Distributed Monitoring and Controlling Using Microcontroller and Virtual Internet Protocol

Ahmad Ashari
Electronics and Instrumentation Laboratory, FMIPA, Gadjah Mada University (UGM)
Sekip Utara Yogyakarta – Indonesia. Tel.+62-274-902382
e-mail: ashari@ugm.ac.id

Abstrak

Saat ini aplikasi berbasis jaringan TCP/IP (transmission control protocol/internet protocol) berkembang dengan pesat. Dengan jaringan TCP/IP fungsi pemantauan dan pengendalian akan lebih mudah serta dapat diakses secara luas. Setiap komputer pribadi (PC) dapat dihubungkan dengan banyak mikrokontroler, namun masih terdapat kesulitan untuk mengakses langsung pada masing-masing mikrokontroler ini melalui jaringan TCP/IP. Pada penelitian ini dikaji tentang pemanfaatan IP virtual pada PC yang terhubung dengan beberapa mikrokontroler, sedang setiap mikrokontroler dihubungkan ke perangkat sensor untuk pemantauan suhu dan relay untuk pengendalian. Dengan cara seperti ini, setiap rangkaian mikrokontroler dapat diakses secara langsung menggunakan IP yang diperjanjikan padanya. Hasil pengujian menunjukkan bahwa sistem ini secara keseluruhan dapat berfungsi sebagai sistem pemantau dan pengendali tersebar yang mempunyai kemampuan merekam dan menampilkan hasil pemantauannya dalam bentuk grafik.

Kata-kata kunci: *IP virtual, mikrokontroler, pemantau dan pengendali tersebar*

Abstract

Nowadays, the applications based on TCP/IP (transmission control protocol/internet protocol) network are growing rapidly. By this application monitoring and control functions are easily and widely accessible through networks. Every PC (Personal Computer) can be connected to several microcontroller, but there are difficulties to access microcontroller directly through TCP/IP networks. This research study the use of virtual IP on a PC that connected with several microcontrollers, where each microcontroller is connected to a sensor device to monitoring temperature and a relay for controlling. In this method, each set of the microcontroller can be accessed directly using the IP. The results show that this system can be a distributed monitoring and controlling system which has the ability to record and display the results in the form of graphs.

Keywords: *virtual IP, microcontroller, distributed monitoring and controlling*

1. Introduction

Nowaday, applications based on TCP/IP (Transmission Control Protocol/Internet Protocol) network are growing rapidly which is impact to the world of electronics and instrumentation. Monitoring and control functions will be easier if it is based on TCP/IP network. Because the monitoring and control functions that can be connected into the network TCP/IP that can be widely accessible.

TCP/IP network in a computer network using standard protocols are popular and widely used both for Internet and Intranet. By using the TCP protocol that implement connection-based communication, the validity of transmitted data can be guaranteed because if there is corrupted data packet, the packet will be resent. TCP/IP is very suitable when applied to a control system and monitoring system that gave priority to the validity of the data. TCP/IP model does not entirely refer to the Open Systems Interconnection (OSI) seven layer models but rather use their own models. In the model of TCP/IP each layer (layer) has a specific function including addressing function [1].

Linux operating system has the ability to have many IP addresses on one network interface is called IP Aliasing [2]. By leveraging IP Aliasing on Linux and make the microcontroller as the equipment being controlled and monitored, then the microcontroller can have a Virtual IP, so that controlling and monitoring functions can be performed in a distributed manner. Based on the description above, the implementation of server applications for microcontroller-based Virtual IP TCP/IP is required. This application will translate the address and data from TCP/IP protocol into a data format in accordance with the microcontroller.

IP aliasing is the process of adding an IP address on a Network Interface Card. Referring to the Layer 1 and 2 models of TCP/IP, IP aliasing allows a Media Access Control (MAC) address has more than one IP address. On the Linux operating system IP aliasing is implemented in the kernel starting 1995 [2].

