

자바를 이용한 마이크로 프로세서 시뮬레이터 개발에 관한 연구

김영민\*, 허 원, 기장근  
공주대학교 전기전자정보공학과

A study on the Development of a Microprocessor Simulator Using JAVA

Young-min Kim\*, Won Ho, and Jang-geun Ki  
Dep of Electrical · Electronic · Information Engineering, Kongju National University

**Abstract** - Educational software tends to be monotonous without the interaction between instructors and students. An unidirectional teaching system can't overcome these barriers because the learning process is monotonous without the bidirectional communication. The advanced network system can solve these problems.

In this study, a bidirectional microprocessor simulator is developed and a new teaching model is proposed using the simulator. It was named JMPS (Java Microprocessor Simulator), which is a software to teach the microprocessor principles and applications.

It has two advantages. First, it can teach the principles and applications of microprocessor effectively. Second, it runs on Internet and provides easy communication between a teacher and students.

**Key Words** : Microprocessor, Simulator, Network, JAVA, Educational Software.

1. 서 론

현재 여러 교육분야에서 컴퓨터를 이용한 교육이 이루어지고 있다. CAI 시스템은 일반적으로 컴퓨터를 이용한 학습 방법을 전체적으로 일컫는 말로서 지금까지의 CAI 시스템은 문자 중심의 화면 구성에서 오는 불편함과 진행방식의 단조로움 때문에 학습자를 지루하게 하는 단점이 있으며, 또한 교육자와 학습자간의 정보 전달 방법에서 한계를 드러내고 있다. 이와 같은 문제점을 해결하기 위하여 최근에 급속히 발전하는 네트워크 시스템을 이용할 수 있다. 네트워크 시스템의 기능을 활용하면 학습자와 교육자간의 양방향 정보 전달이 가능한 교육 시스템을 개발할 수 있다.

본 논문에서는 JAVA를 이용하여 개발한 JMPS(JAVA Microprocessor Simulator)에 대하여 설명하고자 한다. JMPS 시스템은 마이크로 프로세서의 한 종류인 PIC16C84의 교육용 시뮬레이터로서 네트워크 상에서 마이크로 프로세서 응용회로 설계, 저장, 전송, 시뮬레이션 결과 확인 등 개발에서 응용까지가 가능한 네트워크용 마이크로 프로세서 교육용 패키지이다. 또한 JMPS는 인터넷상의 홈페이지에서 실행 가능하며 웹브라우저로 접속하여 실행할 수 있다.

2. 본 론

2.1 JMPS의 개요

JMPS(Java Microprocessor System) 시스템은 마이크로 프로세서와 디지털 소자가 포함된 디지털 회로를 설계하고 이를 컴퓨터 상에서 시뮬레이션 할 수 있는 가상 시뮬레이터 시스템으로 JAVA 언어를 사용하여 개발되었다. JMPS의 특징은 일반 마이크로 프로세서 시뮬레이터의 기능 제공은 물론, 네트워크를 통한 운영이 가능하

도록 설계되어 학습자와 교육자간의 학습 정보 전달이 가능하다는 것이다.

기존의 CAD 도구와는 달리 실시간으로 시뮬레이션 결과를 확인 할 수 있고, 또한 LED나, 7-segment, 부저 등의 소자를 연결하여 시뮬레이션 할 수 있어 시각적, 청각적인 기능을 이용하여 교육적 효과를 높일 수 있는 장점을 가지고 있다.

2.1.1 JMPS의 구조

JMPS 시스템은 그림 1과 같은 기능적인 구조로 설계되었다. 그림 1에서 GUI 블록은 그래픽을 통하여 사용자와의 인터페이스를 담당하는 부분으로 이 인터페이스를 통해 SE(Schematic Editor)를 구동함으로써 사용자는 필요한 심볼을 생성하고, 생성된 심볼들의 입력포트와 출력포트를 상호 연결하게 된다. 여기서 심볼은 디지털 회로를 설계할 때 사용되는 모든 소자로 마이크로 프로세서인 PIC16C84와 AND 게이트, OR 게이트 등의 논리 게이트뿐만 아니라 TTL 칩 소자, LED, 7-segment 소자 등의 광범위한 소자들을 포함한다. ERC(Electric Rule Checker) 블록은 설계된 회로의 전기적인 규칙들을 검사하는 블록으로 에러가 발생하면 벨 소리와 함께 화면 하단에 에러 메시지를 나타낸다. 또한 에러가 없을 경우 각 심볼간의 연결관계를 설정해주고 이에 대한 정보를 CP(Circuit Processor) 블록에 넘겨줌으로써, 시뮬레이션시 CP 블록이 각 심볼의 입력 값에 따른 출력 값을 계산할 수 있도록 해준다. Clock 블록은 독립적인 thread가 수행하는 루틴으로 정해진 주기마다 클럭의 출력값을 0→1→0을 반복하여 발생한다.

