

7 YEAC10 シミュレータの作成と改良

白澤 裕朔

指導教員 飯坂 ちひろ

1. テーマ設定理由

1.1 背景

YEAC10 実習機器の代用、改良していくたいと思
いこのテーマにした。

昨年の先輩が作成した YEAC10 シミュレータの改
良や機能追加をしていきたい。

1.2 目的

普段授業で扱っている Java の知識をより良くし
ていきたい。

YEAC10 の機能を理解し、使用者が利用しやすい
ように機能を追加する。

2. YEAC10 について

2.1 YEAC10 の仕様

表 1 YEAC10 の仕様一覧

語	長	8 ビット
		オペレーションコード 3 ビット
		アドレスコード 5 ビット
データの形式		命令語、データ 各 1 語
命 令 数		9 種類(LDA, STA, ADD, SUB, JMP, CMP, DIA, DOA, HLT)
メモリ容量		32 語
アドレス方式		1 アドレス方式
アドレス修飾		直接アドレス修飾
数 値		2 進
演 算 方 式		並列二進演算方式
ク ロ ッ ク		クロックは三種類 FAST (約 0.2 msec.) SLOW (約 1sec.) MANUAL (任意のタイミング)

2.2 YEAC10 の概要

教育用ディジタル電子計算機実習装置 YEAC10
は、コンピュータの基本である「繰り返し動作」
に基本を置き、計算機の仕組みと情報の流れ、情
報の処理をハードウェア、及びソフトウェアの両
面より理解するために設計されたコンピュータで
ある。

3. 研究概要

3.1 開発環境

Eclipse 3.6.1

IBM によって開発された統合開発環境 (IDE) の
一つ。高機能ながらオープンソースであり、Java
をはじめとするいくつかの言語に対応する。
Eclipse 自体は Java で記述されている。

3.2 初期画面

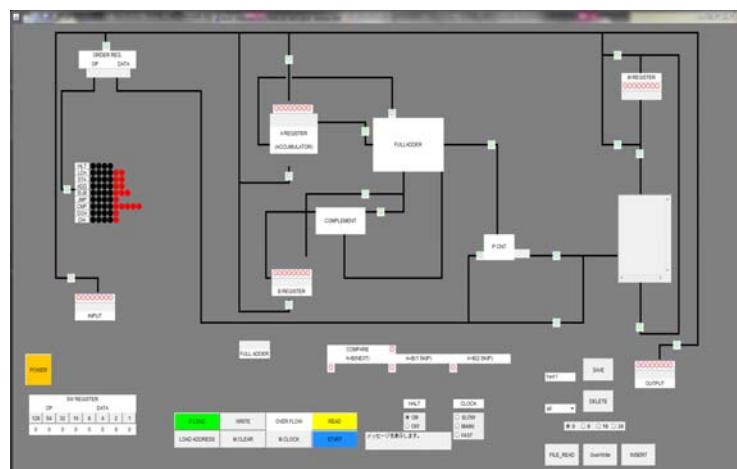


図 1 全体画面

画面作成では画面サイズ及び各ボタンサイズの変
更、またファイル保存、命令削除、ファイル読み
込み、命令上書き、命令文挿入の各機能用にボタ
ンを配置、メッセージ表示用テキストフィールド
を設けた。

3.3 各命令説明

表 2 命令一覧

命令	OP	動き
ロード	001	アドレス部で表示された番地の値を A レジスタにロードする。
ストア	010	A レジスタの値をアドレス部で示された番地へ格納する。
加算	011	A レジスタの値にアドレス部で示された番地の値を加えて、結果を A レジスタにロードする。 但し、加算結果が 255 を超えた場合はオーバーフローとなる。
減算	100	A レジスタの値にアドレス部で示された番地の値を引いて、結果を A レジスタにロードする。 但し、減算結果が 0 より小さくなつた場合はオーバーフローとなる。
分岐	101	アドレス部に示された番地に移動。
比較	110	A レジスタの値とアドレス部で示された番地の値を比較し、結果によりそれぞれ分岐する。 A レジスタの内容が大きい時… 次の命令を実行する A レジスタの内容と等しい時… 一命令をスキップする A レジスタの内容が小さい時… 二命令をスキップする

