c dan UART.c. Secara umum program UART.c berfungsi untuk melakukan inisialisasi komunikasi serial (UART) dalam mikrokontroler. Setelah dilakukan inisialisasi, program akan melakukan perulangan untuk membaca data yang berupa array dari *unsigned char*. Program ini akan mengirim report setelah

eksekusi pengiriman data selesai dilakukan dan akan melakukan perulangan menunggu data baru diterima. Flowchart dari program yang diletakkan dalam mikrokontroler ini dapat dilihat pada Gambar 2.

Skema dari rangkaian untuk aplikasi mikrokontroler dapat dilihat pada Gambar 3, dimana dalam rangkaian ini diterapkan sebuah sistem minimum dari IC ATMEGA 16 dengan antar muka serial RS-232.

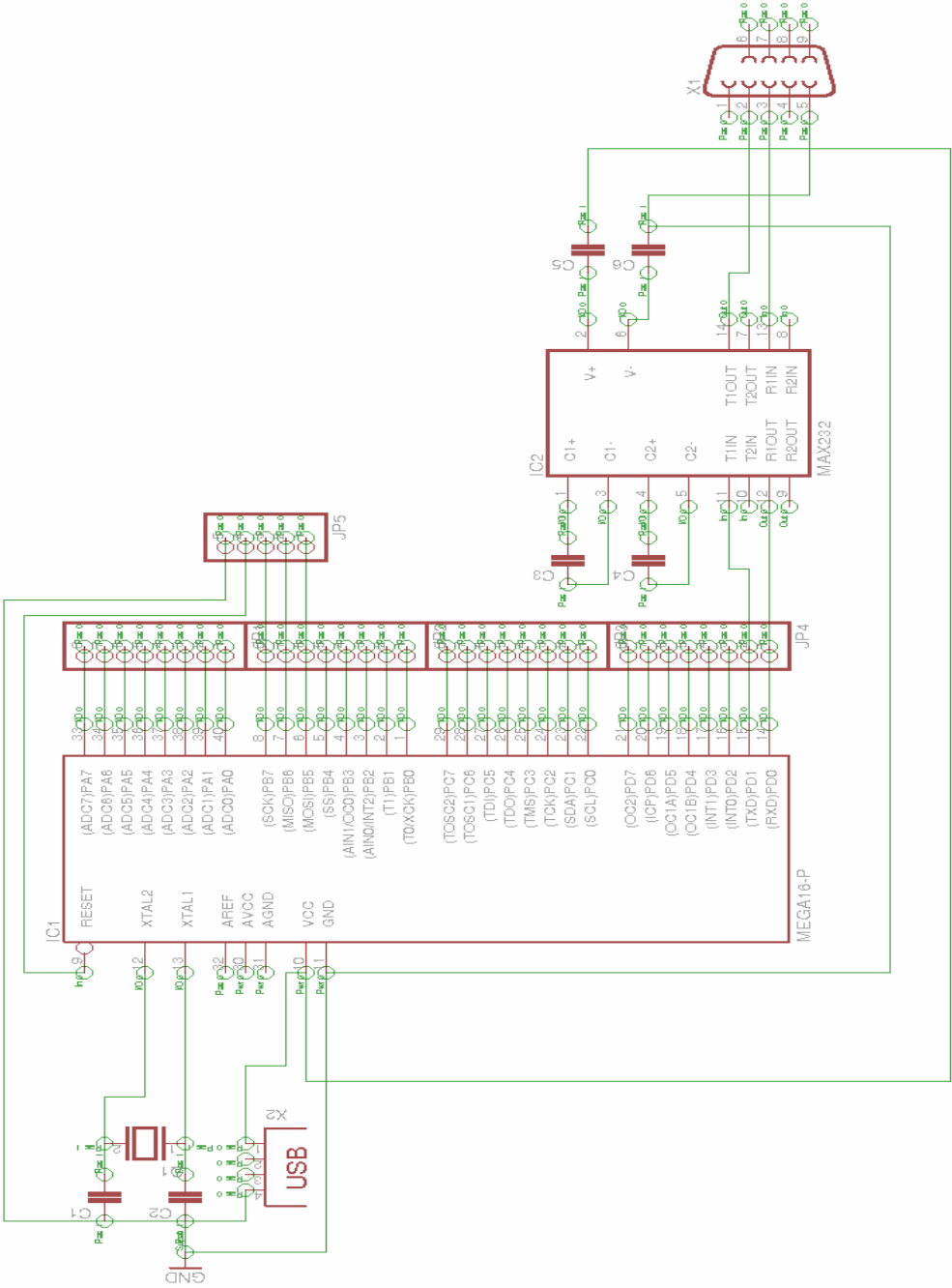


Gambar 2. Flowchart program yang di letakkan dalam mikrokontroler

Sistem minimum ini hanya mengandung:

