

Web ベース 8085 マイクロプロセッサシミュレータの開発と その LMS への実装

村松一弘[†]

LMS (Learning Management System) は、多くの高等教育機関において講義、掲示板、レポート提出、小テストなどで利用されている。本論文では、8085 マイクロプロセッサトレーナーを利用した実習に焦点を絞り、LMS の機能を実験・実習に拡張する。具体的には、マイクロプロセッサトレーナーの動作を模擬する 8085 マイクロプロセッサシミュレータを PHP (Hypertext Preprocessor) 言語で開発する。PHP 言語で開発する理由は、Moodle (Modular Object-Oriented Dynamic Learning Environment) をベースにした本学の LMS が PHP で構築されているからである。次に、このシミュレータを本学の LMS に活動モジュールとして実装する。最後に、実際にマイクロプロセッサ関連の科目において、マイクロプロセッサトレーナーとマイクロプロセッサシミュレータを併用し、双方の比較検討を行う。その結果、シミュレータの方がプログラム入力簡単で使いやすく、ネットワークがあればどこでも利用できるが、ハードウェアが理解する上では実機としてのトレーナーも重要であるという評価が得られた。

Development of Web-based 8085 Microprocessor Simulator and Its Implementation on LMS

KAZUHIRO MURAMATSU[†]

At present, LMS (Learning Management System) such as Moodle (Modular Object-Oriented Dynamic Learning Environment) is used for lecture, tutorial, and assignment at many institutions of higher learning. Focused on practices of an 8085 microprocessor, in this paper, the function of LMS is extended to practices or experiments. Firstly, an 8085 microprocessor simulator is developed in PHP (Hypertext Preprocessor) language, because the Moodle-based LMS of our college has been built on PHP. Next, the 8085 microprocessor simulator is implemented on the LMS. Finally, the students of the college use both of the trainer and the simulator in practices of the 8085 microprocessor, and evaluate them. As a result, it is evaluated that the simulator is easy to use as compared with the trainer, and that it is usable wherever computer networks are accessible. However, the trainer is evaluated to occupy an important position to learn hardware of the microprocessor.

1. はじめに

近年、高等教育機関において LMS (Learning Management System) が普及しつつある。例えば、オープンソースソフトウェア Moodle (Modular Object-Oriented Dynamic Learning Environment) は 2012 年 12 月現在、世界中の 72,000 以上のサイトで約 64,000,000 のユーザが利用している[1]。また Sakai CLE (Collaboration and Learning Environment) は 350 以上の教育機関で利用されている[2]。本学でも Moodle 1.9 をベースとした VLE (Virtual Learning Environment) が 2011 年に導入され、講義ノートや実習ノートの公開、アナウンス、掲示板、レポート提出、小テストなどに活用されている。

一方、理工系の高等教育機関では実験・実習が重視されており、これらを通じて講義の理解を深め、実践的な技術を習得する。教育機関によっては、授業時間の半分近くが実験・実習に割られることもある。しかし、これらの実習装置や実験器具は、当然ながらその授業時間にのみ利用できるため、予習で使用したり、実験レポートを作成する際

に再実験をすることはできない。また、これらの実験・実習の装置や器具は多岐に渡り、その維持管理には多くの費用と労力を要する。

そこで筆者らは、前述の実習装置を模擬するシミュレータを開発し、これを LMS に実装することにより、ユーザが LMS 上で事前にこのシミュレータを使って実験・実習のプロセスを理解し、事後に実験・実習の復習を可能にする研究開発を始めた。さらには、授業時間中にこのシミュレータを利用して実習装置をシミュレータに置き換えることにより、実習装置の維持管理のコストを無くすることが可能かどうかを検討する。すなわち、LMS の機能を実験・実習に拡張することが本論文の目的である。

このような LMS 上でのシミュレータ開発には、電気工学の実験など複数の事例がある[3][4][5]。これらはシミュレータを開発してそれを教育現場に適用した事例である。本論文では、実習装置と同等以上の機能を有するシミュレータを開発し、それを実習に活用するだけでなく、実習現場において実習装置とシミュレータの双方を利用することにより、実習装置と比較して LMS 上でのシミュレータの長所・短所や有用性を明らかにする。

ただし、理工系の高等教育における実験・実習は多岐に渡るので、本学で開講されているマイクロプロセッサ関係

[†]王立ブータン大学科学工科大学カレッジ
College of Science and Technology, Royal University of Bhutan

