

N-014

計算機アーキテクチャ理解支援システム
Computer architecture understanding support system

鈴木 裕也†
Yuya Suzuki

棚橋 純一‡
Junichi Tanahashi

1 はじめに

情報科学を学ぶ上で、計算機の構成とその振る舞いを関連付けて理解することは重要である。しかし、紙の上だけでは計算機の振る舞いをうまく表記することが困難であり、理解は容易ではない。

そこで、計算機を用いて紙面上では表わせない手法を採用することで、計算機アーキテクチャの理解を支援するシステムが作成できるのではないかと考え試作を行った。

2 計算機理解支援の現状と問題点

計算機は与えられた機械命令をフェーズと呼ばれる単位時間で順次処理することにより動作を行っている。

筆者の一人が行った講義「計算機アーキテクチャ」では教科書[1]のみでの理解は困難と考え、フェーズ毎の振る舞いを理解しやすくするため、機械命令の処理過程をまとめた表1を用意した。さらに計算機の装置と装置間のバスによるつながりを描いたデータバスと呼ばれる図1も用意し、両者を突合せて用いることにより、計算機の構成と振る舞いを理解させる方法を採用した。

表1 機械命令処理過程表 (一部)

命令長	命令名	F0	F1	F2	F3	F4	F5
1バイト命令	ALU	PC → D	D → MAR (M読出し)	M → MDR MDR → D PCINC	D → IRH 解読	Rs → R Rd → L	演算 結果 → D
1バイト命令	INC/DEC	PC → D	D → MAR (M読出し)	M → MDR MDR → D PCINC	D → IRH 解読	Rd → L 定数1 → R	演算 結果 → D
1バイト命令	SFT	PC → D	D → MAR (M読出し)	M → MDR MDR → D PCINC	D → IRH 解読	Rd → L	シフトセル
2バイト命令	BRCS (分岐不成立)	PC → D	D → MAR (M読出し)	M → MDR MDR → D PCINC	D → IRH 解読	PC → D	D → MAR (M読出し)

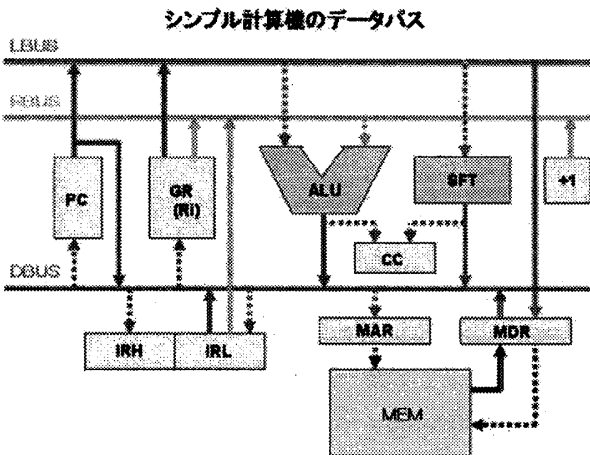


図1 データバス図

しかし、それだけの工夫を行っても生徒の理解は芳しくなかった。理由として、機械命令の処理は時々刻々と変化していくため、動きのない図と表だけでは理解しづらかったと考えられる。

3 計算機理解支援システムの実現

3.1 基本方策

上記の問題点を解決する方策として、アニメーションを取り入れることが考えられる。動きのない図と表を逐次変化させる処理の動きに対応させることで理解が容易になるのではないかと考え、理解支援システムの実現を目指すこととした。

3.2 必要な機能

理解支援システムとして必要な機能は次の通りである。

- (1) データバス図上に処理状態をアニメーションで表示
- (2) 多数ある機械命令から、表示したい命令を任意に選択
- (3) フェーズによる処理過程の理解を促すため、表示するフェーズをボタンで任意に選択
- (4) アニメーション表示だけでなく、同時に処理内容をテキストとして表示
- (5) 表示テキストから装置や機械命令の意味を容易に検索

3.3 一次試作システム

上記の機能を実現するシステムを Java により作成した。実行画面を図2に示す。動作手順は以下の通りである。

- ① 理解したい機械命令をリストから選択
- ② 選択した命令のフェーズに応じたアニメーション、テキストが自動表示
- ③ ボタンを押下してフェーズを進退させる
- ④ 違う命令を表示する場合は①に戻る

なお、表示されたテキストの特定の語句にはリンクが張られており、クリックすることで別ウィンドウに語句の内容詳細が表示される。

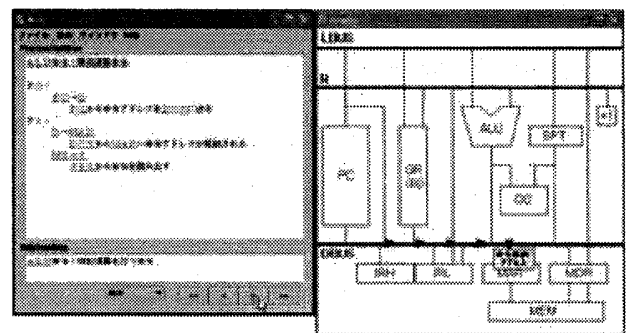


図2 一次試作システムの実行画面

† 中京大学 大学院 情報科学研究科

‡ 中京大学 情報科学部

今回の試作により、各機械命令の振る舞いをアニメーションとして表示し、どのような処理が行われているかを表現することはできた。

しかし、計算機アーキテクチャを理解するにはそれら機械命令の集まり、つまりプログラムとして組み込まれた一連の機械命令による動作を理解する必要があると考え、システムの発展に取り組むこととした。

