논문 2011-48CI-1-14

가상 로봇 교육 시스템 설계 및 구현

(Design and Implementation of a Virtual Robot Education System)

웅 홍 우*, 소 원 호**

(Xiong Hongyu and Wonho So)

요 약

본 논문에서는 레고 마인드스톰 NXT 로봇을 이용한 프로그래밍 교육을 위한 가상 로봇 교육 시스템 (VRES; Virtual Robot Education System)을 설계하고 구현한다. 제안된 시스템을 통하여 프로그램 학습자는 소스 코드를 편집, 컴파일, 그리고 로봇에 다운로드하여 자신의 실행 코드를 동작시킨다. 로봇을 관찰하기 위하여 시스템은 웹 카메라를 포함하고 있어 모니터링 서비스를 제공한다. 따라서 학생들은 자신의 프로그램을 다운로드한 로봇의 동작을 자세하게 검증할 수 있으며 필요시 디버깅 할 수 있다. 추가로 간단한 사용자 친화적 프로그래밍 언어와 이에 대한 컴파일러를 설계한다. 이러한 도구를 이용하여 학습자는 자바 언어보다 쉽게 NXT 로봇 프로그램을 생성하여 테스트할 수 있다. 교수자는 시스템에서 제공하는 직접 제어 모드를 이용하여 수업 주제를 위한 로봇의 제어와 관리가 가능하다. 그럼으로, 제안된 시스템은 학생들이 정규 수업 또는 방과 후에 인터넷과 웹브라우저를 사용하여 로봇 프로그래밍을 학습할 수 있도록 지원할 수 있다.

Abstract

Virtual Robot Education System (VRES), which is for programming education with a Lego Mindstorm NXT robot, is designed and implemented. Through this system, program learners can edit source code, compile, download it into the robot, and run their executive program. In order to observe it, the system includes web cameras and provide monitoring services. Thus, students are able to verify the operation of robot into which they download their program in detail and to debug if necessary. In addition, we design a new simple user-friendly programming language and a corresponding compiler for it. With those tools, learner can more easily create programs for NXT robot and test them than Java language. A educator can control and manage the robot for the subject of a class with direct control mode of our system. Therefore, the proposed system is able to support students to learn robot programming during or after regular classes with web browsers through Internet.

Keywords: Web based System, Multimedia, Robot education, Computer education, Robot programming

I. Introduction

Recently, there are a lot of web applications as education services running in the Internet. Those web-based applications eliminate the limits of time and space and provide a convenient user interfaces for self-study. Those applications are used for simulation, engineering, environment, and so on[1~4].

In [1], a web-based education system for predicate logic, for instance, is showed and describe it operation. In addition, it shows a basic frame of web-based education system. A JINI (pronounced genie), which is a network architecture for the construction of distributed systems in the form of modular co-operating services, is used to remotely control the LEGO Mindstorms RCX robot. But that is

* 정회원, 순천대학교 과학정보융합학과
(Dept. of Science Information Conversion, Sunchon National University)
** 정회원-교신저자, 순천대학교 컴퓨터교육과
(Dept. of Computer Education, Sunchon National University)
※ This work was supported in part by Mid-career Researcher Program through NRF grant funded by the MEST (No. 2009-0086676)
접수일자: 2010년8월9일, 수정완료일: 2010년12월30일

not a real web-based robot application[2].

Sagiroblu and et al. implement a web robot learning system in which Bluetooth communication module is bulit-in[3]. Cedazo and Lopez proposed a platform for web-based laboratories include robotics [4]. These are the existing systems for robot control or education systems for robot programming study. But it is difficult to observe robot activities and test learners' codes and to debug. If there is a web-based environment with those critical gadgets for useful robot education, each student can use the system and access the robot at school or even at home. And finally, the utilization of the robot and its system also be improved.

As a commercial product, Lego Mindstorms Robot is very popular in robotics, design, and mechanical educations. The latest version is called NXT. That is one of very expensive products. It, therefore, is very difficult for each learner to access a NXT robot and to program it[5]. According to the existing works, however, it is practicable to design and implement a web-based environment for java robot programming [6~8]. Supported by an open source project called LeJOS which provides a Java based firmware can be deployed in NXT robot. With LeJOS, Java program can be used to control NXT robot. LeJOS also provides some application programming interface in Java and a collection of developing tools. There is a Java programming library called LeJOS NXJ provides a lot of packages for the robotic programming with NXT robot[9].

