

A Study on the Utilization of Open Source Hardware Platform for Convergence IT Education

Seong-Yeol Kim*

Abstract

In this paper, we suggest a method utilizing OSHW(Open Source HardWare) in order to raise up students who are competent in IT convergence and integration as a basic research to improve the university software education. Software education cannot be too much emphasized in the age of big change of Fourth Industrial Revolution. It hardly seems to have changes in the software education area of university where has to train competent technicians to be deployed into the industrial field, although software education is planned even in elementary, middle, and high school. In this situation, we expect that utilizing OSHW in software education can result in gaining a meaningful effect. However, we don't have various and systemic approach which use it as a education system component, unlike the response and necessity OSHW market.

Therefore in this paper we suggest models which constitute software education environment based on OSHW and exemplify how to use it in each model. In addition, we compare and analyze each model in order to give a criteria to choice one of them according to the condition.

▶ Keyword : Convergence IT Education, Open Source Hardware, Software Education

1. Introduction

2016년 초, 세계경제포럼은 '제 4차 산업혁명'이라는 단어를 화두로 내놓았고 전 세계는 이 단어에 대하여 신속하고도 민감한 대응을 하고 있는 것으로 보인다. 기존의 틀을 깨고, 과학기술의 경계를 허물며 새롭게 등장한 기술과 서비스들은 이 단어의 등장을 예고하고 있었다. '교육·학문·예술 등 인간의 이성 및 감성적 능력을 포함하는 문화적 영향력인 소프트파워를 통한 공장과 제품의 지능화'라는 의미의 이 현상은 각 개체 간 소통, 협력, 진화를 이루어가며 융합될 것이다[1]. 이러한 변화에 대한 대응은 다양한 분야에서 시도될 것이며 특히 교육 분야의 대응은 어느 때보다도 중요한 사안이 될 것이다. 또한 현대사회에서 소프트웨어의 중요성이 강조됨에 따라 각국의 수장들이 소프트웨어 교육의 중요성을 강조하고 있을 뿐만 아니라 국내에서는 Naver, 국제적으로는 구글, 페이스북, 마이크로소프트,

아마존 등의 지원을 받는 Code.org 등이 소프트웨어 교육(프로그래밍, 코딩, 소프트웨어개발, 알고리즘교육 등)을 강조하고 있다[2]. 이는 교육기관에서의 학습자들이 소프트웨어 활용 능력뿐만 아니라 소프트웨어를 직접 설계하고 구현할 수 있는 능력을 키워야함을 의미하고 있다[3]. 또한 [4]는 첨단과학기술의 급격한 발달과 글로벌의 가속화로 양극화되는 사회에서 창의·융합형 인재를 양성하는 것이 오늘날의 교육트렌드라 기술하고 있다. 부연으로 물건을 빠른 시간 안에 정교하게 만들어 내는 산업사회에서는 지식을 알기 쉽게 빨리 가르치는 것이 좋은 교육이었다면, 융합시대의 인재양성을 위한 교육방향은 흥미와 자신감을 갖고 창의적으로 탐구하는 사람, 새로운 도전을 즐기는 사람을 양성할 수 있는 교육이 좋은 교육이라 전하고 있다.

[5]는 전 세계에서 소프트웨어를 중심에 둔 컴퓨터 교육의 변화가 진행되고 있음을 언급하면서 미국의 'CSTA K-12 Computer Science Standards', 영국의 'the national

• First Author: Seong-Yeol Kim, Corresponding Author: Seong-Yeol Kim
*Seong-Yeol Kim (sykimnat@nate.com), School of Information Technology, Ulsan College
• Received: 2016. 12. 05, Revised: 2016. 12. 23, Accepted: 2017. 01. 06.
• This work was supported by Ulsan College. Research Grant(2014).

curriculum in England: Computing', 한국의 '2015개정 교육과정', 이스라엘의 '컴퓨터과학 교육과정' 등을 비교 분석하였다. 이를 통하여 핵심의 협력(Collaboration), 컴퓨팅사고력(Computational Thinking), 컴퓨팅연습과 프로그래밍(Computing Practice and Programming), 컴퓨터와 커뮤니케이션 장비(Computers and Communication Devices), 커뮤니티와 세계화, 윤리적 영향력(Community, Global and Ethical Impacts) 등의 공통 키워드를 도출하였다.

[6]은 최근 IT 업계에서 오픈소스 소프트웨어에 이어 오픈소스 하드웨어(Open Source Hardware, OSHW)가 새로운 기술 혁신 트렌드로 받아들여지고 있다고 하였다. OSHW는 Software의 소스 코드에 해당되는 설계와 디자인을 공개하고 관련 정보를 공유하여 더욱 나은 제품 개발을 촉진하고 있다. OSHW는 기본적인 지식만 가지고 있어도 새로운 SW와 HW 제작을 시도할 수 있다는 점에서 혁신적인 HW 제작의 대중화를 견인하는 요인으로 주목받고 있다. 현재 아두이노(Arduino), 라즈베리 파이(Raspberry Pi), 비글본(BeagleBone) 등의 제품이 시장에 선 보이고 있으며 앞으로도 다양한 제품이 등장할 것으로 예상된다. 오픈소스 소프트웨어가 프로그램, 소스 코드 등을 공유함으로써 소프트웨어 시장을 견인한 것처럼, OSHW는 새로운 시장을 견인하는 중요한 요소가 될 수 있을 것으로 생각된다. 이에 따라 시장에서는 OSHW를 다양한 제품과 서비스에 접목하려는 시도들이 진행되고 있다. 구글의 아두이노를 위한 API 지원, Ford의 차량용 인테페이스 플랫폼 개발을 위한 '오픈XC(OpenXC)' 프로젝트[7], [8]에 기술된 바와 같은 다양한 시도가 진행되고 있다. 영국은 학교 교육용으로 다양한 OSHW 도입을 진행해 오고 있다. [9]은 시장에서 확실한 성공에 대한 사례가 없음에도 OSHW가 일반인부터 대기업에 이르기까지 다양한 부문에 걸쳐 주목을 받기 시작한 것은 분명하며, 그에 따라 관련 시장의 규모 역시 한동안 성장할 것으로 내다보았다.

