# Lego Mindstorms NXT 상에서 SimMusic Language 구현

신수용, 허유정, 김현수
이화여자대학교 컴퓨터공학과
e-mail : suyongshin92@gmail.com,
yooooouj@gmail.com,
hyunsue77@gmail.com

# Implementation of SimMusic Language on Lego Mindstorms NXT

Suyong Christina Shin, Yujeong Heo, Hyunsoo Kim
Dept. of Computer Science and Engineering, Ewha Womans University

**요 약**

본 연구는 Lego Mindstorms NXT 상에서 음악을 재생할 수 있도록 하는 SimMusic language 를 정의한다. 재생하고자 하는 악보는 SimMusic language 로 작성되고, Lego Mindstorms NXT 로 조립된 SimMusic player 는 SimMusic 프로그램을 읽고 음악을 재생한다. SimMusic player 를 통해 하나의 악보가 재생되는 일련의 과정은 컴퓨터 구조에서 프로그램이 수행되는 과정을 기반으로 구현되었기 때문에, 본 연구는 비전공자도 쉽게 컴퓨터 과학의 기초를 다질 수 있는 계기가 된다.

## 1. 서론

본 연구는 악보를 재생하는데 필요한 명령어들을 포함하는 SimMusic language 를 정의하고, 이 언어로 작성된 프로그램을 SimMusic player 가 이해할 수 있는 색깔 테이프로 변환해주는 SimMusic assembler 를 구현한다. 또 색깔 테이프를 읽고 음악을 재생하는 SimMusic player 를 Lego Mindstorms NXT 를 활용하여 구현한다.

이어서 본 논문은 연구를 하는데 필요한 배경 지식을 소개하고, 설계 내용과 이를 바탕으로 한 구현 내용을 설명한다. 마지막으로 본 연구의 수행 결과와 기여 내용을 설명하고 한계점을 제안하며 결론을 낸다.

## 2. 배경 지식

본 연구는 Lego Mindstorms NXT 와 NXC 를 활용한다. Lego Mindstorms NXT (이하 NXT)는 LEGO 사에서 출시한 제품이다. NXT 는 LEGO 블록 뿐 아니라 모터나 다양한 센서들을 포함하고 있어 사용자가 원하는 형태로 자유롭게 조립될 수 있을 뿐 아니라, 사용자가 원하는 기능을 수행할 수 있도록 프로그래밍 가능하다. NXT 를 제어하는데 다양한 언어들이 쓰이지만 본 연구는 NXC 를 활용한다[1]. NXC 는 Not Exactly C 의 약자로 C 와 유사한 프로그램 구조를 갖고 있으며 NXT 를 제어하는데 필요한 다양한 함수와 자료 구조를 포함하고 있다.

본 연구와 유사한 연구로 Krzysztof Kapusta 가 구현한 LEGO Mindstorms Tape reader / player (그림 1 참조) (이하 Tape reader)가 있다[2]. Krzysztof Kapusta 는 출력하고자 하는 문자들을 흑백 테이프로 변환하고, NXT 로 조립된 Tape reader 가 변환된 흑백 테이프를 읽고, NXT 화면에 대응하는 문자를 출력한다. Tape reader 의 경우에는 따로 언어를 정의하지 않고 아스키코드를 활용하여 문자를 흑백 테이프로 변환하는 매우 단순한 시스템이다.

(그림 1) Krzysztof Kapusta 의 Tape reader / player

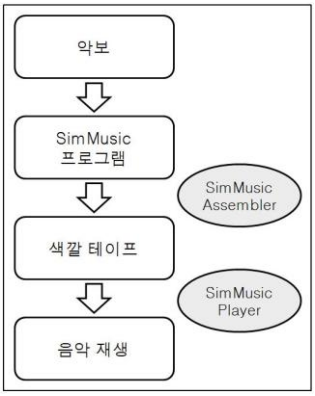
## 3. 설계

본 연구는 아래 (그림 2)와 같은 일련의 과정을 거쳐 음악이 재생되도록 한다. 재생하고자 하는 악보를 SimMusic language 를 이용하여 SimMusic 프로그램을 작성한다. 작성된 프로그램을 입력으로 SimMusic assembler 가 대응하는 색깔 테이프를 출력한다. 마지막으로 SimMusic player 가 색깔 테이프를 읽고 명령어들을 수행하며 음악을 재생한다(그림 2 참조).

### 3.1 SimMusic language

SimMusic language 는 어셈블리 수준의 언어로 악보를 재생하는데 쓰이는 5 개의 명령어들을 포함한다.

SimMusic 어셈블리 언어 정의에 ARM 어셈블리 언어의 공부가 많은 도움이 되었다[3].




