

# Design and Development of Robot Command Card for Coding Learning

Sun-Gwan Han\*

## Abstract

In this paper, we propose a design and development of instructional cards to understand the grammar of coding, solving the problems and extending the computational thinking in the robot-driven environment. First, we designed the input/output module of the robot to process the coding grammar through the function analysis of the robot. And we designed the module of command card to learn coding grammar using color sensors. We have proven the validity of the designed instruction card by examining the experts to see if it is suitable for coding grammar learning. Designed robot and command card were developed with 28 cards and sensor robot. After applying the developed robot and command card to the elementary school students, the questionnaire showed that students grow the understanding and confidence of coding. In addition, students showed an increased need for programming learning.

▶ Keyword: Software Education, Educational Robot, Coding Education, Command Card

## I. Introduction

2018년 소프트웨어 교육이 중학교에 정규교과로 전환되고 2019년부터 초등학교에서 수업에 적용되어 정보교과가 활성화되면서 다양한 교육적 접근이 시도되고 있다. 1990년대에 Basic, Pascal과 같은 스크립트 언어기반의 프로그래밍 교육을 시작으로 이후 Visual Basic과 Delphi등의 GUI기반 비주얼 프로그래밍 언어 그리고 2003년 Scratch 언어가 발표되면서 블록형 명령 프로그래밍이 소프트웨어 교육의 주요한 코딩도구가 되었다[10]. 이후 컴퓨터 없이 컴퓨터 과학을 배우는 언플러그드 컴퓨팅 교육과 컴퓨팅 기기와 연동되어 물리적인 형태의 교육이 진행되는 피지컬 컴퓨팅 교육 방법들도 등장하였다[16].

이와 함께 교육용 로봇은 피지컬 컴퓨팅 교육의 한 부류로 발전하며 공학적 설계와 조립의 틀에서 벗어나 소프트웨어 교육에서 중요시하는 알고리즘과 코딩을 학습하기 위한 기능을 갖추고 다양한 형태로 개발되고 있다[1,2,13,14].

그러나 교육용 로봇이 가진 하드웨어적 제약과 제한적인 로직의 제어 그리고 비용적인 부담에 의해 학습자들에게 적절한 알고리즘 사고와 창의적 문제해결의 경험을 제공하는데 한계를 보였다. 특히 소프트웨어교육에서 핵심으로 다루고 있는 코딩

과 프로그래밍의 문법적인 학습을 지원하는데 많은 제약이 따라 정규 교육에서 활용하기에는 부적합하였다.

따라서 본 연구에서는 한 대의 교육용 로봇으로 코딩에서 다루는 대부분의 문법을 이해하고 알고리즘에 대해 학습하는 명령카드를 설계하고 구현하고자 한다. 또한 학습자들에게 직접 적용하여 코딩의 문법적인 학습이 현장 교육에서 활용할 수 있는지에 대한 가능성을 제공하고자 한다.

## II. Backgrounds

### 1. Robot and Education

교육용 로봇과 관련된 선행연구를 살펴보면 교육용 로봇의 개발, 교육용 로봇의 교육적 효과, 교육용 로봇의 유형 분석, 교육용 로봇의 교육적 전략 등에 관련된 내용들이 주를 이루고 있다.

교육용 로봇은 영어회화나 음악 지도 등의 학습처럼 교사나 사용자의 교육을 지원하는 교육보조로봇과 학습자의 문제해결

• First Author: Sun-Gwan Han, Corresponding Author: Sun-Gwan Hang  
\*Sun-Gwan Han (han@gin.ac.kr), Dept. of Computer Education, Gyeongin National University of Education  
• Received: 2017. 11. 01, Revised: 2017. 12. 05, Accepted: 2018. 01. 10.

력과 창의성을 신장시키기 위해 로봇을 직접 제작하거나 기능을 제어하는 교구재로봇으로 구분한다[2]. 교육용 로봇을 세부적으로 분류하는 방법을 서순식 외의 연구에서 살펴보면 다음과 같다[12].

Table 1. Classification of Educational Robots[12]

| Type       | Characteristic                                                                                                                                                                |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Completion | Most of the robot structure is preconfigured. Especially when the operation method can not be changed                                                                         |
| Expandable | It consists of completed module and small number of parts, and if elements such as operating method are packaged                                                              |
| Assembled  | If the user can assemble with components and decide how to operate                                                                                                            |
| Open       | It is provided in the form of a board, and other elements may be used in connection with other products, such as an open structure (for example, Arduino, sensor board, etc.) |
| Etc        | Other than the above-mentioned type                                                                                                                                           |

로봇의 활용 유연성 측면에서 보면 개방형이 가장 확장성이 뛰어나며 수업의 시간과 제작의 부담 측면에서는 완성형이 가장 쉽게 접근할 수 있다고 보았다.

