

# 이동 로봇 모듈의 RTC 미들웨어 개발

문용선\* · 배영철\*\* · 노상현\*\*\* · 조광훈\*\*\*\* · 박용구\*\*\*\*\*

## The Development of RTC Middleware of Mobile Robot Module

Yong-seon Moon\* · Young-chul Bae\*\* · Sang-hyun Roh\*\*\* · Kwang-hoon Cho\*\*\*\* · Yong-gu Park\*\*\*\*\*

### 요약

본 연구에서는 로봇 미들웨어 기술 중 국제표준으로 채택된 RTC를 이용하여 이동 로봇 시스템의 구성 요소에 대한 컴포넌트를 개발하였다. 또한 미들웨어 네트워크를 기반으로 이동 로봇 구성 요소를 서로 연결하여 하나의 시스템을 구현하였으며 실제 이동 로봇인 NRLAB 02를 이용하여 구현한 이동 로봇 시스템의 성능 시험을 수행하였다.

### ABSTRACT

In this paper, we developed component for constituent of mobile robot system using RTC which is adapted as an international standardization among the robot middleware technology. We implemented one system by connecting the f elements of mobile robot constituent to each other and performed performance test on mobile robot system using real mobile robot NRLAB 02 that were implemented.

### 키워드

OpenRTM, Middleware, Mobile Robot, Component, Modularize

## I. 서론

최근 로봇 기술의 급격한 발전에 따라 로봇에 요구되는 기능들이 늘어나고 있으며 이러한 증가하고 있는 요구를 충족하기 위해 다양한 기능의 하드웨어 구성 요소가 개발되고 있다. 따라서 현재는 로봇 시스템에 있어서 단순 구성의 하드웨어로는 로봇에 요구되는 기능을 만족하기 어렵게 되었다.

하나의 로봇 시스템은 다양한 기능을 수행하는 하드웨어의 조합으로 구성될 수 있으며 어플리케이션

에 따라 여러 가지 형태나 기능을 가진 로봇이 개발되고 있지만, 그 구조는 일반적으로 이동 요소, 센서 요소, 액추에이터 요소등 기본적인 요소 기기의 집합체이며 공통적인 요소 기기가 사용되고 있다. 그렇기 때문에 개개의 로봇 개발에 필요한 고유한 기능도 있지만 로봇 간에 서로 공유할 수 있는 요소들이 많이 있다. 이러한 공통적인 요소를 재사용하기 위해서는 구성 요소의 컴포넌트화 혹은 모듈화가 필요하며 각 컴포넌트나 모듈은 네트워크로 연결이 가능해야 한다.

\* 순천대학교 정보통신공학과(moon@sunchon.ac.kr)

\*\* 교신저자 : 전남대학교 전기전자통신컴퓨터공학부(ycbae@chonnam.ac.kr)

\*\*\* 순천대학교 전자공학과(rsh@urc.kr)

\*\*\*\* (주)메크로시스템 엔지니어링(choicemylife@hanmail.net)

\*\*\*\*\* 레드윈테크놀로지(주)(pyg@urc.kr)

접수일자 : 2010. 03. 03

심사완료일자 : 2010. 04. 02

최근에는 모듈화된 구성 요소를 기반으로 상위의 소프트웨어 레벨에서의 통합의 필요성이 증대되고 있다. 소프트웨어 레벨에서의 통합이란 사용하는 제어 컴퓨터의 운용환경, 제어언어 등에 관계없이 공용으로 적용할 수 있는 통합 소프트웨어 개발 환경을 구축하여 제어 소프트웨어 환경의 호환성, 재사용성을 극대화하고자 하려는 것이다. 이러한 요구조건에 따라 개발된 것이 미들웨어 기술을 이용한 컴포넌트 기반의 통합 소프트웨어 기술이라 할 수 있다.

미들웨어 통합 소프트웨어 기술이란 운영체제와 사용자 어플리케이션 사이에 공용으로 적용할 수 있는 미들웨어 레벨을 두고 이를 기반으로 하드웨어에 표준화된 컴포넌트 구조를 정의함으로써 실제 사용자 및 개발자는 시스템 운영환경 및 개발 언어에 상관없이 표준화된 컴포넌트를 수집 및 재구성함으로써 다양한 서비스를 구현할 수 있다. 이러한 로봇 미들웨어 소프트웨어 프레임워크로는 CORBA 기반의 OpenRTM, uPnP 기반의 iRSP, HTTP/DSSP 기반의 MSRDS 등을 들 수 있으며 그 외에도 OpenRDK, MIRO, OROCOS 등이 있다[1-5].