In this paper, we consider the powerful functions of NXT robot and the drawback of its cost, access, and monitoring on-line. In addition, Java based tools are used to constructed a virtual robot education system (VRES). Thus, some users including a few learners and an educator can create source code in Java, and then compile it simultaneously. Even though Java language can be used to create various programs, it is a little difficult for beginner to study computer or robot program. It means other simple way is needed for beginners. So a new script-like language is designed to control NXT robot, and an interpreter is provided to convert the script to Java object code in our system.

The rest of this paper is organized as follows. Section Ⅱ describes the framework of system, contains the function modules and the user interfaces. Section Ⅲ illustrates the implementation of some function modules in detail. In section Ⅳ, some issues related to the usage of the system are explained. Finally, the conclusion and future work is presented in section Ⅴ.

Ⅱ. System Design

Figure 1 depicts the proposed VRES and the environment for it. This education system as a server can support lots of students who may locate in different spaces. With this system, users can create a robot program, compile it, and download its executable file into control NXT robot. And users can monitor the state of robot remotely through video and audio on the web on real-time.

1. Module Design

The functions required for VRES are provided by following modules: Login, File Manager, User Manager, Editor, Compiler, Loader, Monitor, Robot

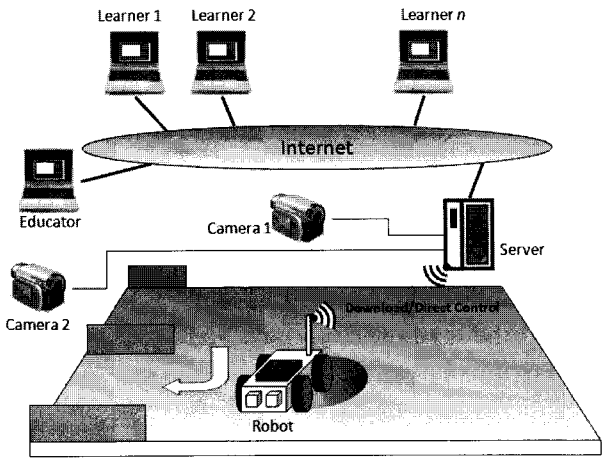

그림 1. 가상 로봇 교육 시스템 환경
Fig. 1. The environment of Virtual robot education system.

Access Controller, Instant Commander, and Responder (Figure 2). ***Login*** guarantees the user is a registered user through ID and password authentication. There are two types of users who use this system. One is normal user used by student as learner. Another is the administrator for this system used by teacher as educator.

File Manager can show files belong to the login user. And the owner of the files can open, save, rename, or delete them by this module.

User Manager Module is used to add, delete, or modify a user of this system. This module is accessed by an administrator, specially an educator in class.

Editor is used to edit the source code on the web page. Through this function, a learner can also debug his program having compile errors or runtime ones.

Compiler can convert a source code organized in a simple user-friendly programming language into Java code, and compile this code into Java byte code. In this paper, the simple programming language is called NXJScript(NXJS) which is constructed for programming beginner.

Loader is used to download an executable program file to the robot, and run it in the robot. For this function, VRES system uses a wireless communication interface of Bluetooth between a remote robot.

Monitor is implemented in Java and its functions have to be embedded into VRES server and user client system. The server can capture two video and audio data when the remote robot runs. And then it send those multimedia data to clients through each connection. The client can play video and audio on the web. With this mechanism, users are able to observe the action of robot and verify their program operations.

Robot Access Controller is designed to assign the robot for users. It provide two kind of schemes as robot access control. For *Auto-Mode*, it is based on FCFS (First-Come, First-Service) queue system working in the server, users follow the FCFS mechanism to access the robot and they have to release the control for the first waiting user after the end of execution. Under *Manual-Mode*, only one user who has been assigned can access robot. It means he is able to download his executable files, run and test it. In case of educator, he also use direct control module and control robot in real time. Other users, however, only observe the robot activities without the access.

Instant Commander provides the instant controlling interface for the robot. An educator can control the robot instantly. Responder show users the responses related to user's operation. It works as a common interface for other modules to communicate with the end users.