2. Robot in Software Education

교육용 로봇이 교실에서 활용된 것은 매우 오래 전부터이다. 로봇의 공학적 특성을 바탕으로 기계의 원리와 조립 그리고 기능의 교과적 원리와 개념을 융합하는 STEAM교육으로까지 전개 되면서 다양하게 발전하여 왔다. 소프트웨어교육의 등장으로 단순히 완성된 로봇의 제어나 조립의 공학적 설계를 벗어나 컴퓨팅 사고의 신장으로 소프트웨어적인 부분과의 연계에 많은 개발자들과 교육자들이 집중하고 있다[2,3,4,11,12].

이은경은 교육용 로봇을 활용하여 컴퓨팅 사고를 신장시키기 위한 교수학습모형을 제안하였다. 교육용 로봇을 이용하여 과제의 탐색, 문제해결의 지원,과제의 해결, 지식의 통합의 4단계로 활동하도록 학습의 단계를 구성하였다[2]. W. Allan은 로봇을 활용한 교육적 활동에서 컴퓨팅 사고를 신장시키기 위한 교수학습전략을 제안하였다. 이 전략에서 각 학습의 단계는 학습자들이 다양한 디바이스와 교구재를 포함한 기술을 탐색하고 기능적인 수정을 통해 학습을 확대시켜간다. 또한 컴퓨팅 사고와 관련된 개발 프로젝트를 진행하면서 추상화와 자동화의 개념을 학습하도록 교수학습전략을 제시하였다[13]. 한국교육학술정보원은 매년 운영하고 있는 교육용 로봇 실험학교의 주제를 최근 몇 년간은 소프트웨어 교육과 연계하여 그 효과성과 인식의 변화를 분석하고 있다[15].

로봇의 기능과 동작을 제어하는 것은 소프트웨어적인 코드에 의해서 가능하다. 즉 로봇을 작동하고 제어하기 위해서는 프로그래밍 언어 또는 그에 상응하는 소프트웨어적인 도구가 요구된다. 소프트웨어교육의 관점에서 교육용 로봇을 제어하는 유형을 분석하면 <Table 2>와 같다[12].

Table 2. Classification of robots for SW education

| Type            | Characteristic                                                                                                  |
|-----------------|-----------------------------------------------------------------------------------------------------------------|
| Button          | When the user presses a physical device (such as a button) attached to the robot to control the operation       |
| Card            | When a robot moves by inserting various kinds of cards for controlling the movement of the robot into the robot |
| Apps            | Controlling the robot's movement with a controller-type app                                                     |
| Sensor          | When the robot's motion is controlled by using the sensor attached to the robot                                 |
| Block-based EPL | When a robot's motion is coded with a block-based programming language (such as Scratch)                        |
| Text based EPL  | If you are coding a robot's movement with a text-based programming language (C, Python, etc.)                   |
| Etc             | Other than the above-mentioned classification                                                                   |

<표 2>의 로봇 제어 유형은 최근 소프트웨어 교육에서 중요시하고 있는 코딩의 알고리즘과 컴퓨팅 사고의 신장을 위해 한 가지 제어 방법의 유형보다는 여러 가지 제어 방법의 유형을 복합적으로 적용하고 있다.

본 연구에서 제안하는 카드 코딩 로봇의 제어방식은 센서 제어 방식과 앱제어 방식 그리고 블록 기반 EPL을 통한 제어 방식을 모두 통합하여 적용하되, 코딩의 문법 구조를 이해하고 컴퓨팅 사고의 신장을 위해 카드를 이용한 제어 방식에 집중하여 설계하고 구현하도록 하였다.

III. Analysis and Design

1. Structure Design of Educational Robot

소프트웨어교육을 위한 로봇의 기본 기능은 다음과 같은 조건을 요구한다. 이러한 조건은 대표적인 EPL인 스크래치와 엔트리 프로그램의 명령 문법을 만족하는 기능을 포함한다. 전체적인 장치는 피지컬 컴퓨팅 모델을 따라 중앙처리장치와 입출력 장치로 구성된다.

