

コンピュータの動作理解のための教育支援システム ——その教材としての評価——

香山瑞恵*・清水道夫*

An Application Software for Understanding the Behaviors of a Computer :
Its Development and Evaluation as Teaching Material

Mizue KAYAMA* and Michio SHIMIZU*

Abstract: We have made an application software for demonstrating the behaviors of a model computer proposed by Nakamura and Shimizu. We have been using this software for teaching computer literacy to beginners and have evaluated its usefulness.

As a result we found that this software was helpful to the learners in understanding the basic behaviors of the model computer and the roles of the elements of the model computer such as the adder and the address decoder. Our software turned out to be also effective in helping the learners to realize the necessity of programming in a thoroughly logical and efficient way.

1. はじめに

パソコン・リテラシー教育において、その教育方法や使用する教材に関して、様々な報告がなされてきている¹⁾。我々も、パソコン・リテラシーを習得させるための教材として、日本語ワードプロセッサや表計算、そしてプログラミング言語などを用い、その使用方法や有効性について、学習者からの授業評価などをもとに検討してきた²⁾。その中で、学生のリテラシー習得を支援するための教材の一つとして、コンピュータ・シミュレータの導入を考えた。

コンピュータの内部構造の理解を支援する目的

で開発されているシミュレータについては、いままでも多くの報告が出されている^{3,4)}。また、内部動作の理解支援を試みているシミュレータも報告されているが、それらの多くは内容的にも機能的にも専門性が高く、用いているモデル自体も高性能かつ多機能なものを想定していることが多い⁵⁾。現状では、初学者がコンピュータの動作原理に関する基本的な知識を習得できるようなものはあまりみられない。

そこで、今回、中村・清水が提案した3ビットのモデル・コンピュータ⁶⁾を使用して、コンピュータの内部動作に伴うデータの流れをCRT画面上で動的かつ視覚的に理解できるようなシミュレータを作成した⁷⁾。

このシミュレータをパソコン・リテラシー習得過程にある学生に教材として使用させて、その有

*〒380 長野市三輪8-49-7 長野県短期大学

*Nagano Prefectural College, 49-7 Miwa 8-chome, Nagano 380, Japan.

効性および使用法について検討したので報告する。

は、文献6)で詳しく説明しているのでここでは省略する。

2. シミュレータの概要

2.1 モデル・コンピュータとシミュレータ

図1はモデル・コンピュータである。これは、メモリ、アドレス・デコーダ、アキュムレータ、加算器、プログラム・カウンタ、命令レジスタ、命令デコーダの6つの要素とそれらを結ぶデータ・バス、コントロール・バス(制御線)そしてアドレス・バスの3つの経路を含んでいる。このモデル・コンピュータの動作原理と各用語の説明

今回のシミュレータは、このモデル・コンピュータで考えられている動作を実際の動きとして表現したものである(図2)。ここでは、特に、その動作に伴うデータの流れを可視化し、さらにその動作を理解し易いように説明を加えた。

なお、今回のシミュレータの開発に際しては、PC9801を動作環境とし、言語はBORLAND C[®] Ver2.1を用いた。また、システム全体のファイル・サイズは約64KBであった。

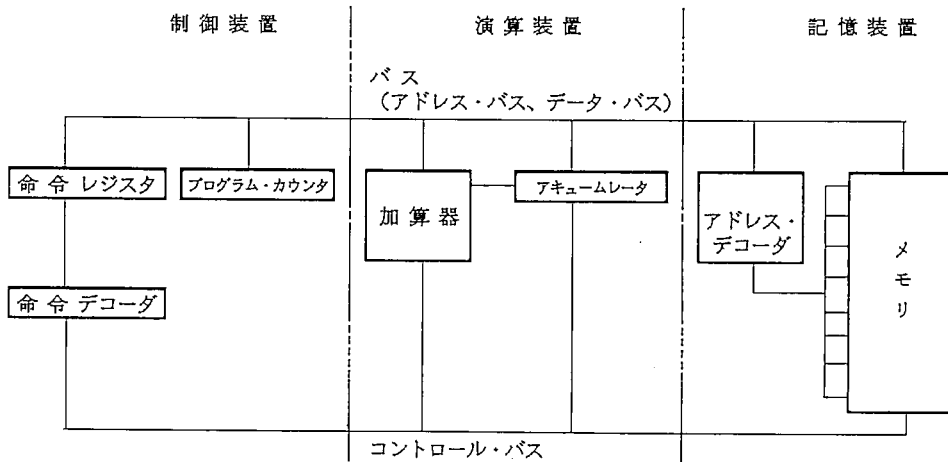


図1 モデル・コンピュータ
Fig. 1 Model computer (中村・清水 1988)