본 연구에서는 로봇 미들웨어 분야에서 국제표준화된 규격인 RTM의 RT-컴포넌트를 이용하여 이동 로봇 시스템의 구성 요소를 컴포넌트화하여 하나의 이동 로봇 시스템을 구현하고자 한다.

행과 관리를 위한 미들웨어 매니저, 컴포넌트 프로그램 코드의 골격을 생성하는 RT 템플레이터(template), 그리고 RTC의 실행과 연결을 위한 툴로 구성되어 있다. 컴포넌트 개발자는 컴포넌트 프레임워크 안에서 개발한 것과 구현하고자 하는 기능을 조합함으로써 RTC를 생성하게 된다. 그림 1에 이상의 과정을 나타내었다.

로봇 시스템에는 이렇게 생성되는 컴포넌트가 적어도 2개 이상의 컴포넌트를 결합하는 것에 의해 만들어지게 된다.[6]

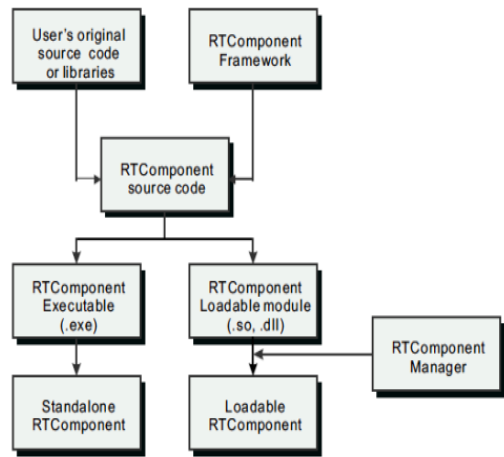


그림 1. RTC 개발 순서도  
Fig. 1 RTC development flow chart

## II. OpenRTM을 이용한 이동로봇 시스템 구조 설계

### 2.1 OpenRTM기반의 구성요소 컴포넌트화

RTC는 OpenRTM에서 컴포넌트에 대한 규격이며 RTC 표준은 지능형 서비스 로봇의 소프트웨어 분야에서는 세계 최초의 표준 규격이 되며, 이 규격 안은 2007년 12월 회의에서 OMG 표준안으로 채택 되었다.

RT 미들웨어는 CORBA와 같은 분산객체기술을 이용하여 로봇 시스템을 구성하는 각 소프트웨어 요소들을 컴포넌트화하고 이들을 조립해 하나의 로봇 시스템을 구성할 수 있도록 소프트웨어 기반을 제공하는 로봇용 미들웨어 기술이다.

OpenRTM은 RT 미들웨어의 실행과 관련되는데, 컴포넌트를 만들기 위한 프레임워크와 컴포넌트의 실행

### 2.2 이동로봇 플랫폼의 구성요소

본 연구에서 사용한 이동 로봇은 레드윈테크놀러지(주)에서 개발된 NRLAB 02 이동 로봇 플랫폼이다. NRLAB 02 이동 로봇 플랫폼은 실시간 모션제어 네트워크인 이더캐트(EtherCAT; Ethernet for Control Automation Technology)을 기반으로 개발된 자율주행 로봇으로 내부 구성 요소들이 이더캐트를 기반으로 모듈화되어 있는 네트워크 기반 제어 시스템 구조를 가진다. 그림 2는 NRLAB 02 이동 로봇 플랫폼의 구성 요소를 나타낸다.

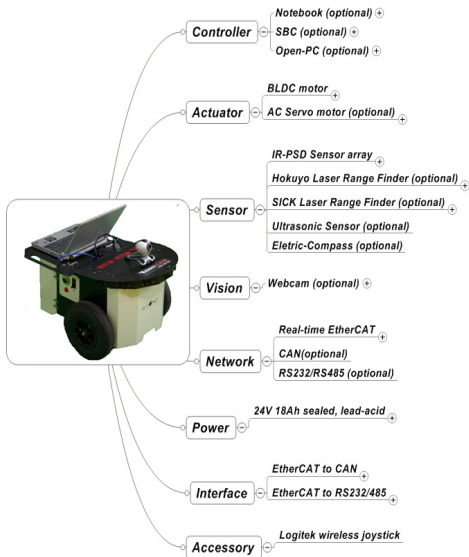


그림 2. NRLAB02 이동로봇 플랫폼 구성 요소  
Fig. 2 The element of constituent of NRLAB02 mobile robot platform

### 2.3 OpenRTM을 적용한 이동 로봇 시스템 구조 설계

본 연구에서는 통합 소프트웨어 환경을 제공하는 OpenRTM을 적용하여 이동 로봇 시스템의 컴포넌트 구조를 설계하였다. 그림 3에 구현할 OpenRTM 기반의 이동 로봇 시스템에 대한 RTC 구성의 개념적인 구조를 나타낸다.