위와 같이 환경의 변화와 새로운 기술의 등장은 참신하고 혁신적인 요소로 다가오고 있으며 기존의 IT 교육 환경에서는 새로운 도전이라 할 수 있다. 제 4차 산업혁명은 소프트웨어 중심 사회로의 변화를 의미하는 것으로 지속적으로 IT 및 컴퓨터과학을 전공한 소프트웨어 인력이 요구됨을 의미한다. 이에 각국은 전술한 바와 같이 소프트웨어 교육을 위한 다양한 접근을 준비하고 있다. 우리나라도 초중고를 중심으로 한 소프트웨어 교육이 진행될 예정이다. 장기적인 관점에서 필수적으로 진행되어야 할 내용이지만 바로 산업 현장에 투입되어야 하는 인재를 양성하여야 하는 대학교의 소프트웨어 교육에 대하여서는 별다른 변화가 없어 보인다. 더불어 대학교의 관련 학과를 전공하는 신입생 수는 매년 감소하고 있는 추세이다[10]. 여기에 전공 학생들마저 소프트웨어 교육을 기피하고 있는 것이 현실이다. 또한 변화되는 사회는 기존의 디지털 중심적인 교육시스템으로부터 아날로그와 디지털 처리 능력을 겸비한 인재, 물리적/논리적 양단간의 경험을 가진 융합형 인재 육성의 필요성을 요구하고 있다. 이에 OSHW를 소프트웨어 교육에 활용하는 것은

좋은 해결책이 될 수 있을 것으로 생각된다. 하지만 OSHW가 시장의 반응 및 필요성과는 다르게 교육시스템의 요소로, 또는 교육에 활용을 위한 체계로의 접근이 다양해 보이지 않는다.

본 연구는 대학 소프트웨어 교육방법의 개선을 위한 기초 연구로서 융합 IT 인재를 길러내기 위하여 OSHW를 활용하는 구체적인 툴(Tool)과 방안을 제시하고자 하였다. 다양한 OSHW 도구를 이용하여 환경을 구성하고 학습자를 대상으로 경험하게 하여 효과성이 있다고 판단된 방식에 대하여 교육요소를 구성하고 교육에 활용하는 접근법을 제시하였다. 이를 위하여 2장에서 소프트웨어 교육 현황과 연구, OSHW에 대하여 기술하고, 3장에서는 구체적인 OSHW를 이용한 교육환경 구성과 이용 방법에 대하여 제시하였다. 그리고 각 방식에 대한 비교 분석을 통하여 상황에 맞는 구성을 선택할 수 있도록 하였다. 4장에서 결론을 맺는다.

II. Preliminaries

1. Related works

1.1 Software Education Status

대화형 컴퓨터 시스템 즉, PC가 일반화 되면서 소프트웨어 교육 환경에도 많은 변화가 있었다. 분석, 설계 시트를 구성하고 코딩으로 옮기는 환경에서 문제를 바로 키보드를 통한 코딩으로 옮기는 환경으로 바뀌면서 분석, 설계를 진행하는 과정이 생략되고 모니터를 통한 결과에 바로 대응하는 방식이 되었다. 이는 문제를 스스로 분석하고, 설계하고, 구현하는 과정을 깨뜨리고 단순히 교육교재의 내용을 따라치는 '따라치기'라는 소프트웨어 교육의 병폐를 야기하고 있다. 이는 소프트웨어 교육의 본래 목적을 달성하지 못하고, 학습자에게는 교육의 목표와 의미를 이해하지 못하게 하는 현상을 발생시키고 있다. 모니터와 키보드만으로 구성된 소프트웨어 교육 환경은 학습자들에게 지겨움과 어려움을 제공하고 있다. 결과적으로 창의적이고 도전적으로 진행되어야 하는 교육이 학습자들의 관심에서 멀어지게 되는 것이다. 이를 극복하기 위한 다양한 시도가 진행되고 있는데 대표적인 것으로 다음과 같은 방법들이 있다.

- EPL(Educational Programming Language)
- LEGO MindStorm
- Robot System
- Physical Computing

하지만 EPL은 [10]에서 보인 바와 같이 제한적인 학습 범위를 갖는다는 단점과 여전히 디지털 환경에서 벗어나지 못한다는 단점이 있다. LEGO MindStorm와 Robot System을 이용한 교육은 EPL의 단점을 어느 정도 극복할 수 있다고 생각되나 역시 벤더가 제공하는 제한적인 학습 환경의 문제와 부품의 호환성이 떨어지는 단점이 존재한다. 또한 이 시스템들은 초기 도입 비용도 과다하다는 문제점도 있다.