Atmel AVR microcontroller has been widely used for interesting applications [3]. The use of Linux to control the Atmel microcontroller via a serial or parallel port interface can also be done [4], and the programming for Atmel microcontroller can performed in Linux using C programming language [5]. Integration of multiple sensor networks in a single virtual sensor network has been carried out by Lei et al. [6].

Network programming by using Java programming language allows a program based on client-server applications [7], [8], in which the program can run in all operating systems, not only Linux operating system. Accessing database server from Java programming language need a JDBC connector [8], and the Web application that accessing database can be performed with Java Programming Language [8] or with PHP scripting language [9].

2. Research Method

2.1. System Architecture

System for monitoring and controlling distributed applications using the microcontroller and the virtual IP, the general pledge to work with the principle of one each IP address to an address of serial port that is connecting to the circuit microcontroller. This application is also accompanied by a web-based application that can be accessed from a Browser. Globally, the architecture of this system shown in Figure 1.

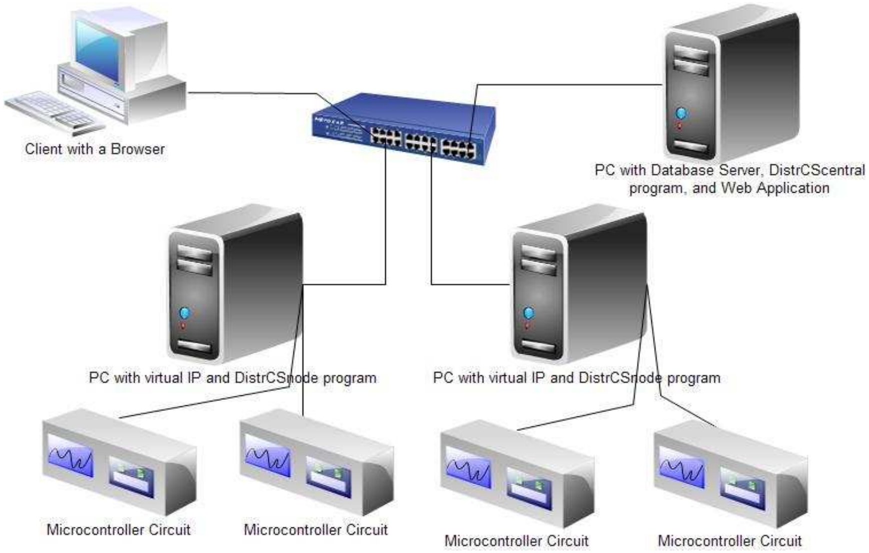

Client with a Browser
PC with Database Server, DistrCScentral program, and Web Application
PC with virtual IP and DistrCSnode program
PC with virtual IP and DistrCSnode program
Microcontroller Circuit
Microcontroller Circuit
Microcontroller Circuit
Microcontroller Circuit

Figure 1. The Architecture of Distributed Monitoring and Controlling System

2.2. Program Structure

There are three applications have been developed for this system, i.e.: First, a web-based applications using PHP. Second, an application using Java for the server is namely DistrCScentral (Distributed Control System central). Third, an application that also using Java for on each node is namely DistrCSnode (Distributed Control System node). The application has IP Virtual that connected to the microcontroller circuit via serial port.

A Client computer via browser can be used for monitoring the temperature and provide controlling commands through the server for the node which is monitored and controlled. This is using a Web based application that has a structure which is shown in Figure 2.