2. Interface Design

The web interface for both users and administrators is shown in Figure 3. There are a lot of elements in that interface frame: buttons, name list of the files, java applet, and text area. First, the user information and basic programming functions' area includes *Login*, *File Manager*, *Editor*, *Compiler*, and *Loader* modules. Second, the monitoring services' area is for *Monitor* module with which users can activate multimedia interface. The robot access control services are used to request and to release the access of robot. Through the project directory and files for each user and the source code area, source and executable files are managed and edited respectively. Finally, the user interactive console for progress help users to review the response of VRES and its services.

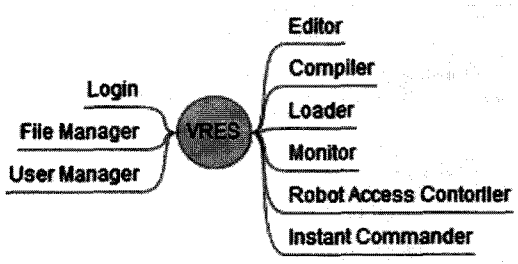

그림 2. VRES 함수 모듈
Fig. 2. The function modules for VRES.

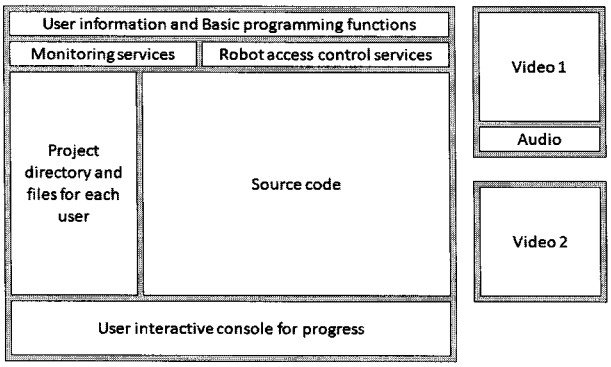

그림 3. VRES 웹 GUI
Fig. 3. Web GUI for virtual robot education system (VRES).

Figure 4 shows the frame of system console which can be only used by system administrator. In administrator functions area, an educator as a administrator use *User Manager* module so that he can create a new user, see online users, and check robot state. With the Bluetooth communication open and close, and arrows, the educator create data channel and directly control the remote robot and enable it to move a specific position for a class.

III. VRES Implementation

In this paper, the apache web server is used to provide web service. And it uses PHP dynamic web pages to support interactions with clients. A lot of web pages containing PHP, HTML, and JavaScript program files are created for VRES function modules. Login, File Manager, Editor, User Manager, and Responder, these modules can be implemented by very conventional methods. The following paragraphs describe the details about the implementation for Compiler, Loader, Monitor, Robot Access Control, and Instant Commander these five modules.

Compiler: When users click the compile button on the web page, the php program will invoke three external executable files in the server to do compiling. The schema is showed in Figure 5. There are three external programs, nxjsc, nxjc, and nxjlink in the server. NXJ Script (NXJS) is a simple language proposed in this project. It is designed as a

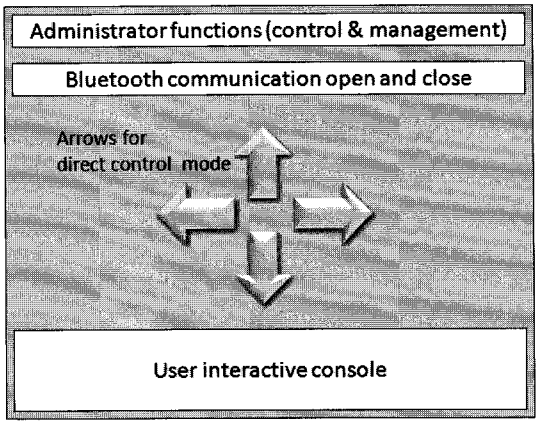

그림 4. 관리자 직접 제어를 위한 웹 GUI
Fig. 4. Web GUI for the direct control by administrator.

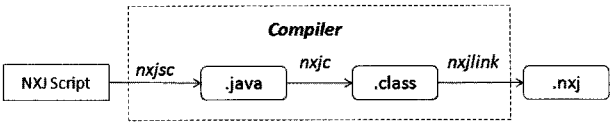

그림 5. 컴파일 스키마
Fig. 5. Schema of compile.

표 1. 명령어 집합
Table 1. Instruction set.

