4V-2

アセンブリ言語学習シミュレータの開発

黒田順子　　早川栄一
拓殖大学　工学部　情報工学科
r28432@st.takushoku-u.ac.jp　　hayakawa@cs.takushoku-u.ac.jp

1.　はじめに

情報工学を学ぶにあたってプログラム格納式の理解は不可欠であり、機械語と１対１に対応したアセンブリ言語を学ぶことによってプログラム格納式の理解を深めることができる。

アセンブリ言語を学習する際、処理の仕方を理解するための効果的な方法としてトレースを行うという方法がある。トレースを行う上では、学習者自身でトレース結果をまとめた表を作成することによってトレースを行うこともできるが、シミュレータを利用することによって効果的に学習ができる。

しかし、一般的な学習シミュレータ[1]は、レジスタの値などの最小限の情報しか表示せず、それ以外の情報はコマンド入力などによって出力させることもできるが、過去のデータやプログラムは表示されないため、学習者がプログラム全体を把握しながらトレースの実行を行うことができない。

また、シミュレータからの一方的な情報提供しか行われず、学習者が本当に理解できたかを測ることはできない。そこで本研究では、本学で H8 アセンブリ言語の講義を履修している学生を対象として、学習者への情報提供に注目したシミュレータを開発・提供する。

2.　システムの特徴

本研究では次に挙げる二つの特徴を持つシステムを提供する。

(1)　必要に応じた情報の表示

全体のプログラムや過去のトレース結果といった学習者にとって必要な情報を常に表示できるようにする。
一般的なシミュレータでは実行直後のレジスタの値しか表示しないが、過去のレジスタの値の遷移を見たい場合もある。

そこで本システムではトレースの実行結果を元に実行ごとのレジスタの値をまとめたものをトレース表として表示し、トレース表にはレジスタの値だけでなく、メモリの値も表示できるようにする。さらに、本研究では実行中に１度もデータが書き換わらないレジスタなど、学習者やプログラムによって表示の需要が変わるレジスタなどの値について、表示/非表示を選択できるようにする。

また、アセンブリ言語ではループの明確な定義がないので、学習者がアセンブリ言語に慣れていないうちはループ外へ分岐する命令を書き忘れる、ループ外へ分岐する命令の参照レジスタがループ内で書き換えられていないといったことをしてしまい、無限ループに陥りやすくなる。

そこで本研究ではループ範囲を表示することで学習者がループに関するエラーをしても検出しやすくなるようにする。

(2)　テスト機能の提供

プログラムを実行してトレース結果を表示するだけでは、学習者が本当に理解しているかを確かめることはできない。そこで、本研究ではトレース結果に基づいたトレーステストを行うことによって、学習者が理解しているかを確認できるようにする。

3.　設計

本システムで提供するシミュレータの表示画面実行例を図 1 に示す。画面はⅠ:プログラム表示部、Ⅱ:初期値・表示設定部、Ⅲ:実行部、Ⅳ:メイン表示部から構成している。

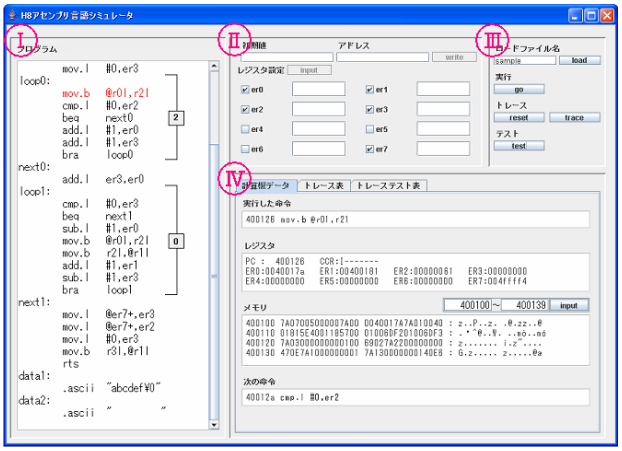

図１　表示画面実行例

3.1　プログラム表示部

ここではプログラムの表示を行う。

トレース実行中は直前に実行した命令について色を変えて表示する。3.4 の計算機部でも直前に実行した命令を表示する箇所はあるが、同じ命令が複数箇所に存在した場合、計算機部の表示からは命令の場所を限定できない。そこで、プログラム表示部でも実行した命令の箇所を明示することによって実行箇所を限定できるようにする。

また、プログラム表示部ではループ範囲の表示も行っている。プログラム内に繰返し処理をされている箇所[2]があれば、その部分をループ範囲として表示し、その範囲と処理した回数を表示する。

Development of a simulator for supporting assembly language learning.
Junko Kuroda , Eiichi Hayakawa
Faculty of Engineering , Takushoku University