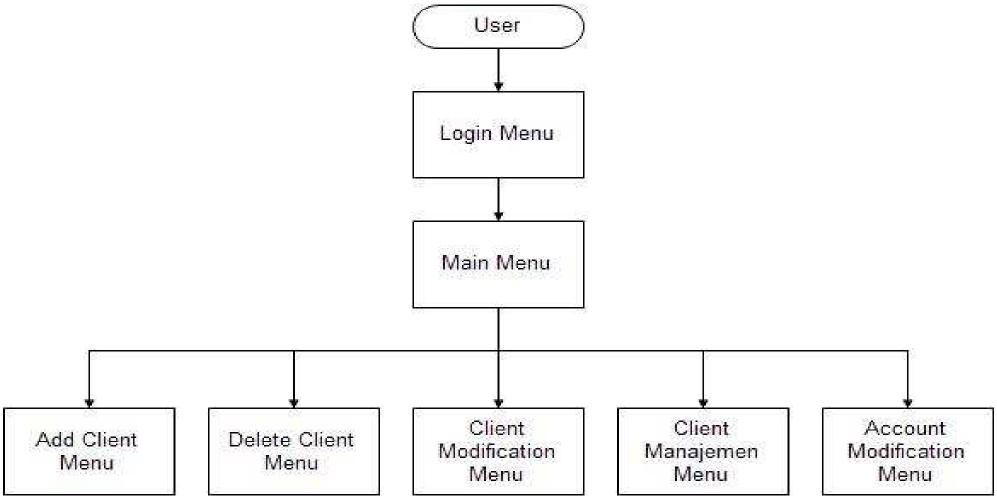


Figure 2. Diagram Structure of Web Application

DistriCScentral application program is a program that serves to link the database on the server with the node that is monitored and controlled. This application uses the Java language and requires mysql jdbc libraries to regular communication with the database mysql server.

DistriCSnode application program is a program that serves to portend an IP address to an address of serial port that is connecting to the circuit microcontroller. This application uses the Java language and requires a Java RXTX library for communication with serial port. This application has 2 main functions, such as: handle connection with computer monitors and controllers (server), and handle the interface to the circuit microcontroller through the serial port.

From the above descriptions, its need to develop three classes that support these functions, i.e.:

1. *Main* Class
   The main class is the class that is automatically executed when the application program is executed. This class has the following functions: opening the port by *Socket* class, create a connection in the port number1234 and executing the class that handles the communication with the computer which monitoring and controlling via the stream protocol.
   At the time the program starts, the first stage is to find the IP of the computer server. If the IP of the server is not found, then the process will be terminated. If IP of the server has been found, then socket on the port 1234 will be created. The object socket connection is made to perform data communication between computer server and all computer node.
2. *Input Handler* and *Output Handler* class
   These classes handle input and output data streams via connection that made for monitoring and controlling. Data received from the stream input of the string. The data is then sent to a microcontroller for processing via a serial interface handling class. Handling Input and Output class has a function of observation to the data input of serial interface handling class. If there is incoming data from serial port, this class of computer monitors and sends it to the controller via the output stream. Flow diagram of the input and output handling classes depicted in Figure 3.
3. *Serial Port Daemon* class
   Serial Port Daemon class is run from Input Handler and Output Handler class. This class handles the interface with a microcontroller via a serial port interface, it’s have the following function:

a) Identify the port.
b) Determining the parameters of the serial port baud rate, data bits, stop bits and parity bits. This application uses the 9600 baud rate and 8N1 (8 data bits, no parity bit, and 1 stop bit) format.
c) Sending data to the microcontroller through the serial port output stream.
d) Receiving data from the microcontroller through the serial port input stream.

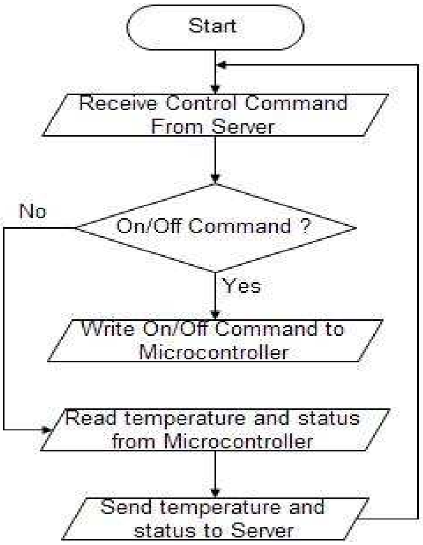

Start
Receive Control Command From Server
On/Off Command ?
No
Yes
Write On/Off Command to Microcontroller
Read temperature and status from Microcontroller
Send temperature and status to Server