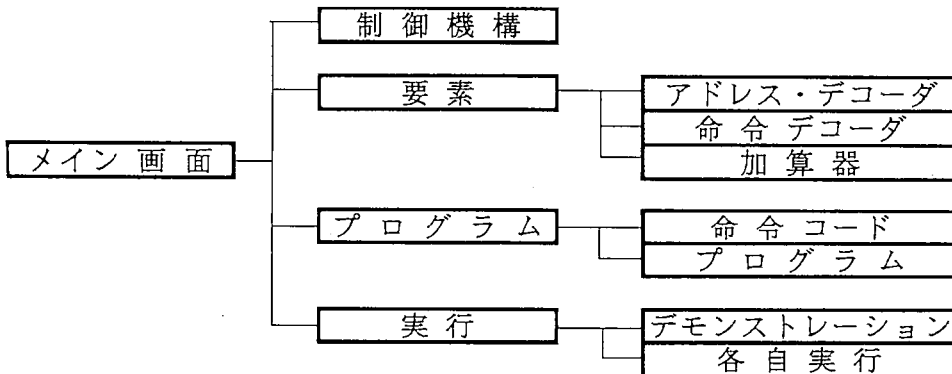


図2 シミュレータの構成
Fig. 2 Elements of application software

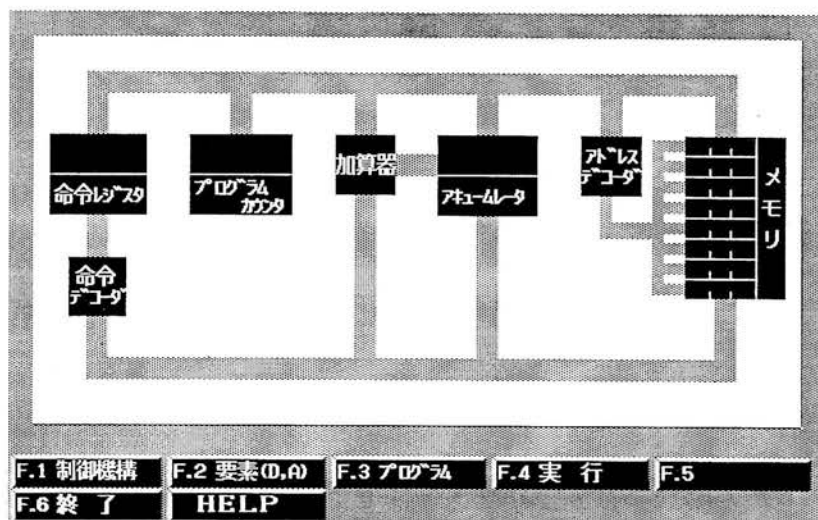


図3 メイン画面
Fig. 3 Main picture

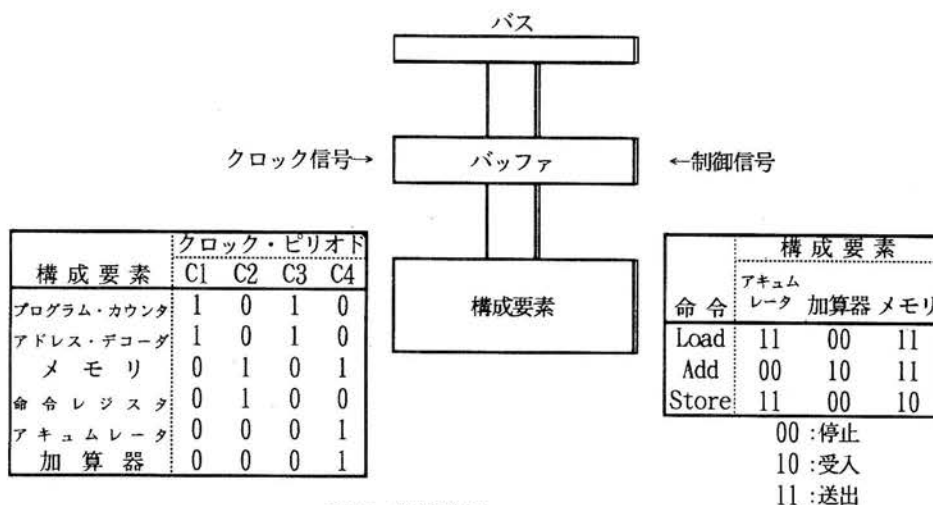


図4 制御機構
Fig. 4 Mechanism of operation

2.2 シミュレータの画面表示

シミュレータが呈示するそれぞれの画面とその持つ意味は次の通りである。

メイン画面：シミュレータ起動時には、まずこのメイン画面が表示される（図3）。以後、画面下部に示されている「制御機構」、「要素」、「プログラム」、「実行」の各メニュー項目への参照が終了