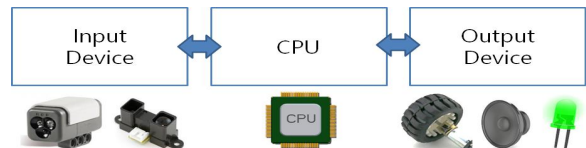


Fig. 1. Design Overview of Educational Robots

첫째, 로봇은 조립형태가 아닌 피지컬 컴퓨팅 교구의 형태로서 입력 장치와 출력 장치 그리고 제어와 연산을 위한 중앙처리장치로 구성된다.

둘째, 입력장치는 명령 카드를 인식하기 위한 컬러 센서 2개와 거리센서 2개로 구성된다. 자이로센서나 움직임센서 등의 입력장치는 명령카드로 코딩하는 것과 관계없으므로 선택사항이다.

셋째, 출력장치는 이동을 위한 모터와 바퀴, 사운드를 위한 스피커, 색표현을 위한 LED로 구성된다.

넷째, 중앙처리장치는 명령카드의 입력처리와 로봇동작의 출력처리를 한다. 특히 명령의 제어와 연산 모듈을 가진 ROM과 명령카드의 입력 절차를 기억하는 RAM 메모리 영역의 크기가 제한된다.

## 2. Design of Robot Command Card

### 2.1 Design Overview

교육용 로봇은 학습자 수준, 구입비용, 컴팩트한 형태와 구조 그리고 센서의 크기를 만족해야 하기 때문에 로봇제어 명령카드의 설계시 이러한 하드웨어의 환경을 고려해야 한다. 특히 입력 센서 중 컬러센서는 그 수가 2개로 한정되어 있어 명령카드의 종류가 제약이 있고 주변 빛의 밝기와 빛이 비추는 방향에 따라 인식률이 영향을 주기 때문에 색상의 선택과 명령어 구성의 카테고리를 적절하게 설계해야 한다.

또한 코딩 문법의 적용을 위해 출력 장치의 특징을 고려하고 명령의 절차적 처리를 위한 RAM의 용량 또한 고려하여 명령카드를 설계하였다.

### 2.2 Analysis of Coding Grammar

프로그래밍 언어의 코딩 문법은 프로그래밍 언어의 특성에 따라 다양하기 때문에 표준화된 문법을 따르기 어렵다. 교육용 프로그래밍 언어(Educational Programming Language, EPL)는 이러한 코딩 문법의 다양성을 배제하고 학생들이 필수적으로 배워야 하는 문법 구조만을 제공한다.

따라서 본 연구에서는 전 세계적으로 가장 많이 사용되며 MIT 미디어랩에서 개발한 EPL인 스크래치(Scratch, <http://scratch.mit.edu>)를 기준으로 코딩 문법을 분석하였다.

Fig 2의 스크래치의 실행 화면을 살펴보면 상단 중앙에 명령어 군이 범주화되어 구분되어 있다. 교육용 프로그래밍에서 필요한 명령들을 기능별로 묶어 블록형태로 구성되었다. 각 명령어군은 Table 3에서 보는 것처럼 색상값으로 구분하여 나뉘었으며 총 10개 군으로 이루어져있다.

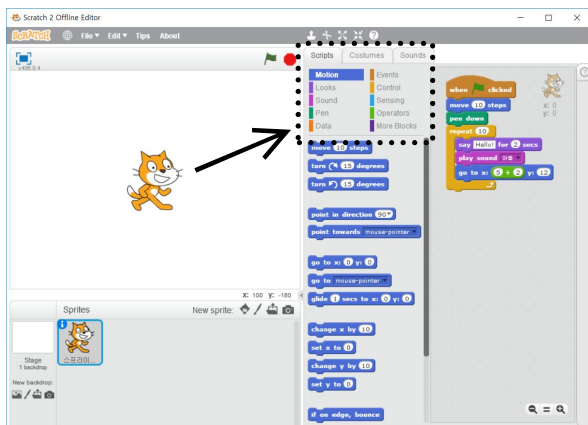


Fig. 2. Screen of Scratch

로봇의 제어를 위한 명령 카드는 Table 3의 스크래치 명령군의 10개에 속한 문법들을 이해할 수 있도록 설계할 필요가 있다. 이 명령 중 명령카드를 모두 개발하는 것이 아니라 명령카드의 논리적인 조합과 활용하여 개념적으로 접근할 수 있는 것과 실제적인 사용을 통해 물리적으로 접근하도록 융통성 있게 설계할 수 있다. 스크래치 명령어군의 분석을 통해 선정된 명령어는 Table3의 오른쪽에 제시하였다.