Figure 3. Flowchart of Input Handler and Output Handler class

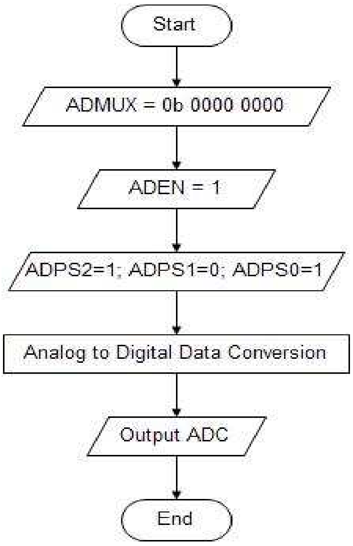


Figure 4. Flowchart of conversion process from analog to digital

Microcontroller applications program written using C language and compile with AVR-GCC compiler [5]. This application handles the analog signal conversion from a temperature sensor with an analog to digital (ADC) of microcontroller and send/receive data to/from PC via serial port communication. Figure 4 shows the flowchart of this program.

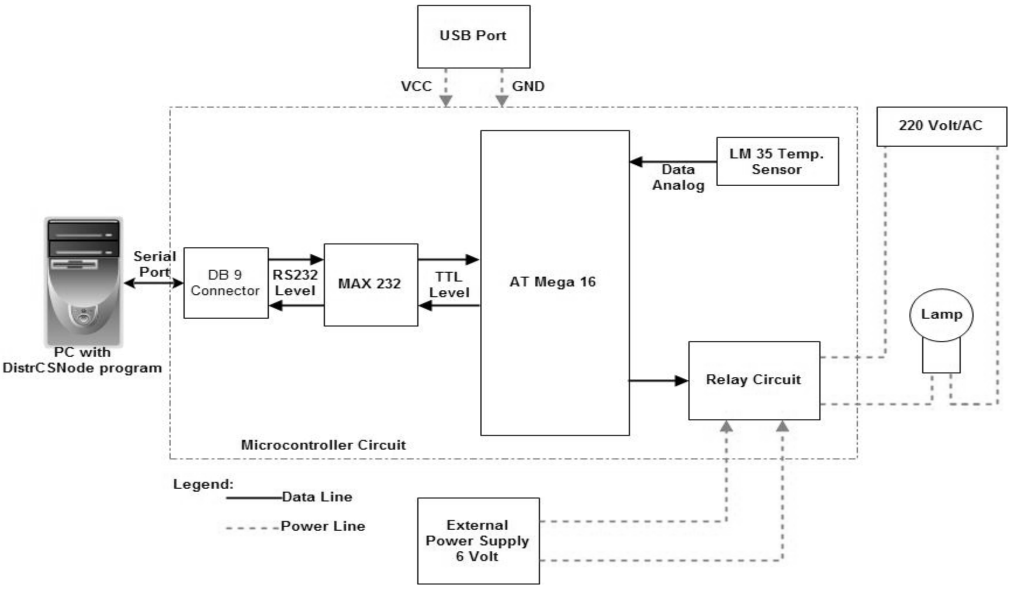


Figure 5. Hardware Block Diagram

In order for microcontroller can communicate with the computer, first must be performed an initialization. Initialization is performed to determine the baud rate and activate the serial input and output ports. Baud rate microcontroller is set by giving a specific value off the register in microcontroller AT Mega 16.

2.3. Hardware Block Diagram and Microcontroller Circuit

The hardware used includes a microcontroller AT Mega 16 [10], the IC MAX232, LM35 temperature sensor and the Lamp module. In this application there are two levels of logic, ie RS232 levels and TTL levels. To bridge the two levels of logic are used circuit MAX232. The series is a bridge between the serial ports of a computer with a microcontroller. Microcontroller handles serial input circuit which is already in the TTL logic level, to handle the voltage readings from the temperature sensor and relay control circuit via the output port. Relay circuit is a series of transistors as switches which become relay drivers. This circuit uses two power supply is V_{CC} and ground from the USB port and external power supply. Use the USB port as a voltage source aims to provide ease of installation and during the process of writing a program on the microcontroller. Scheme of the hardware can be shown in Figure 5.