の科目を対象を絞り、これらの科目の実習で利用されている 8085 マイクロプロセッサトレーナーを模擬する 8085 マイクロプロセッサシミュレータを開発する。そしてこのシミュレータを本学の Moodle 1.9 をベースとした LMS に実装し、マイクロプロセッサ系の科目の実習で活用する。

以下、§2 で 8085 マイクロプロセッサトレーナーの解説、§3 で 8085 マイクロプロセッサシミュレータの開発、§4 でマイクロプロセッサシミュレータの LMS への実装、§5 で学生による評価について述べる。§6 がまとめと今後の課題である。

2. 8085 マイクロプロセッサトレーナー

現在、本学のマイクロプロセッサ関連の科目でアセンブリ言語によるソフトウェア制御を習得するために利用されている実習装置は、Minmax Electronics 社製の 8085 マイクロプロセッサトレーナーである[6]。8085 マイクロプロセッサはインテルにより 1976 年に開発された 8 ビットのマイクロプロセッサで、現在では実用に供されていない。しかし、アーキテクチャが簡単でプログラミングの概念を習得するのに必要最低限の命令セットが用意されているので、教材としてはふさわしく、現在でも 8085 マイクロプロセッサに関する教育用書籍が出版されている[7][8]。

マイクロプロセッサトレーナーの外観を図 1 に示す。ユーザは、アセンブリプログラムをトレーナーに添付されているマニュアルを用いて 16 進数のマシン語に翻訳し、翻訳されたマシン語を 16 進キーボードで入力する。入力されたプログラムは、EXEC キーを押して一括実行される。あるいは、STEP キーを押すごとに 1 ステップずつ実行される。そして各ステップで、レジスタ、プログラムカウンタ、スタックポインタの内容が表示される。

3. 8085 マイクロプロセッサシミュレータ

前述のように、8085 マイクロプロセッサは優れた教材であるので、そのシミュレータは GNU Sim8085, Gsim85, 8085 SimuKit など多数存在する[9][10][11][12]。これらは Windows ベースあるいは Linux ベースで開発されており、Web プログラミング言語で開発されていないので、LMS との親和性は無い。そこで、筆者らは e-Learning システム Moodle 上に実装することを念頭に置き、PHP (Hypertext Preprocessor) 言語で 8085 マイクロプロセッサシミュレータを開発した。

開発に際して、以下の設計方針を採用した。

- 前節で述べたトレーナーは 16 進数のマシン語入力であるが、シミュレータではユーザが記述内容を理解しやすいアセンブリプログラムを入力する。
- アセンブリプログラムの入力ミスを低減させるため、入力があり得ない項目は入力不可能にする。



図 1 8085 マイクロプロセッサトレーナー

Figure 1 The 8085 Microprocessor Trainer.

- プログラム実行モードとして、RUN モードと STEP モードの 2 モードを用意する。RUN モードはバッチ処理であり、STEP モードはプログラムの 1 ステップごとに実行結果を確認できる。
- ユーザがタブレット端末で利用することを考慮し、キーボードを使用せず全てマウス入力にする。

開発した 8085 マイクロプロセッサシミュレータは、メインモジュール、実行モジュール、GUI (Graphical User Interface) モジュールから構成される (図 2)。

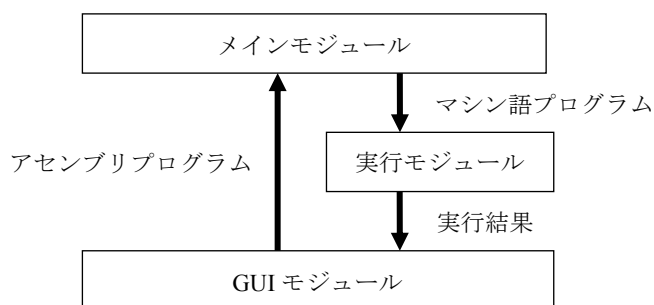


図 2 8085 マイクロプロセッサシミュレータのモジュール構成とデータの流れ

Figure 2 Modules and data flows in the 8085 Microprocessor Simulator.