1. Clock generator yang dibangkitkan oleh kristal eksternal sebesar 4 MHz.
2. Sumber Tegangan 5 Volt yang diperoleh dari port USB.
3. Port Data dan Port Alamat yang masing-masing menggunakan Port B dan Port A dari mikrokontroler ATMEGA 16.
4. Antar Muka Serial RS-232. Untuk transmisi serial dengan jarak yang cukup jauh, diperlukan teknik tambahan untuk memastikan integritas data. Hal tersebut disebabkan oleh

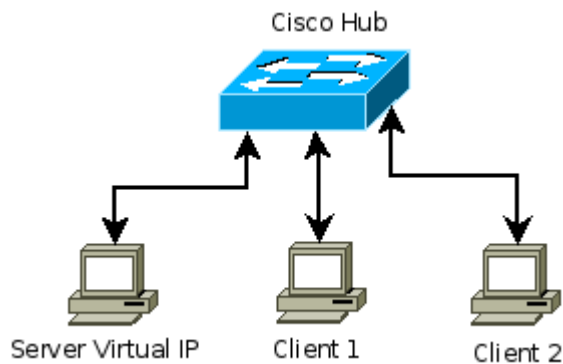
adanya penurunan tegangan logika dan juga adanya derau dalam proses transmisinya. Sehingga terjadi kerancuan terhadap logika tinggi dan logika rendah. Dengan standar RS-232 (ADL-232), sebuah logika 1 diwakili dengan tegangan -12 V DC, sedangkan logika 0 diwakili oleh +12 V DC. IC yang digunakan adalah MAX232 yang melakukan konversi tegangan 0 – 5 Volt ke standar komunikasi yang kompatibel dengan RS232 pada transmitter dan mengkonversi balik ke tegangan 0 – 5 Volt di receiver.



Gambar 3. Skema rangkaian untuk aplikasi Mikrokontroler

### 3. IMPLEMENTASI SISTEM

Aplikasi Server Virtual IP dan Aplikasi Mikrokontroler diimplementasikan dalam sebuah jaringan LAN dengan topologi seperti Gambar 4.



Gambar 4. Topologi LAN untuk aplikasi server virtual IP

Aplikasi mikrokontroler, yang merupakan rangkaian sistem minimum ATMEGA 16, dihubungkan ke Server Virtual IP melalui port serial RS-232. Perangkat lunak yang digunakan untuk implementasi Server Virtual IP adalah sebagai berikut:

1. Sistem Operasi GNU/Linux Debian Lenny kernel 2.6.26-486
2. JRE 6 update 7 dengan library RXTX 2.1-7
3. Netbeans 6.1
4. Aplikasi Java “Server Virtual IP”

Sedangkan perangkat lunak yang digunakan untuk implementasi Client 1 adalah sebagai berikut:

1. Sistem Operasi Windows XP Service Pack 2
2. JRE 6 update 7
3. Netbeans 6.1
4. Aplikasi Java “Client TCP”.

Dan perangkat lunak yang digunakan untuk implementasi Client 2 adalah sebagai berikut:

1. Sistem Operasi GNU/Linux Debian

Lenny kernel 2.6.26-486

2. JRE 6 update 7
3. Netbeans 6.1
4. Aplikasi Java “Client TCP”

Pemilihan 2 client dengan 1 sistem operasi yang berbeda dimaksudkan untuk pengujian portabilitas aplikasi java yang di buat. Agar semua komputer dapat berkomunikasi, kartu jaringan harus dikonfigurasi terlebih dahulu. Implementasi Server Virtual IP dilakukan pada sebuah jaringan lokal (Local Area Network / LAN). Pada sebuah jaringan lokal, komputer server dan client harus berada dalam jangkauan IP yang sama. Jangkauan IP pada sebuah jaringan ditentukan oleh nilai netmask. Konfigurasi kartu jaringan dari komputer – komputer yang digunakan adalah sebagai berikut:

a) Server

Untuk menjadi sebuah server virtual IP, kartu jaringan dari komputer server harus memiliki lebih dari satu IP. Sistem operasi Linux memungkinkan sebuah kartu jaringan dengan sebuah alamat MAC memiliki lebih dari satu IP. Metode multi IP pada sebuah kartu jaringan disebut IP aliasing. Pada sistem operasi GNU/Linux Debian IP aliasing dilakukan dengan mengkonfigurasi file `/etc/network/interface` dengan konfigurasi sebagai berikut:

```
allow-hotplug eth0
iface eth0 inet static
address 192.168.0.226
netmask 255.255.255.0
network 192.168.0.0
broadcast 192.168.0.255
gateway 192.168.0.1
dns-nameservers 192.168.0.2
iface eth0:1 inet static
address 192.168.0.227
netmask 255.255.255.0
auto eth0:1
```

Dari konfigurasi di atas maka konfigurasi IP utama dan kedua sebagai berikut:

1. Alamat IP utama : 192.168.0.226 dengan netmask : 255.255.255.0
2. Alamat IP kedua : 192.168.0.227 dengan netmask : 255.255.255.0

Dengan dua IP di satu kartu jaringan maka Server Virtual IP mampu mengalami dua buah mikrokontroler. Untuk menjalankan konfigurasi tersebut, service jaringan perlu direstart terlebih dahulu dengan perintah:

```
#/etc/rc.d/networking restart
```

Alamat IP utama (192.168.0.226) merupakan alamat IP dari Server1 yang akan dialamatkan ke mikro 1, sedang alamat IP kedua (192.168.0.227) digunakan oleh server 2 yang akan dialamatkan ke mikro 2.

- b) Client1 dengan sistem operasi Windows  
Konfigurasi jaringan di sistem operasi Windows dilakukan dari menu Control Panel – Networks Connection – Local Area Connection – Properties – Internet Protocol (TCP/IP), dengan konfigurasi alamat IP: 192.168.0.171 dan netmask: 255.255.255.0
- c) Client2 dengan sistem operasi Linux  
Konfigurasi jaringan dilakukan pada file `/etc/network/interface` dengan konfigurasi alamat IP: 192.168.0.224 dan netmask: 255.255.255.0

### Pengujian Sistem

Aplikasi Server Virtual IP berjalan di atas Java Virtual Machine. Dalam percobaan ini JVM berjalan di atas sistem operasi GNU/Linux Debian Lenny. Aplikasi Server Virtual IP tidak memerlukan aplikasi lain selain JVM, walaupun dapat juga dijalankan didalam aplikasi Netbeans 6.1 yang tentunya terdapat JVM didalam sistem operasinya. Aplikasi Server Virtual IP adalah aplikasi daemon, sehingga setelah dieksekusi aplikasi akan berjalan di background. Tetapi untuk mempermudah pengamatan terhadap perilakunya, aplikasi ini akan menampilkan status notifikasi dari serial

port, penutupan port, waktu kirim reply ke serial port dan waktu terima data dari client. Aplikasi Server Virtual IP dijalankan dengan perintah:

```
#java Main
```

yang setelah dieksekusi memberikan tampilan seperti Gambar 5.

Pengujian aplikasi dilakukan untuk menyelidiki kebenaran dan kemampuan aplikasi Server Virtual IP dan aplikasi Mikrokontroler yang meliputi empat hal, yaitu:

1. Kebenaran aplikasi Server Virtual IP menerima data, memberi header alamat, mengirimkannya melalui serial port ke mikrokontroler serta mengirim data reply respons dari serial port.
2. Kebenaran aplikasi mikrokontroler menerima data dari serial port, melakukan penerjemahan header alamat, melakukan eksekusi perintah kirim data dan memberi respon ke PC setelah perintah kirim data selesai dilakukan.
3. Menguji waktu eksekusi dari Server Virtual IP dan Aplikasi Mikrokontroler. Kecepatan yang diuji adalah kecepatan eksekusi perintah penerjemahan alamat serta pengiriman data ke aplikasi mikrokontroler.
4. Menguji waktu respon dari client – Server Virtual IP – Aplikasi Mikrokontroler – Server Virtual IP – client. Kecepatan yang diuji meliputi waktu transfer data, waktu eksekusi dan juga waktu transfer reply dari aplikasi mikrokontroler.

