

퍼스널 로봇을 위한 통합 개발 환경(RDC)의 개발

김주민*, 김홍열*, 김대원*, 양광웅**, 김홍석**, 이호길**
 명지대학교 정보제어공학과*, 한국생산기술연구원**

The Integrated Development Tool(RDC:Robot Design Center) for Personal Robot

Joomin Kim*, Hongryeol Kim*, Daewon Kim*, Kwangwoong Yang**, HongSeok Kim**, Hogil Lee**
 Department of Information Control Engineering Myongji University*, KITECH**

Abstract - In this paper, we propose the RDC(Robot Design Center) for a network-based personal robot system. The RDC is developed under the new software framework for personal robots. The configuration for each modules in robot and functions of the RDC are described. The concept of standardized and modular program is presented and finally, developing procedures are proposed.

1. 서 론

기존 산업용 로봇 시장의 포화에 따른 새로운 로봇 시장의 필요성[1]과 IT(information technology)기술의 발전에 맞물려 로봇산업의 새로운 패러다임(paradigm)으로 퍼스널 로봇(personal robot) 분야가 각광받고 있다. 퍼스널 로봇 분야에 대한 많은 국·내외 연구가 이루어지고 있고 시제품도 발표되고 있다. 국내 연구에서 새로운 퍼스널 로봇 프레임워크[2]를 제안한바 있으며, 제안된 프레임워크를 기반으로 국가과제가 수행되고 있다[3].

제안된 새로운 퍼스널 로봇 소프트웨어 프레임워크는 독립적으로 운영 가능한 형태의 하드웨어적인 모듈(module)을 기반으로 네트워크를 통해 연결되어 하나의 퍼스널 로봇을 구성하게 된다. 모듈 프로그램은 행위(behavior) 단위로 구현하며 모듈의 독립성과 호환성을 위해 모듈은 RT-Linux 상에 통신 미들웨어와 RVM(Robot Virtual Machine)이 미들웨어로 동작한다. 모듈의 행위들은 RVM상에서 동작할 수 있도록 RPL(Robot Programming Language)로 구현되어지며 바이트 코드화 되어 모듈에 적재 된다[4][5][6].

본 논문에서는 앞서 제안된 새로운 퍼스널 로봇 소프트웨어 프레임워크를 위한 개발 환경(RDC: Robot Design Center, 이하 'RDC'라 칭함)의 구성 및 기능에 대해 설명하고, RDC를 이용한 개발 및 운영 절차를 제안하고자 한다. 2.1에서는 RDC의 시스템 구성, 그래픽/텍스트 에디터, 컴파일러의 소프트웨어 컴포넌트(software component)에 대한 기능과 개발 단계별 화면에 대하여 기술한다. 2.2에서는 표준화된 모듈프로그램의 개발을 위한 RDC의 역할에 대하여 기술한다. 2.3에서는 RDC를 이용한 개발절차와 운영방법을 제안하며 마지막으로 결론 및 향후계획에 대하여 기술한다.

2. 본 론

2.1 RDC의 구성 및 기능

RDC는 시스템 구성, 그래픽/텍스트 에디터, 컴파일러의 기능적인 소프트웨어 컴포넌트로 구성되어 있다. 그림1은 RDC의 실행화면으로, RDC는 작업단계별로 구현된 모듈 정보수집, 모듈 행위정보 관리, 행위 에디터, 모듈 정보 등록으로 명명된 창을 통해 퍼스널 로봇 개발을 위한 사용자 환경을 지원한다.

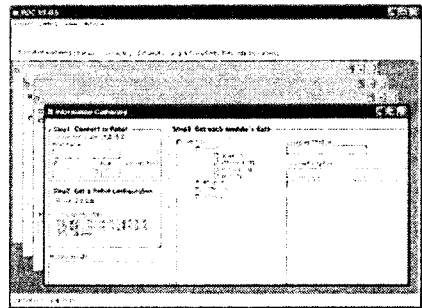


그림 1. RDC 실행화면

모듈 정보수집 창에서는 무선 랜(wireless LAN) 환경과 서버/클라이언트(server/client) 통신방법을 사용하여 퍼스널 로봇(서버)에 접속하고 모듈 정보를 수집한다. 수집되는 모듈정보로는 모듈 간 네트워크 구성정보, 모듈 내 행위 정보, 행위 간 연결 정보, 모듈 내 RVM 상태들이며 서버 기능을 담당하는 모듈은 무선 랜이 장착된 모듈이 되며, 외부 네트워크(robot-RDC 통신)와 내부 네트워크(퍼스널 로봇 내부의 모듈 통신) 통신은 통신 미들웨어가 처리한다.

모듈 행위정보 관리 창에서는 수집된 모듈 정보를 프로젝트 단위로 관리하고 모듈정보를 변경할 수 있다.