Table 3. Command Category of Scratch

| Category    | Description                                                                                                            | Selected Command    |
|-------------|------------------------------------------------------------------------------------------------------------------------|---------------------|
| Motion      | Moves sprites and changes angles and change X and Y values                                                             | Moving<br>Pause     |
| Looks       | Controls the visuals of the sprite; attach speech bubble, change of background, enlarge or shrink, transparency, shade | Light<br>Line       |
| Sound       | Plays audio files and programmable sequences                                                                           | Sound               |
| Pen         | Draw on the portrait by controlling pen width and color. Allows for turtle graphics.                                   | Moving              |
| Data        | Variable and List usage and assignment                                                                                 | Number(n)           |
| Events      | Contains event handlers placed on the top of each group of blocks                                                      | Start-End           |
| Control     | Conditional if-else statement, "forever", "repeat", and "stop"                                                         | Loop<br>Conditional |
| Sensing     | Sprites can interact with the surroundings the user has created                                                        | Function            |
| Operators   | Mathematical operators, and-or statement that compares sprite positions                                                | -                   |
| More Blocks | Custom procedures (blocks) and external devices control                                                                | Function            |

### 2.3 Design of Command Cards

명령카드의 설계는 명령 로직 설계와 인터페이스 설계로 구분된다. 명령로직 설계는 2.2절에서 분석한 것처럼 EPL 명령어군의 개념을 대부분 포함하도록 해야 한다. 로봇을 제어하는 명령은 입출력 장치에 종속되어 매우 제한적이므로 명령어의 타입도 다음과 같이 3가지로 구분하였다.

Table 4. Command Type

| Command Type      | Script                     | Command Block |
|-------------------|----------------------------|---------------|
| 0-Address Command | start, halt<br>clear, show |               |
| Control Command   | for { }<br>if { }          |               |
| Single Command    | go 30<br>sound do          |               |

연산명령, 논리명령, 다중 값을 가진 명령은 사용자가 논리적으로 적용하거나 여러 번 반복하면 되므로 명령카드의 설계에서 제외한다.

명령어 카드의 인터페이스 설계는 로봇의 입력 센서 즉, 컬러센서 2개의 위치를 고려하여 카드 상단에 2개 컬러 박스를 배치하고 하단에 명령어 텍스트를 추가하였다. 또한 사용자가 직관적으로 카드의 명령을 이해할 수 있도록 카드의 중앙에 기호와 그림으로 명령어의 의미를 나타냈다.

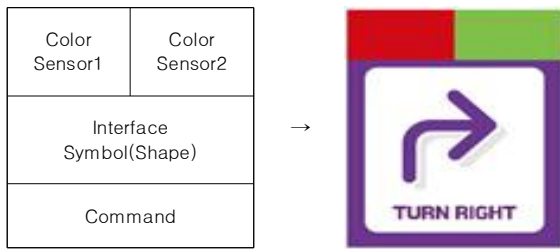


Fig. 3. Interface Design of Command Card

컴퓨팅 사고의 정의와 개념은 J. Wing은 추상화와 자동화로 Google은 분해, 패턴인식, 추상화, 알고리즘, 자동화로 구분하였고, CSTA는 더 많은 하위요소를 제시하며 학자마다 다양한 개념을 제시하고 있다[3,11,13]. 이러한 분석은 사고력의 개념으로 접근하였고 코딩의 문법적인 특징과 개념을 명확하게 제시하고 있지 않다. 본 연구에서 제시하는 컴퓨팅 사고의 개념은 코딩을 통한 신장에 있으므로 코딩 문법과 역량에 집중할 수 있는 정의와 개념을 선택할 필요가 있다. Table 5와 같이 MIT Media Lab의 Scratch 팀에서 제공하고 있는 컴퓨팅사고의 개념(Computational Thinking Concept)은 코딩의 문법을 명확하게 제시하고 있고 스크래치 프로그램의 용이성을 제공하기 때문에 그 기준[5]에 따라 제어구조와 명령어 구조를 설계하였다.