以下に、各モジュールを解説する。

3.1 メインモジュール

メインモジュールは、GUIモジュールから送られたアセンブリプログラムをマシン語プログラムに翻訳する。例えば、メインモジュール中の以下のプログラムは、命令‘HLT’をマシン語‘76H’に翻訳する。

```

if(!empty($hlt)) {
    $mnemonic[$insert] = "HLT";
    $mc[$insert] = 0x76;
    $codeinput = "opcode";
    if($insert == $addr) $addr++;
    $insert = $addr;
}
    
```

翻訳されたマシン語プログラムをクライアント側のクッキーに格納する。クッキーのサイズ制限から、マシン語レベルで4096バイト以下のプログラムが入力可能である。4096バイトというサイズはアセンブリプログラムとしては約2000行の長さに相当し、教育用プログラムとして十分なサイズであると考えられる。クッキーに格納されているマシン語プログラムは、プログラムを実行する際に実行モジュールに引き渡される。

また、アセンブリコードを入力する際に、次に入力可能な項目の情報をGUIモジュールに渡す。前述のメインモジュールのプログラムにおいて、\$codeinput = "opcode"; がGUIモジュールに引き渡される情報である。

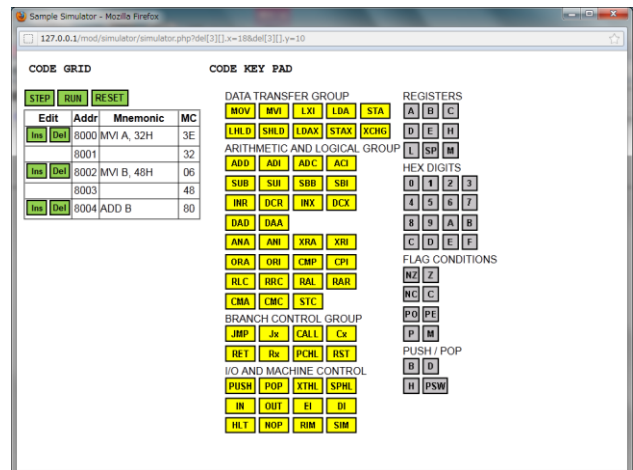
3.2 実行モジュール

実行モジュールでは、メインモジュールから渡されたマシン語プログラムを実行し、プログラムの各ステップでのレジスタ、プログラムカウンタ、スタックポインタなどの値をGUIモジュールに送る。実行モードには、ランモードとステップモードの2種類がある。ランモードでは、RUNボタンによりプログラムを一括して実行し、全てのステップでのレジスタなどの値をGUIモジュールに同時に引き渡す。一方ステップモードでは、プログラムはステップごとに実行され、各ステップでの実行結果がGUIモジュールに送られる。STEPボタンにより次のステップに進む。

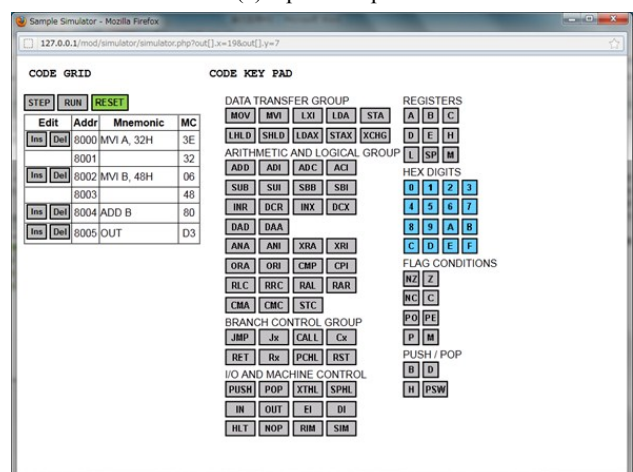
3.3 GUIモジュール

ユーザはGUIモジュールを通じてアセンブリプログラムを入力し、入力されたアセンブリプログラムはメインモジュールに引き渡される。図3に入力画面を示す。アセンブリコードは、処理の内容を指示するオペコードと処理の対象を指示するオペランドから構成される。図3(a)がオペコードの入力画面で、図3(b)がオペランドの入力画面である。ボタンの配置は文献[13]を参考にした。

図3に示すように、入力可能な項目に対応するボタンのみを色付き表示で有効にし、入力が不可能な項目のボタンを灰色表示で無効にすることにより、入力ミスを低減させる設計にした。GUIモジュールの中で入力可能な色付き



(a) オペコード入力
 (a) Opcode Input.



