

Sistem Pengendalian dan Pengukuran Melalui Web Berbasis Sistem Operasi Linux

I. Firmansyah, Z. Akbar, B. Hermanto dan L.T. Handoko
iman002@lipi.go.id

Pusat Penelitian Fisika, Lembaga Ilmu Pengetahuan Indonesia
Kompleks Puspiptek Serpong, Tangerang 15310, Indonesia
Tlp. 021-7560556

Abstract- The paper describes the design and the application of web-based automation and measurement. The system consists of several devices such as AVR/MCS51 microcontrollers, sensors, actuators and personal computer (PC) as a web server which is installed by Linux operating system. All the components are connected through communication link so that they can communicate each other while sending or receiving the data. The clients or users can access the devices over web browser by sending request to the web server, thus the web server receives and processes the request by activating the CGI program which has responsibilities for sending and receiving the command to the microcontroller through parallel port. The microcontroller will finally generate an action based on the command received such as reading the temperature from the sensors or controlling the DC motor. By integrating the system, then we can build and develop the system which can be controlled and measured over web or Internet such as LIPI Public Cluster and Web-based Modular Wireless Robot.

Keywords: Linux, Measurement, Automation, Microcontroller, Web

I. PENDAHULUAN

Semakin pesatnya perkembangan teknologi informasi tidak hanya dirasakan dalam dunia hiburan multimedia tetapi juga dalam bidang pengendalian dan pengukuran khususnya bagi penerapan di kawasan industri maupun laboratorium. Berbagai peralatan pengendali dan pengukuran yang biasanya hanya dapat bekerja secara individu dan bahkan tidak dapat dimonitor secara jarak jauh, dengan menggunakan protokol TCP/IP diharapkan selain dapat saling berkomunikasi satu sama lain juga dapat dimonitor dan dikendalikan dari jarak jauh melalui web^[2]. Keuntungan lainnya yaitu kita dapat mengkases berbagai peralatan tadi dari manapun kita berada selama kita masih terkoneksi ke jaringan internet.

Salah satu solusi untuk melaksanakan proses pengendalian dan pengukuran berbasis web yaitu dengan menggunakan sebuah PC yang berfungsi sebagai *master* maupun *web-server*, dirangkai dengan beberapa mikrokontroler sebagai *slave* yang berfungsi untuk melakukan pengukuran dan pengendalian. *Web-server* tersebut berkomunikasi melalui *I/O port* dengan peralatan lain dalam hal ini mikrokontroler. Untuk kasus berikut

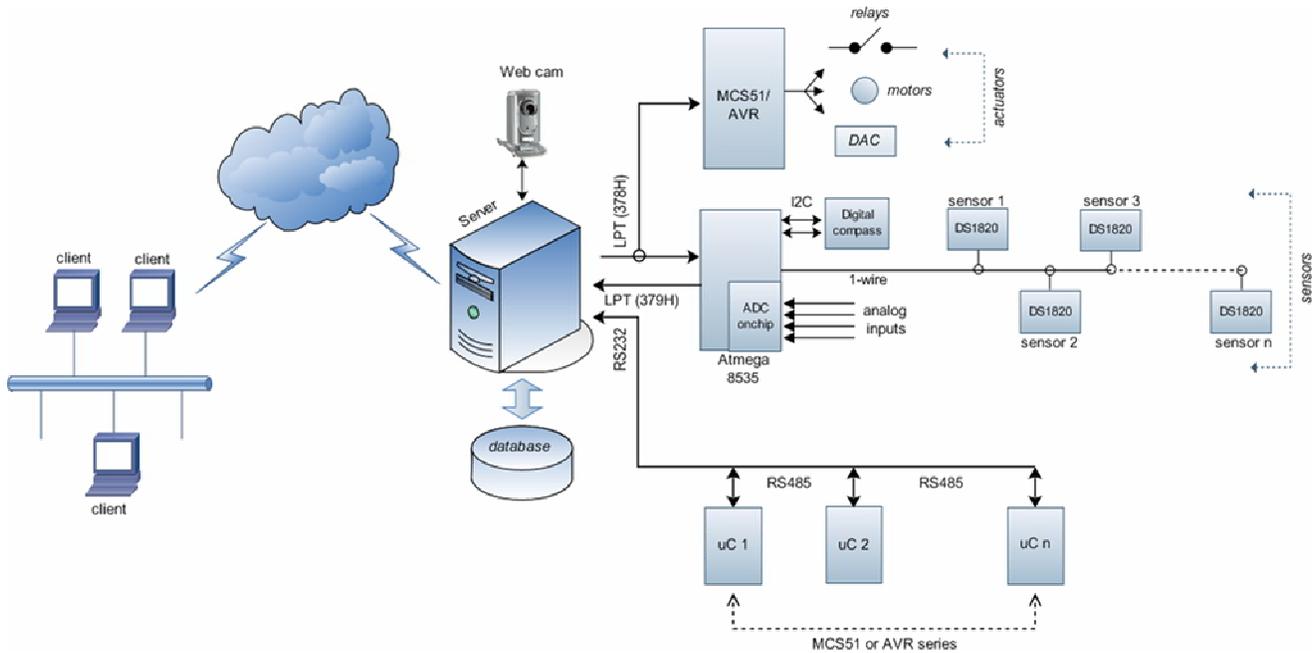
digunakan port parallel sebagai *channel* komunikasi meskipun pada dasarnya kita dapat menggunakan port lain seperti port serial, USB atau bahkan GPIB.