1.2 Research on Software Education

서론에서 전술한 바와 같이 소프트웨어 교육의 중요성은 강조되고 있으며 이에 따라 다양한 연구가 진행되고 있는 것으로 보인다. [11][12][13][14]에서와 같이 소프트웨어 교육을 위한 연구가 진행되었으나 주로 초중고 교육기관에 중심을 두고 있다. 물론 대학교육에 중심을 두고 연구된 결과도 있다. [15]은 NCS를 기반으로 정보기술분야의 교육과정에서 반영될 수 있는 실무적인 방안을 제시하였다. [16]은 전문대학 컴퓨터과의 특성화를 위한 NCS기반 교육과정을 제시하였다. 이들 연구는 교육과정에 중심을 두고 있고 기술되어 있다. 초중고 교육기관을 대상으로 한 연구들이 정책, 교육과정, 방법 등 다방면에서 다양하게 진행되고 있는 반면에 대학교육에서의 소프트웨어 관련 교육은 많은 연구가 진행되고 있지 않는 것으로 보인다. 융합인재육성이라는 키워드에 맞추어 SW교과의 교양 편성 또는 비전공자의 SW교과 수강을 위한 노력들이 진행되고 있다. 하지만 전공 학습자들이 이탈하지 않고 융합인재로 육성될 수 있는 소프트웨어 교육환경에 대한 접근이 더 필요할 것으로 생각된다.

2. Trend of OSHW

오픈 소프트웨어가 세상의 변화를 주도한 것처럼 오픈하드웨어 플랫폼은 다양한 가능성을 잠재하고 있다. 따라서 이에 대한 적절한 교육 방안이 마련되고 제공되어야 할 충분한 필요성이 존재한다. 단순히 소프트웨어만 활용하는 기존 소프트웨어 중심의 학습에 OSHW 플랫폼을 활용하는 것은 효과적인 교육 방법일 뿐만 아니라 타 분야에 대한 관심과 적응도를 높일 것으로 판단된다. 소프트웨어와 하드웨어를 연계한 학습은 알고리즘 향상, 문제해결 능력, 융합적 사고력 등을 크게 신장할 수 있으며, 궁극적으로 학습태도 및 학업 성취력을 높이는 것으로 보여지고 있다. 본 연구를 수행하며 효과적으로 사용되어진 3종의 OSHW는 다음과 같다.

2.1 Typical OSHW

2.1.1 Raspberry Pi

컴퓨터 기초과학교육을 증진시키기 위해 영국의 라즈베리파이 재단에서 만든 손바닥만 한 크기의 싱글 보드 컴퓨터로서 HDMI포트를 통한 모니터 연결과 USB포트를 통한 마우스와 키보드의 연결이 가능하다[17]. 라즈베리파이는 2012년 3월 첫 판매가 이루어졌으며, 2013년에는 라즈베리파이 모델 A, B가 각각 출시되었고, 2015년에는 쿼드코어 CPU를 장착한 라즈베리파이2 모델 B가 출시되었다. 모델 A와 B의 차이점은 하드웨어 구성요소에 따라 구분된다. 2016년에는 무선 랜과 블루투스 기능이 내장된 라즈베리파이3 모델 B가 출시되었으며, 전력적인 면에서 많은 기능 개선이 이루어졌다. 현재 라즈베리파이를 활용한 IoT기반의 스마트 홈 및 빅 데이터 분석 등과 같은 연구에 적극적으로 활용되고 있다[18]. 본 연구에서는 2016년에 출시된 라즈베리파이3 모델 B를 기준으로 하고 있

며, 그 하드웨어 구조와 사양은 Fig.1과 Table 1에 나타내었다.

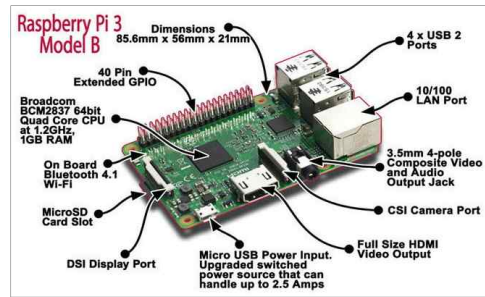


Fig. 1. Raspberry Pi 3 Model B Structure[19]

Table 1. Raspberry Pi 3 Model B Specification[19]

Device Category	Raspberry Pi 3 Model B
CPU	1.2GHz 64-bit quad-core ARMv8
GPU	Videocore IV
Memory	1GB
Storage	microSD card slot
Connectivity	4xUSB, HDMI, Ethernet, 3.5mm audio jack, Wi-Fi, Bluetooth
OS	Linux, Windows 10 IoT core
Connectors	Camera interface(CSI), GPIO, SPI, I2C, JTAG

2.1.2 Arduino

아두이노는 2005년 이탈리아 북부 이브레아(Ivrea)라는 도시에서 IT융합을 교육하던 전문대학원인 IDII(Interaction Design Institute Ivrea)에서 예술을 전공한 학생들이 자신의 작품을 손쉽게 제어하기 위해 고안된 단일 마이크로컨트롤 보드를 말한다. 아두이노는 여러 개의 입출력 포트를 통해 LED, 센서 및 모터 등과 같은 부품들을 제어함으로써 상호작용이 가능한 다양한 형태의 장치를 개발할 수 있다. 아두이노 통합개발환경(Arduino IDE)은 소스코드를 작성하고 편집할 수 있으며, 소스코드를 아두이노 하드웨어가 이해할 수 있는 명령어로 컴파일하여 USB포트를 통해 업로드가 가능하다. 성능과 용도에 따라 다양한 종류의 아두이노 보드가 존재하며 이 중 아두이노 우노 보드가 가장 많이 사용되고 있다[20]. 본 연구에서는 아두이노 우노를 기준으로 하고 있으며, 하드웨어의 구조와 사양은 Fig.2와 Table 2에 나타내었다.