Table 5. CT Framework of Media Lab[5]

| Category       | Concept                                                                                                      |
|----------------|--------------------------------------------------------------------------------------------------------------|
| CT Concept     | Sequence, Loop, Parallelism, Event, Conditional, Data, Broadcast, Function, Operator                         |
| CT Practice    | Experimenting and iterating<br>Testing and debugging<br>Reusing and remixing<br>Abstracting and modularizing |
| CT Perspective | Expressing, Connecting, Questioning                                                                          |

이론적으로는 2개의 컬러센서를 이용하여 CMYK값을 입력받게 되면 총 256개(24\*24)의 명령카드를 만들 수 있으나 두 색의 간섭과 빛의 밝기, 그리고 빛의 입사각에 따라 컬러센서가 인식하지 못하는 경우가 있어 두 개의 컬러값들을 서로 보색(complementary color)의 관계로 제한을 두어 64개(24\*4) 이하의 명령어 카드까지 만들 수 있다. CT 프레임워크에서 제공하고 있는 CT개념을 통해 로봇을 제어하는 명령어 카드는 총 28개를 설계하였다. 그 목록은 다음 Table 6과 같다.

Table 6. List of Command Card

| Category    | num | Command                                   | I/O                                             |
|-------------|-----|-------------------------------------------|-------------------------------------------------|
| Event       | 2   | Start-End, Hand                           | -                                               |
| Moving      | 4   | Forward, Backward, Turn Right, Turn Left, | Motor                                           |
| Pause       | 1   | Wait(n)                                   | wait for moving, sound, led                     |
| Sound       | 1   | MusicOn-Off(n)                            | Speaker                                         |
| Light       | 1   | LightOn-Off(n)                            | LED                                             |
| Loop        | 3   | While2, While3, While-WhileEnd(n)         | -                                               |
| Conditional | 2   | If-IfEnd(n), IfHand, IfDirect             | Proximity sensor                                |
| Number(n)   | 10  | 0~9 Numbers                               | parameter for loop, sound, led, condition, wait |
| Function    | 3   | Function-FuntionEnd, Grid, LineTracing    | -                                               |
| Correction  | 1   | Calibration                               | -                                               |
| Total       | 28  |                                           |                                                 |

### 3. Validity Analysis

설계된 명령어 카드가 코딩의 주요문법에 적절하고 학습자들이 컴퓨팅 사고를 신장하기에 적용가능한지에 대한 검증을 위해 전문가들의 자문을 구하여 내용 타당성 검사를 실시하였다. 전문가 집단은 컴퓨터교육과 교수 2명, 컴퓨터 공학 전공 박사 2명, SW선도 학교 교사 5명, 로봇 개발자 1명으로 구성되었다.

내용 타당도 검사 문항은 5점 척도로 구성된 질문지로 총 10문항을 구성하여 디자인 전반에 관한 문항과 각 명령어 카드의 적합성 그리고 인터페이스에 대한 적합성 문항에 대하여 질문하였다.

타당도 분석은 내용 타당도 지수(CVI; Content Validity Index)로 확인하였는데 이론적으로 CVI 값이 0.75이상인 경우 타당성이 있다고 보고 있다. 각 검사 문항의 CVI값은 7번과 9번 문항을 제외하고 .75이상의 값을 보여 타당성이 높게 나와 제안한 명령어 카드를 이용한 코딩 학습 방법을 구현하고 현장 교육에 적용하는데 무리가 없다는 결과를 보였다.

Table 7. Test results of Content Validity(N=10)

| No | Item                                          | CVI  |
|----|-----------------------------------------------|------|
| 1  | Suitability of Command Card Design            | .79* |
| 2  | Suitability of Event, pause, calibration Card | .82* |
| 3  | Suitability of Moving Card                    | .87* |
| 4  | Suitability of Sound Card                     | .85* |
| 5  | Suitability of Light Card                     | .82* |
| 6  | Suitability of Loop Card                      | .77* |
| 7  | Suitability of Conditional Card               | .67  |
| 8  | Suitability of Number Card                    | .82* |
| 9  | Suitability of Function Card                  | .66  |
| 10 | Suitability of Command Card Interface         | .81* |

\* : CVI > .75

7번과 9번의 경우 조건문과 함수에 관련된 명령어 카드에 관한 검사문항인데 조건문의 복잡성과 조건 표현의 제한으로 인한 적용의 어려움으로 나타났다. 또한 함수 명령의 경우 굳이 로봇의 제어 명령에 포함시킬 필요가 있는가에 대한 의견이 있었고, 실제 적용시 학습자가 함수 안의 명령어들을 기억하는 인지적 부하가 코딩의 재미를 감소시킬 수 있다는 우려의 의견을 제시하였다. 하지만 보통 타당도 검사에서 CVI값을 .6이상인 경우에도 타당성이 매우 높지는 않지만 적용하기에는 적당하다는 이론적인 관점에서 조건문과 함수문의 명령카드도 구현하는데 무리가 없을 것으로 판단된다.