Untuk mengurangi biaya operasional khususnya dari segi perangkat lunak, maka dipergunakan sistem operasi Linux. Suatu distro Linux sudah terdiri dari berbagai aplikasi yang sangat lengkap mulai dari program aplikasi untuk perkantoran, multimedia, komputasi atau bahkan untuk pengaksesan perangkat keras. Pada perancangan berikut digunakan distro Ubuntu sebagai sistem operasi beserta paket aplikasi seperti Apache *web-server*, PHP, *CGI script*, GCC, Camserv, Java dan aplikasi lainnya. Dari segi perangkat keras digunakan mikrokontroler tipe keluarga MCS51 dan keluarga AVR ATmega8535 yang telah terintegrasi dengan internal ADC 10-bit, sedangkan sensor dan aktuator yang digunakan dalam perancangan sistem ini yaitu sensor suhu DS1820, LM35, relay AC dan motor DC meskipun pada dasarnya kita dapat menggunakan berbagai sensor dan aktuator lainnya sesuai dengan keperluan.

II. PERANCANGAN SISTEM

Sistem pengukuran dan pengendalian berbasis web terdiri dari sebuah PC, beberapa mikrokontroler yang berfungsi sebagai interface antara PC dengan peralatan luar baik itu sensor maupun aktuator^[1] yang diperlihatkan pada gambar.1. Dalam sistem berikut sebuah PC digunakan sebagai *web-server* yang akan berkomunikasi dengan beberapa peralatan yang terhubung ke mikrokontroler. *Web-server* tersebut akan menerima *request* dari *client* berupa perintah seperti menggerakkan aktuator atau bahkan perintah untuk membaca sensor, yang kemudian hasilnya akan dikirim oleh *web-server* ke *client* melalui *web-browser* sehingga akan tercipta suatu sistem pengukuran dan kendali berbasis web.

Untuk data dalam jumlah yang tidak terbatas, kita dapat menyimpan berbagai data ke dalam suatu data base, karena pada rancangan ini kita menggunakan PC. Dikarenakan dalam perancangan ini terdapat beberapa mikrokontroler, sensor dan aktuator sehingga diperlukan protokol yang digunakan dalam proses transmisi data. Berikut beberapa protokol yang digunakan dalam proses perancangan sistem :



Gambar. 1. Perancangan sistem pengendalian dan pengukuran berbasis web

1-wire

Protokol komunikasi 1-wire dikembangkan oleh Dallas Semiconductors^[4] untuk mengurangi penggunaan pengkabelan yang berlebih pada saat membangun perangkat sensor yang saling terhubung atau terdistribusi, sehingga kita dapat menghubungkan berbagai sensor hanya dengan satu kabel meskipun pada kenyataannya diperlukan satu kabel tambahan sebagai *ground*. Supaya protokol 1-wire ini berjalan dengan baik, maka digunakan suatu resistor pull up 4K7. Pada gambar.1 diperlihatkan pemakaian sensor suhu DS1820 yang bekerja dengan menggunakan protokol 1-wire. DS1820 mempunyai 64-bit ROM yang terdiri dari family code (8-bit), *serial number* (48-bit) dan CRC (8-bit). Adanya *serial number* menjadikan setiap sensor DS1820 unik dan dapat diidentifikasi.