이동 로봇 시스템의 주요 기능을 그룹화하여 관리할 수 있는 RTC를 설계하고 다시 각 세부 요소로 나누어 하나의 하드웨어에 하나의 RTC가 구성되도록 설계하였으며 화살표는 데이터의 흐름을 나타낸다. 각각의 RTC가 수행하는 세부적인 기능은 다음과 같다.

주제어기(main controller) RTC는 유저가 이동 로봇을 제어하기 위한 메인 컨트롤러로서 이동 명령, 영상 정보 수신 등의 전반적인 기능을 수행하며, 조이스틱 RTC는 조이스틱으로부터 데이터를 받는 기능을 수행한다. IMG 프로세서 RTC는 카메라 RTC로부터 받은 영상 데이터를 이미지 처리 기법을 통해 필요한 요소들을 추출하는 기능(에지 검출, 필터 기법)을 수행하며, 카메라 RTC는 USB 카메라로부터 영상 데이

터를 얻는 기능을 수행한다. 센서 RTC는 여러 센서로부터 데이터를 받아 일괄적으로 주제어기에 전달하는 기능을 수행하며, IR 센서 배열(sensor array) RTC는 7개의 IR 센서로부터 각각의 거리 값을 받아 센서 RTC에 전달하는 기능을 수행한다. 드라이브(drives) RTC는 주제어기에서 내려진 제어 명령에 따라 전·후진과 같이 이동 로봇을 구동하기 위해 모터 RTC에 필요한 명령을 생성하여 전달하는 기능을 수행하며, 모터1 RTC와 모터2 RTC는 각각 하나의 액추에이터를 구동하는 기능을 수행하도록 설계하였다.

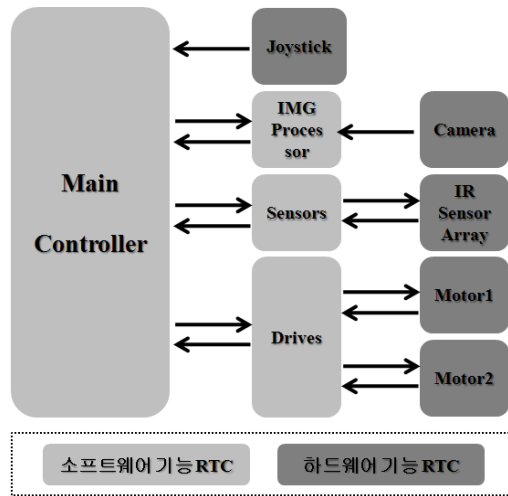


그림 3. 이동로봇 RTC의 개념적인 설계  
Fig. 3 Concept design of mobile robot RTC

## III. OpenRTM을 이용한 이동 로봇 시스템 구현

### 3.1 RTC Builder를 이용한 RTC 구현

OpenRTM을 이용한 이동 로봇 시스템 구성 요소의 RTC 개발은 OpenRTM 1.0 Release 버전을 이용하였다. 그림 4는 RTC Builder를 이용한 기본 코드의 생성을 나타내었고, 그림 5는 C++ 개발 환경으로 생성된 기본 코드의 프로젝트를 나타내었다.

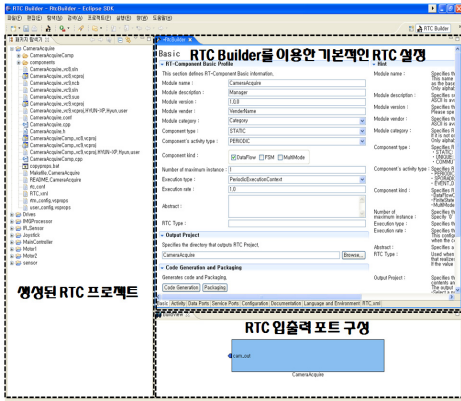


그림 4. RTC Builder를 이용한 RTC 기본 코드의 생성  
Fig. 4 The creation of RTC basic code using RTC Builder