3.4 操作概要

操作の一例を以下に示す。

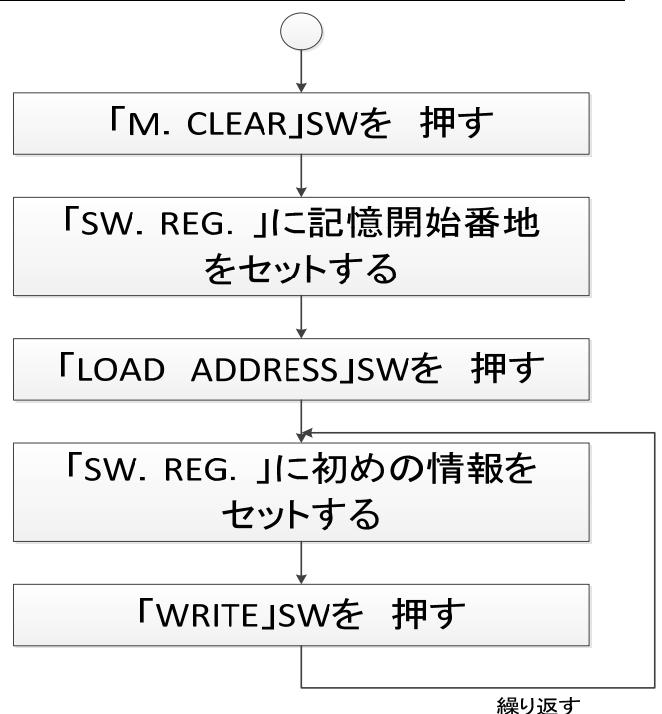
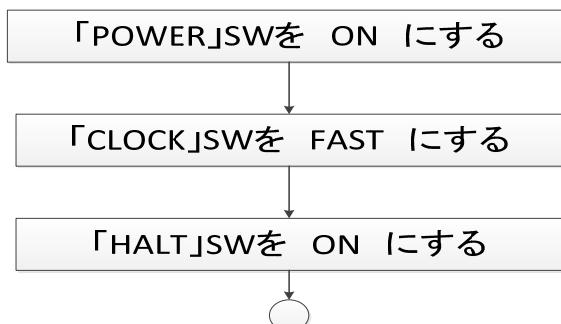


図 2 プログラム・データ等をメモリに書き込む

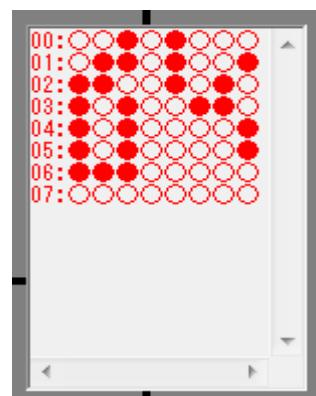


図 3 書き込まれたデータの例

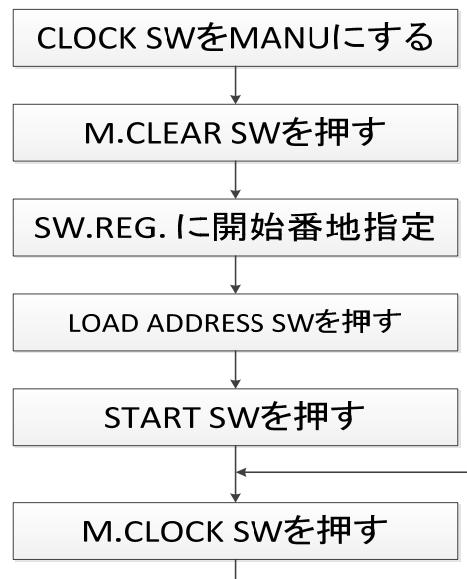


図 4 MANU で命令を実行する場合

3.5 変更点・改良点

- ・画面サイズ 1250×700 から 1920×1080 に変更
- ・ボタンのサイズ・各表示部分を画面サイズに合わせて変更
- ・ボタンカラー変更
- ・各テキストエリア、テキストフィールド、ラベルを書き変え不可に変更
- ・メモリ切り替えを可能にした。
- ・POWER ボタンで電源のオンオフを再現。



