

## コンソールパネルを持つ機械語教育用マイコンの 開発と授業への応用

重村 哲 至<sup>†,††</sup> 古川 達 也<sup>††</sup>  
相 知 政 司<sup>††</sup> 林 敏 浩<sup>†††</sup>

電気・電子・情報系の学生にノイマン型コンピュータの動作原理を教えるために、実機を用いた機械語教育を行うことが提唱されてきた。しかし、現在のコンピュータ・システムは教育用には高度で複雑すぎる。そこで、筆者らは、高専や大学において、機械語教育に使用する演習用のマイコンを設計・開発した。開発したマイコンが備える、内部を2進数で観察できるコンソールパネル、命令セットアーキテクチャ、入出力装置、クロス開発環境は、どれも教育用に配慮がされたものである。また、学生が個人で所有できるように小型・安価に実装した。実際に徳山工業高等専門学校の授業で4年間使用し、教育効果があることが判明した。

### Development and Application of Educational Microcomputer System to Instruct Machine Language Programming with a Console Panel

TETSUJI SHIGEMURA,<sup>†,††</sup> TATSUYA FURUKAWA,<sup>††</sup> MASASHI OHCHI<sup>††</sup>  
and TOSHIHIRO HAYASHI<sup>†††</sup>

To teach students the principle of the von Neumann-type computer, the machine language education with a real computer has been advocated conventionally. However, present computer systems are too advanced and complex to educate them. According to the demand for the higher education, the authors have designed and implemented an educational microcomputer system to satisfy the demand. The implemented computer has a console panel, that can be used to observe the inside, a set of instructions, I/O devices and cross development environments that are appropriate for education. Moreover, it has been implemented small size and at a low price so that the student might own it individually. It has been found out that there is a feasible education effect because of the practical usage in some classes for four years at Tokuyama College of Technology.

#### 1. はじめに

近年のコンピュータシステムは高度化・複雑化し、下位レベルの構造を隠蔽して学習を進めないと、目的の領域を理解することが難しい。そのため、高級言語を使用して学習し、ライブラリやOS、ハードウェアをブラックボックスにすることが多い。しかし、ハードウェア技術に立脚し計算機科学や計算機工学を学ぶ初学者にとって、ハードウェアとソフトウェアの境界

にある「ノイマン型コンピュータの動作原理」を正しく理解し、その可能性と限界を知ることが重要である。また、本原理を高専や大学の低学年時に理解することができれば、他の関連学習項目の基礎として活用できると考えられる。

動作原理を理解する手段として、機械語プログラミングを繰り返すことで、機械語プログラムの視点からコンピュータの内部動作を観察する方法が考えられる。機械語プログラミングの過程で、2進法による命令とデータの表現、プログラム内蔵方式、命令の逐次実行等、ノイマン型コンピュータの原理に関する理解が深まるものとする。本方法なら、論理回路やステートマシンの学習していない低学年の学生でも、コンピュータの動作原理に由来する可能性と限界を知ることが可能である。すでに、中西の報告<sup>1),2)</sup>、清水らの

† 徳山工業高等専門学校

Tokuyama College of Technology

†† 佐賀大学大学院工学系研究科

Graduate School of Science and Engineering, Saga University

††† 香川大学総合情報基盤センター

Information Technology Center, Kagawa University

報告<sup>3),4)</sup>でも、計算機の動作原理を教えるために機械語を教育し、成果があがっていることが報告されている。

機械語を学習するための演習環境として、シミュレータを用いる方法と実機を用いる方法が考えられる。シミュレータ<sup>5)-7)</sup>には(1)計算機の内部を詳細に可視化できる(2)手軽に利用できる(3)目的に応じアーキテクチャや抽象化のレベルを選んで実装できる等、優れている点がある。

一方で実機は「実物のランプやスイッチ」を用い2進数(デジタル回路)で動いていることを分かりやすく表現できる、外部に入出力装置を接続し入出力の意味を明確にできる、実行時間を正確に認識できる、実物が存在しハードウェアの実装イメージを実感しながら学べる等の点で優れていると考えられる。