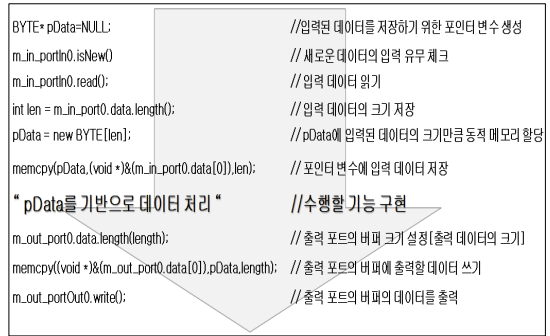


그림 6. RTC 기능 구현을 위한 기본적인 프로그램 작성 순서

Fig. 6 Basic program sequence for RTC function implementation

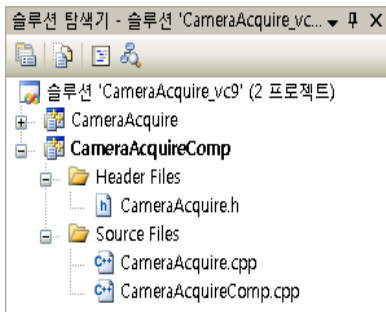


그림 5. 생성된 카메라 RTC의 기본 코드 프로젝트  
Fig. 5 Basic cord project of created camera RTC

이렇게 생성된 기본 코드의 프로젝트 파일의 구성은 헤더파일인 CameraAcquire.h과 수행할 코드를 작성할 CameraAcquire.cpp파일과 메인 함수가 있는 CameraAcquireComp.cpp파일이 생성된다. 이렇게 생성된 프로젝트에 수행하고자 하는 기능을 추가하여 RTC를 개발하게 되는데, 그림 6에 RTC가 입력된 데이터를 기반으로 데이터를 처리하여 출력으로 보내는 소스 코드에서 사용되는 함수와 흐름을 나타내었다.

실제 작성된 카메라 RTC의 소스 코드를 그림 7에 나타내었는데 카메라 RTC는 입력 포트는 없고 출력 포트만 있기 때문에 카메라로부터 영상 데이터를 획득하여 출력 포트를 통해 IMG 프로세서 RTC에 전달하는 기능을 수행한다.

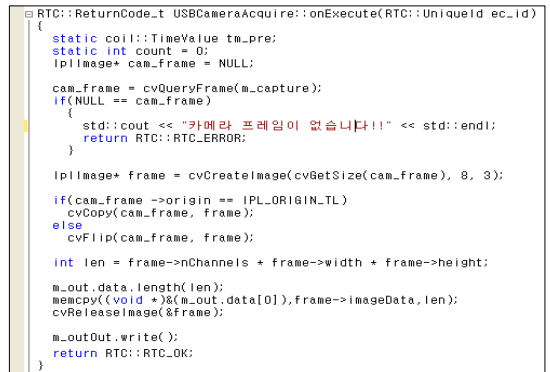


그림 7. 카메라 RTC가 수행할 메인 기능의 소스 코드  
Fig. 7 Camera RTC will perform source code of main function

주제어기 RTC는 사용의 편리성을 위해 MFC와 연동하여 GUI로 구성하였으며 카메라 영상 정보와 센서 데이터를 보는 부분과 조이스틱과 버튼을 이용한 주행 제어 부분이 있다. 주제어기 RTC의 실행 화면은 그림 8과 같다.

Motor1과 Motor2 RTC는 이동 로봇 시스템에서 이동 로봇의 성능 개선을 위해 액추에이터와 같은 하드웨어가 변경되더라도 재사용할 수 있도록 RTC에 환경 설정 변수를 추가하여 RTC의 재개발 없이 액추에이터의 물리 모델과 PID 계인을 변경할 수 있게 하였다.