Table 8. Control Command Type of Scratch by CT Framework

| CT Concept  | Code Example                                                                                                                   | Card Design                                                         |
|-------------|--------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------|
| Sequence    | <pre> move 10 steps pen down say Hello! for 2 secs play sound 아웅 turn 15 degrees                     </pre>                    |                                                                     |
| Loop        | <pre> repeat 10 say Hello! for 2 secs play sound 아웅 turn 15 degrees                     </pre>                                 |                                                                     |
| Conditional | <pre> if touching edge? then move 10 steps say Hello! for 2 secs else turn 15 degrees play sound 아웅                     </pre> |                                                                     |
| Data        | <pre> set score to x position change score by 1                     </pre>                                                     |                                                                     |
| Broadcast   | <pre> broadcast message1 when I receive message1 show go to x: -131 y: 90                     </pre>                           |                                                                     |
| Function    | <pre> define function number1 change score by number1                     </pre>                                               |                                                                     |
| Event       | <pre> when space key pressed turn 15 degrees next costume                     </pre>                                           | Logical processing:<br>User processes various execution events      |
| Parallelism | <pre> when clicked clear when clicked show set y to 0 play sound 아웅                     </pre>                                 | Logical processing:<br>Two or more robots are driven simultaneously |
| Operator    | <pre> set y to 5 + 3 * 2                     </pre>                                                                            | Logical processing:<br>User performs the operation                  |

## IV. Development and Application

### 1. Development of Robot Command Card

교육용 로봇 명령 카드의 설계를 바탕으로 코딩 학습을 위한 명령 카드를 개발하였다. 개발된 명령어 카드는 CT 프레임워크의 CT개념 범주에 맞게 Table 8과 같다. CT 개념 범주에 속한 명령 중 순차문, 반복문, 조건문, 변수, 브로드캐스팅, 함수 부분은 명령 카드로 구현을 하였고 이벤트, 병렬처리, 연산자 부분은 굳이 명령 카드로 구현하지 않고 사용자의 실행을 통해 이벤트로 처리하거나 논리적 처리 방법으로 2대 이상의 로봇을 동시 구동시켜 병렬처리에 대한 개념을 익히도록 하였다.

또한 연산식의 경우도 문제를 해결하는 알고리즘의 논리적 상황에서 연산을 통한 처리를 하거나 변수에 사용된 순자를 이용하여 수행하도록 하였다. Table 8의 가운데 표는 스크래치 프로그램과의 비교를 통해 카드 명령어와의 실제 사례를 제시한 것이다.

### 2. Development of Educational Robot

명령 카드를 이용한 코딩 학습을 위한 로봇은 3장의 로봇 설계에서 기술한 바와 같이 입출력 장치와 내부에 연산과 제어를 위한 중앙처리장치를 탑재하였다. Fig 4는 교육용 로봇의 구성과 명령코딩의 이미지를 보여준다.

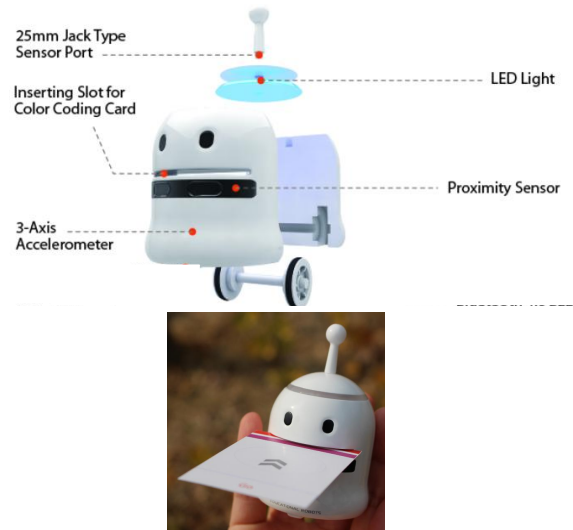


Fig. 4. Educational Robot for Command Card

입력장치는 명령 카드를 인식하기 위한 컬러센서를 로봇 중앙에 삽입할 수 있도록 홈을 만들되 입의 모양으로 설계하여 학습자가 친숙한 이미지로 학습에 임하도록 하였다. 또한 명령 카드 입력 홈 아래 거리센서를 배치하여 조건문에 맞는 동작을 하도록 디자인하였다.