すると、再びこの画面が表示される。

「制御機構」：クロック・ピリオド毎の構成要素の状態、つまり各構成要素とバスとの接続状態を示す表と、命令毎のアキュムレータ、加算器そしてメモリの状態、つまり命令毎に決められている情報の送り手と受け手を示す表、とが表示される（図4）。

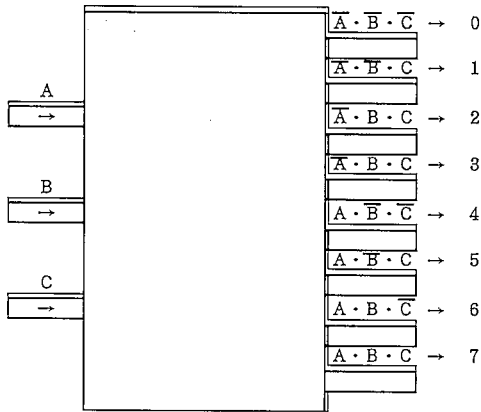


図5 アドレス・デコーダ
Fig. 5 Address decoder

「要素」：この画面は、『アドレス・デコーダ』と『命令デコーダ』、そして『加算器』の3種類の構成要素の説明を、サブメニューとして備えている。これらは、入力と、その入力に対しておこなわれる論理計算式と、その計算の結果の出力とを示している。図5は『アドレス・デコーダ』を示しており、『命令デコーダ』もこれと同様の論理で出力を得ることができる。

また『加算器』では、2つの数×3ビットの6つの入力から3つの出力を得るまでの過程でおこなわれる論理計算式が示されている(図6)。

「プログラム」：命令、そのコードおよび命令の具体的内容を示す『命令コード』と、2つの数の加算を実行するプログラム、およびそのプログラムがメモリ上に展開されたときの命令コードとそのメモリ番地を示す『プログラム』、との2つのサブメニューを備えている(図7, 図8)。

「実行」：あらかじめ組み込んである加算のプログラムを実行させる『デモンストレーション』と、加算する2つの数をユーザが指定できる『各自実行』との2つのサブメニューがある。そして、それぞれのサブメニューの実行に際しては一定時間毎にクロック・ピリオドが移っていく連続モードと、1クロック・ピリオド毎に停止していくステップモードとの2つの実行モードがある。

なお、このシミュレータでは、コンピュータによる処理の基準である1マシンのサイクルが4クロック・ピリオドからなっており、2クロック・ピリオド毎に命令の取出しと実行がおこなわれる。

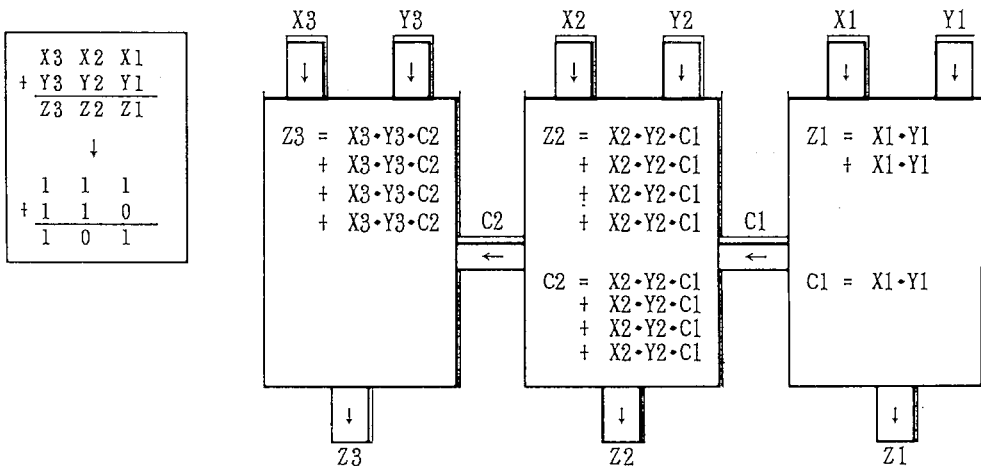


図6 加算器
Fig. 6 3 bits adder and its logical operation

命令	コード	具体的な内容
Load (転送)	0 0 1	メモリ番地の内容をアキュムレータに転送する
Add (加算)	0 1 0	アキュムレータの内容とメモリ番地の内容を加える
Store(格納)	0 1 1	アキュムレータの内容をメモリ番地に格納する
Halt (停止)	1 0 0	動作を停止する

図7 命令コード
Fig. 7 Example of program

C = A + B を計算する

A=5 , B=3 とした場合のプログラム

番地	メモリ	
0	001	Load A } 命令語1
1	101	
2	010	Add B } 命令語2
3	011	
4	011	Store C } 命令語3
5		
6	100	Halt } 命令語4
7		