図 5 メモリ切り替えのイメージ



図 6 サイズ、カラー変更したボタン

3.6 追加機能について

- ・電源を落としてもプログラムのデータを維持できるように、保存・読み込みできる機能を実装する。
- ・プログラムの簡易的な手直しができるように、命令文の上書き、削除、挿入機能を実装する。
- ・プログラム動作時に簡易的なメッセージの表示をする。

以下に各機能の操作説明を示す。

・命令文保存

- SAVE ボタン左のテキストフィールドにファイル名（初期値は「test1」）を入力し、SAVE ボ

タンを押してファイルを保存。メモリ表示用データと処理配列用データをそれぞれ別ファイルに保存。



図 7 保存用ボタンとテキストフィールド

・命令文削除

→DELETE ボタンの左にあるコンボボックスから削除したい行を選択し、DELETE ボタンを押して指定した行だけを削除。all を選択時は全行削除。

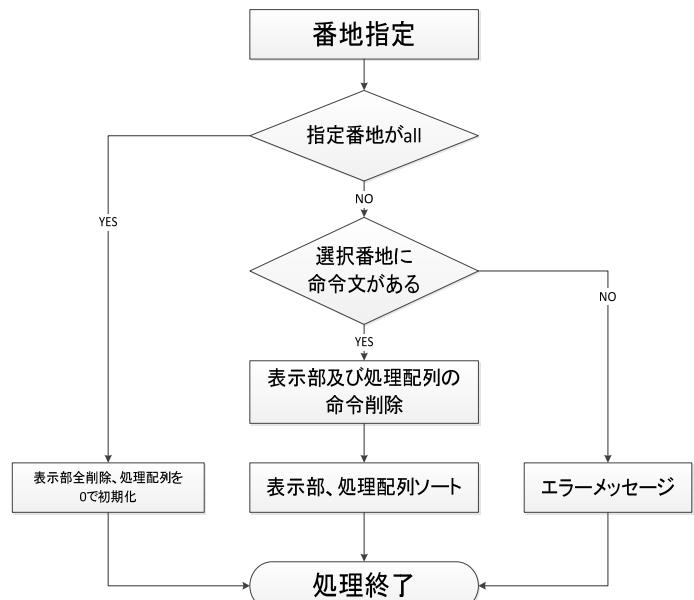


図 8 命令削除の流れ

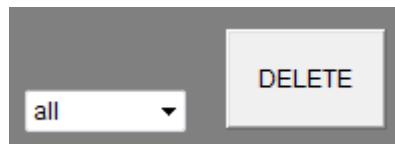


図 9 削除用ボタンとコンボボックス

・命令文上書き

→コンボボックスから命令を上書きしたい行を選択した後、上書きしたい命令文を SW REGISTER に書いて OVER WRITE ボタン押下で命令文の上書きをする。