Instruction	Action of Robot
forward (s)	forward in time s
backward (s)	backward in time s
turnleft (s)	turnlefting in time s
turnright (s)	turnrighting in time s
rotate (s)	rotate with clockwise in time s
wait (s)	do current action for s seconds
stop (0)	stop the robot

line:= *instruction*
instruction:= *keyword (number)*
keyword:= *forward* | *backward* | *turnleft* | *turnright* | *rotate* | *wait* | *stop*
number:= *integer* | *float*

그림 6. NXJS의 문법
Fig. 6. NXJS's grammar.

script language. Each line contains one instruction to control the robot. NXJS currently has seven instructions listed in Table 1.

Figure 6 shows the grammar of NXJS. The program nxjsc is an interpreter created by us in C supported with Flex and Bison[10]. It compile the source code written in NXJS into Java source code as .java file.

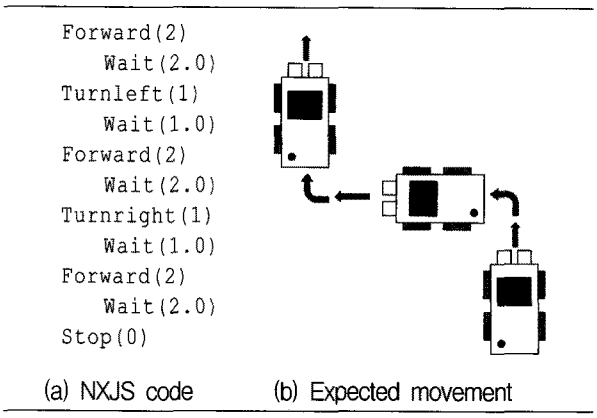

그림 7. NXJS 프로그래밍 예
Fig. 7. A NXJS programming example.

Figure 7 shows an example of source code in NXJS and the expected movement of robot. The external program nxjc and nxjlink are provided by leJOS NXJ command line tools. Command nxjc can compile the Java source code into Java byte code. A successful compiling of nxjc generates a .class file. Command nxjlink links the .class file with other necessary libraries to generate an .nxj file.

Loader: Similar to the compiler when users click the run button in the toolbar, the php program will invoke another external program to download the .nxj file to robot and start running. leJOS NXJ command line tools provide a command tool nxj to download and run the .nxj file. It can be illustrated by Figure 8.

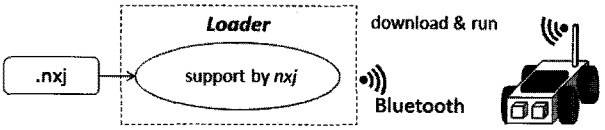

그림 8. 로더 스키마
Fig. 8. The schema of *Loader.*

Robot Access Controller: Figure 9 depicts two type of robot access control modes. The auto mode is based on a FCFS mechanism. The FCFS queue in the server registers all requests from clients. Any time, only the user in the head of queue access the robot and download own executable program into it. After finishing an experiment, the user who controls the robot have to release the robot. In this example, user 1 is using the robot and user 2 and 5 are waiting for the access control. The scheduler server is implemented by java. The applet embedded in the web browser communicates with VRES server by TCP/IP socket. Supervising educator determines the access control mode for VRES and he also set the system to normal mode in which learners have to receive an acceptance from eucator.

Monitor: Java Media Framework (JMF) is a library which provides real time multi-media service on the web[11]. In VRES, a java program uses JMF to capture image and sound. The collected information is sent to the client through the Internet. In our test-bed, two computers are used to provide video and audio services. One computer only captures video, and the second computer captures video and audio simultaneously. Both of two media servers send media information to each user with Real Time Transport Protocol (RTP) when they request this service by using multimedia service buttons[12]. Two videos give users a two-aspect view of the robot. The audio service strengthen the feeling of reality and interactivity. Each user accesses the video and audio service through the Java applet with JMF embedded in the webpage. The service model can be illustrated by Figure 10.

Instant Commander: There is an open source package called icommand which controls NXT robot instantly. In VRES, a command processor as a background job is run at the system server. NXT

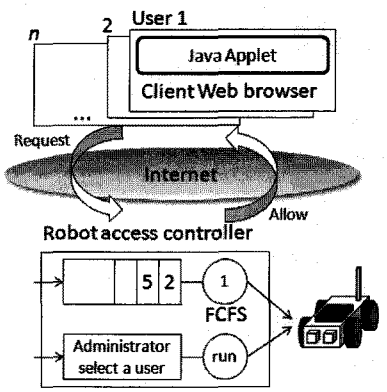

그림 9. 두 가지 로봇 접근 제어
Fig. 9. Two types of robot access controls.