Microcontroller circuits have function to controls data transmission from sensors to a computer and control the lamp. The series of microcontrollers that are used in this application uses an external clock source generated from frequency 4 MHz crystal. The reading of the voltage signal is generated by the LM35 temperature sensor which is read by the port ADC0 pin 40. Control signal ON/OFF generated by the microcontroller via the output port of the PD 7. Control signals are used to control the relay circuit. Schematic circuit microcontroller with LM35 temperature sensor and relay circuit can be shown in Figure 6.

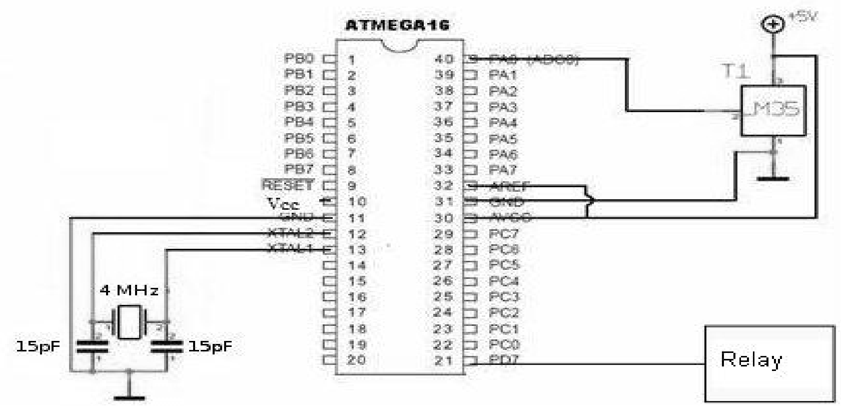


Figure 6. Microcontroller connection with LM35 and relay

2.1. Hardware and Software Configuration

DistrCScentral program and Web based applications running on a notebook computer as server with the following hardware specifications: processor Intel Core 2 Duo, memory DDR2 1 GB, hard disk 160 GB, and network interface card Broadcom Netlink Gigabit Ethernet.

DistrCSnode program running on a desktop computer with hardware specifications: processor Intel Celeron Dual Core, memory DDR2 1 GB, hard disk 80 GB and network interface card RTL 8139 10/100 Mb/s.

All computers (nodes and servers) are connected through a Cisco Switch 10/100 Mbps. The microcontroller circuits are connected to each node through the serial port. The server uses the following software: Windows XP, JRE 6 update 7 with mysql jdbc connector, MySQL 4, Apache Web Server, PHP 4 and DistrCScentral and Web server applications. Node A and node B use the following software: Linux, JRE 6 update 7 with RXTX 2.1-7 library and DistrCSnode application.

Network cards must be configured before all computers can communicate. Implementation of the system performed on a LAN (Local Area Network). On a local network, central computer (server) and the nodes must be within range of IP the same. Range of IP on a network is determined by the value of the netmask. Configuring your network card from the computer - a computer that is used is as follows:

a. Server
   Server for this application can use Linux or Windows operating system, but for this test used the Windows XP operating system with the following configuration.
   a) IP Address : 192.168.0.230
   b) Subnet Mask : 255.255.255.0