Metode pengujian yang dilakukan adalah metode kotak hitam (*black box*), yaitu suatu cara pengujian dengan hanya menguji masukan dan keluaran dari suatu sistem tanpa menguji kebenaran interaksi di dalamnya.

Sesuai dengan tujuan pengujian, tahap pengujian yang dilakukan terhadap aplikasi

Server Virtual IP dan Aplikasi Mikrokontroler terdiri atas tiga macam, yaitu:

1. Pengujian kebenaran transfer data.

Pengujian ini bertujuan untuk membuktikan kebenaran transfer data dari client ke mikrokontroler. Selain itu dari percobaan ini juga dapat dilihat kebenaran penerjemahan alamat IP ke alamat mikrokontroler.

Pengujian dilakukan dengan mengirim data berupa sebuah *unsigned char* (byte) dari aplikasi Client TCP di komputer client ke aplikasi Server Virtual IP di komputer server. Oleh aplikasi Server Virtual IP, data dikirim ke aplikasi Mikrokontroler. Sehingga pengujian dilakukan dengan mengamati port alamat dan port data dari Aplikasi Mikrokontroler dengan alamat IP yang dituju oleh program client dan data yang dikirim melalui program client.

2. Pengujian kecepatan eksekusi.

Pengujian ini bertujuan melihat waktu eksekusi perintah penerjemahan alamat, pengiriman data dan pengiriman reply. Pengujian dilakukan dengan menghitung waktu awal data dari client diterima hingga waktu saat reply dikirim ke client.

Waktu Eksekusi = Jam kirim reply –  
Jam terima data

- a) Jam kirim data adalah waktu pada saat data diterima dari client dihitung mulai 1 Januari 1920 00:00:00 GMT dalam orde milidetik.
- b) Jam kirim reply adalah waktu pada saat aplikasi Server Virtual IP mengirim reply ke client dihitung mulai 1 Januari 1920 00:00:00 GMT.
- c) Waktu eksekusi adalah selisih antara Jam kirim reply dengan jam terima data. Waktu eksekusi merepresentasikan lamanya

waktu untuk memproses pengiriman data ke Aplikasi Mikrokontroler, penerjemahan alamat oleh Aplikasi Mikrokontroler, Pengiriman data melalui port data oleh Aplikasi Mikrokontroler dan proses penerimaan reply oleh Server Virtual IP dari Aplikasi Mikrokontroler. Waktu eksekusi dalam orde milidetik.

Jam kirim reply dan jam terima data diperoleh dengan memanggil method *getTime* dari kelas *java.util.Date*.