また、次のように、実機がシミュレータに比較し不足している点は補うことが可能である(1)ブロック図等を用いた講義を行うことでデータパスの動作を説明できるし、実機でもコンソールパネルを実装し、レジスタやメモリの内容を表示することによって、機械語演習に必要な可視化は可能である(2)工夫によって、小型で手軽な実機を製作することが可能である。(3)FPGAを用いることによって、教育用に工夫した独自アーキテクチャを持つ実機の製作が可能である。

そこで筆者らは、ハードウェア技術に立脚した計算機科学を学ぶ学生には、実機の導入によっていっそう教育効果があがるものと判断し、以下の要件を満たした教育用のコンピュータを開発した<sup>8),9)</sup>。

要件1:実機であること 実機は、ハードウェアが2進数で動いていること、実行時間の認識、入出力命令の理解、ハードウェアを実感して理解できる点で優れていると考えられる。一方、シミュレータは、学生の視点から見れば、コンピュータの画面に表示される文字や図、アニメーションであり、その表示がコンピュータのハードウェアを表現していることが分かり難いところもある。

要件2:コンソールパネルを実装していること 初心者の学生が、機械語を実行しながらコンピュータの内部を観察するためには、デバッガやROMモニター等は適さない。これらは、ソフトウェアを通して間接的に内部を観察する機構である。ソフトウェアの層が入ることによって、ソフトウェアとハードウェアの機能を区別し難くなる。ハードウェアによって、内部を2

進数で観察できる、コンソールパネルが必要である。

要件3:必要な台数を確保できること 円滑な演習を行うために、受講者全員が同時にコンピュータを使用できる必要がある。予算と演習場所の確保が容易な、安価で小型なコンピュータが望まれる。

要件4:容易に理解できる単純な構成であること 低学年で計算機の動作原理を教えることが目的であり、実用的なことより、経験の少ない学生でも無理なく理解できることが重要である。実用的なアーキテクチャは、必要があれば、後修科目で学習すればよい。

中西らの報告<sup>1),2),10)</sup>で述べられた演習用コンピュータや、市販のワンボードマイコンは要件2を満たしていない。清水らの報告<sup>3),4)</sup>では、台数が確保できないためシミュレータを主に使用しており、要件1を満たしていない。

1990年代には神原らの報告<sup>11),12)</sup>や、末吉らの報告<sup>13)</sup>において、計算機の動作原理を初心者理解させることを目的の1つとした、教育用マイクロプロセッサが提案されている。しかし、クロック単位の実行や、FPGAによる学生自らのCPU実装等を考慮した構成であり、回路設計寄りの視点で考えられており本システムと目標が異なる。

最近では、新田らの報告<sup>14)</sup>において、本物の操作感を再現するシミュレータも提案されている。これは、実機を必要台数そろえ維持管理することが困難という前提で設計・実装され、受講者各自に動作環境を提供する当初の目的を達成しているが、要件1に示したハードウェアの理解には寄与していない。

今回、開発したマイコンは、内部をハードウェア制御で直接観察できる独自のCPUとコンソールパネルを備えている。また、小型・安価に実装し十分な台数を確保することが容易である。内部の構成は低学年の機械語演習をするのに必要最低限とし、理解も容易である。さらに、改良を加えながら、授業で4年間の使用実績がある。

本論文では、教育用に設計・実装したマイコンのコンソールパネル、命令セットアーキテクチャ、入出力装置、クロス開発環境、実装について述べ、最後に授業への応用例とアンケートの結果について報告する。

## 2. 機械語教育用マイコンの設計仕様

要件1~要件4を満たし、ノイマン型コンピュータの原理(以下最初の3つ)と、入出力、割込みを理解するために必要な演習環境を提供することを本マイコンの設計仕様とする。