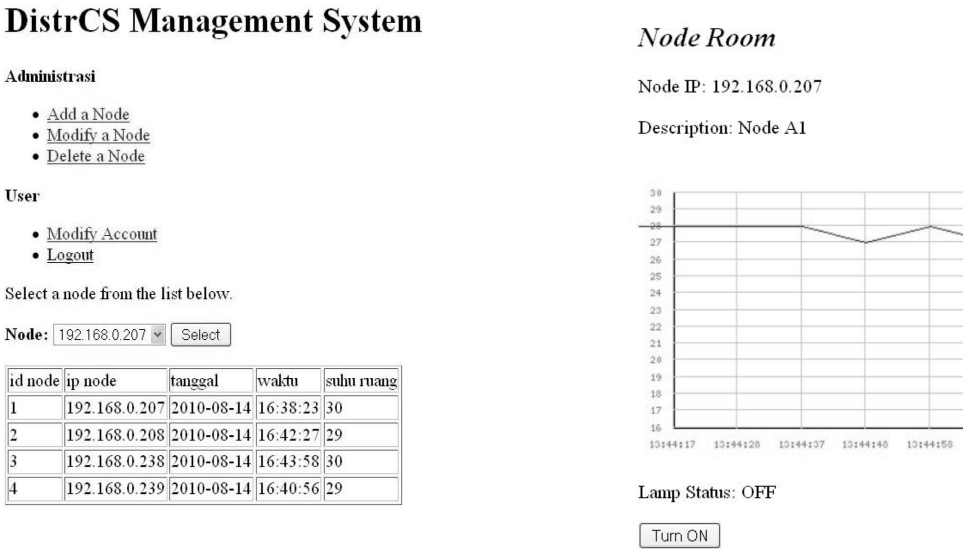
b. Node A
   Linux operating system with the following configuration:
   a) IP Address : 192.168.0.207 (for Node A1)
   b) IP Address virtual : 192.168.0.208 (for Node A2)
   c) Subnet Mask : 255.255.255.0

c. Node B
   Linux operating system with the following configuration:
   a) IP Address : 192.168.0.238 (for Node B1)
   b) IP Address virtual : 192.168.0.239 (for Node B2)
   c) Subnet Mask : 255.255.255.0

3. Results and Analysis

The testing have aims to investigate the validity and capability of this system, which includes three issues such as:

a. The validity of IP address and microcontroller translation in the IP Virtual system and distributed systems.
b. The validity of command sending for the light turns on and off, and the validity of reading the lamp status.
c. The ability of the system in the recording of data per 10 seconds and displays them in graphical form.



DistrCS Management System

Administrasi

- Add a Node
- Modify a Node
- Delete a Node

User

- Modify Account
- Logout

Select a node from the list below.

Node: 192.168.0.207 Select

id node	ip node	tanggal	waktu	suhu ruang
1	192.168.0.207	2010-08-14	16:38:23	30
2	192.168.0.208	2010-08-14	16:42:27	29
3	192.168.0.238	2010-08-14	16:43:58	30
4	192.168.0.239	2010-08-14	16:40:56	29

Node Room

Node IP: 192.168.0.207

Description: Node A1

Lamp Status: OFF

a. b.

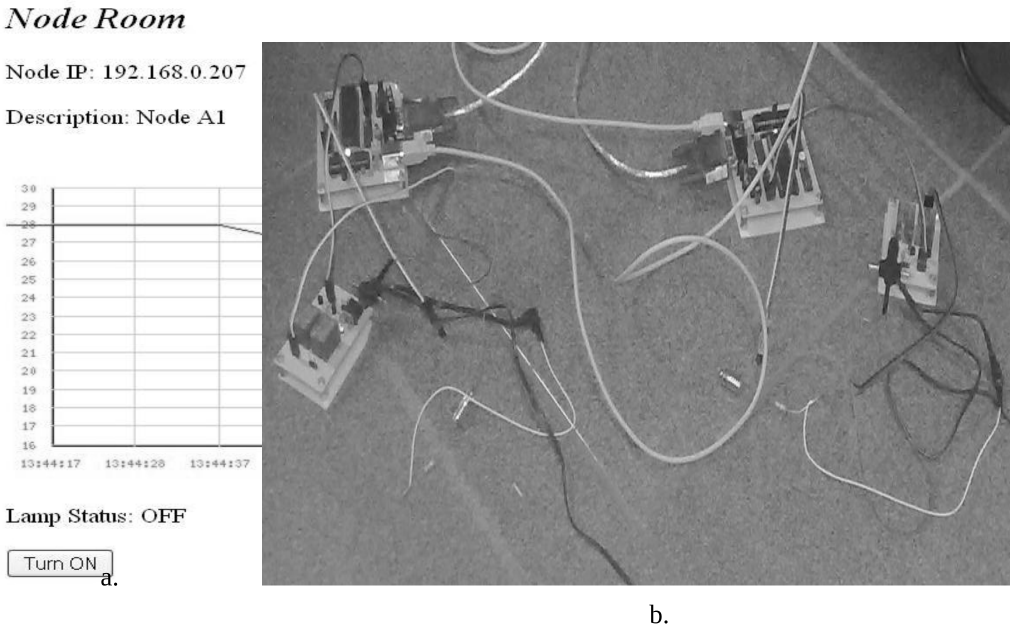
Figure 7. The test results of the validity of the IP address and the microcontroller
a. First page of Web application b. IP and Node A1