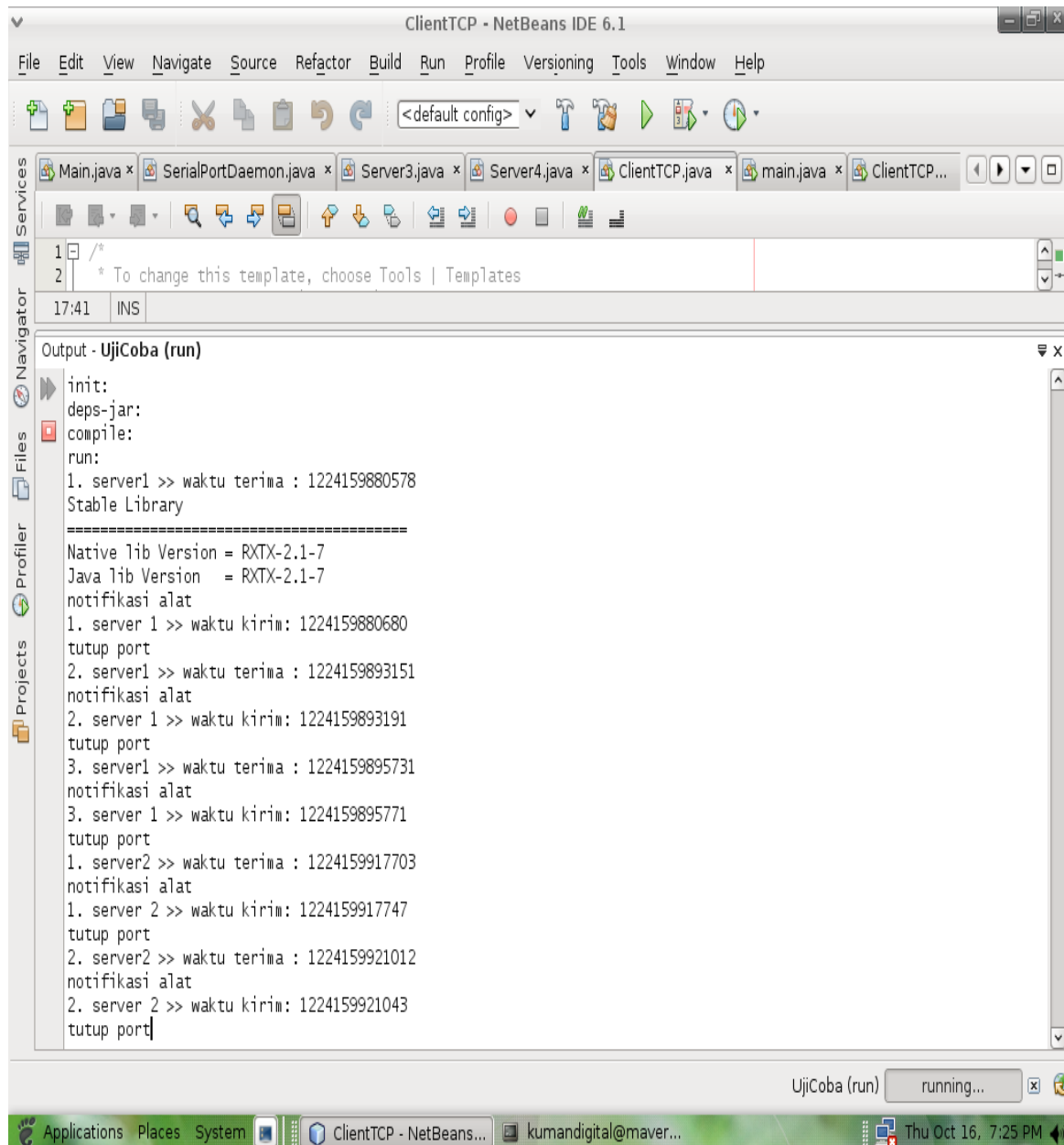
3. Pengujian waktu respon

Pengujian ini bertujuan untuk melihat waktu respon dari proses pengiriman data ke Aplikasi Mikrokontroler hingga reply dari Aplikasi Mikrokontroler diterima oleh client. Waktu respon ini menggambarkan efisiensi dari aplikasi yang diuji. Semakin cepat waktu respon maka akan semakin efisien.

Waktu Respon = Jam terima reply – jam kirim data

- a) Jam kirim data adalah waktu pada saat data dikirim dari client ke Server Virtual IP dihitung mulai 1 Januari 1920 00:00:00 GMT dalam orde milidetik.
- b) Jam terima reply adalah waktu pada saat client menerima reply dari Server Virtual IP dihitung mulai 1 Januari 1920 00:00:00 GMT dalam orde milidetik.
- c) Waktu respon adalah selisih antara jam terima reply dengan jam kirim data. Waktu respon merepresentasikan proses pengiriman dan penerimaan data ke dan dari komputer server melalui jaringan LAN serta waktu eksekusi di Server Virtual IP.





Gambar 5. Tampilan setelah eksekusi aplikasi

### Analisis Hasil Pengujian

Analisis hasil pengujian yang telah dilakukan memberikan tiga hasil, yaitu

1. Analisis dari hasil pengujian terhadap kebenaran transfer data dan penerjemahan IP. Hasil pengujian transfer data dapat

dilihat dari port data Aplikasi Mikrokontroler. Pengujian dilakukan dengan membandingkan bentuk hexa dari karakter yang dimasukkan dari client dengan bentuk hexa dari keluaran port data.

Hasil pengujian ini dapat dilihat pada Tabel 1. Tabel uji kebenaran di atas

memperlihatkan aplikasi Server Virtual IP dan Aplikasi Mikrokontroler dapat berjalan dengan baik dan benar dalam melakukan proses transfer data.

Hasil pengujian penerjemahan alamat IP dilihat dari port alamat aplikasi

mikrokontroler. Pengujian dilakukan dengan membandingkan tabel alamat IP dan port Server Virtual IP dengan bentuk biner dari keluar port alamat. Hasil pengujian ini dapat dilihat pada Tabel 2.