(1)プログラム内蔵方式(2)逐次実行、

昔のコンピュータに装備されていた、ハードウェア制御でコンピュータ内部にアクセスできる、操作パネルのこと。

### (3) 2進法 (4) 入出力 (5) 割込み

割込みは、ノイマン型の原理を学習した後、さらに進んだ内容を学習できるように、あらかじめ仕様を含めた。また、「自宅学習でも使用できるよう安価かつ携帯可能なこと」、「演習効率を高くするためにクロス開発が可能なこと」も設計仕様とした。

## 3. コンソールパネル

コンソールパネルは、学生がコンピュータ内部を観察したり、データやプログラムを入力したりするために使用するユーザインタフェースである。

### 3.1 コンソールパネルの長所

ハードウェア制御で実現されたコンソールパネルには、ソフトウェアが介在しない。そのため、学生が、ソフトウェアの機能とハードウェアの機能を混同することがない。また、ランプやスイッチのONとOFFで情報を表現するので、コンピュータの内部が2進数で動いていることの理解や、メモリ内のビットイメージの体得が容易である。

### 3.2 データ・アドレスの桁数

マイコンは、学生が携帯できるよう安価で小型なものが望まれる。コンソールパネルには、データやアドレスの桁数分、トグルスイッチやランプを並べる必要がある。スイッチは高価な部品である。また、操作性の観点から左右の間隔を1.0cm程度とる必要がある。マイコンを安価で小型なものにするためには、スイッチの個数を、極力、少なくする必要がある。

コンソールパネルで16桁のデータとアドレスを扱うには、16個のトグルスイッチを横一列に配置する必要があり、部品の価格とサイズの両面で不利になる。そこで、8桁のコンソールパネルを実装することにした。そのため、マイコンのCPUも8ビットアドレス、8ビットデータにする。

### 3.3 コンソールパネルの構成

コンソールパネルの構成を図1に示す。ハードウェア制御で実現したので、プログラム実行中でもメモリやCPUの内部を観察できる。

操作性を向上させるために、押しボタンスイッチは電子的な操作音を出すようにした。また、長押しによるリピート機能もハードウェア制御で実装した。

各部の役割は次のとおりである。

(1) アドレスランプは、メモリをアクセスする際、メモリのアドレスを表示するランプである。

(2) データランプは、メモリ、またはレジスタの値を表示するランプである。プログラム実行中もリアルタイムで表示が変化する。

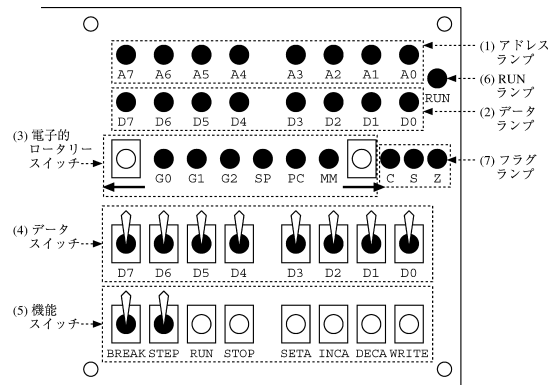


図1 コンソールパネル  
Fig.1 Console panel.

(3) 電子的ロータリースイッチは、データランプに表示するレジスタまたはメモリを選択するスイッチである。6個のランプのうちの1つが点灯し選択したものを表す。G0, G1, G2, SP, PC ランプは同じ名前のレジスタを、MM ランプはメモリを選択していることを表す。左右の押しボタンスイッチでどれを選択するか切り替える。機械式のロータリースイッチは背が高くマイコンの小型化に不利なため、このような構成とした。プログラム実行中も操作可能である。

(4) データスイッチは、データやアドレスの値を入力するための8個のトグルスイッチである。データスイッチにセットした値は、SETAスイッチを押した場合はアドレスランプに書き込まれ、WRITEスイッチを押した場合はデータランプに表示中のレジスタやメモリに書き込まれる。メモリに書き込んだ場合は、アドレスランプの値が自動的に増加する。

(5) 機能スイッチは左から順に、ブレークポイントモード (BREAK)、ステップ実行モード (STEP)、プログラム実行 (RUN)、プログラム停止 (STOP)、アドレスセット (SETA)、アドレスを進める (INCA)、アドレスを戻す (DECA)、書き込み (WRITE) の機能を持つ。RUN、WRITEを除き、プログラム実行中も操作可能である。WRITEは、操作ミスでプログラムの実行結果に影響を及ぼす可能性があるため、実行中は操作不可とした。