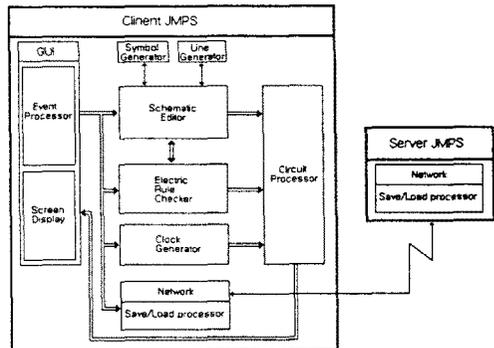


그림 1. JMPS의 구조

2.1.2 JMPS을 이용한 마이크로 프로세서 응용회로 설계 및 시뮬레이션 방법

JMPS에서 마이크로 프로세서를 이용한 응용회로 설계와 시뮬레이션하는 방법은 다음과 같다. 먼저 Schematic Editor에서 응용회로를 설계하고, FileEdit 창을 열어 어셈블러 프로그램을 작성한 다음, MPASM 컴파일러로 컴파일한다. 컴파일 과정에서 에러 메시지가 나오

면 ErrView 창을 열어 에러를 수정한 후, 다시 컴파일을 한다. 이와 같은 작업이 성공적으로 끝나면 Hex code를 PIC16C84의 프로그램 메모리 쪽으로 다운로드 한다. 그 후 ClockStart 버튼을 눌러 시뮬레이션을 시작한다. 이와 같은 과정은 그림 2와 같다.

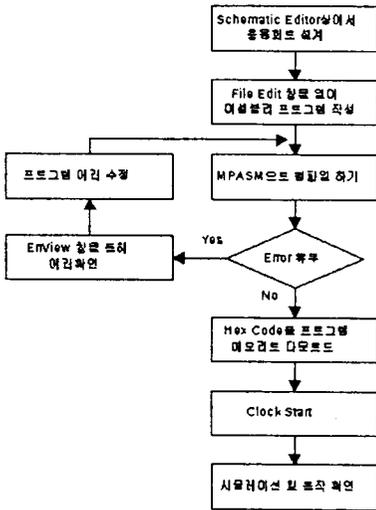


그림 2. JMPS의 시뮬레이션 과정

## 2.2 JMPS를 이용한 코스웨어의 구현

대학교의 마이크로 프로세서 응용 수업에서 실제 마이크로 프로세서(PIC16C84) 실험용 키트를 만들기 전에, JMPS를 이용하여 PIC16C84의 기본 구조와 동작원리를 학습할 수 있는 코스웨어를 아래와 같이 구현했다.

- ▶ 마이크로 프로세서(PIC16C84)의 동작원리 이해  
PIC16C84의 명령어 익히기(총 35개)  
PIC16C84의 명령어를 이용하여 LED Display 과정

- ▶ 응용 부분  
PIC16C84의 제어 레지스터 사용법 익히기  
PIC16C84의 제어 레지스터를 이용한 응용회로 설계

### 2.2.1 마이크로 프로세서(PIC16C84)의 동작원리 이해

마이크로 프로세서(PIC16C84)의 명령어 각각의 동작원리를 이해하고, 여러 개의 명령어를 사용하여 논리 연산, 수식 연산 등을 학습한다. 또한 PIC16C84에 LED를 연결하여 연산 결과를 표현할 수 있다.

- ▶ PIC16C84의 COMF, MOVWF, MOVLW, NOP 명령어를 이용하여 녹색 LED와 적색 LED를 반복해서 점멸하는 예제이다.

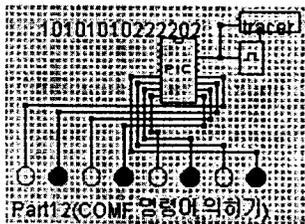


그림 3. COMF 명령어를 이용한 예제

그림 4는 PIC16C84의 Memory 내용과 실제 실행되는 번지를 확인할 수 있는 Memory Map window이다.