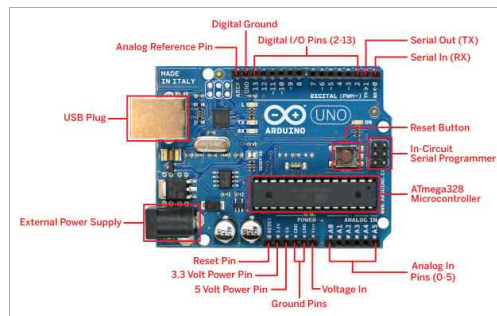


Fig. 2. Arduino UNO Structure[21]

Table 2. Arduino UNO Specification[21]

Device Category	Arduino UNO
CPU	None
GPU	None
Memory	SRAM 2KB, EEPROM 1KB
Storage	32KB
Connectivity	1 x USB, 1 x External Power Supply
OS	None
Connectors	GPIO

2.1.3 Smart Inventor Board

스마트 인벤터 보드는 모터 드라이브 및 적외선 센서 등과 같은 다양한 센서가 내장되어 있으며, 아두이노 IDE, 스크래치, 유니티 3D 및 Snap4Arduino를 기반으로 한 Rokit Brick 등으로 프로그래밍이 가능한 교육용 로봇 제어 보드이다. 기본적으로 내장된 센서 외에 Bluetooth 및 Zigbee 등과 같은 다양한 외부 센서의 사용도 가능하다[22]. 본 논문에서는 다양한 프로그래밍 언어 중 Rokit Brick을 기준으로 하며, 하드웨어의 구조와 사양은 Fig.3과 Table 3에 나타난 바와 같다.

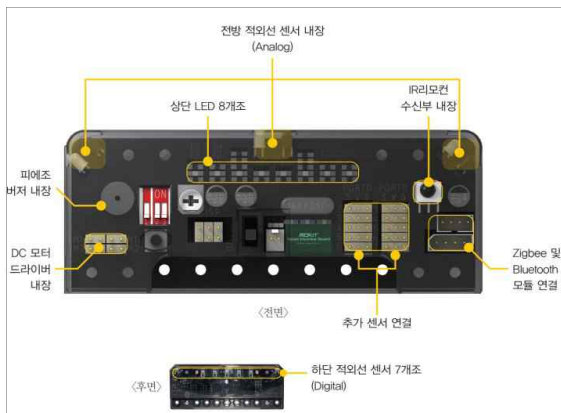


Fig. 3. Smart Inventor Board Structure[22]

Table 3. Smart Inventor Board Specification[23]

Device Category	Smart Inventor Board
CPU	ATmega32
GPU	None
Memory	SRAM 2KB, EEPROM 1KB
Storage	32KB
Connectivity	Top side IR Sensor x 3, Bottom side IR Sensor x 7, IR Receiver
OS	None
Connectors	Bluetooth Port, UART Port, USB Sensor Port, USB Servo Port

III. Configuration Proposal for Software Education

1. Raspberry Pi with Python & Minecraft Pi

라즈베리파이에 리눅스 데비안을 기반으로 하는 라즈비안 운영체제를 설치할 경우 기본적으로 파이썬 프로그래밍 언어와 마인크래프트 파이 에디션이 설치되고, 마인크래프트 API와 파이썬 프로그래밍 언어를 통해 마인크래프트 파이 에디션을 제어할 수 있으며 기본적인 구조는 아래와 같다.

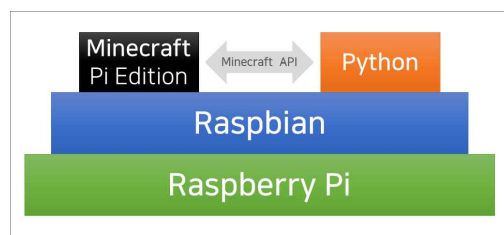


Fig. 4. Structure map

라즈베리파이에 라즈비안 운영체제를 설치하는 방법에는 NOOBS(New Out Of Box Software)와 Raspbian 이미지 파일을 이용하는 두 가지 방법이 있다. NOOBS는 오프라인 설치와 네트워크 설치를 모두 지원하는 NOOBS와 네트워크 설치만 지원하는 NOOBS LITE로 구분된다. NOOBS는 설치 과정이 간단하고 두 개 이상의 운영체제를 쉽게 설치할 수 있으며 복구 기능을 제공한다. 또한 Raspbian 이미지 파일을 이용한 설치 방법에는 RASPBIAN JESSIE WITH PIXEL과 RASPBIAN JESSIE WITH PIXEL의 경우 GUI(Graphic User Interface)와 CLI(Command Line Interface)를 동시에 지원하며 다양한 응용 프로그램의 사용이 가능한 반면 RASPBIAN JESSIE LITE의 경우 CLI만 지원하며 부가 기능이 제외되어 개발에 최적화되었다고 할 수 있다[24]. 본 논문에서는 RASPBIAN JESSIE WITH PIXEL을 이용한 환경설정 방법을 제시하고자 한다.