4 計算機理解支援システムの発展

4.1 二次試作システムの構想

上記の考察に基づき二次試作システムでは、以下二点の機能を追加した。

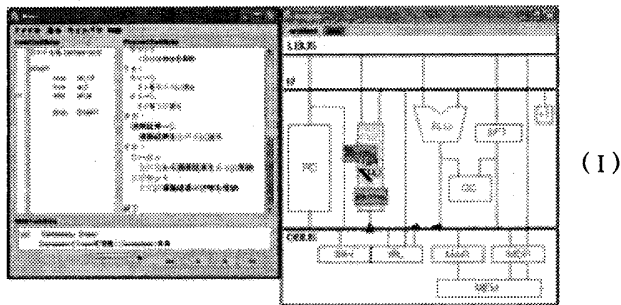
- (1) アセンブリ言語を読み取り、そのアセンブラ命令に対応した機械命令を表示する。ここで計算機構成は先のデータパス図を基礎とするため、使用するアセンブリ言語も単純なものとする。
- (2) レジスタ状況を処理毎に表示し、プログラム実行中のデータの遷移も表示する。

4.2 二次試作システムの実行過程

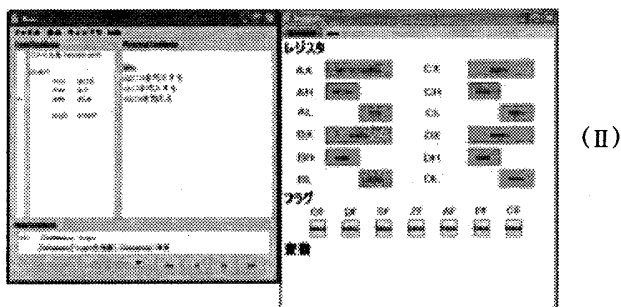
上記の機能を実現した発展システムの実行画面を図3に示す。動作手順は以下の通りである。

- ① ファイルからアセンブリ言語のソースを読み込む
- ② ボタンを押下することによりコードを1行(1命令)読み進める
- ③ アセンブラ命令に対応した機械命令を判読し、一次試作システムの方法で処理過程が自動表示(図3(I))
- ④ 機械命令の表示が終わるまでボタンを押下して進める
- ⑤ 変更されたレジスタ等の状態を自動表示(図3(II))
- ⑥ 上記②~⑤をソースの最後まで繰り返す

なお、③、④の機械命令表示は省略することができる。



(I)



(II)

図3 二次試作システム実行画面

また、一次試作と同様に表示されたテキストの特定の語句にはリンクが張られており、クリックすることで別ウィンドウに語句の内容詳細が表示される。

4.3 二次試作システムの評価

今回のシステムで取り上げた命令の種類は限定的ではあるが、プログラムから命令を1つずつ取り出して解読し、それに対応した機械命令の処理を一次試作と同様にアニメーション表示し、更にレジスタの内容がどう変化するかも表示することにより、実際の計算機の構成とその振る舞いを総合的に理解することが容易になった。

実際のプログラムを読み取りできるようにしたことで、使用者本人が作成したプログラムを用いて計算機の振る舞いを見ることが出来る。さらに、進行制御ボタンを実装したことで、プログラム進行の流れを自由に確認出来、計算機アーキテクチャの理解支援システムとして、効果を発揮すると思われる。

類似の研究例「Visu Sim」[2]はアセンブリ言語で作られたプログラムを実行する過程において、レジスタの視覚化を主体とした計算機教育支援ツールである。

しかし、機械命令については深く言及されておらず、システムとしても計算機の構成を書籍等で学習した上での利用を想定していると思われるため、計算機アーキテクチャを初歩から理解しようとしている人にとって使いやすい環境であるとは言い難い。

本システムは、機械命令からの段階的学習ができ広い範囲での理解ができる。また、語句の意味を表示するシステムなど初歩からでも利用しやすい理解支援システムとなったと考えられる。

5 今後の発展

今後以下の機能充実をはかってゆく予定である。

- (1) 解読できるアセンブリ言語を追加して、複雑な処理を表示できるようにする。
- (2) 現在は計算機内のデータの流れを追っているだけなので、ハードにおける理解支援を追加する。

6 謝辞

本研究を進めるにあたり、実現すべき機能などにつき意見をいただいた研究室のメンバーに感謝します。

参考文献

- [1] 富田真治：「コンピュータアーキテクチャ 基礎から超高速化技術まで」, 丸善株式会社 (2000年9月)
- [2] 今井慈郎, 富田真治, 古川善吾, 井面仁志, 白木渡, 石川浩, 宮武明義：「アセンブリ言語レベルの計算機教育支援ツール」, 第61回情報処理学会全国大会予稿集 (平成12年度後期)
- [3] 鈴木裕也：「計算機アーキテクチャ理解支援システム」, 中京大学情報科学部 2003年度卒業論文 (2004年1月)