(b) オペランド入力
 (b) Operand Input.

図3 アセンブリプログラムの入力画面
 Figure 3 Input Windows of an Assembly Program.

ボタンの表示プログラムの一部を下記に示す。

```

<form action="" .
    htmlentities($_SERVER["PHP_SELF"])
    .'" method="GET">
<input type="image" src="img/MOV.bmp"
    alt="MOV" name="mov[]" />
<input type="image" src="img/MVI.bmp"
    alt="MVI" name="mvi[]" />
<input type="image" src="img/LXI.bmp"
    alt="LXI" name="lxi[]" />
<input type="image" src="img/LDA.bmp"
    alt="LDA" name="lda[]" />
    
```

一方、以下のHTMLプログラムは、入力不可能なボタンの表示プログラムの一部である。

```





    
```

またGUIモジュールは、実行モジュールからプログラムの実行結果を受け取り、それを表示する。図4に、実行モ


```
$string['instructions'] = 'Instructions';  
$string['timeavailable'] = 'Open from';  
$string['timedue'] = 'Open until';  
$string['entersimulator'] = 'Click here to  
run 8085 Simulator';
```

と変更すると、図 5 に示すように simulatorname が '8085 Simulator' , instructions が 'Instructions' , timeavailable が 'Open from' , timedue が 'Open until' と表示される。また、図 6 の起動画面でシミュレータへのリンクが 'Click here to run 8085 Simulator' と表示される。

(4) Icon Form

マイクロプロセッサシミュレータを表すアイコンを図 7 により作成した。これにより、教師がマイクロプロセッサシミュレータを登録した後、コース画面においてシミュレータの実行モジュールが図 8 のように表示される。

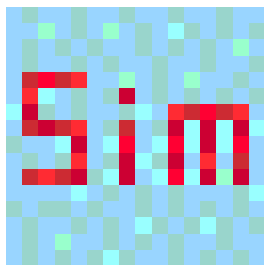


図 7 マイクロプロセッサシミュレータ実行モジュールのアイコン

Figure 7 Icon of Microprocessor Simulator Module

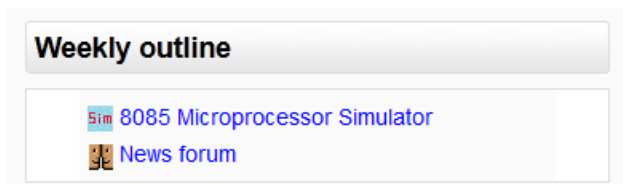


図 8 マイクロプロセッサシミュレータ登録後のコース画面

Figure 8 Course View after Registration of Microprocessor Simulator Module

5. 学生による評価

前節までのように、マイクロプロセッサシミュレータを開発し、本学の LMS に実装した。その上で、実際にマイクロプロセッサの実習に使用し、学生による評価を実施した。具体的には、21 名の学生で構成されるマイクロプロセッサの授業科目で、10 回の実習のうち 6 回で 8085 マイクロプロセッサトレーナーを使用し、残りの 4 回で 8085 マイクロプロセッサシミュレータを利用した。そして、それぞれの長所・短所があればそれらを列記させた。

その結果、3 名以上の学生が指摘したトレーナーとシミュ

レータの長所・短所は以下の通りである。トレーナーの長所として「ハードウェアが理解しやすい (11 名)」「マイクロプロセッサの周辺機器まで理解することができる (7 名)」「アセンブラプログラムを理解するのに有効である (3 名)」という回答を得た。トレーナーの短所としては、「アセンブラプログラムをマシン語に翻訳する必要があるため、時間がかかる (12 名)」「レジスタを表示させるのに手間がかかる (4 名)」「ポータブルではない、すなわち実習室でしか利用できない (3 名)」という回答であった。一方、シミュレータの長所では、「プログラム入力が簡単で使いやすい (17 名)」「ネットワークさえあればどこでも利用できる (8 名)」「出力結果が簡単に得られる (5 名)」という回答であった。シミュレータの短所としては、「周辺機器などのハードウェアの知識が得られない (14 名)」「ネットワークが不調の時に利用できない (5 名)」という意見が寄せられた。