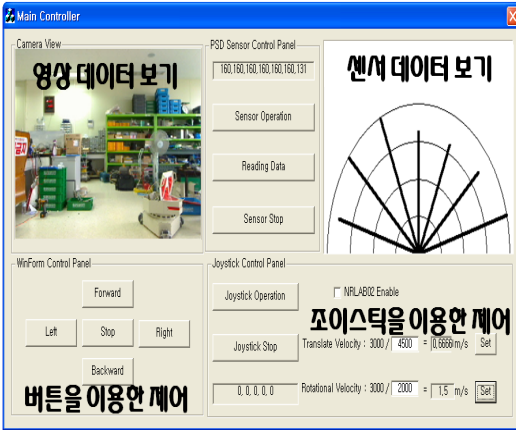


그림 8. 주제어기 RTC 실행 결과  
Fig. 8 The result of execution of main controller RTC

그림 9는 RTC 환경 설정 변수를 추가하기 위한 소스 코드이며, RTC 개발이 완료된 후에 환경 설정 변수의 값을 변경할 경우에는 그림 10에 나타나 있는 RT System Editor의 Configuration View 탭에서 변경할 수 있다.

```

RTC::ReturnCode_t Motor1::onInitialize()
{
    bindParameter("resistance", m_conf_R, "16.4");
    bindParameter("Back-EMF", m_conf_backemf, "3.36");
    bindParameter("Torque", m_conf_torque, "32.1");
    bindParameter("Rotor_inertia", m_conf_inertia, "3.8");
    bindParameter("P_gain", m_conf_p, "1024");
    bindParameter("I_gain", m_conf_i, "0");
    bindParameter("D_gain", m_conf_d, "0");
    return RTC::RTC_OK;
}
    
```

그림 9. RTC 환경 설정 변수 추가를 위한 소스 코드  
Fig. 9 Source code for added RTC environment setting variable

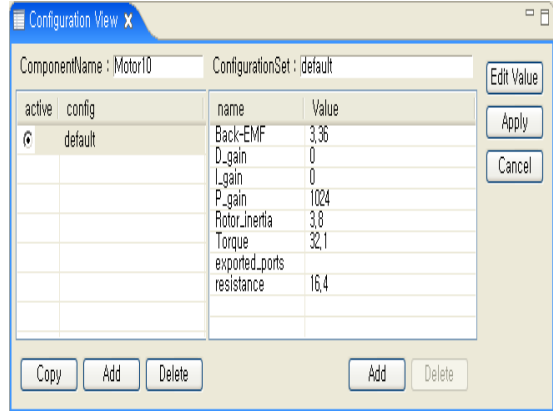


그림 10. RTC 환경 설정 변수를 변경할 수 있는 배열 뷰 탭

Fig. 10 RTC environment setting variable can be change configuration view tap

이동 로봇 시스템 구성 요소의 RTC 구현이 완료되었으며 구현된 RTC를 서로 연결하여 최종적으로 이동 로봇 시스템을 구현하게 된다. RTC의 연결은 CORBA 기반의 네임서버를 이용하여 구현된 RTC를 오픈한 네임서버에 접속시켜 네트워크 기반으로 서로 연결되게 된다.

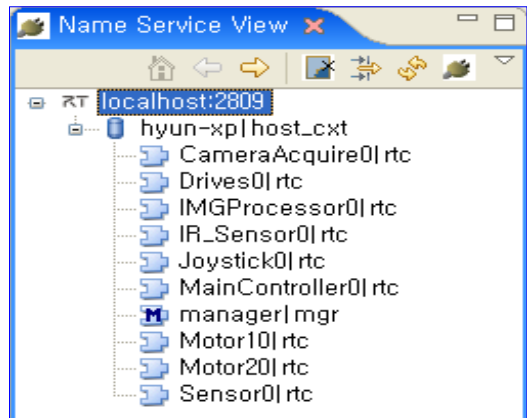


그림 11. 네임서버에 접속한 RTC의 트리 구조  
Fig. 11 Tree structure of RTC connected name server

네임서버에 구현된 RTC가 모두 정상적으로 접속이 되면 System Diagram 창을 이용해 RTC를 서로 연결하여 이동 로봇 시스템을 구현하게 되는데, 그림 10

에 이동 로봇 시스템 구현을 위한 RTC 연결 구성을 나타내었다.