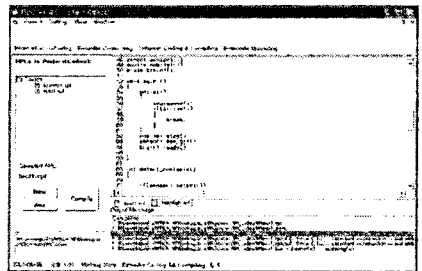


그림 2. 텍스트 에디터 및 컴파일 화면

그림2는 행위 에디터 창의 텍스트 에디터와 컴파일 과정에 대한 화면이다. 행위 에디터 창에서는 모듈 행위를 프로그래밍 하는 작업을 지원하며 행위 에디터는 그래픽 에디터와 텍스트 에디터로 구성된다. 그래픽 에디터는 모듈 간 구성 정보 파일의 편집·관리 및 모듈 프로그램을 심볼릭(symbolic) 프로그래밍 형태로 지원하여 기본 코드를 생성하고 텍스트 에디터를 통해 생성된 기본 코드를 수정·확장한다. 텍스트 에디터에서 모듈 프로그램 개발에 사용하는 언어는 KR-C의 서브셋(subset)으로 구성된 RPL을 사용한다. RPL로 작성된

모듈 프로그램은 컴파일 과정을 거쳐 모듈의 RVM 상에서 실행 가능한 바이트코드로 생성된다.

모듈 정보 등록 창에서는 변경된 모듈 정보 및 바이트코드가 된 모듈 프로그램들을 해당 모듈로 등록(upload)하며 등록정보를 관리한다.

2.2 표준화된 모듈 프로그램 개발

그림3은 제안된 퍼스널 로봇시스템 구성 예이다. 각 모듈은 네트워크를 통해 연결되고 통신 미들웨어에 의해 관리되어진다. 통신 미들웨어는 모듈 프로그램 내에서 사용되는 모듈 간 메시지/함수호출/이벤트의 네트워크 전송을 처리하며, 서버/클라이언트 방식과 퍼블리셔/서브스크라이버(publisher/subscriber) 방식[7]을 지원한다.

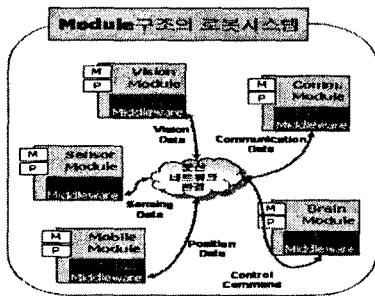


그림 3. 제안된 퍼스널 로봇시스템 구성 예

그림4는 통신 미들웨어의 구조이다. 통신 미들웨어는 다양한 통신 방식을 지원하도록 설계되어 있다. 이것은 RVM과 연동되어 통신 및 OS에 독립적인 모듈 표준화·독립화를 구현할 수 있는 RDC의 중요 요소기술이다.

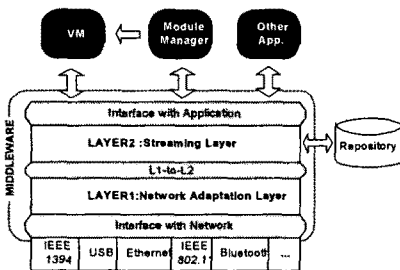


그림 4.통신 미들웨어의 구조

통신 미들웨어는 퍼스널 로봇에서 사용하는 모든 네트워크 관련 기능을 수행하는 에이전트(agent)로서의 역할을 수행하고 RVM은 모듈의 하드웨어와 소프트웨어를 분리하는 역할을 수행하게 된다. 이 두 가지 요소기술을 통해 모듈 프로그램을 표준화할 수 있다.

그림5는 표준화된 모듈 프로그램 개발의 예이다. Brain, Vision, Sensor, Mobile의 4가지 모듈로 구성된 퍼스널 로봇에서 Brain 모듈에 구현된 프로그램의 각 코드 라인에 대응되는 모듈 함수의 관계를 도식화한 것이다. 실선으로 표시된 부분은 네트워크 통신에 의해 기능 수행이 이루어지는 것을 의미한다.

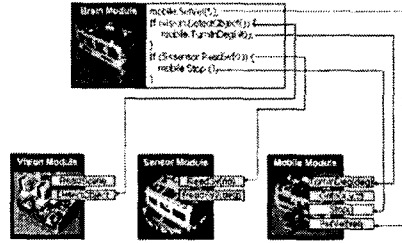


그림 5. 표준화된 모듈 프로그램 개발 예

2.3 개발환경을 이용한 개발 및 운영절차

RDC를 이용한 모듈 소프트웨어 개발은 크게 플랫폼 개발, 표준화된 모듈 개발, 모듈 통합 설치의 3가지 단계로 이루어진다.