図8 プログラム
Fig. 8 Operation code

また、プログラムの実行に際しては、構成要素、バスおよび制御線はクロック・ピリオドに伴い表示色に変化してくので、学生は加算動作を視覚的に理解することができる(図9)。

3. 教材としてのシミュレータの使用とその結果

3.1 対象者

このシミュレータを教材として使用した学生は、本研究室の1992年度のゼミ生である2年生7名で、全員女子学生である。使用した時期は1992年9月下旬から10月上旬にかけてであった。

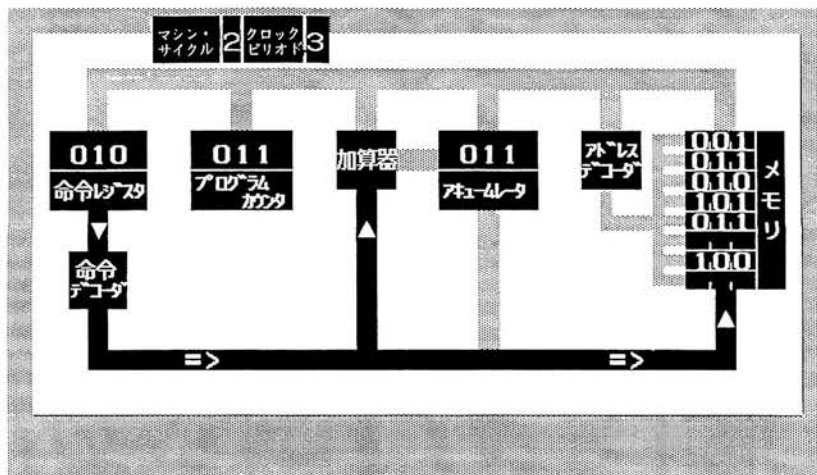


図9 加算動作のシミュレーション
Fig. 9 Simulation of addition behavior

この7名の学生は、1年次に開講されている2単位の情報処理演習Iで日本語文書処理や表計算を学習してきており、2年になってゼミ形式の情報処理演習IIを選択し、パソコン・リテラシー習得に対しては積極的な姿勢をもっている。また、シミュレータを使用するまでの学生の先行知識としては、アプリケーションソフトの活用方法、OSに関する基本的な事項、プログラミング言語の基本的知識、および基本的なアルゴリズムの概念などがある。

これらの学生に対して、シミュレータを演習教材として提供し、与えた課題を解決する過程における学生の反応を観察するとともに、シミュレータへの学生の評価などを求めた。

なお、本稿をまとめるに際して、1993年度に使用したゼミ生の意見も若干参考とした。

3.2 使用方法

今回計画したシミュレータの使用方法は、まず、シミュレータの機能と操作を説明し、その後コンピュータの内部動作を理解できるように課題を与えて解決させ、最後に学生のシミュレータに対する各自の評価や意見を求める、というものであった。具体的には下記の通りである。

(1) シミュレータの機能と操作の説明

学生に対して、最初にシミュレータの各構成要素の機能、およびマシン・サイクルとクロック・ピリオドによる制御のしくみ、つまり制御機構などについて説明する。そして次に、メイン画面下部に示されているメニューから目的とする項目を選択し、その項目と対応しているキーを押すことでシミュレータを操作できることを示す。

(2) アドレスの決定方法と加算の方法とに関する動作の確認

アドレスの決定方法については、3つの入力により0から7までの8つのアドレスが決定されていく過程とその結果とを確認させる。ここでは3つの入力に対する論理演算の結果として、それぞ

れのアドレスが決定されることを学生各自に理解させる。具体的には3つの入力を各々0, または1に変化させ、論理式が当てはめられている8つの出力線のうち、どこから信号が出されるかを確かめる。

また、加算の方法については、2つの数の和の計算がどのように実行されているかを各自に確認させる。すなわち、入力される2つの数は、そのどちらもが3つのビット毎に分解され、3ビット×2つの数で6つの入力信号に分けられる。その入力信号は各ビット毎に与えられる論理式によって計算がおこなわれ、結果を確認することができる。ここで与えられる論理式は図10に示す通りである。

(3) コンピュータの内部動作とその意味に関する課題、およびその解決

学生は、このシミュレータに組み込まれている1桁の整数の加算の実行過程を画面上で確認しながら、与えられる次の2つの課題を解決していく。1つは、クロック・ピリオド毎に、その時働いている要素と、その働きが送出または受入のどちらであるか、を考えることである。もう1つは、クロック・ピリオド毎のバス上にある情報の意味を

$$\begin{array}{r} X_3 \ X_2 \ X_1 \\ + \ Y_3 \ Y_2 \ Y_1 \\ \hline Z_3 \ Z_2 \ Z_1 \end{array}$$