Tabel 1. Hasil pengujian kebenaran transfer data

Char	Port Data (biner)	Port Data (hexa)	Hasil
1	0011 0001	0x31	benar
2	0011 0010	0x32	benar
3	0011 0011	0x33	benar
4	0011 0100	0x34	benar
5	0011 0101	0x35	benar
a	0110 0001	0x61	benar
b	0110 0010	0x62	benar
c	0110 0011	0x63	benar
d	0110 0100	0x64	benar
e	0110 0101	0x65	benar

Tabel 2. Hasil pengujian penerjemahan alamat IP

IP:port	Alamat Mikrokontroler	Keluaran port alamat
192.168.0.226:1234	Server '1' = 0000 0001	0000 0001
192.168.0.227:1235	Server '2' = 0000 0010	0000 0010

2. Dari hasil uji pada Tabel 2 memperlihatkan bahwa aplikasi translasi alamat IP ke alamat mikrokontroler berjalan dengan baik dan benar. Hal ini terlihat dari Alamat Mikrokontroler = keluar port alamat. Analisis hasil pengujian terhadap waktu eksekusi Hasil pengukuran waktu eksekusi seperti pada Tabel 3 adalah hasil olah data dari waktu terima data dan waktu kirim reply yang merupakan output dari Server Virtual

IP. Output tersebut merupakan hasil dari penerimaan data dari Client 1 dan Client 2 dengan waktu kirim data yang acak.

Dari data waktu eksekusi di atas diperoleh rata – rata waktu eksekusi adalah 39,75 milidetik. Sehingga kecepatan rata-rata eksekusi transfer data yang dapat dilakukan oleh server adalah:

$$\text{Kecepatan server} = 1 / 39,75 \text{ milidetik} \\ = 25 \text{ kali transfer data / detik}$$

3. Analisis hasil pengujian terhadap waktu respon

Hasil uji waktu respon adalah hasil olah data dari waktu kirim dari client dan waktu

terima reply oleh client dari Server Virtual IP. Setiap program client akan menampilkan hasil waktu respons saat dijalankan seperti tampak pada Gambar 6.

Tabel 3. Hasil pengukuran waktu eksekusi program

No	Server	index	Waktu Kirim Replay	Waktu Terima Data	Waktu Eksekusi
1	2	1	1224163564372	1224163564330	42
2	2	2	1224163566477	1224163566437	40
3	2	3	1224163568653	1224163568614	39
4	2	4	1224163570897	1224163570859	38
5	2	5	1224163575773	1224163575732	41
6	1	1	1224163585837	1224163585798	39
7	1	2	1224163587949	1224163587909	40
8	1	3	1224163590057	1224163590017	40
9	1	4	1224163592165	1224163592125	40
10	2	6	1224163592549	1224163592509	40
11	1	5	1224163594273	1224163594233	40
12	2	7	1224163594797	1224163594756	41
13	1	6	1224163596381	1224163596341	40
14	2	8	1224163596905	1224163596865	40
15	1	7	1224163598489	1224163598449	40
16	2	9	1224163599157	1224163599116	41
17	1	8	1224163600597	1224163600557	40
18	2	10	1224163601721	1224163601688	33
19	1	9	1224163602705	1224163602665	40
20	2	11	1224163603833	1224163603793	40
21	1	10	1224163604813	1224163604773	40
22	2	12	1224163605941	1224163605902	39
23	1	11	1224163606921	1224163606881	40
24	2	13	1224163608117	1224163608076	41
Rata-rata esksekusi					39,75

Olah data waktu respon diamati dari saat masing-masing client dijalankan, dengan hasil pengamatan untuk Client 1 pada Tabel 4 dan Client 2 pada Tabel 5. Waktu respon dipengaruhi oleh dua faktor. Faktor pertama adalah waktu eksekusi. Faktor yang kedua adalah kecepatan transfer data dalam jaringan LAN. Waktu transfer data, adalah waktu tempuh roundtrip dari client ke server dan ke client

lagi. Dilihat dari perbandingan hasil uji waktu eksekusi dan waktu respons, faktor yang paling besar mempengaruhi waktu respons adalah waktu transfer data dalam jaringan. Hal itu terlihat dari perhitungan berikut:

$$\begin{aligned} \text{rata-rata waktu transfer 1} &= \text{rata-rata respon Client1} - \text{rata-rata eksekusi} \\ &= 1205,82 - 39,75 = 1166,07 \text{ milidetik} \end{aligned}$$

```

C:\WINDOWS\system32\cmd.exe - java ClientTCP
Enter char:
1
From Server: mikro 2 data terkirim waktu respon: 32
Enter char:
2
From Server: mikro 2 data terkirim waktu respon: 1344
Enter char:
3
From Server: mikro 2 data terkirim waktu respon: 1218
Enter char:
4
From Server: mikro 2 data terkirim waktu respon: 1063
Enter char:
5
From Server: mikro 2 data terkirim waktu respon: 1297
Enter char:
6
From Server: mikro 2 data terkirim waktu respon: 1296
Enter char:
7
From Server: mikro 2 data terkirim waktu respon: 1266
Enter char:
8
From Server: mikro 2 data terkirim waktu respon: 1344
Enter char:
9
From Server: mikro 2 data terkirim waktu respon: 1094
Enter char:
0
From Server: mikro 2 data terkirim waktu respon: 1219
Enter char:
a
From Server: mikro 2 data terkirim waktu respon: 609
Enter char:

```

Gambar 6. Tampilan waktu respon saat program client dijalankan

Tabel 4. Waktu respon pada Client1

No	Respon (mdet)
1	1032
2	1344
3	1218
4	1063
5	1297
6	1296
7	1266
8	1344
9	1096
10	1219
11	1089
Rata-rata	1205,82

Tabel 5. Waktu respon pada Client2

No	Respon (mdet)
1	43
2	74
3	39
4	38
5	41
6	41
7	41
8	233
9	41
10	33
11	302
12	375
13	352
Rata-rata	127,15

rata-rata waktu transfer 2 = rata-rata respon Client2 – rata-rata eksekusi = 127,15– 39,75 = 87,75 milidetik

Dari hasil di atas dapat dilihat bahwa waktu transfer data lebih besar dari waktu eksekusi. Waktu transfer data sangat dipengaruhi oleh lalu lintas data didalam jaringan yang digunakan.

#### 4. KESIMPULAN

Dari hasil penelitian ini dapat diambil kesimpulan berikut:

1. Telah berhasil dibuat suatu sistem aplikasi server virtual IP untuk mikrokontroler yang dapat menerjemahkan alamat IP di jaringan langsung ke mikrokontroler dengan pemetaan satu ke satu.
2. Hasil pengujian terhadap sistem ini menunjukkan bahwa transfer data dan penerjemahan IP telah berjalan dengan benar.
3. hasil pengukuran terhadap waktu diperoleh bahwa waktu eksekusi rata-rata dari program adalah 39,75 mdet dan waktu respon terhadap client rata-rata 1205,82 mdet untuk Client 1 dan 127,15 mdet untuk Client 2.

Akan dilakukan penelitian lanjutan untuk mengetahui apakah sistem ini dapat diterapkan pada penggunaan mikrokontroler untuk pemantauan dan pengendalian, dengan cara rangkaian mikrokontroler nya dihubungkan pada suatu pemantau berupa sensor dan alat pengendali berupa relay.

#### Ucapan Terima Kasih

Ucapan terima kasih disampaikan kepada Program D3 Elins yang telah mendanai penelitian ini dan kepada Dipto Djatmiko yang telah membantu dalam pengujian.

#### DAFTAR PUSTAKA

- Deegan, P., 2005, *Developing for the Atmel AVR Microcontroller on Linux*, <http://www.linuxjournal.com/article/7289>
- Hallot, E.R., 2004, *Java Network programming 3rd edition*, Prentice Hall
- Korbel, S., Janes, V., 2004, Interesting Applications of Atmel AVR Microcontrollers, IEEE – Euromicro Symposium on Digital System Design (DSD'04), August 2004, France.
- Lei, S., Xu, H., Xiaoling, W., Lin, Z., Cho, J., Lee, S., 2006, VIP Bridge: Integrating Several Sensor Networks into One Virtual Sensor Network, IEEE – International Conference on Internet Surveillance and Protection (ICISP'06), August 26-29, France.
- Pillay, H., Yokley, J., 2001, *Setting up IP Aliasing on A Linux Machine Mini-HOWTO*, <http://tldp.org/HOWTO/IP-Alias/index.html>
- Tanenbaum, A.S., 1996, *Computer Network 3rd edition*, Prentice Hall