(6) RUNランプは、CPUがプログラムを実行中であることを表す。CPUが正しくない命令 (未定義命令) を実行した場合は、点滅して異常を知らせる。

(7) フラグランプは、CPUのC (Carry)、S (Sign)、Z (Zero) フラグの値を表示する。プログラムの実行中も表示が変化する。

### 4. 命令セットアーキテクチャ

コンソールパネルの制限から、8ビットデータ、8ビットアドレスの新しいCPUを独自に設計した。命令セットアーキテクチャは、学生が学習しやすく、かつ、一般的に用いられる他のCPUを学習するときの基礎となる知識が得られるように定めた。

8ビットアドレスではメモリ空間が小さすぎるため、教育用としても不十分ではないかとの懸念があった。そこで、徳山高専情報電子工学科の低学年で行われていたアセンブラ演習の課題を調べ、256バイトのメモリ空間でも演習に十分であることを確認した。

#### 4.1 CPU

CPUのブロック図を図2に示す。将来、高学年の学生が回路全体を理解できるよう、最小限の回路構成とした。設計と仕様変更の工数を節約するために、マイクロプログラム制御方式を採用した。

レジスタは8ビットの汎用レジスタ3本(G0, G1, G2)、スタックポインタ(SP)、プログラムカウンタ(PC)である。命令の実行結果で変化するフラグは、桁上げ(C)、符号(S)、零(Z)の3種類である。フラグに、割り込み許可ビット(E)を合わせた4ビットはPSW(Program Status Word)と呼ばれる。PSWは、メモリに退避・復元ができるようデータパス上に配置されている。

IR(Instruction Register)には実行中の機械語命令の第1語を格納する。IRとマイクロプログラム制御部の間に置かれるROMは、機械語命令と対応するマイクロプログラムの実行開始番地の対応表である。すべての未定義命令にエラー処理用のマイクロプログラムを対応させることで、ハンドアセンブラや打ち込みの間違いで書き込まれた未定義命令を、実行時に発見することができる。未定義命令を実行した場合は、CPUを停止し3.3節(6)で説明したようにする。

#### 4.2 命令体系

機械語命令は29種類、レジスタ-メモリ演算を基本としレジスタ間演算命令は用意しない。アドレッシングモードは、ダイレクト、インデクスト、イミディエイトの3種類、直交性の高い命令体系になっている。命令は、1バイトのものと2バイトのものがある。命令の形式を図3に、一覧表を表1に、1から10の合計を求めるプログラムの例を図4に示す。

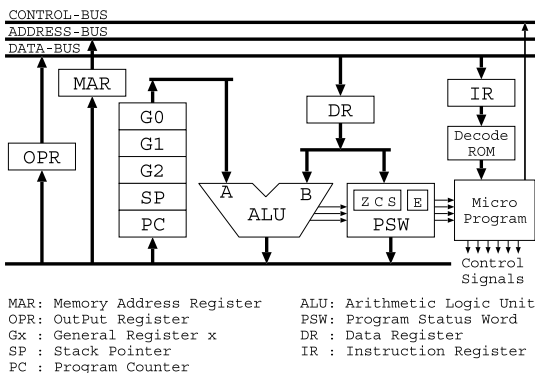
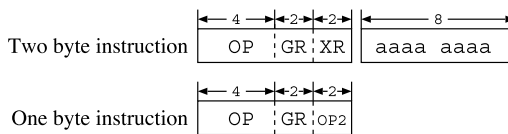


図2 CPUのブロック図  
 Fig.2 CPU block diagram.



OP(Ope-code)  
 GR(0:G0,1:G1,2:G2,3:SP)  
 XR(0:Direct,1:G1 Indexed,2:G2 Indexed,3:Immediate)  
 aaaa aaaa(Address or Constant)

図3 命令形式

Fig.3 Instruction format.