<p>C1: 1位の桁上がり C2: 10位の桁上がり</p>

$$Z_1 = X_1 \cdot \overline{Y_1} + \overline{X_1} \cdot Y_1$$

$$C_1 = X_1 \cdot Y_1$$

$$Z_2 = X_2 \cdot \overline{Y_2} \cdot \overline{C_1} + X_2 \cdot Y_2 \cdot \overline{C_1} + \overline{X_2} \cdot Y_2 \cdot C_1 + X_2 \cdot Y_2 \cdot C_1$$

$$C_2 = X_2 \cdot Y_2 \cdot \overline{C_1} + X_2 \cdot \overline{Y_2} \cdot C_1 + \overline{X_2} \cdot Y_2 \cdot C_1 + X_2 \cdot Y_2 \cdot C_1$$

$$Z_3 = X_3 \cdot \overline{Y_3} \cdot \overline{C_2} + \overline{X_3} \cdot Y_3 \cdot \overline{C_2} + \overline{X_3} \cdot \overline{Y_3} \cdot C_2 + X_3 \cdot Y_3 \cdot C_2$$

図10 加算を表す論理式
Fig. 10 Logical expression for addition

考えることである。具体的には、メモリのアドレス番号となるプログラム・カウンタの値なのか、メモリから呼び出されたコードなのか、それともアキュムレータからメモリへ送られるデータなのかを考えることである。

ここでは、付録に示すような動作表を用いて、この2つの課題を解決させていく。

(4) シミュレータに関する評価や意見の記述

学生に対して、このシミュレータを用いた演習を通して理解できたこと、解らなかつたこと、およびシミュレータとして改善すべきことなどを自由記述形式で求めた。

3.3 シミュレータの有効性に関する学生の評価や意見

シミュレータを用いた演習の結果、理解できたこととして学生が記述した主なものを、内容別に整理すると次の通りである。

アドレス・デコーダの機能：

「ある2進数が与えられた時、アドレス・デコーダによって唯一のアドレスが決められること」(学生B)

「ある2進数が与えられた時、アドレス・デコーダでANDとORの組み合わせによって唯一のアドレスが決ること」(学生A)

「シミュレータを見ているだけでは解らなかつたが、アドレス・デコーダに3ビットの2進数が与えられた際に実際の値を入れて計算し、アドレスが決定されていくことを確かめたことで、どのようにして必ず唯一つのアドレスが決定されていくのかが解ったこと」(学生D)

加算器の機能：

「3ビット加算器によって2進数の和が計算されるしくみ」(学生B)

「加算器で、入力した数の計算がどのような手順でおこなわれているか、ということ」(学生G)

「加算器に具体的な数を入れて計算してみると、その結果は自分で出した答えと間違いなく合うこ

と」(学生D)

情報の送受と関連した内容：

「クロック・ピリオド毎に、情報の送受側と受入側が1ヶ所ずつあって、その他の要素はその時バスとは接続されていないこと」(学生A)

「要素がバスに接続されている時だけ、情報の送受と受入がおこなわれて、それぞれ1ヶ所ずつであること」(学生B)

命令とコンピュータの動作：

「メモリの中に命令とデータが対となって入っていて、命令は順次命令レジスタに送られ、命令レジスタを通してそれぞれ必要部分に働きかけることによってデータがアキュムレータへ送られたり、加算器へ送られたりするということ」(学生D)

「命令とデータは対となっていて、例えば命令はメモリの0, 2, 4, 6番地に入っていて、その命令にしたがってプログラム・カウンタで指定されるアドレスのデータをアキュムレータに送ったり、アキュムレータから受けたりすること」(学生C)

「LOADがメモリからアキュムレータへの転送で、STOREがアキュムレータからメモリへの格納であること」(学生C)

その他：

「結果を出すのにこれだけのことがなされているということ」(学生F)

「コンピュータは、0と1の組み合わせで必要な動作をおこなっているということ」(学生D)

「各構成要素の機能」(学生A)

「こういう形で加算動作を目にしたことで、6つの構成要素が連動して計算の答えを出していることを知り、今まで漠然としていたコンピュータの構造が解ったこと」(学生G)

「コンピュータの仕組みの大まかなところが、なんとなく解った」(学生F)

「本当になんとなくコンピュータについて理解

したという感じなので、もっと理解したい」(学生F)

4. シミュレータの使用とその有効性に関する考察

シミュレータを操作している学生の行動に関する観察や、シミュレータに対する学生の評価や意見などを通して、今回のシミュレータの使用方法の問題、有効性およびシミュレータ自体の問題点がいくつか明らかとなった。それらについて整理すると次の通りである。