I2C (Inter Integrated Circuit)

Protokol I2C atau *Inter Integrated Circuits*^[3] dikembangkan untuk proses pertukaran data diantara komponen atau sensor yang saling berbeda seperti EEPROM, RAM, ADC, DAC, RTC atau bahkan modul kompas digital (CMPS03). Bus I2C terdiri dari dari dua kabel yaitu *Serial Data Line* (SDA) dan *Serial Clock Line* (SCL) yang masing-masing terhubung ke Vcc dengan resistor *pull-up* sehingga menghasilkan suatu komponen atau sensor yang saling terhubung. Karena suatu bus I2C ini dapat menghubungkan satu *master* dengan beberapa *slave* yang berlainan jenis, maka *master* bertanggung jawab dalam menggunakan protokol I2C sehingga dapat berkomunikasi dengan berbagai *slave* yang berbeda.

RS485

Untuk mengatasi kekurangan dari saluran RS232 dari segi jarak tranmisi dalam proses pertukaran data, dapat digunakan saluran RS485. Dengan menggunakan saluran RS485, selain

jarak transmisi data dapat lebih jauh juga dapat lebih tahan dari gangguan *noise* dikarenakan saluran RS485 menggunakan saluran transmisi ganda atau *differential*^[3]. Untuk memungkinkan dalam satu jaringan RS485 terdapat beberapa *node* yang dapat saling berhubungan, diperlukan suatu IC *multipoint* RS485 *transceiver* seperti SN75176 buatan Texas Instrument. Karena saluran RS485 menggunakan pengkabelan yang dipakai secara bersama-sama, maka sebagai akibatnya proses komunikasi data bersifat *half duplex* yang artinya pada suatu waktu hanya satu *node* yang diperkenankan baik untuk mengirim atau menerima data.

III. PROTOKOL KOMUNIKASI

Proses pengiriman data dari *server* ke mikrokontroler melalui port I/O (port parallel, serial atau USB) dapat menggunakan proses yang sederhana dalam bentuk pengiriman data perintah terlebih dahulu, diikuti dengan mengeksekusi waktu tunda atau *delay* yang dimaksudkan untuk memberikan kesempatan kepada mikrokontroler untuk mengeksekusi program sesuai dengan perintah, dan diakhiri dengan proses pembacaan data kembali.

Proses tersebut berlaku untuk kasus sederhana dimana peralatan atau sensor yang akan dikontrol maupun dimonitor mempunyai jumlah yang terbatas. Biasanya perintah pembacaan dan pengendalian dapat dibedakan dengan memberikan perintah mulai dari 0 desimal sampai dengan 255 desimal, sehingga secara total dapat diperoleh 256 jumlah perintah yang berbeda. Untuk kasus yang lebih kompleks, digunakan protokol seperti terlihat pada gambar 2.



Gambar. 2. Format pengiriman data serial pada sistem multi slave

Sistem yang lebih kompleks khususnya yang terdiri dari beberapa sensor atau mikrokontroler, akan lebih baik jika menggunakan saluran komunikasi serial RS232 yang dihubungkan ke saluran RS485 untuk meningkatkan jarak tempuh transmisi data. Penggunaan saluran komunikasi RS232 akan lebih memudahkan dikarenakan hampir semua mikrokontroler telah terintegrasi dengan UART di dalam chip IC-nya.

Gambar 2 memperlihatkan format data serial yang dapat digunakan untuk berkomunikasi antara *master* tunggal dengan *multi slave*^[6], terlihat bahwa data terdiri dari beberapa *byte* yang tiap *byte*-nya mempunyai fungsi masing-masing sesuai keperluan. *Byte* pertama berupa “start of transmission” (STX), diikuti dengan “device number” (DN) yang menandakan nomor tiap peralatan atau sensor yang akan diakses, sebagai contoh DN=1 untuk mikrokontroler ke-1, DN=2 untuk mikrokontroler ke-2. *Byte* selanjutnya yaitu “data type” (DT) yang menandakan tipe data seperti “1” untuk membaca sensor ke-1 dan sebagainya. Sedangkan dua *byte* terakhir dapat digunakan untuk pengecekan kesalahan *byte* atau dapat digunakan untuk keperluan lainnya dan *byte* penanda bahwa format perintah data telah dikirim (ETX=end of transmission).