シミュレータが使いやすいという意見が多かった。これは、トレーナーではアセンブラプログラムからマシン語への翻訳を学生がマニュアルを使って実施するのに比べて、シミュレータではアセンブラプログラムだけの入力だけで済み、学生の負担が少ないからであると考えられる。同様に、結果を表示させるのにトレーナーではやや煩雑な操作が必要になるのに比べて、シミュレータでは RUN ボタンあるいは STEP ボタンのマウスクリックで済み、この点でも負担が少ない。しかし学生の回答から、実機としてのトレーナーを使用することはハードウェアを理解する上で重要な位置を占めている。それゆえ、マイクロプロセッサトレーナーをマイクロプロセッサシミュレータで完全に置き換えることは不可能で、実習ではトレーナーとシミュレータを併用し、トレーナーで得られた結果を補助教材としてのシミュレータで確認するような利用が良いと考える。

6. まとめ

8085 マイクロプロセッサシミュレータを PHP 言語で開発し、Moodle をベースにした本学の LMS に活動モジュールとして実装した。そして、実際にマイクロプロセッサに関連した実習に利用し、これまで使用してきた 8085 マイクロプロセッサトレーナーと比較検討を行った。評価の結果、シミュレータの方がプログラム入力が簡単で使いやすく、ネットワークがあればどこでも利用できるが、ハードウェアを理解する上では実機としてのトレーナーも重要であるという結論が得られた。

今後は、シミュレータで入力したアセンブリプログラムを保存する機能、マシンサイクルの計算機能、出力ポートからのデータ表示機能などの機能拡張を行う予定である。

謝辞 Web ベース 8085 マイクロプロセッサシミュレータの開発に協力いただいた、王立ブータン大学科学工科カレッジ電気工学科卒業研究生の Anil Lama, Karma Dugyel, Lhazin Wangmo, Sangay Zangmo, Sonam Wangmo, Tshering Wangchuk, Sonam Dhendup の諸氏に感謝する。またマイクロプロセッサシミュレータの本学 LMS への実装に協力いただいた情報工学科スタッフの Yeshi Wangchuk 氏に感謝の意を表す。

参考文献

- 1) Moodle .org: Moodle Statistics, <http://moodle.org/stats>
- 2) Sakai Project: Collaboration and learning - for educations by educators, <http://sakaiproject.org>
- 3) Donzellini, G. and Ponta, D.: A Simulation Environment for e-Learning in Digital Design, *IEEE Trans. IE*, Vol.59, No.7, pp.3078-3085 (2007).
- 4) Fischer, W. and Lindemann, A.: Circuit simulation in a research oriented education of power electronics, *Proc. 11th Workshop on Control and Modeling for Power Electronics (COMPEL 2008)*, pp.1-5 (2008).
- 5) Ribeiro, T., Faria, N., Morgado, L., Simões, P., Rodrigues, P. and Leite, L.: Integrating a Military Air Traffic Control Simulator with an LMS, *Proc. Third International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES)*, pp.132-135 (2011).
- 6) MINMAX ELECTRONICS: MICRO-8085, <http://minmaxelectronicsindia.appspot.com/MICRO-8085.html>
- 7) Ganguly, K.A. and Ganguly, A.: Microprocessors and Microcontrollers 8085, 8086 and 8051, Alpha Science Int. Ltd., Oxford (2012).
- 8) Mathur, K.S.: Microprocessor 8085 and Its Interfacing, PHI Learning, New Delhi, 2nd edition (2011).
- 9) GNUM8085: 8085 Simulator for Linux and Windows, <http://gnusim8085.org>
- 10) GSim85: An 8085 Microprocessor Simulator, <http://gsim85.sourceforge.net>
- 11) Chehab, A., Hanna, S., Kabalan, Y.K. and El-Hajj, A.: 8085 microprocessor simulation tool "8085 SimuKit", *Computer Applications in Engineering Education*, Vol.12, No.4, pp. 249-256 (2004).
- 12) 8085 Simulator Project: 8085 Simulator Free Download, <http://8085simulator.codeplex.com>
- 13) Gaonkar, S.R.: Microprocessor Architecture, Programming, and Applications with the 8085, Prentice Hall, New Jersey, 5th edition (2002).
- 14) Moodle.org: NEWMODULE Documentation - MoodleDocs, http://docs.moodle.org/dev/NEWMODULE_Documentation
- 15) Moore, J. and Churchward M.: Moodle 1.9 Extension Development, Packt Publishing, Birmingham (2010).