4.1 シミュレータの使用法の改善

用語集の整備：今回のシミュレータを説明するのに使われている用語を解説した用語集が必要である。教員が当たり前として使っている用語の中には、学生にとって理解に苦しむものも少なからず含まれているようである。用語についての説明は使用開始時にふれる程度なので、その時点で「そのように言うもの」として考えてしまい、その後のシミュレータでの演習を通して、各々の用語の持つ意味などをそれ以上考えようとしなない。動作の説明よりも用語の説明を先にする方が適切であると考えられた。

補助資料の整備：画面で説明する内容に係わる資料が学生の手元に必要である。各構成要素やプログラムについての説明は、一画面毎にしか表示されない。そこで、学生の手元に補足資料があれば、ある画面の内容を、他の要素や説明項目などに関連づけて、もしくは比較して考えることができるからである。

なお、1993年度のシミュレータの使用に際しては、学生の理解をより容易にするために必要と思われる情報をあらかじめ紙面に用意した。これにより、例えば、画面上にプログラムが示されている時、その意味を確認するために各命令に関する情報を記した資料を参照することもできる。これは、1992年度の調査結果をふまえ、取り入れたものである。

その結果、紙面で補足的な情報を提供したことは、学生の理解を進める上でかなり有効であると考えられた。もちろんこれは、情報の提示にマルチウインドウ方式を採用すれば必要なくなることではある。しかし、実際の教育現場で学生が使用しているCRTの多くが画面の解像度が変更できない現状では、限られた画面の中にたくさんの情報をオーバーラップさせながら提示することは必ずしも有効ではない。また、学生の視認性の問題や担当者の開発効率から考えても紙面による補足情報の提示の方が実際的である。

課題説明の工夫：課題の与え方を十分に吟味することは当然であるが、説明にも配慮が必要である。例えば今回の課題の1つである動作表についていえば、これを完成させたことでコンピュータの動作が解った、と報告した学生もいた。しかし、全体的にはクロック・ピリオド毎のバスの情報を記入する動作表に対する説明が不十分であったためか、どういった観点でシミュレータを見て、どういった内容を動作表へ記入すればいいのか、すぐには理解できなかったようである。

教員の直接指導：コンピュータの動作原理の理解を支援するためにシミュレータを学生に提供する場合でも、教員は学生の質問や要請に対して、即座に回答できるような位置にいる必要がある。例えば、画面上のどこを特に注意して観察するか、といった指示をだすなどして、学生の理解をより容易にするような配慮が必要である。

4.2 学生の意見をもとにしたシミュレータの有効性

コンピュータの内部動作に対する理解：学生が自分で加算する数値を決められる「各自実行」では、「デモンストレーション」とは違った数値を用いた加算を実行することができる。そのため、自分が入力した数値が確かにメモリ内に書き込まれることに気づき、コンピュータの内部における動作を多少は理解できた学生もいた。

デコーダの機能：入力される値をあらかじめ与えられている論理演算にあてはめ、学生自身が実際に計算をおこなったことで、単なる機械的な作業によって出された結果の意味が、アドレス番地であったり、構成要素に対する制御であることに改めて気づいた、と報告したものは少なくない。なかには、「3ビットの2進数ABCが与えられた時、アドレス・デコーダによって唯一のアドレスが決定されるのを確かめて感動した」と書いた学生Eもいた。

加算器の機能：加算器の機能を視覚的に理解したことで、コンピュータがおこなう加算という動作と人間がする足し算という行動との違いに気がつく学生もいる。すなわち、人間が何気なくおこなっていることでも、加算器というハードウェアで実現するためには、多くの手順と処理が必要であることに気がつく。「2進数の和の計算はあのような手順をふんでいるのかと感心した」という学生Eの意見がそのよい例である。

コンピュータのアルゴリズムの認識：学生はプログラムを組む際に、文法的な矛盾がなく、能率的な手順で記述していくことを求められるが、どうしてそのようなことが必要なのか、に関しては明確な認識をもってはいない。しかし、今回シミュレータを使用して、加算のプログラムを実行させたことで、例えば「あのような簡単な加算でもあのような厳密な手順をふんで計算をしていて、しかもそれを瞬時にやっつけるということに感動した」と指摘する学生Eもいた。このことから考えて、このシミュレータはコンピュータのアルゴリズムを理解させる一つの有効なアプローチであり、プログラムの一つ一つの処理が文法的矛盾がなく記述され、実行能率が良いように並べられていなければならない、ということを確認させることができると思われる。