表1 命令表

Table 1 Instruction table.

Mnemonic	1'st byte		2'nd byte	Flags	States*	Operation
	OP	GRXR				
NO	0000	00 00	_____	—	3	No Operation
LD	0001	GR XR	aaaa aaaa	↑	5/7/7	GR <- [EA]
ST	0010	GR XR	aaaa aaaa	—	-7/7	[EA] <- GR
ADD	0011	GR XR	aaaa aaaa	↑	5/7/7	GR <- GR + [EA]
SUB	0100	GR XR	aaaa aaaa	↑	5/7/7	GR <- GR - [EA]
CMP	0101	GR XR	aaaa aaaa	↑	5/7/7	GR - [EA]
AND	0110	GR XR	aaaa aaaa	↑	5/7/7	GR <- GR & [EA]
OR	0111	GR XR	aaaa aaaa	↑	5/7/7	GR <- GR   [EA]
XOR	1000	GR XR	aaaa aaaa	↑	5/7/7	GR <- GR ^ [EA]
SHLA	1001	GR 00	_____	↑	4	GR <- GR << 1
SHLL	1001	GR 01	_____	↑	4	GR <- GR << 1
SHRA	1001	GR 10	_____	↑	4	GR <- GR >> 1
SHRL	1001	GR 11	_____	↑	4	GR <- GR >>> 1
JMP	1010	00 XR	aaaa aaaa	—	-5/6	PC <- EA
JZ	1010	01 XR	aaaa aaaa	—	4/5/6	if Zero PC <- EA
JC	1010	10 XR	aaaa aaaa	—	4/5/6	if Carry PC <- EA
JM	1010	11 XR	aaaa aaaa	—	4/5/6	if Sign PC <- EA
CALL	1011	11 XR	aaaa aaaa	—	-6/7	[--SP] <- PC, PC <- EA
IN	1100	GR 00	0000 pppp	—	8	GR <- IO[P]
OUT	1100	GR 11	0000 pppp	—	8	IO[P] <- GR
PUSH	1101	GR 00	_____	—	6	[--SP] <- GR
PUSHF	1101	11 01	_____	—	6	[--SP] <- PSW
POP	1101	GR 10	_____	—	6	GR <- [SP++]
POPF	1101	11 11	_____	↑	6	PSW <- [SP++]
EI	1110	00 00	_____	—	4	Enable Interrupt
DI	1110	00 11	_____	—	4	Disable Interrupt
RET	1110	11 00	_____	—	6	PC <- [SP++]
RETI	1110	11 11	_____	—	6	PC <- [SP++], EI
HALT	1111	11 11	_____	—	4	Halt

\*Number of states : Immediate / Direct / Indexed

1990年頃にCP/M上で行っていた演習課題を調査した。コンソールパネル関連の回路は省略してある。命令の第2語はアドレスまたは定数なので、未定義命令のチェックは第1語だけでよい。

Adr	Code	Label	Instruction
00	13 00	START	LD GO,#0
02	17 01		LD G1,#1
04	57 0B	LOOP	CMP G1,#11
06	A4 10		JZ STOP
08	24 13		ST G1,N
0A	30 13		ADD GO,N
0C	37 01		ADD G1,#1
0E	A0 04		JMP LOOP
10	20 14	STOP	ST GO,SUM
12	FF		HALT
13			
13	00	N	DS 1
14	00	SUM	DS 1

図 4 プログラム例

Fig. 4 Example program.

Memory Map		I/O Map	
Addr	Explanation	Addr	Read/Write
00   DB	RAM	0	Data-Sw/b0: Buzzer
DC	Interrupt Vector 0	1	Data-Sw/b0: Speaker
DD	Interrupt Vector 1	2	SIO-Data/SIO-Data
DE	Interrupt Vector 2	3	b7:Tx Ready, b7:Tx STI b6:Rx Ready/ b6:Rx STI
DF	Interrupt Vector 3	4	Available/Available
E0   FF	ROM(IPL)	....	....
		F	Available/Available

図 5 メモリ・入出力マップ

Fig. 5 Memory and I/O map.