IR1=00000010000110	IR2=000000000000000	EXECaddr=-1	Wreg=11011001
RAM[0][0]=00000000	RAM[1][0]=00000000	RAM[0]=1683	RAM[1]=300f
RAM[0][1]=00000000	RAM[1][1]=11111111	RAM[2]=0085	RAM[3]=3000
RAM[0][2]=00001001	RAM[1][2]=00001001	RAM[4]=0086	RAM[5]=1283
RAM[0][3]=00011000	RAM[1][3]=00011000	RAM[6]=0186	RAM[7]=300f
RAM[0][4]=00000000	RAM[1][4]=00000000	RAM[8]=0086	RAM[9]=008c
RAM[0][5]=00000000	RAM[1][5]=00001111	RAM[a]=030c	RAM[b]=1d03
RAM[0][6]=11011010	RAM[1][6]=00000000	RAM[c]=2808	RAM[d]=0086
RAM[0][7]=00000000	RAM[1][7]=00000000	RAM[e]=2807	RAM[f]=0000
RAM[0][8]=00000000	RAM[1][8]=00000000		
RAM[0][9]=00000000	RAM[1][9]=00000000		
RAM[0][a]=00000000	RAM[1][a]=00000000		
RAM[0][b]=00000000	RAM[1][b]=00000000		
RAM[0][c]=11011010	RAM[1][c]=11011010		
RAM[0][d]=00000000	RAM[1][d]=00000000		
RAM[0][e]=00000000	RAM[1][e]=00000000		
RAM[0][f]=00000000	RAM[1][f]=00000000		

그림 4. Memory Map Window

- ▶ 입력 포트에 스위치를 달아 스위치의 ON, OFF 상태에 따라 PIC16C84의 출력 포트의 출력값을 제어할 수 있는 예제이다. 다음 예제는 PORTA.0 비트의 상태값에 따라 LED의 Display가 좌우로 변하는 제어회로이다.

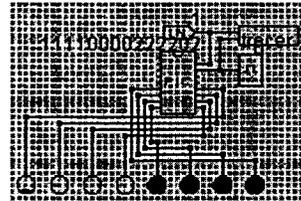


그림 5. PIC16C84의 출력 포트 제어회로의 예

### 2.2.2 응용 부분

응용 부분은 마이크로 프로세서(PIC16C84)에서 CPU의 기능과 주변 장치를 제어하는 제어 레지스터를 활용하여 여러 가지 응용 회로설계 및 프로그램을 학습하는 부분이다. 그 예는 다음과 같다.

- ▶ PIC16C84에서 PORTB를 전부 출력 포트에 설정한 다음, 각 포트에 LED를 연결하여 최상위 비트(MSB)에서 최하위 비트(LSB)로 Display하는 예제이다.

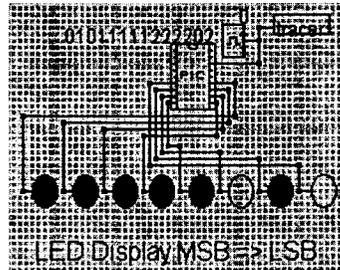


그림 6. MSB에서 LSB로 LED Display

- ▶ 그림 7은 일정한 시간 간격을 두고 신호가 바뀌는 신호등의 동작원리를 이해할 수 있는 예제이다. 그림 8은 PIC16C84의 출력포트의 TRACER 소자를 연결하여 시뮬레이션한 결과 파형이다.

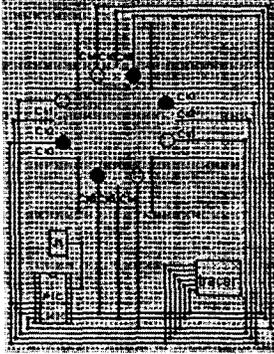


그림 7. 신호등 제어회로

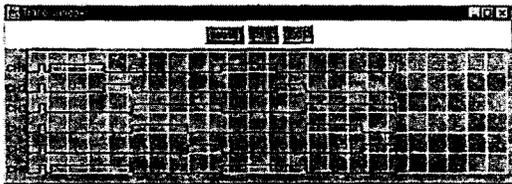


그림 8. 신호등 제어회로 시뮬레이션 결과

▶ PIC16C84를 이용하여 16진 Up/Down Counter의 설계에 대한 예제이다. 외부 입력 신호(PORTA.0)를 '0'으로 입력하면 0에서 F까지의 Up Counter로 동작하고, 반대로 외부 입력 신호(PORTA.0)를 '1'로 하면 F에서 0까지 Down Counter로 동작하는 회로이다.