4.3 今回のシミュレータの改良すべき点

既に述べたような使用上の問題点に加え、シミュ

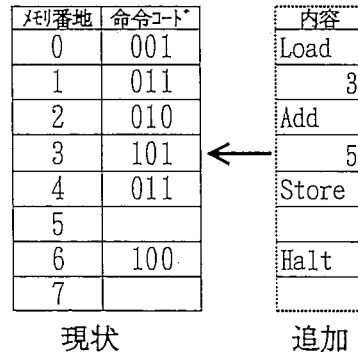


図11 プログラムの表示方法の改良
Fig. 11 Improvement of program demonstration

ュレータそのものについても改良すべき問題点が考えられる。その主な改良点は次の通りである。プログラムの表示方法の改良：アセンブラ形式の記述、つまり命令コードの意味を理解しやすい形で示した方がよい。例えば図11に示すように、

```
Load    5
Add     3
Store   □
Halt
```

などを組み込まれている加算のプログラムと対応づけられる形で示すならば、今どんな動作がおこなわれているのか、これからどんな動作がおこなわれるのか、といったコンピュータの動作に対する理解が容易になる、と指摘する学生もいたからである。

シミュレートできるプログラムの拡大：あらかじめ組み込まれている2つの数の和の計算プログラム以外のプログラムをシミュレートすることはできない。そのため、自分の作ったプログラムの働きを、各クロック・ピリオド毎に確認するような試みはできない。もし、このようなことができるシミュレータであれば、本学のような初学者のみを対象に使用するにしても、プログラムの動作を、単に実行された結果のみで把握するのではなく、その一つ一つの命令コードの意味とコンピュータ上での構成要素の動作とを合わせて確認でき、理

解が深まるということが考えられる。

日頃の学生の観察からも、パソコン・リテラシーの習得過程にある学生にとって、今使える命令が提示されている場合に、その命令の範囲内で、このコンピュータにとって文法的な矛盾がなく、実行能率の良いプログラムを発見していくような学習は、必要なことと考えられる。学生は、自分がコンピュータにさせたい内容をコンピュータのシンタックスに合うように矛盾なく記述することに困難を覚えることが多いことはこれまでに報告している⁸⁾。その主な原因は、コンピュータの動作原理を理解していないためであることが観察されており、こうしたシミュレータの使い方は、それに対する一つの解決策と考えられるからである。論理式の計算過程のシミュレートの実現：デコーダや加算器の説明画面で、信号の入力から出力にいたるまでにおこなわれる論理式の計算もシミュレートできる方がよいと考えられる。そうすることにより、コンピュータの内部で実行されていて通常は見えない計算過程を目の当たりにすることができる。その結果、学生は自分のした手計算の過程とコンピュータの計算過程とを比較することができ、単純な0と1の論理演算の組合せのみで計算が実現されていることが確認できる。

バスの表現方法の改良：シミュレート画面でのバスの表現形態と説明画面でバスからデコーダなどへの入力の表現形態との整合性を持たせることも必要である。図12に示す通り、メイン画面の中で1本の経路として示されているバスは、実際には数値を3ビットに分解し、3本の独立した線からしかるべき構成要素へ各々のビットを入力している。今回のデコーダの説明画面では、それら3入力をA, B, Cといった変数に置き換えている。そして、それらの変数が、デコーダの中でおこなわれる論理計算の式に用いられていることは、画面を見ているだけでは解りにくいようであった。

また、制御線の表示方法を改良することも必要

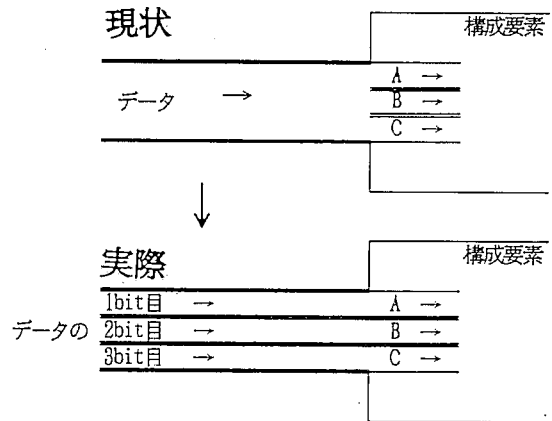


図12 バスの表現方法の改良
Fig. 12 On demonstration of bus

である。今回のシミュレータは、バスと制御線とが同じ形態で表示されていたので、次の動作の対象となる要素をアクティブにするという制御線の役割を理解しにくい学生もいたからである。実行モードの改良：任意のマシン・サイクルおよびクロック・ピリオドでの動作を確認できるようにしたほうがよい。それは、ある一つの動作を確認するために、マシン・サイクル1, クロック・ピリオド1から始め、目的のサイクルまでステップ実行していく場面が多く観察されたからである。画面操作の自由度の拡大：加算動作の実行をシミュレートしていく画面では、その途中でもシミュレートソフト自体を終了させることや、他の説明項目へ移ることができた方がよい。今回のシミュレータでは加算動作のシミュレート画面においては、一旦実行を始まるとマシンサイクル4, クロック・ピリオド4になるか、またはHalt命令がデコードされるまで動作を中断することができない。また、意図しない動作をさせてしまった際も取り消しすることができず、その動作が完了するのを待たなければならない。したがって、少なくとも一つ前の過程には動作途中からでも戻れるようにしておくことが望ましい。