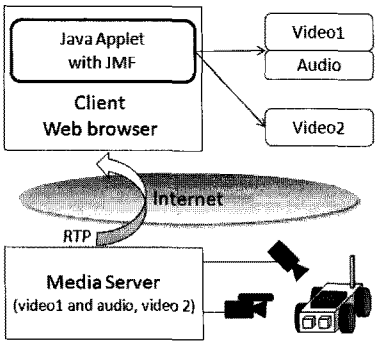

그림 10. 모니터 서비스 모델
Fig. 10. The service model of Monitor.

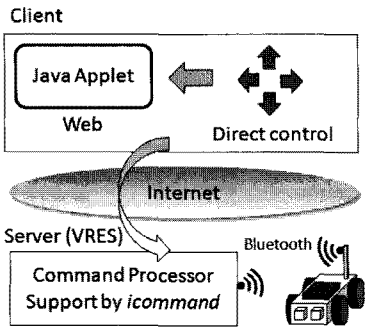

그림 11. 인스턴트 명령 서비스 모델
Fig. 11. The service model of instant commander.

robot connects with that command processor through Bluetooth device module. A Java applet embedded in the web page communicates with the command processor. When an arrow button in the applet is clicked, the corresponding action command will be sent to the command processor. Finally, the action of NXT robot will follow that command. Figure 11 shows us the model of this approach.

IV. Operation and Discussion

In order to use VRES and carry on a class, all students and a teacher need to use web browser. In our VRES, we recommend the google chrome and the MS explorer. All users, typically, have to install a Java Runtime Environment (JRE) in their computer. It is necessary for the applets in VRES. Another important configuration is related to the security of Java applet. The Java applet cannot access local system properties and cannot create socket connect to another machine by default. Users should configure

```
grant codeBase "http:// IP_OF_VRES_SERVER
/vres/class"{
        permission java.security.AllPermission;
};
grant codeBase
"http://IP_OF_VRES_SERVER/vres/class/lib/jmf.jar"
{
        permission java.security.AllPermission;
};
```

그림 12. Java.policy 파일 내용
Fig. 12. The contents of java.policy file.

the security policy file to add the permission for the applets of VRES to access the information of local host. And add the permission for those applets to create new socket connections. Normally, that configuration file is `JRE_HOME/lib/security/java.security` and policy.url.*n*=http://URL_TO_VRES/java.policy have to be appended at the end of the file. The *n* of policy.url.n should be changed according the number of existing policy.url in `java.security` file. If it is the first time to modify this file ***n*** would be 3. The Uniform Resource Locator (URL) indicates a java.policy file exists in the VRES root directory.

The content of `java.policy` file in the server is showed in Figure 12. For exactly mean of this file, refer to the Java official webpage to see the directions of policy file. After the configuration, the Media· Service Applet, Robot Queue System Applet, and Instant Operation Applet can access the corresponding servers.

Figure 13 shows an example of user web browser

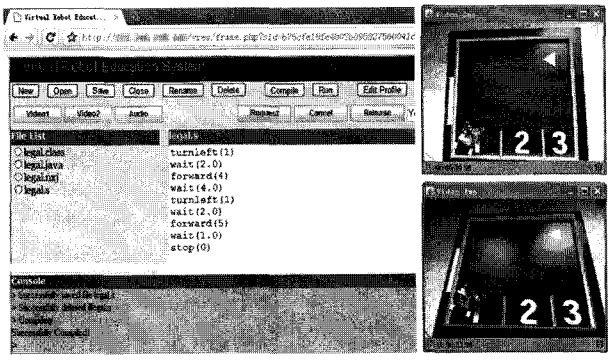

그림 13. 학습자와 교수자를 위한 GUI 실행
Fig. 13. GUI execution for a learner and a educator.