### 4.3 メモリ空間

図 5 にメモリ・入出力マップを示す。メモリ空間は 1 語 1 バイト、全体で 256 バイトである。00H 番地から DBH 番地は自由に使用できる RAM 領域、DCH 番地から DFH 番地に割り込みベクタ、E0H 番地から FFH 番地には IPL (Initial Program Loader) ROM 領域が配置される。IPL に関しては、後述する。

### 4.4 I/O 空間

図 5 に示すように I/O 空間は全体で 16 バイトである。I/O 空間は IN 命令、OUT 命令でアクセスする。0H 番地から 3H 番地に、内蔵の入出力装置が配置される。4H 番地から FH 番地は拡張用領域である。

### 4.5 割り込み

4 種類 (INT0~INT3) の割り込みを使用できる。メモリ空間の割り込みベクタ 4 バイトが各割り込みに対応し、各々に 1 つの割り込み処理ルーチンを定義できる。

4 つの割り込みは、禁止/許可を DI/EI 機械語命令で一括して設定する。NMI 割り込みは用意していない。割り込み処理ルーチンでは PUSHF/POPF 機械語命令を用い、PSW の退避/復元をする必要がある。

## 5. 入出力装置

入出力装置を演習で効果的に使うことができれば、文字コードや実行時間等を教えるためにも、また、学生の興味を引き出すためにも役立つ。

### 5.1 コンソールパネル

コンソールパネルを入出力装置として使用することができれば、マイコン本体だけで入出力の演習が可能になる。しかし、コンソールパネルに 2 つの役割を持たせると、意味が分かり難くなる危険性がある。

まず、アドレスランプとデータランプのどちらか一方を、プログラム実行中は出力 (表示) 装置として使用することを検討したが、上記の理由によって実現しなかった。逆に、プログラム実行中もレジスタやメモリの値をコンソールパネルに表示し続けることで、OUT 命令を習っていない学生も内部状態を実時間で確認できるので、動作原理の理解に役立ち教育効果の向上が望めると判断した。

一方、データスイッチは、プログラム実行中コンソールパネルの役割で使用することはない。そこで、混乱は少ないと判断し、データスイッチの値を IN 命令で読み出せるようにした。これによって、本体だけで IN 命令の簡単な演習をすることが可能になった。

### 5.2 シリアルポート

文字コードの概念を勉強するために、文字を使用するユーザインタフェースが必要である。しかしマイコンのサイズとコストを考慮すると、本体に文字入出力用のユーザインタフェースを実装することは困難である。そこで、マイコンとシリアルケーブルで接続したパーソナルコンピュータ (PC) を、文字端末として使用することにした。

シリアルポートの制御は、ポーリングと割り込みのどちらにも対応できる設計とした。

### 5.3 スピーカ

OUT 命令の初歩的な演習を行うために、プログラムで制御できるブザー音源とスピーカを実装した。

また、ブザー音源とは別に、スピーカに直接接続したスピーカ・ポートも用意した。これは、プログラムの実行時間を意識させるための演習に使用できる。プログラムが、スピーカ・ポートに出力する '0'、'1' を一定間隔で変化させることで、時間間隔に応じた高さの音がスピーカから発せられる。学生は、表 1 に示した機械語命令のステート数を数えて、実行時間を調整する。

## 6. クロス開発環境

機械語プログラミング演習の効率を良くするために、PC上で動作するクロスアセンブラと、機械語プログラムをPCからアップロードする機能を準備した。

### 6.1 クロス開発環境の必要性

学習を始めた初期の段階では、機械語をハンドアセンブルで作成しコンソールパネルから打ち込むことで、メモリ内のビットイメージをしっかりと認識することが重要である。しかし、ビットイメージを理解できた後では、このようなプログラミング環境は手間がかかりすぎる。

「機械語プログラマの視点からコンピュータを観察する」ためには、たくさんのプログラムを作成し十分なプログラミング経験を積むことが必要である。プログラム作成を効率良く行うための環境も必要になる。