출력장치는 하단의 바퀴를 제어하기 위한 모터와 후면의 스피커 그리고 머리 상단에 띠 모양의 LED를 배치하여 전체적인 이미지를 유치등 학습자의 수준에 맞게 디자인하였다.

### 3. Content for Robot Coding

개발된 로봇과 명령 카드를 초등학교 현장 교육에 적용할 수 있도록 학습 콘텐츠를 개발하였다. 개발된 콘텐츠는 학습자들이 직관적으로 이해할 수 있도록 개발된 코딩 카드를 일련의 단계로 나열하는 형태로 개발하였으며 일련의 단계를 알고리즘적 사고로 이해하며 직접 실습하면서 문제를 해결하도록 하였다. Fig 5는 학습 콘텐츠의 예를 보여준다.



Fig. 5. Content for Robot Education

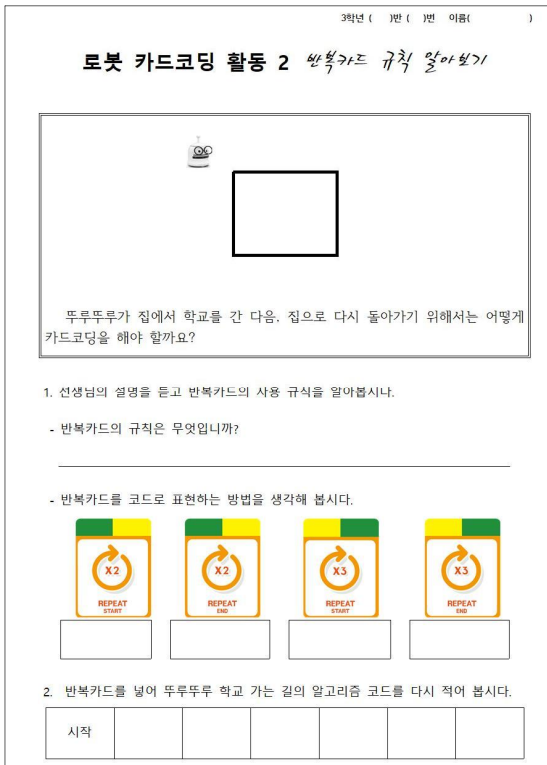


Fig. 6. Worksheet for Robot Education

또한 콘텐츠의 학습 후에 로봇 코딩 명령 카드의 기능과 사용방법을 일반적으로 안내하는 직접교수법 대신 학생이 스스로 반복문의 특징과 규칙을 발견하도록 구성주의적인 학습 전략에 사용할 수 있도록 워크시트 문항지를 개발하였다. Fig 6은 위

크시트의 예를 보여준다.

## V. Conclusions

한국로봇산업진흥회에서 실시한 ‘로봇산업의 실태조사 보고서’에 따르면 국내의 로봇의 산업시장이 2012년에 이미 2조 1300억원이 넘어섰고[7] World Robotics 2013에 따르면 전세계 로봇시장은 2014년부터 2017년까지 매년 6%씩 성장을 기록하고 있다고 보고하였다[17]. 서비스 로봇 시장인 교육용 로봇의 시장 또한 교육/연구용 로봇 비중이 62.6%에 육박하고 있어 교육에서의 로봇 연구와 활용이 활성화 되고 있다[7].

한편, 소프트웨어 중심사회 실현 정책으로 시작된 소프트웨어교육의 현장 적용으로 2015 개정 교육과정에 ‘소프트웨어 교육’이 정보 교과에 반영되어 초등학교에서는 ‘실과’ 교과에서 적용되고 중학교는 독립교과로서 ‘정보’교과가 2018년 실시되며, 고등학교에서는 일반선택 과목으로서 ‘정보’교과를 이수하게 하였다[8,9]. 이러한 소프트웨어 교육의 활성화에 따른 교육용 로봇의 교육적 연구와 개발 또한 활성화 되고 있다.

이에 본 연구는 기계 조작과 조립 형태의 공학적인 로봇 활용 교육의 방법을 벗어나 논리적인 알고리즘과 문제해결 능력을 키우는 방법으로서 로봇을 통해 코딩 능력을 향상시키는 명령 카드의 설계와 개발에 관한 연구를 제시하였다. 연구의 결과는 다음과 같다.

먼저, 교육환경에서 사용되는 EPL의 코딩 문법의 분석을 통하여 명령 카드를 설계하고 실제적인 명령 카드를 개발하였다. 명령 카드의 코딩 수업 적합성에 대한 전문가 10명의 내용 타당도 검사를 통해 설계된 명령 코딩의 방법이 대부분 CVI 값 .75 이상을 보여 연구의 타당성을 높였다.