3.2 初期値・表示設定部

ここでは実行するプログラムの初期設定及びトレースに表示するレジスタの指定を行う。

アセンブリ言語を用いた場合、学習者はプログラム内で初期値を設定するときにアドレスの指定をすることができないため、ラベルを利用しないと初期値を活用しにくい。さらに、レジスタの初期値はプログラム内では設定できない。そこで、本システムでは指定したアドレスやレジスタに初期値を設定する機能を用意する。

また、プログラム内では使用していないなどの理由からトレース表に表示する必要のないレジスタも存在する。そういったレジスタなどの表示/非表示もここで選択できる。

初期値・表示設定は、トレース実行前に設定する必要があり、トレース実行中に設定を変えても実行中のプログラムに反映させることができない。そこで、本システムではトレース実行中に初期値・表示設定をできないようにしている。

3.3 実行部

本システムを実行するための部分である。ここでプログラムのロード、実行、トレース、トレースのテストを行う。

また、トレース実行中に初期値を変えて最初からやり直すために、リセット機能を用意する。

3.4 メイン表示部

ここでは本システムでメインとなる“計算機部”、“トレース部”、“トレーステスト部”の表示を行う。これらはお互いを見比べて検討する必要がないので、タブで選択できるようにしている。この 3 箇所の設計を次に示す。

(1) 計算機部

本システムは計算機のシミュレータであるため、計算機としてどのような情報を保持しているのかを示す必要がある。

そこで、計算機部では次に挙げる実際の計算機でも保持する情報を表示する。

- 直前に実行した命令の表示
- レジスタの値の表示
- 指定したメモリの表示
- 次の命令の表示

(2) トレース部

トレース部ではトレース実行をした命令までの範囲でトレース表を作成し、表示をする。トレースを実行し、毎回値を確認することでもレジスタやメモリの状態遷移を理解することができるが、トレースを実行していくうちに過去の遷移を忘れてしまい、何度もトレースをしなおすということがある。

そこでトレースを実行するごとにトレース表を作成、表示させることで、1度の実行で過去の遷移を繰り返し確認できるようにする。

トレース表は命令、PC、CCR、レジスタを表示するが、3.2 でチェックを入れなかったレジスタについては表示されない。

また、トレース表内で書き変わった場所を学習者が見つけやすくするために、値が書き換わった箇所について色を変えて表示する。

(3) トレーステスト部

トレーステスト部では、トレース表を元に学習者がトレースを理解できたかを確認するためのテストを行う。

4. 実現

本システムの全体構成を図 2 に示す。

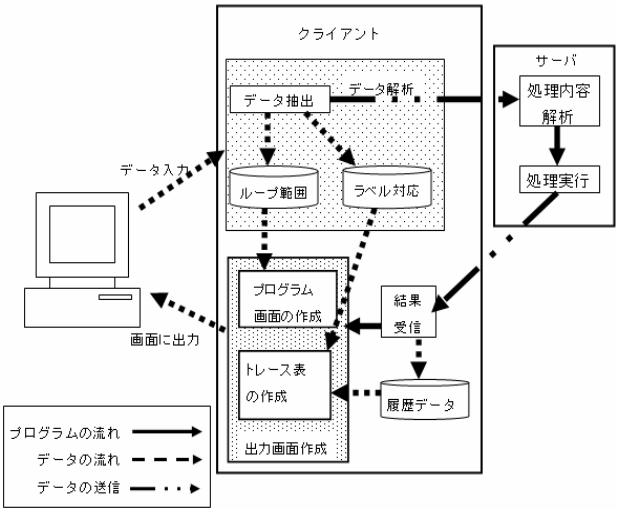

図 2 全体構成

本システムはメモリ・レジスタの管理と出力情報の管理を分割して行うためにクライアントサーバ方式を採用し、クライアント部は Java 言語で、サーバ部は C 言語で開発している。

機械語のプログラムデータ、レジスタやメモリのデータはサーバ側に保存されている。サーバ側はプログラムのロード、レジスタなどのデータを読み出す際に要求を送られる。

クライアント側ではそれらのデータを元に表示情報を作成し、表示をする。表示情報を作成する際、ロードされたプログラムからループ範囲と変数として使用されているラベルを決定し、表示するときにそれらのデータを元に表示する。

また、毎回トレースを実行する度にその結果を履歴として保存する。トレース表に出力する際に、履歴データを表に変換して表示させる。

5. おわりに

本稿ではアセンブリ言語の学習を支援するシミュレータについて述べた。本研究によってループ範囲の表示、トレース表の表示が可能になった。

今後の課題は次のとおりである。

(1) 色変え表示機能の実現
(2) ラベルをトレース表に反映させる

参考文献

[1] H8/300Hシミュレータ
http://home.t04.itscom.net/tokuse/pib/pH8.html
[2] 中田育男:“コンパイラの構成と最適化”朝倉書店 1999