IV. KONFIGURASI PERANGKAT LUNAK

Supaya semua komponen peralatan baik itu mekanik maupun sensor dapat dihubungkan satu sama lain atau bahkan dapat dikendalikan melalui jaringan khususnya melalui *web browser*, salah satu hal yang paling utama yaitu diperlukannya sebuah *server* utama yang dapat menerima *request* dari *client* dan mengeksekusi dan mengirim data hasil olahan sesuai *request*. Kelebihan penggunaan *web browser* pada aplikasi ini yaitu, dari sisi *client* tidak diperlukan program khusus yang diperlukan ketika mengakses berbagai peralatan dan sensor, disamping itu akan memberikan keleluasaan atau kebebasan dalam penggunaan sistem operasi pada *client*.

Pada aplikasi berikut, digunakan sistem operasi Linux beserta semua program aplikasinya seperti sistem operasi Linux Ubuntu, Apache web-server, GCC, Java, PHP dan berbagai program aplikasi pendukung lainnya. Di samping aplikasi tersebut, juga digunakan CGI (common gateway interface) yaitu suatu aplikasi yang memungkinkan suatu *client* mengirim *request* untuk mengeksekusi program aplikasi yang terdapat dalam *server*^[6], seperti terlihat pada gambar 3. Dengan menggunakan CGI maka akan dihasilkan suatu tampilan web yang dinamis, seperti pada kalkulator *online* dan kamus *online*. Pada kasus berikut CGI digunakan untuk mengeksekusi program yang dapat mengakses *hardware* seperti membaca dan menulis sebuah data ke port parallel. Untuk mengakses perangkat keras yang terdapat pada *server* seperti port parallel dan serial atau bahkan USB, digunakan GCC (GNU C Compiler). Contoh berikut diperlihatkan suatu potongan

program yang digunakan untuk mengakses port parallel menggunakan GCC, pemilihan port parallel sebagai sarana input dan output data dikarenakan kemudahan dalam pemrogramannya.

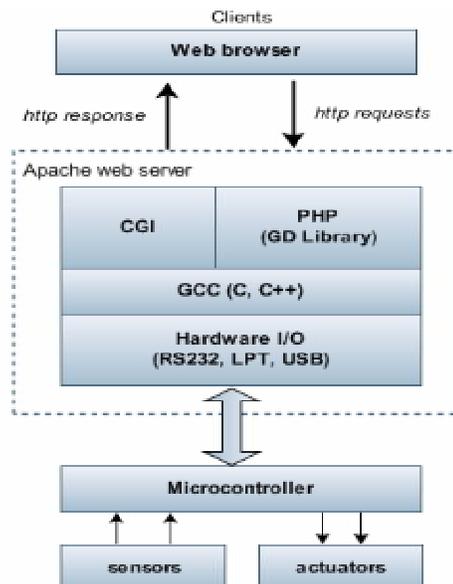
```
//send value to base address (0x378H)
outb(unsigned char value, base);
```

```
//read value from base address
int input;
input = inb(base);
```

Dengan menggunakan CGI, program C tersebut dapat diakses melalui *web browser* sehingga akhirnya dihasilkan suatu aplikasi pengukuran dan pengendalian berbasis web. Tag HTML berikut digunakan untuk mengeksekusi program CGI dengan nama *file* *read_data.cgi* melalui *web browser*

```
<form action="/cgi-bin/read_data.cgi"
method="get" name=" ">
<input value="Read Data" type="submit">
</form>
```

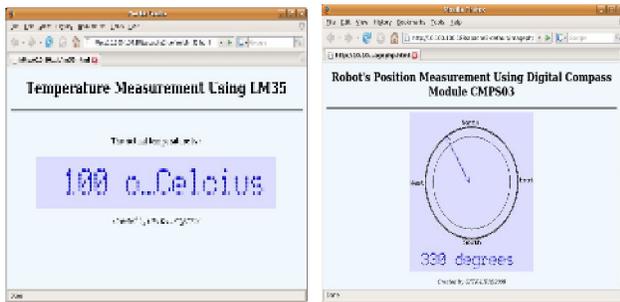
Untuk menghasilkan tampilan data hasil pengukuran lebih terlihat interaktif, maka dapat digunakan berbagai program aplikasi yang dapat merubah data tadi menjadi tampilan dalam betuk format JPEG, PNG atau GIF. Selain dengan menggunakan *Applet* dari JAVA dengan JNI-nya, salah satu metode lain yaitu dengan menggunakan *library* GD. Pada prinsipnya *library* ini dibuat menggunakan C dan digunakan untuk menghasilkan tampilan grafik, *chart* atau tampilan lainnya. *Library* GD tersebut dapat dipanggil dengan menggunakan program Perl, PHP, C, dsb.