5. おわりに

学生の評価にもある通り、今回のシミュレータは、コンピュータの内部動作やそれに伴うデータの流れが時間と共に変化していく過程を視覚的に理解する上で有効である。また、一部の学生に対しては、コンピュータのアルゴリズムに関心を向けさせる結果にもつながった。

しかし、既に指摘したような有効性をより確かなものとするには、シミュレータを使用しているときの学生の行動を詳細に観察し、それに基づいて教員が直接適切な指導をしなければならない。

また、どのようなシミュレータを設計し、どのように使用するかは、学生に何を理解させ、何を習得させたいかによって当然異なるので、教員は、シミュレータの設計およびその使用方法を十分に吟味する必要がある。すなわち、自分が進めようとするパソコン・リテラシー教育全体の戦略の中におけるシミュレータの位置付けを明確に意識していなければならない。

さらに、シミュレータ自体の改善すべき点は、既に述べたことのほかにもいくつかある。例えば、このシミュレータが3ビットマシンであるということ的前提とするならば、操作方法の改善点は次のようなことが考えられる。まず、画面上の知りたい・視たい場所で、知りたい・視たい情報が得られるようなキーボードレス・オペレーション機能を付加することである。具体的には、メニュー項目から個々の要素を選択することで表示していた各構成要素の動作説明を、要素を直接指し示すことで実現できるようにすることも、画面の内容を理解させるのに有効な手段ではないかと考えられる。

しかし、今回のシミュレータは、コンピュータ内部の「動作」を中心に開発されている。そのため、上記のようにキーボードレス・オペレーショ

ン機能を付加することや、実際のコンピュータの構成要素の機能に則した形で、このシミュレータの個々の構成要素を表現する方向に拡張していくのであれば、今回のシミュレータを核とした改良では不十分で、シミュレータ自体の再開発が必要である。すなわち、シミュレータの構成要素一つ一つの存在と役割、必要なデータなどを実際のコンピュータに則して表現できるような形で、シミュレータを開発する必要がある。

我々も既に、これまでシミュレータを教材として使用して得た経験および学生の評価や意見をふまえて、構成要素一つ一つをオブジェクトとして表現し、シミュレータ自体の再開発に着手している。その使用結果について別の機会に報告したいと考えている。

参考文献

- 1) 佐藤隆博：“教育情報工学入門”，コロナ社（1989）。
- 2) 香山，清水：“表計算ソフト習得上の問題点の分析”，CAI学会誌，Vol.10，No.2，pp.63～70（1993）。
- 3) 前田，辻野，都倉：“計算機構造の教育支援システムとその評価”，電子情報通信学会教育工学研究会報告，ET-89-149，pp.43～48（1989）。
- 4) 梶浦，宮地：“問診型エラー診断機能付き計算機シミュレータの開発”，CAI学会誌，Vol.9，No.3，pp.110～119（1992）。
- 5) 榊本保治：“CPU動作学習用CAI（1）”，電子情報通信学会秋季大会，1-157，（1991）。
- 6) 中村，清水：“教養のための情報科学入門”，近代科学社（1988）。
- 7) 清水，香山：“2値論理に基づくコンピュータ動作原理の教育支援システム”，日本教育工学会第8回大会，pp.418～419（1992）。
- 8) 香山，清水：“習得過程にある学生の表計算ソフトに対するイメージ”，CAI学会第18回全国大会，pp.275～278（1993）。

付 録

コンピュータの内部動作理解のための課題として用いた動作表等

氏名

[1]

クロック・ピリオド毎に、各々の構成要素がどのような状態になっているかを確認しましょう。

要素がバスに接続され、かつデータ送出的とき……………1

要素がバスに接続され、かつデータ受入のとき……………1'

要素がバスから切り放されているとき……………0

マシン・サイクル	1				2				3				4			
クロック・ピリオド	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
プログラム・カウンタ															—	—
アドレス・デコーダ															—	—
メモリ															—	—
命令レジスタ															—	—
アキュムレータ															—	—
加算器															—	—

[2]

各クロック・ピリオド毎のバスの上の情報について下の表を完成させます。

その時バス上にある情報を2進数で表し、その意味を記入してください。

マシン・サイクル	クロック・ピリオド	プログラム・カウンタ	バス上の情報	
			2進数	意味
1	1			
	2			
	3			
	4			
2	1			
	2			
	3			
	4			
3	1			
	2			
	3			
	4			
4	1			
	2			
	3			
	4			