(그림 2) SimMusic 을 이용한 악보 재생 과정

명령어 PN 은 악보의 음표 하나를 재생한다. 명령어 JM 은 프로그램 카운터를 명시된 인덱스로 이동한다. 단, 이동할 인덱스는 현재 프로그램 카운터 보다 뒤에 있어야 한다. 명령어 RM 은 특정 구간을 특정 횟수 반복한다. 단, 반복할 구간에 대한 offset 은 현재 프로그램 카운터 보다 뒤에 해당하며, 중첩 반복 문은 허용되지 않는다. 명령어 PM 은 음악을 명시된 시간만큼 잠시 중단 한다. 명령어 SM 은 음악 재생을 중단한다(표 1 참조).

| Mnemonic | Instruction |
|---|---|
| PN | Play note |
| JM | Jump |
| RM | Repeat music |
| PM | Pause music |
| SM | Stop music |

(표 1) SimMusic 명령어요약

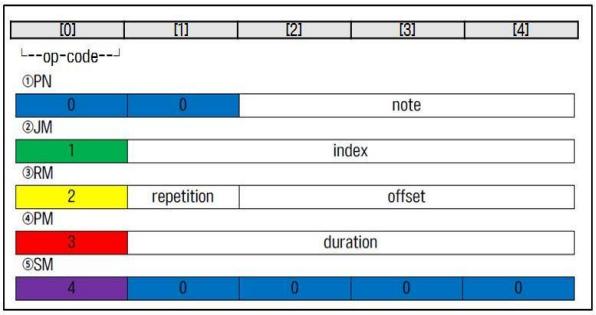
### 3.2 SimMusic assembler

SimMusic assembler 는 SimMusic language 로 작성된 프로그램을 SimMusic player 가 이해할 수 있는 5 진수 및 대응하는 5 개의 색깔을 포함하는 테이프로 변환한다. SimMusic language 의 각 명령어는 5 진수-5 digit 길이로 정의된다. 이 때, 가장 첫 번째 digit 은 op-code 를 명시하며 나머지 digit 들은 operation 에 따라 의미하는 바가 다르다(그림 3 참조).

본 연구는 5 진수를 5 가지의 색깔과 짝 지어 정의한다. 5 진수의 0 은 파란색, 1 은 초록색, 2 는 노란색, 3 은 빨간색, 마지막으로 4 는 보라색으로 대응된다.

### 3.3 SimMusic player

SimMusic player 는 SimMusic assembler 가 출력한 색깔 테이프를 읽고 명령어들을 수행한다. 색깔 센서를 통해 SimMusic player 는 테이프의 색깔을 입력 받고, 각 색깔에 대응되는 5 진수를 2 차원 배열에 초기화한다. 배열의 초기화를 마치면, 배열에 초기화 된 명령들을 한 줄씩 읽고 op-code 에 대응되는 명령어를 수행한다.




(그림 3) SimMusic 명령어 형식 요약

## 4. 구현

앞서 설명한 설계를 기반으로 SimMusic assembler 는 JAVA 로 작성되었고 JVM 이 설치 되어있는 PC 환경에서 실행 가능하고, SimMusic player 는 NXC 로 작성되었고 NXT 에서 실행 가능 하다. SimMusic player 의 외형은 Lego Mindstorms NXT 8527 과 LEGO MINDSTORMS education 9797 을 활용하였고, 색깔 센서는 HiTechnic 사의 NXT Color Sensor V2 를 이용하였다.

실제로 "Let It Go" 라는 곡을 대상으로 테스트를 진행하면, 자연광이 충분한 환경에서 SimMusic player 는 테이프를 읽고 음악을 올바르게 재생됨을 본 연구에서 확인했다.

## 5. 결론

본 연구는 컴퓨터 구조에서 프로그램이 수행되는 과정을 기반으로 NXT 상에서 악보가 재생되는 과정을 구현하기 위해, SimMusic language 를 정의하고 SimMusic assembler 와 SimMusic player 를 작성하고 SimMusic player 의 외형을 조립한다. 결과적으로 본 연구를 계기로 악보를 새로 정의된 언어로 표현 가능해졌을 뿐 아니라, 본 시스템의 구현 과정을 이해함으로써 비전공자도 컴퓨터 과학의 기초를 이해할 수 있게 되었다. 하지만 본 연구에서 정의한 언어는 매우 간단한 명령어들만 포함하고 있기 때문에, 복잡한 악보를 언어로 표현하는 건 사실상 불가능하다. 또 NXT 의 센서 민감성과 부 정확성으로 인해 색깔 테이프를 읽어 들이는 과정에서 오류가 발생할 위험이 크다는 한계점이 있다.

본 연구는 미래창조과학부 및 정보통신기술진흥센터(IITP)의 서울어코드활성화지원사업 (IITP-2016-R06131610040001002)의 연구결과로 수행되었음.

## 참고문헌

[1] 홍선학, 『NXC 로 즐기는 Lego Mindstorms NXT』, 이지테크(2007)

[2]http://legorobot.pl/project.php?pr=tapereader (2016.07.01)

[3] Stephen Bo Furber, 『ARM System -on-chip Architecture 』Pearson Education,(2010)