From the testing that has been done, the results obtained as follows:

a) The test results of the validity of the IP address and the microcontroller (see Figure 7); it's shown that the IP address and Microcontroller have been read correctly.

b) The test results of the status of the Lamp was *off* in Figure 8 b and the button for sending command to turn on the Lamp in Figure 8 a, also the Lamp is *on* in Figure 9 b and the button for sending command to turn back off the Lamp in Figure 9 a. Its shows that the reading of status of the Lamp and sending commands to turn on or turn off the Lamp was work properly.

c) The test on the system's ability for data recording and display it’s in graphics form can be shown in Figure 10 (for the Node B2). These result shows that reading/monitoring the temperature was recorded every 10 second.



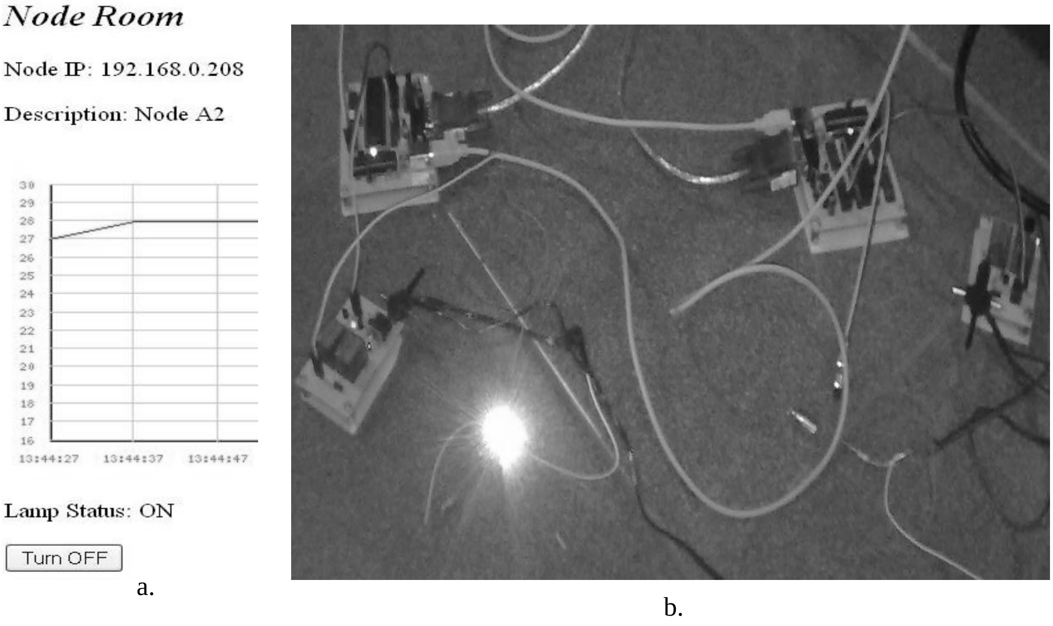
Node Room

Node IP: 192.168.0.207

Description: Node A1

Lamp Status: OFF

Figure 8. The test results of the status of Node A2,
a. status of the lamp, (b) the lamp is off for Node A1



Node Room

Node IP: 192.168.0.208

Description: Node A2

Lamp Status: ON

Figure 9. . The test results of the status of Node A2,
a. status of the lamp, b. the lamp is on for Node A2