Fig. 5. Configuration Order

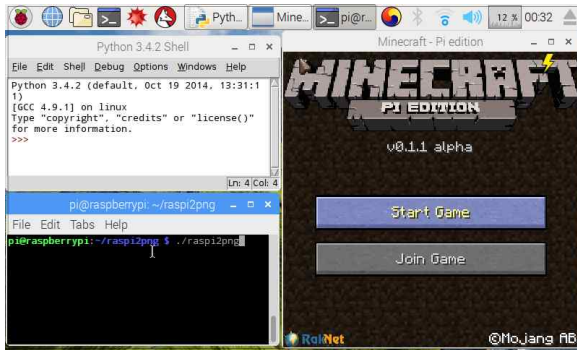


Fig. 6. Python & Minecraft Pi Edition

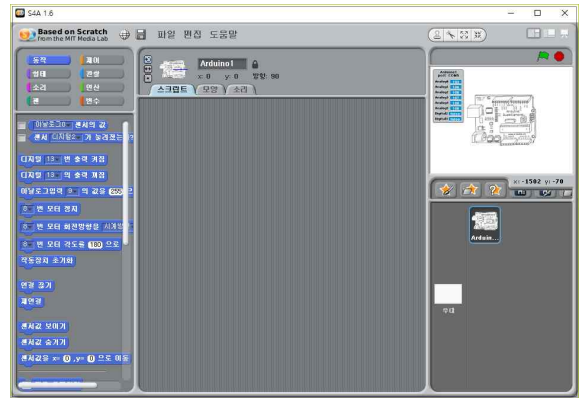


Fig. 9. S4A(Scratch for Arduino)

2. Scratch for Arduino

대표적인 교육용 프로그래밍 언어인 스크래치를 확장한 S4A(Scratch for Arduino)는 S4AFirmware16을 통해 아두이노에 연결된 센서(Sensor) 및 액츄에이터(Actuator)를 제어할 수 있으며 기본적인 구조는 아래와 같다.

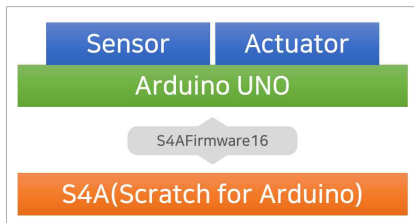


Fig. 7. Structure map

현재 스크래치 기반의 S4A(Scratch for Arduino)는 사용자들에게 무료로 제공되며 아두이노 통합개발환경을 이용해 아두이노에 S4AFirmware16 펌웨어를 설치함으로써 아두이노에 연결된 다양한 센서 및 액츄에이터 제어가 가능해진다. 본 논문에서는 다양한 아두이노 보드 중 아두이노 우노를 이용한 환경 설정 방법을 제시한다.

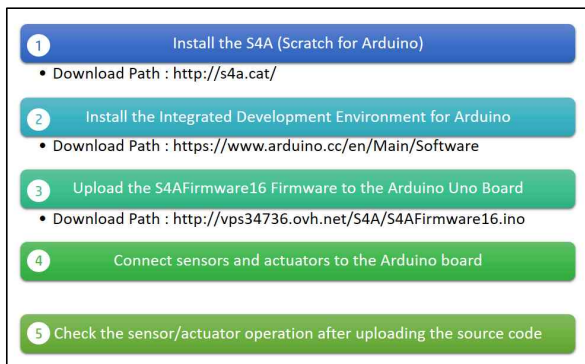


Fig. 8. Configuration Order

3. Smart Inventor Board with RokitBrick CODRONE

로킷브릭 코드론은 버클리 대학에서 교육용으로 개발한 Snap과 Snap4A 프로그래밍 언어를 기반으로 만들어 졌으며 퍼마타 프로토콜을 통해 스마트 인벤터 보드를 제어할 수 있다. 또한 스마트 인벤터 보드와 연결된 BLE(Bluetooth Low Energy) 링크 보드를 통해 무선으로 드론을 제어할 수 있으며 기본적인 구조는 아래와 같다[25].

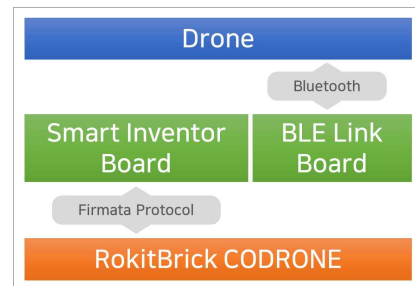


Fig. 10. Structure map

드론 제어를 위해 Snap을 기반으로 개발된 로킷브릭 코드론은 로컬 시스템에 설치용으로 제공된다. 스마트 인벤터 보드와 BLE 링크 보드는 스마트 인벤터 보드에 있는 블루투스 모듈 연결 포트를 통해 연결할 수 있으며, 드론은 BLE 링크 보드와 블루투스로 연결되어 제어가 가능하다. 아래에 환경설정 방법을 제시하였다.

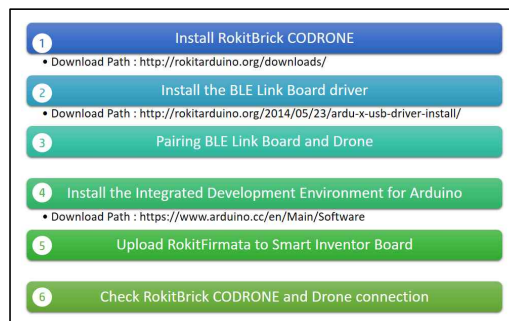


Fig. 11. Configuration Order



Fig. 12. RokitBrick CODRONE



Fig. 14. Before execution

IV. Implementation and Analysis

1. Implementaion of Mainly Instruction by eachSystem

1.1 Raspberry Pi with Python & Minecraft Pi Edition

파이썬 프로그래밍 언어의 경우 문법이 매우 직관적이고 쉬워 빠르게 익힐 수 있으며 소스코드가 매우 간결한 특징을 가지고 있다. 또한 마인크래프트 API를 이용하여 파이썬을 통한 마인크래프트 파이 에디션 게임을 직접 제어해 보도록 한다.

```

if_while2.py - /h...hile2.py (2.7.9)
File Edit Format Run Options Windows Help
from mcpi.minecraft import Minecraft
from time import sleep
mc=Minecraft.create()
while True:
    x,y,z=mc.player.getPos()
    block_beneath=mc.getBlock(x,y-1,z)
    if block_beneath == 2:
        mc.setBlock(x,y,z,38)
    else:
        mc.setBlock(x,y-1,z,2)
    sleep(0.1)
Ln: 14 Col: 11
    
```

Fig. 13. if and while statements in Python

Fig. 13의 소스코드는 if와 while문을 사용한 예제이다. 실행 시 마인크래프트 파이 에디션 게임의 캐릭터가 이동한 경로 상에 풀(grass)이 존재할 경우 풀 위에 꽃이 나타나게 되고, 풀이 아닐 경우 풀로 변경 후 꽃이 나타나게 된다. 실행 전, 후의 화면은 아래와 같다.