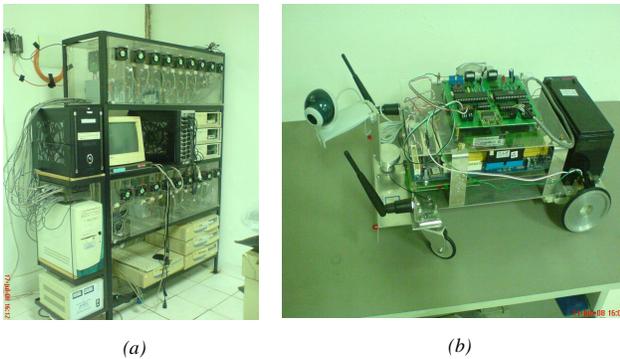
Gambar. 3. Alur program akses perangkat keras melalui web browser

Pada contoh aplikasi berikut diperlihatkan beberapa contoh aplikasi peralatan yang dapat dimonitor bahkan dikendalikan melalui jaringan. Untuk mengakses sistem tersebut hanya diperlukan *web browser* seperti terlihat pada gambar 4. Pada contoh pertama yaitu suatu sistem pengendali *power node* pada Public Cluster LIPI^[7] yang ditampilkan pada gambar 5.(a). Aplikasi ini terdiri dari sebuah *server* yang dapat mengendalikan *power* dari tiap-tiap *node*, jumlah maksimum *node* komputer yang dapat dikendalikan melalui jaringan tersebut yaitu 48 *node*. Aplikasi ini bertugas mengendalikan aktuator berupa relay AC yang terhubung ke tiap *node* komputer secara terpisah.

Pada aplikasi yang kedua memperlihatkan suatu prototipe robot yang dapat diakses melalui jaringan secara nirkabel^{[7][8]}, seperti terlihat pada gambar 5.(b). Tugas dari perancangan robot ini yaitu ditujukan dalam proses pengukuran yang membutuhkan mobilitas tinggi, dimana si pengguna dapat melakukan pengukuran tanpa berinteraksi langsung dengan objek yang akan diukur. Robot ini dapat digunakan pada pengukuran suatu gas berbahaya atau pengukuran di tempat yang mempunyai tingkat radiasi yang tinggi.



Gambar. 4. Ilustrasi pengukuran menggunakan web browser



(a)

(b)

Gambar. 5. (a).Public Cluster LIPI (b).Robot Nirkabel Modular berbasis web

Pemanfaatan teknologi informasi beserta sistem operasi Linux dalam proses otomatisasi dan pengukuran khususnya untuk penerapan di lingkungan laboratorium atau pun industri akan mempermudah pengguna dari segi fleksibilitas dalam mengakses semua peralatan dimanapun kita berada selama masih terhubung ke jaringan internet, tanpa harus berhadapan secara langsung dengan peralatan yang akan dioperasikan. Di samping itu penggunaan sistem operasi Linux yang bersifat *Open Source* akan mengurangi biaya dalam hal pembelian *software*.

REFERENSI

- [1] Miodrag Bolic, Vujo Drndarevic, Branko Samardzic, "Distributed measurement and control sistem based on microcontroller with automatic program detection," Elsevier, June 2000.
- [2]. M. Can Filibeli, Ozgur Ozkasap, M. Reha Civanlar, "Embedded web server-based home appliance network", Journal of Network and Computer Application, 2005.
- [3] Dhananjay V.Grade, "Programming and Customizing the AVR microcontroller, McGraw-Hill, 2001.
- [4] Claus Kuhnel, "BASCOM Programming microcontroller with easy ", Universal Publisher, 2001.
- [5]. Matthew Chapman, "The final word on 8051", 1994.
- [6]. First Regional Workshop on Distributed Laboratory Instrumentation Systems, Volume II.
- [7]. I. Firmansyah, B. Hermanto, Hadiyanto and L.T. Handoko, "Real-time control and monitoring system for LIPI's Public Cluster", 2007.
- [8]. LIPI Wireless Robot, <http://robot.teori.fisika.lipi.go.id>.
- [9]. LIPI Networked Robot, <http://opennr.teori.fisika.lipi.go.id>.