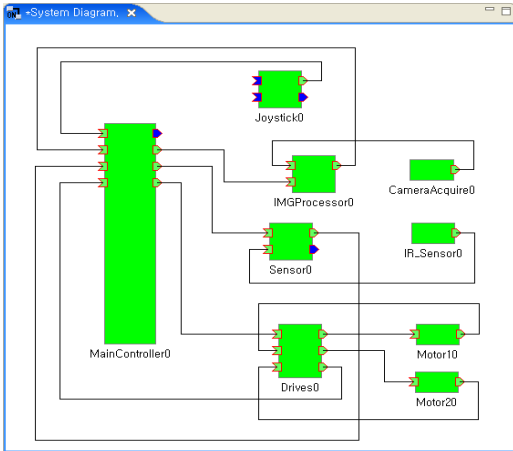


그림 12. 이동 로봇 시스템 구현을 위한 RT 컴포넌트 연결

Fig. 12 RT component connection for system implementation of mobile robot

#### IV. 결론

본 연구에서는 로봇 미들웨어 기술 중 국제표준으로 채택된 OpenRTM을 이용하여 이동 로봇 시스템의 구성 요소를 RTC로 만들고 미들웨어 네트워크를 기반으로 이동 로봇 구성 요소의 RTC를 서로 연결하여 하나의 이동 로봇 시스템을 구현하였으며 실제 이동 로봇인 NRLAB 02를 이용하여 구현한 이동 로봇 시스템의 성능을 시험하였다.

시험 결과 RT 컴포넌트로 구현된 이동 로봇 시스템은 잘 구현되었으며, NRLAN 02 이동 로봇을 이용한 동작 테스트 역시 만족할 만한 수준이었다. 그러나 이동 로봇의 기능을 동작시키는 정도에 머물러 있어 시험 결과에 대한 정량적인 수치는 얻지 못하였다.

앞으로 정량적 목표를 세워 테스트 결과에 대해 신뢰성을 부여하고, 이동 로봇에 필요한 다양한 기능(다양한 센서의 사용, 주행 알고리즘, 로봇 위치 인식 등)을 수행할 수 있는 RTC를 개발하여 진보된 이동 로봇 시스템을 구현하고자 한다.

#### 감사의 글

The research was supported by the "GRRC" Project of Gyeonggi Provincial Government, Republic of Korea.

#### 참고 문헌

- [1] Donald E. Knuth. The Art of Computer Programming. Four volumes. Addison-Wesley, 1973.
- [2] Slovakian-Hungarian Joint Symposium on Applied Machine Intelligence and Informatics, "Virtual Master Device", 2007.
- [3] Daniele Calisi, OpenRDK : a modular framework for robotic software development, 2008.
- [4] Herman Bruyninckx, "Open robot control software: the OROCOS project". In Proceedings of Int. Conf. of Robotics and Automation (ICRA'01), 2001.
- [5] S. Sablatnog, S. Enderle, and G. Kraetzschmar, "Miro middleware for mobile robot applications", IEEE Transaction on Robotics and Automation, 2002.
- [6] OPENRTM-AIST Document "<http://www.openrtm.org/>"
- [7] OMG "Robotic Technology Component Specification Version 1.0", 2008.
- [8] REDONE TECHNOLOGIES "NRLAB02 User Guide", 2009.

#### 저자 소개



#### 문용선(Yong-seon Moon)

1983년 2월 조선대학교 전자공학과 졸업(공학사)

1989년 2월 조선대학교 대학원 전자공학과 졸업(공학박사)

1992년 - 현재 : 순천대학교 정보통신공학부 교수

현재 : 레드윈테크놀로지(주) 기술이사

※ 관심분야 : 산업통신망 및 로봇, 실시간 모션 제어



**배영철(Young-chul Bae)**

1984년 2월 광운대학교 전기공학과 (공학사)

1986년 2월 광운대학교대학원 전기공학과 (공학석사)

1997년 2월 광운대학교대학원 전기공학과(공학박사)

1986년 ~ 1991년 : 한국전력공사

1991년 ~ 1997년 : 산업기술정보원 책임연구원

1997년 ~ 2006년 : 여수대학교 전자통신전기공학부 부교수

2006년 ~ 현재 : 전남대학교 전기전자통신컴퓨터공학부 교수

※ 관심분야 : Chaos Control and Chaos Robot, Robot control 등



**박용구(Yong-gu Park)**

2009년 2월 순천대학교 자동차공학과 (공학사)

2009년 2월 ~ 현재 : 레드윈테크놀러지(주) 연구원

※ 관심분야 : 로봇 설계, 모터 설계



**노상현(Sang-hyun Roh)**

2007년 2월 순천대학교 전자공학과 (공학사)

2009년 2월 순천대학교 전자공학과(공학석사)

2009년 ~ 현재 : 순천대학교 전자공학과 (공학박사 재학 중)

※ 관심분야 : 로봇 제어, 모터 제어, 산업통신망



**조광훈(Kwang-hoon Cho)**

2003년 2월 순천대학교 전자공학과 졸업 (공학사)

2003년 2월 (주)메트로 시스템 엔지니어링 입사

※ 관심분야 : 자동화 시스템, 산업통신망