Fig. 15. After execution

1.2 Arduino with S4A(Scratch for Arduino)

스크래치는 대표적인 블록 기반의 교육용 프로그래밍 언어로서 문제해결능력향상에 도움이 되는 알고리즘 교육의 도구로 널리 사용되고 있다. 기존 스크래치의 기능을 확장한 스크래치포 아두이노를 이용하여 아두이노에 연결된 액추에이터와 사운드를 직접 제어해 보도록 한다. Fig.16의 LED 액추에이터는 구성의 편의성과 학습자들의 흥미를 위하여 본 연구 진행시 제작된 것이다.

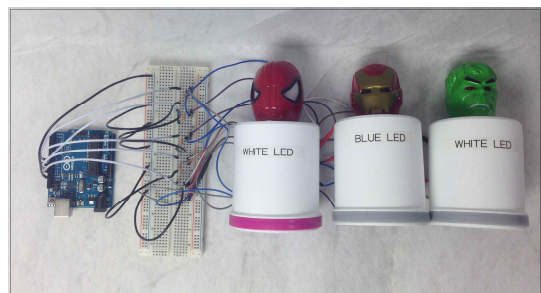


Fig. 16. 4 LEDs blinking with play sound project

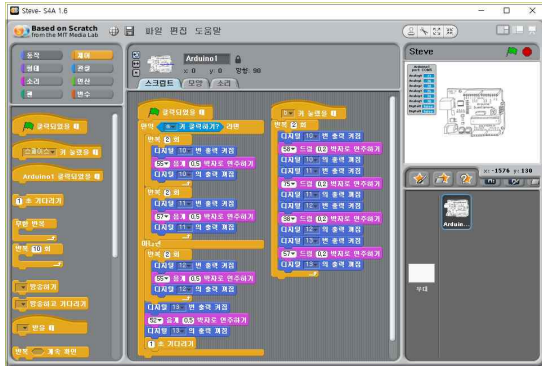


Fig. 17. Conditional and repetitive statements block in S4A

Fig.16에서 브레드보드를 통해 4개의 LED를 아두이노 보드와 연결 후 Fig.17 S4A 스크립트 입력 창에 조건문 블록과 반복문 블록을 활용하여 키보드의 'a' 입력 시 '술술라라' 음계가 재생됨과 동시에 LED가 순차적으로 켜졌다 꺼지며, 그 이외의 경우에는 '술술미' 음계가 재생됨과 동시에 LED가 순차적으로 켜졌다 꺼지게 된다. 또한 키보드의 'b' 입력 시 미리 녹음된 드럼 악기 소리가 재생됨과 동시에 LED가 켜졌다 꺼지게 된다.

1.3 Smart Inventor Board with RokitBrick CODRONE

스크래치와 더불어 대표적인 교육용 프로그래밍 언어인 Snap 및 Snap for Arduino를 확장한 로킷브릭 코드론을 활용하여 드론을 제어해 보도록 한다.



Fig. 19. Repetitive statements block in RokitBrick CODRONE

Fig.18에서는 조건문 블록을 이용하여 키보드의 'w' 입력 시 드론이 상승하고 's' 입력 시 드론이 하강하게 된다. throttle값이 양수 일 때는 드론이 상승하고 음수 일 때는 하강하게 되는데 만약 throttle값이 100을 초과할 경우 throttle값을 100으로 유지하고, -100미만일 경우 throttle값을 -100으로 유지하게 된다. 또한 Fig.19에서는 키보드의 's' 입력 시 드론이 이륙하고 이륙 명령 1.5초 후 좌회전(yaw : -60) 한 후 오른쪽으로 이동(roll : 100)하고 1.2초 후 이 과정을 4회 반복하게 된다. yaw값이 양수 일 때는 드론이 오른쪽으로 회전하고 음수 일 때는 왼쪽으로 회전하게 되며, roll값이 양수 일 때는 오른쪽으로 이동하고 음수 일 때는 왼쪽으로 이동한다. Fig.20은 구성에 사용된 드론과 스마트 인벤터 보드이다.