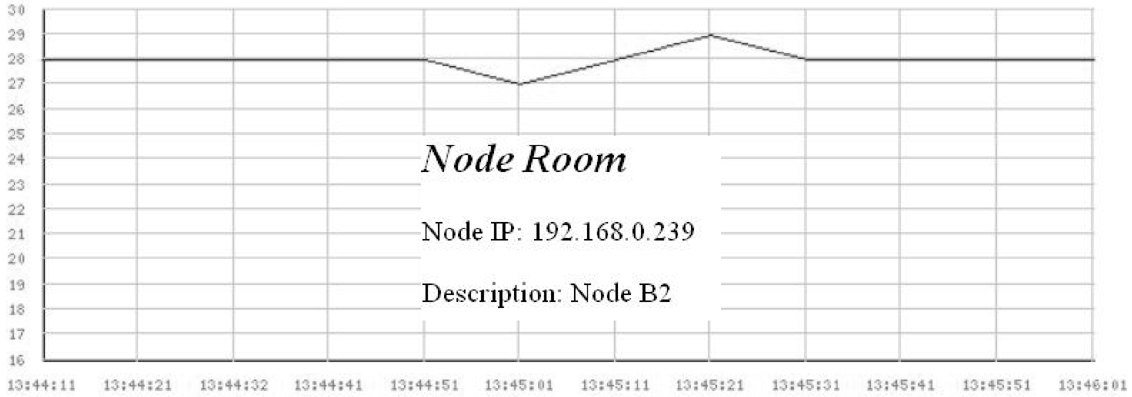


Figure 10. Graphic of Recorded Data every 10 second for Node B2

4. Conclusion

From the results can be concluded that the distributed monitoring and controlling system by using a microcontroller and a virtual IP can mapping a single IP address on the PC directly to the appropriate microcontroller. It can be easily accessed through the network based on TCP/IP for monitoring temperature and controlling the spread. It can also record and display temperature monitoring every 10 seconds in a graph.

Acknowledgment

Acknowledgements submitted to Studies Program S1 Electronics and Instrumentations FMIPA UGM which has funded this research, also to my both colleges Dipto and Hisham who has helped in configuration and testing the system.

References

[1] Tanenbaum AS. Computer Network. 4th edition. New York: Prentice Hall. 2003.
[2] Zhang S, Chen T, Cai X. *The Design of an Embedded System Based on ARM Microprocessor and Implementation of an IP-sharer with IP tables of Linux*. IEEE International Conference on Control and Automation (ICCA). Guangzhou. 2007: 207-210.
[3] Korbel S, Janes V. *Interesting Applications of Atmel AVR Microcontrollers.* IEEE Euromicro Symposium on Digital System Design (DSD'04). France. 2004.
[4] Deegan P. Developing for the Atmel AVR Microcontroller on Linux. *Linux Journal*. 2005; 17 (130): 10.
[5] Slapovskaya YP, Maslov DM. *Upgrading of laboratory facilities Microcontrollers and automation*. International Conference on Actual Problems of Electron Devices Engineering (APEDE). Saratov. 2010: 363-369.
[6] Lei S, Xu H, Xiaoling W, Lin Z, Cho J, Lee S. VIP Bridge: *Integrating Several Sensor Networks into One Virtual Sensor Network*. IEEE International Conference on Internet Surveillance and Protection (ICISP). France. 2006.
[7] Hallot ER. Java Network programming. 3rd edition. New York: Prentice Hall. 2004.
[8] Hall M, Brown L. Core Web programming. 2nd edition. New York: Prentice Hall. 2001.
[9] Lane D, William HE. Web Database Application with PHP and MySQL. 2nd edition. O'Reilly. 2004.
[10] Chao M, Qingli L, Zhongyuan L, Yu J. *Low cost AVR microcontroller development kit for undergraduate laboratory and take-home pedagogies*. 2nd International Conference on Education Technology and Computer (ICETC). Shanghai. 2010; 1: 35-38.