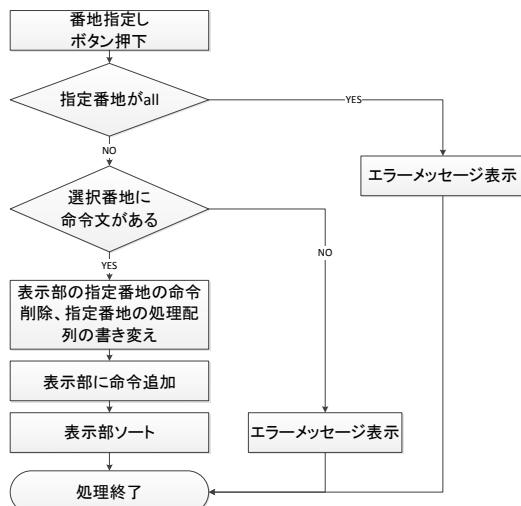


図 10 命令文上書きの流れ

・命令文の挿入

→挿入したい行をコンボボックスで指定した後、命令文を SW REGISTER に書き、INSERT ボタン押下し命令文を指定番地に挿入。

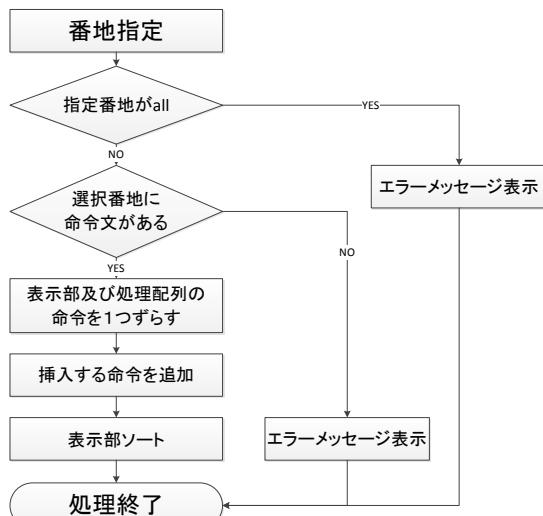


図 11 命令文挿入の流れ

・命令ファイル読み込み

→命令ファイル（命令文保存機能で保存したもの）のファイル名を SAVE ボタン左のテキストフィールドに書き込み、FILE_READ ボタンを押すことでメモリ部分のテキストエリアにプログラムの表示、処理用配列にデータの挿入する。

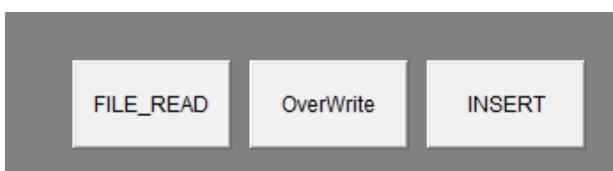


図 12 各追加機能用ボタン

・メッセージ表示

→例外が発生した場合、テキストフィールドにエラーの原因と思われるメッセージ、また一部機能完了メッセージを表示。以下に例を示す。

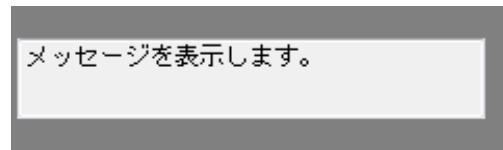


図 13 初期メッセージ

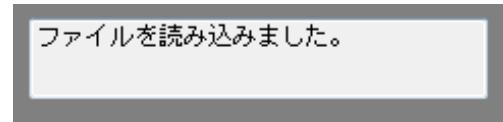


図 14 成功時のメッセージ

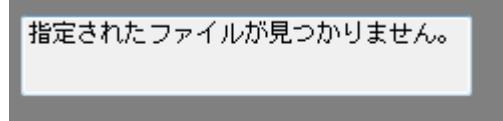


図 15 エラーメッセージ

4. 終わりに

追加機能としてはテーマ発表の際に発表した、データ保存機能、命令文の挿入・上書き・削除機能についてすべて完成させることができた。

また、メッセージの表示やボタンサイズ変更など、作成中に気付いた点を使いやすいうように改善することができた。

しかし一方で、ボタン配置等の関係で画面サイズを大きくしてしまった事により、モニタの大きさによっては対応できないという問題が発生してしまった。

5. 参考文献

やさしい Java

高橋 麻奈 著

JavaDrive

<http://www.javadrive.jp/>

テックノート

<http://javatechnology.net/>

Java AWT 入門

<http://wisdom.sakura.ne.jp/system/java/awt/index.html>