플랫폼 개발단계에서는 전문화된 모듈 제조 업체의 고유한 기술력에 의해 모듈에 내장된 기능에 대한 인터페이스 및 기능 검증을 한다. 내장되는 기능들은 모듈 API로써 모듈에 롬(ROM)에 이식 되어진다. 표준화된 모듈 개발 단계는 상호 운영성과 소프트웨어 통합을 보장하는 모듈을 개발하는 것으로서 제조하는 업체에 의해 모듈 API를 기반으로 하는 모듈 행위 또는 모듈에 의존적이지 않는 일반적인 형태의 행위를 개발한다. 개발된 행위들은 RVM 상에서 동작되며 RPL로 작성한다. 모듈 통합 설치 단계에서는 퍼스널 로봇시스템을 이루고 있는 모듈의 구성 정보를 작성 및 관리한다. 모듈에서 지원하는 API 및 행위들의 조합을 통해 퍼스널 로봇의 동작 및 기능을 구현하고 모듈에 등록하는 기능을 수행한다.

그림6은 RDC 개발모형을 도식화 한 것이다. 퍼스널 로봇의 각 모듈은 제공되는 내장 기능에 따라 API들로만 구성될 수도 있고, 시나리오 레벨의 행위 까지 내장될 수도 있다. 모든 행위들은 RVM상에서 운영되며 API들은 RVM에서 사용되어질 수 있도록 제작된다.

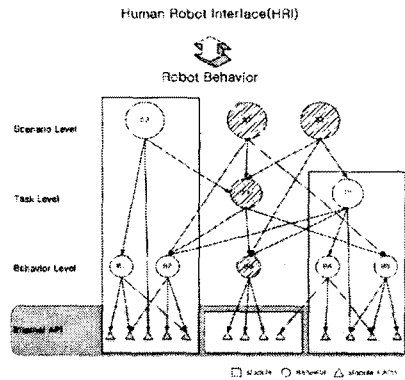


그림 6. RDC 개발모형

3. 결 론

새로운 퍼스널 로봇 소프트웨어 프레임워크를 위한 개발 환경인 RDC는 시스템 구성, 그래픽/텍스트 에디터, 컴파일러의 기능적인 소프트웨어 컴포넌트로 구성되어 있다. 각 컴포넌트들은 작업단계별 창을 통해 구현되어져 있고 플랫폼 개발, 표준화된 모듈 개발, 모듈 통합 설치의 3가지 개발 절차들로 활용되어진다.

RDC를 통해 개발되는 모듈 프로그램은 네트워크로 연결된 모듈을 행위기반으로 동작될 수 있도록 RPL로 작성된다.

RDC를 기반으로 플랫폼 개발 단계에서는 모듈의 호환성 및 기본 기능 테스트를 수행하고, 표준화된 모듈 개발 단계에서는 모듈 프로그램의 개발 및 테스트를 한다. 마지막으로 모듈 통합 설치단계에서는 모듈간 정보를 구성하고 행위를 조합·확장한다.

현재 RDC는 시험 버전을 기반으로 퍼스널 로봇의 프로토타입(prototype) 개발에 사용되고 있다. 제안된 네트워크로 분산 모듈화 된 퍼스널 로봇의 구조는 사용된 네트워크의 성능, 모듈 하드웨어 성능, 그리고 모듈프로그램의 작성 방법에 따라 전체적인 퍼스널 로봇의 성능이 좌우될 수 있기 때문에 RDC의 추가 기능으로서 모듈 프로그램 동작 평가를 위한 방법과 기능구현이 필요하다.

[참 고 문 헌]

- [1] "차세대 로봇 기반 기술 개발 - 부제: 차세대신기술개발사업을 위한 산업분석 보고서", 산업자원부, 2001
- [2] Hongryeol Kim, Daewon Kim, HongSeok Kim, and Hogil Lee, "Toward the Personal Robot Software Framework", *ICCAS Conf.*, pp. 2307-2312, 2002
- [3] 차세대신기술개발사업 퍼스널 로봇 기반기술개발. available on line: <http://www.personalrobot.org>
- [4] 김홍석, "퍼스널 로봇을 위한 시스템엔지니어링기술 현황", 제2회 퍼스널로봇 기반기술개발 Workshop, pp. 159-174, 2003
- [5] 한국생산기술연구원, 성균관대학교, 인하대학교, 명지대학교, "표준형 로봇디자인센터(Robot Design Center)의 프레임워크 개발", 제2회 퍼스널로봇 기반기술개발 Workshop, pp. 175-193, 2003
- [6] Rodney A. Brooks, "A Robust Layered Control System for a Mobile Robot", *IEEE Journal of Robotics and Automation*, Vol. RA2, No. 1, pp.14-23, 1986.
- [7] Ragnathan Rajkumar, Mike Gagliardi, and Luisa, "The Real Time Publisher/Subscriber Inter Process Communication Model for Distributed Real Time System: Design and Implementation", *First IEEE Real-Time Technology and Applications Symposium*, 1995.