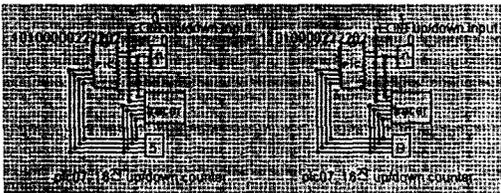


그림 9. 16진 Up/Down Counter 설계 회로

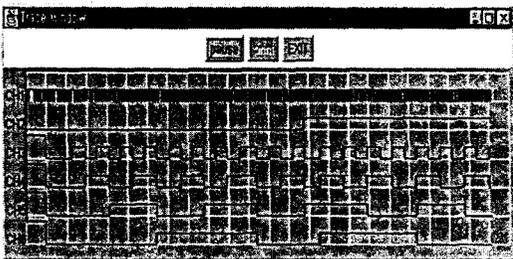


그림 10. 16진 Up/Down Counter 시뮬레이션 결과

그림 10에서 Ch2의 출력 과정에서 중간 지점에서 0에서 1로 변한 부분을 볼 수 있다. 이 부분에서 16진 Up Counter를 Down Counter로 변환시키는 부분이다.

### 3. 결 론

본 논문에서는 JMPS의 개발과 이를 이용한 WWW상에서의 대학교 1학기용 코스를 개발하였다. 이 코스를 이용하여 학습자는 인터넷을 통해 마이크로 프로

세서의 기본 개념부터 응용분야까지 학습 할 수 있다. 앞으로 네트워크 기능을 향상하여 JMPS는 학습자간의 대화식 교육을 가능하도록 개발할 것이다. 또한 JMPS를 바탕으로한 아날로그 회로 시뮬레이터의 개발과 다른 종류의 마이크로 프로세서의 기능의 추가를 목표로 기능을 개선하여 가고 있다. 궁극적으로 JMPS는 소자의 상태, 선 연결, 작성한 프로그램 등에 대한 객체화 된 정보를 가지고 학습자가 설계한 회로의 객체를 분석하여 학습 성취도를 판단할 수 있는 학습 종합 평가 시스템으로 개발하고자 한다.

### [참 고 문 헌]

- [1] Partrick Chan, Rosanna Lee, Douglas Kramer, "The Java Class Libraries Second Edition", Vol.1-2,1-2,2-1,2-2, Addison Wesley, 1998
- [2] 김형주 역, "객체지향", 교학사, 1998
- [3] 양오, "디지털 시스템 설계 및 응용", 복두출판사, 1998
- [4] Nataraj Nagaratnam, Brian Maso, Arvind Srinivasan, "JAVA Networking and AWT API Superbible", The Waite Group, 1996
- [5] 김종상, 이동호, "컴퓨터 구조", 회중당, 1992
- [6] Robert Sedgewick, "Algorithms in C++", Addison Wesley, 1992
- [7] 김광훈, 김영민, 허원, "JAVA를 이용한 변압기, 유도전동기, 교육 및 설계 패키지 개발에 관한 연구", 98년도 대한전기학회 하계학술대회 논문집 A, pp. 312~315, 1998
- [8] 허원, "공학교육을 위한 대화식 교육 시스템의 개발", 공학교육과 기술 제 4권, pp. 46~51, 1997
- [9] 김영민, 이종혁, 허원, "JAVA를 이용한 동기기 교육, 설계 패키지 개발에 관한 연구", 97년도 대한전기학회 추계학술대회 논문집, pp. 16~18
- [10] Won Ho, Joung-Huck Lee, "Development of an Interactive Education System for Engineering Education", ICEE, pp. 640~647, 1997
- [11] J.Hennessy, "RICS Microprocessor", IEEE Micro, p27, 1996
- [12] J.E Smith and G.S Sohi, "The Microarchitecture of superscalar processors", Univ.of Wisconsin, Aug 20, 1995
- [13] "Microchip DataBook", Microchip.co
- [14] <http://www.sun.com>
- [15] <http://jino.kongju.ac.kr>