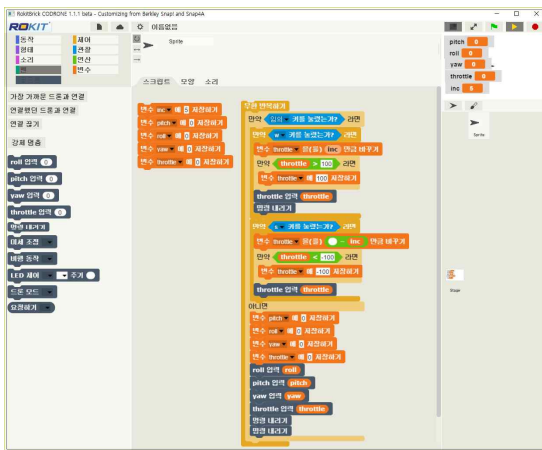


Fig. 18. Conditional statements block in RokitBrick CODRONE

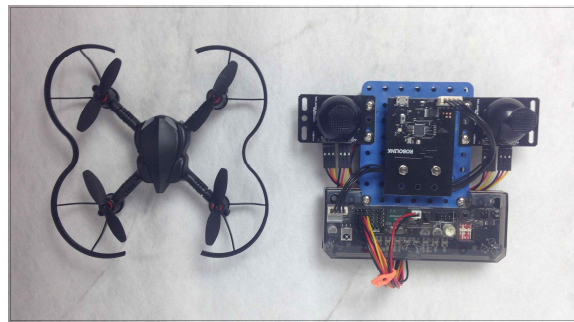


Fig. 20. Drone and Smart Inventor Board

2. Comparison and Analysis

일반적으로 OSHW에 대한 정보는 보드 자체의 특성에 대한 비교를 그 중심으로 하고 있다. 본 논문에서는 소프트웨어 교육을 위한 OSHW 환경을 구성하고 이를 기반으로 구체적 전개 가능성을 보였다. Table.4는 전술한 세 가지 OSHW 플랫폼의 특징을 비교분석한 것이다. 그 비교 값은 제안된 세 가지 방식의 상대적 평가이다.

Table 4. Three model comparison analysis

Model Item	1)	2)	3)
OSHW	Raspberry Pi3 Model B	Arduino UNO R3	Smart Inventor Board
OS	Raspbian(based on Debian)	N/A	N/A
PC OS	N/A	Windows 10	Windows 10
Language	Python	Scratch	Snap
Develop Environment	· Python 2 IDLE · Python 3 IDLE	S4A (Scratch for Arduino)	RokitBrick CODRONE
Extra Device	N/A	· LED · Breadboard · Jump wire · Resistor	· Drone · BLE Link Board
Construction Cost	Low	Low	Relative High
Construction Difficulty Level	Low	Middle	Middle
Convenience	Low	Middle	Middle
Perception Level of the Result	Middle	Middle	High
Response Time	Immediate	Immediate	Immediate

- 1) Raspberry Pi with Python & Minecraft Pi Edition
- 2) Arduino with S4A(Scratch for Arduino)
- 3) Smart Inventor Board with RokitBrick CODRONE

제안된 세 가지 플랫폼은 각각의 장단점을 가지고 있다. 우선 OSHW의 운영체제 유무에 따라 소프트웨어 교육환경을 위한 별도의 컴퓨터를 요구하는 2),3)과는 다르게 1)은 별도의 컴퓨터를 필요로 하지 않는다. 개발환경은 모두 오픈소스로서 인터넷을 통하여 제공 받을 수 있어 별도의 비용이 요구되지 않는다. 라즈베리파이는 별도의 구성요소를 제어할 수 있으나 제안된 1)의 플랫폼은 별도의 하드웨어적인 기타 구성요소를 요구하지 않는다. 이에 반하여 2)와 3)은 별도의 구성요소가 요구된다. 교육을 위한 플랫폼 구축의 용이성은 1)이 가장 뛰어나다. 또한 1)은 구축비용에 있어서도 장점이 있다. 그러나 개발환경의 편의성에 있어서는 1)이 상대적으로 떨어지는 것으로 판단된다. 학습자가 작성한 프로그램에 대하여 그 반응을 인지하는 정도는 2), 3)에서 높게 나타난다. 기존의 환경에 비하여 물리적 또는 아날로그적으로 표현되기 때문이라고 판단된다. 하지만 1)의 경우에도 게임의 형태로 나타나기 때문에 기존의 환경에 비하여서보다 높은 인지성을 갖는다. 특히 3)의 경우 결과에 대한 인지도가 높게 나타나지만 구축비용이 상대적으로 높고 결과를 확인하는데 물리적공간과 전원 공급의 제약 사항을 갖는다.

V. Conclusions

OSHW 플랫폼은 다양한 분야에 활용될 수 있는 특징으로 인하여 IT와 타 분야의 융합에 주도적인 역할을 하고 있다. 따라

서 단순히 소프트웨어만 활용하는 기존 학습 환경보다는 OSHW 플랫폼을 활용하는 것은 효과적인 교육 방법일 뿐만 아니라 타 분야에 대한 관심과 적응도를 높일 것이라 판단으로 본 연구는 수행되었다. 연구의 개시 전인 2013년부터 매년 OSHW를 기반으로 하는 캡스틴디자인을 수행하여 왔다.

Table 5. List of Capstone Design

Year	Works
2013	Construction of Access control system using RFID and Arduino board
	Development and application of USB driver for Excel data processing
2015	Building Ardu-crane using Open source based Arduino
	DIY project using Open platform based hardware and Arduino
	Smart Car control based on Open Source Hardware
2016	Development of Education Contents using Arduino and Papercraft
	Design and Construction of Remote controlled Flying Balloon using Open source based Arduino

이를 통하여 OSHW를 이용한 소프트웨어 교육이 알고리즘 향상, 문제해결 능력, 융합적 사고력 등을 크게 신장할 수 있으며, 궁극적으로 학습태도 및 학업 성취력을 높이는 것을 정성적으로 파악할 수 있었다. 하지만 캡스틴디자인은 비교과과정으로 진행될 뿐만 아니라 제한된 학생만이 참여할 수 있다는 한계가 있다. 따라서 정규교과에서 기존의 소프트웨어 교육환경을 개선하여야 한다. 연구를 진행하는 동안에도 사회 여러 부분에서 기존과는 다른 소프트웨어 및 알고리즘 교육에 대한 도전이 시작되었으며 이는 본 연구가 충분한 필요 동기를 가지고 시작되었음을 반증하는 것으로 판단할 수 있었다.