또한 명령 카드의 적용을 위한 로봇을 개발하고 실제적으로 수업 현장에서 활용할 수 있는 카드 28개를 개발하였다. 개발된 교육용 로봇과 명령 카드는 수업 적합성 확인을 위해 초등학교 3학년을 대상으로 적용하였고 수업 프로그램을 작성하여 16시간 운영하였다. 수업 결과에 대한 학생 설문 결과 코딩의 이해도와 코딩의 자신감 그리고 프로그래밍 언어 학습의 욕구가 긍정적으로 나타났다.

본 연구를 통하여 교육용 로봇이 단순한 제어와 기능의 공학적 학습 도구로서 뿐만이 아니라 창의적인 문제해결을 위한 컴퓨팅 사고(Computational Thinking)의 신장 도구로서 활용될 수 있는 가능성을 제시하였다.

향후 연구로는 로봇의 센서 기능을 추가하여 코딩의 문법에 적용하고 28개의 명령 카드의 범위를 확장하는데 주력할 필요가 있다. 또한 개발된 명령 카드를 활용한 수업에서 학생들의 컴퓨팅 사고가 어떤 측면에서 신장되고 어떤 부분을 보완해야 하는지에 대한 수업 전략적인 측면과 수업 결과에 대한 평가 연구가 필요하다.

## REFERENCES

- [1] D. Catlin, J. Woollard, Educational Robots and Computational Thinking, Proceedings of 5th International Conference Robotics in Education, 144-151, 2014.
- [2] E. K. Lee, "Robot Programming Teaching and Learning Model to Enhance Computational Thinking Ability", Korea National University of Education, Doctoral Thesis, 2009.
- [3] J. M. Wing. "Computational thinking and thinking about computing". Philosophical transactions of the Royal Society, Vol. 366, pp.3717-3725, 2008 October.
- [4] J. M. Wing. "Computational Thinking", Communication of the ACM, Vol. 49, No. 3, pp.33-35, 2006.
- [5] K. Brennan, C. Balch, M. Chung, "Creative Computing Guide Book", Harvard University, 2015.
- [6] K. Brennan, M. Resnick, "New frameworks for studying and assessing the development of computational thinking". In Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada. pp.1-25, 2012.
- [7] Korea Robot Industry Promotion Association, "Report on the actual status of the robot industry", 2015.
- [8] Ministry of Education, KERIS, "Guidelines for Operating Software Education", 2015.
- [9] Ministry of Science, ICT and Future Plan, "SW-centered Society and Software Education", SW Policy Reports, 2015.
- [10] S. K. Han, Teaching and Learning Strategies for SW Education, Creative Computing Issue Report 2015-1, 2015, <http://computing.or.kr>
- [11] S. K. Han, M. Y. Ryu, "Software Education", Seoul: Saengreung-Press, 2016.
- [12] S. S. Seo, I. K. Jung, Y. J. Yang, "Comparative Study on Effects and Recognition of Robot-Based SW Education", Korea Education & Research Information Service Report, KR 2017-3, December 2016.
- [13] W. Allan, B. Coulter, J. Denner, J. Erickson, I. Lee, J. Malyn-Smith, F. Martin, "Computational thinking for youth". White Paper for the ITEST Small Working Group on Computational Thinking (CT), 2010.
- [14] Y. K. Bae, "A Study of the Robot Programming Instructional Strategies Considered Gender Differences". The Journal of Korea Association of Computer Education. Vol. 10, No. 4, pp.27-37, 2007 December.
- [15] Korea Education & Research Information Service, <http://lib.keris.or.kr>
- [16] T. Bell, Unplugged CS site: <http://csunplugged.org>
- [17] World Robotics, "Report of 2013 Service Robots",

[www.bara.org.uk/pdf/2013/World-Robotics-2013.pdf](http://www.bara.org.uk/pdf/2013/World-Robotics-2013.pdf)

## Authors



Sun-Gwan Han received the Ph.D. degrees in Computer Science and Engineering from Inha University, Korea, in 2002. Dr. Han joined the faculty of the Department of Computer Education at GyeongIn National University of Education, Incheon, Korea, in

2002. He is currently a Professor in the Department of Computer Education, GyeongIn National University of Education. He is interested in Computational Thinking Education, STEAM, Computer Education, Software Education and Artificial Intelligence.