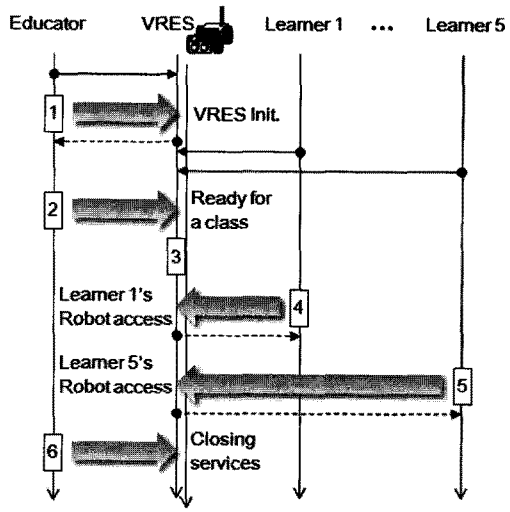

그림 14. VRES를 이용한 로봇 프로그래밍 실습
Fig. 14. A scenario for a robot programming class with VRES.

in which a source code in NXJS is created and file management is progressed as shown in the console area, and then a compiling the source code is completed. Two video modes are run so that the user observe the remote robot and its state. In menu bar, there are many buttons, specially Request, Cancel, and Release are related to access the robot. The Cancel is to give up getting access. According to the design requirements of previous section, this interface is implemented and tested.

In order to evaluate this system, we need to keep a procedure for convenient progress of a class. Figure 14 show a typical scenario for this goal. 1)An educator firstly should login to VRES and initialize the system such as diagnosing robot and all function modules of the system, making user account, and tuning the monitoring devices. When the server is ready learners can be allowed to login to the system and to use it. 2) For a class, the educator controls the robot using instant commander module and move it at a resonable place for the class. In addition, he determines the rule of robot access. 3)And then, the system and robot is enabled for this setup and controls. Through 4) and 5), learners access the robot and test their own robot programs and release the robot. Finally, educator close all services for VRES.

V. Conclusions and Discussion

In this paper, we introduce a web-based education system called VRES and assist learners with robot programming environment on-line. The VRES includes user and file management, web-based programming functions, and monitoring modules. For the simultaneous usage of the system, it use two robot access control modes and have the instant commander module which help to directly control the remote robot. Therefore students access this system without the restriction of space and time. Specially, we develop a user-friendly script programming language NXJS for users who are not familiar with programming. It facilitates the practice for students with the NXT robot.

In the near future, the VRES will be opened as an open platform and it will provide the robot programming tools and monitor service to every student. And we want that VRES provides teachers of different schools web-based programming services without limitation of access.

References

[1] Hatzilygeroudis, I., Giannoulis, C., Koutsojannis, C. "A Web-Based Education System for Predicate Logic," *Fourth IEEE International Conferenceon Advanced Learning Technologies*, 2004.

[2] Gonzalez, D. "Distributed Platform for Control of Robots at Distance," *6th World Multiconference on Systemics, Cybernetics and informatics*, 2002.

[3] Sagiroglu, S., Yilmaz, N., & Wani, M. A. "Web Robot Learning Powered by Bluetooth Communication System," *5th international Conference on Machine Learning and Applications*, 2006

[4] Cedazo, R., & Lopez, D. "Ciclope: FOSS for Developing and Managing Educational Web Laboratories," *IEEE Transactions on Education*, 50(4), 352 - 359. 2007

[5] "Lego Mindstorms NXT," Retrieved from http://en.wikipedia.org/wiki/Lego_Mindstorms_NXT, 2009.

[6] Hazel, S. "Codepad." Retrieved from

http://codepad.org, 2009
[7] Khen, G., Khen,D., & Koubi, G. 'Coderun," Retrieved from http://coderun.com, 2009
[8] Foord, M., "Trypython." Retrieved from http://www.trypython.org, 2009.
[9] Solorzano, J., & Andrews, Paul., "LeJOS Java for LEGO Mindstorms," Retrieved from http://lejos.sourceforge.net, 2009.
[10] Vermeir, D., "An Introduction to Compilers," Retrieved from http://star2.vub.ac.be/~dvermeir/courses/compilers/compilers.pdf, 2009
[11] Sun Microsystems, Inc., "Java Media Framework API Guide," Retrieved from http://java.sun.com/javase/technologies/desktop/media/jmf/2.1.1/guide/JMFPreface.html, 1999.
[12] Schulzrinne, H., Casner, S., Frederick, R., & Jacobson, V., "RTP: A Transport Protocol for Real-Time Applications," Retrieved from http://tools.ietf.org/html/rfc3550, 2003.

저 자 소 개

웅 홍 우(정회원)
2010년 8월 순천대학교 컴퓨터교육정보학 석사
2010년 현재 순천대학교 과학정보융합학과 박사과정
<주관심분야: 무선센서망 MAC, 라우팅 프로토콜, 컴퓨터 및 로봇 기반 교육>

소 원 호(정회원)-교신저자
대한전자공학회 논문지
제43권 TC편 1-14 참고