본 연구는 기존의 대학 소프트웨어 교육을 개선할 수 있는 방안을 제시하고자 하였으며 본 논문에서는 그 구체적인 방법의 일부를 제시하였다. 또한 제시된 구성에 대한 비교를 통하여 각 교육현장에서 요구되는 제약 또는 상황에 따라 교육환경을 선택할 수 있는 기본 정보를 제공하고 하였다. 현 시대적, 사회적 환경 변화에 대응하는 소프트웨어 중심적 인재 육성을 위해서는 지속적인 방안들이 제시되어지고 시도되어야 할 것으로 생각된다. 추후에는 제안된 플랫폼을 기반으로 더 구체적인 콘텐츠를 개발하고 이를 교과과정으로 구성하는 연구를 진행하고자 한다.

REFERENCES

- [1] What is the fourth industrial revolution in the 21st century?, <http://www.ulalalab.com>
- [2] Code.org, <https://code.org/>
- [3] Sangjin An , Youngjun Lee, “Educational Objectives in Computing Education,” Journal of the Korea Society of Computer and Information, Vol.21, No.1, pp.181-189, Jan.2016.

- [4] Hyun, Hye Jung, Kim, Kyung Hoon, "A suggestion for the development and utilization of experience education program converging open source hardware and design," KOREA SCIENCE & ART FORUM Vol.14, pp.527-537, Dec.2013.
- [5] Seong-Won Kim, Youngjun Lee, "Development of a Software Education Curriculum for Secondary Schools," Journal of the Korea Society of Computer and Information, Vol.21, No.8, pp.127-141, Aug.2016.
- [6] Yoo, Jae-pil, "Open Source Hardware Platform (OPHW) Trends and Forecasts," Internet&Security Focus, pp. 24-50, Aug.2013.
- [7] Ford 'Automotive hardware is also open source', http://www.zdnet.co.kr/news/news_view.asp?artice_id=20130111093830
- [8] Innovative and incredible Raspberry Pie Project 10 things, <http://www.itworld.co.kr/print/92451>
- [9] Policy Research Division Fusion Policy Research Department, 'Features and popularization of new keyword 'open source hardware' in ICT market', Trends and prospects, Vol.67, pp.46-56, Korea Broadcasting & Communications Agency, Oct.2013.
- [10] Park Hyo-min, "Global Software Education Status and Training Tools Trends," INTERNET & SECURITY FOCUS September 2014, pp.40-56. Sep.2014.
- [11] In Sik Kim, Chang Ho Yun, Jong Won Park, Chul Sang Yoon, Yong Woo Lee, "Vocational education in the high school for the development of software industry," The Proceeding of Korean Society For Internet Information, Vol.15, No.2, pp.37-38, Oct.2014.
- [12] Jin-Youn Kim, Hye-Yeon Huh, Young-Min Kim, Ki-Soo Kim, "Analysis On The Awareness And The Needs Of Technology Teachers For Sw Education On Secondary School," The Korean Journal Of Technology Education, Vol.15, No.3, Pp.50-72, Dec.2015.
- [13] Yohan Hwang, Kongju Mun, Yunebae Park, "Study of Perception on Programming and Computational Thinking and Attitude toward Science Learning of High School Students through Software Inquiry Activity: Focus on using Scratch and physical computing materials," The Korean Association for Research In Science Education, Vol. 36, No. 2, pp. 325-335, March 2016.
- [14] Jinsoo Kim, "Study on SW Education and STEAM Education using Arduino for Technology Subject," The Korean Journal Of Technology Education Vol.15, No.1, pp.22-48, Apr.2015.
- [15] Hyeyoung Kim, "A Study of NCS(National Competency Standards) Curriculum Development in the Field of Information Technology," The Journal of Internet Electronic Commerce Resarch, Vol. 15, No.4, pp.85-99, Aug.2015.
- [16] Dong-Yeon Kim, "A Case Study on the Curriculum Development for the Specialization of Computer Department," Korean Technology Education Association for Research, Vol.15, No.2, pp.153-176, Aug.2015.
- [17] Raspberry Pi, <http://terms.naver.com/entry.nhn?docId=2454688&cid=42346&categoryId=42346>
- [18] Raspberry pie (Doosan encyclopedia), <http://terms.naver.com/entry.nhn?docId=3404406&cid=40942&categoryId=32829>
- [19] Raspberry Pi 3, http://www.icbanq.com/P007115245//?catg_code=120
- [20] Arduino (Doosan encyclopedia), <http://terms.naver.com/entry.nhn?docId=2835912&cid=40942&categoryId=32828>
- [21] Arduino Introduction, <http://www.hardcopyworld.com/ngine/arduino/index.php/archives/932>
- [22] ROKIT Smart Inventor Board, <http://rokitarduino.org/smartinventor/>
- [23] Smart Inventor Board, <http://rokitarduino.org/2015/06/25/smart-inventor-board/>
- [24] Raspbian Downloads, <https://www.raspberrypi.org/downloads/>
- [25] ROKIT BRICK Codrone, <http://rokitarduino.org/rokit-brick-codrone/>

Authors



Seong-Yeol Kim received the B.S., M.S. and Ph.D. degrees in Computer Science from Chosun University, Korea, in 1994, 1996 and 2000, respectively. Dr. Kim joined the faculty of the School of Information Technology at Ulsan College,

Ulsan, Korea, in 2002. He is currently a Professor in the School of Information Technology, Ulsan College. He is interested in information security, distributed system, sensor network, IoT, virtualization, cloud computing, and big data.