### 6.2 クロスアセンブラ

クロスアセンブラを、種類のPCで実行できるように、Javaで記述した。ハンドアセンブルと本質的に同じであることが分かりやすいように、絶対番地形式の機械語を直接出力するようにした。また、図4に示すアSEMBルリストを生成し、ハンドアセンブルと同じであることを確認できるようにした。

### 6.3 アップロード環境

機械語プログラムをPCからアップロードするために、マイコンにシリアルポートとIPL、PCにアップロードプログラムを準備した。

IPLは、シリアルポートから機械語プログラムを受信する簡単なプログラムであり、メモリ空間の最後の部分32バイトに格納される。IPLはシリアルポートから、格納アドレス、プログラム長、機械語をバイナリ形式で受け取る。

## 7. 実装

完成したマイコンを図6に示す。マイコンは1枚のプリント基板上に実装することができた。基板の大きさは、VHSビデオカセットと同じサイズにし、市販のビデオカセットケースに収納するようにした。基板の右半分にコンソールパネル、左半分にランプのドライバ回路、電源回路、シリアルポート、拡張ポート、FPGA等を実装した。

マイコンのCPU、メモリ、周辺回路等のロジック回路は、FPGAチップ1つに集積できた。FPGAの設計にはVHDLを用い、約2,000行の記述量だった。

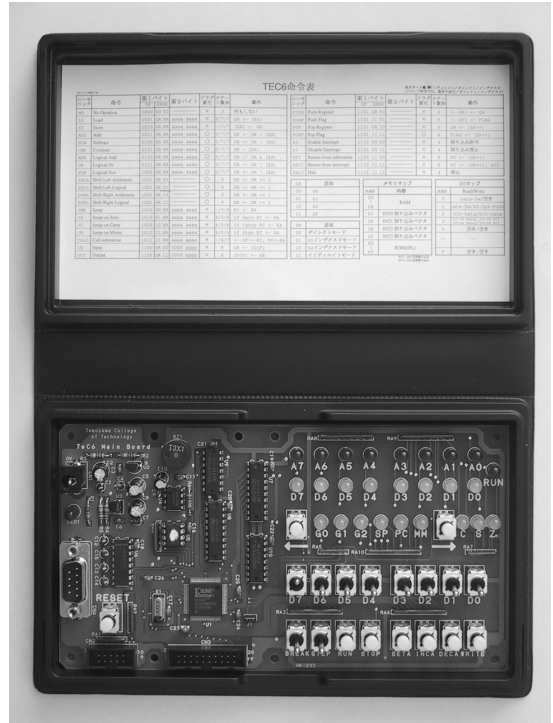


図6 教育用マイコン

Fig. 6 Educational microcomputer.

ケース蓋の裏に命令表が貼りつけてある。本体とケース、それにACアダプタを持ち歩けば、学校でも家庭でも機械語の演習が可能である。

1台あたりの部品代も安く、学生が個人的に購入可能な金額である。安価なので、従来のように必要台数の確保で悩むことはない。また、小型なので自宅に持ち帰り自宅学習で利用することも可能である。すでに、徳山高専用を含め、約300台が製造されている。

開発したマイコンは、補助記憶装置を持たないのでプログラムの保存ができないという問題があり、メモリのバッテリーバックアップを検討した。しかし、学生の使用方法を観察すると、「最初からもう一度やり直す」ために電源を切っていることに気が付いた。電源を切ることで電子回路が初期状態に戻ることに気づいているのである。これは、バッテリーバックアップをしなかったことによる利点である。クロス開発環境を用いれば、プログラムを保存する必要はないので、バッテリーバックアップは実装しなかった。

表 2 カリキュラム例  
Table 2 Example of curriculum.

授業内容	回数
ガイダンス	1
情報の表現 (2 進数, ビット, 文字等)	4
論理演算 (演算と論理回路)	1
マイコンの組み立て (ハンダ付け)	5
コンピュータの基本構成 (CPU, バス, メモリ, I/O)	1
マイコンの基本構成と操作	1
命令実行の流れ (命令フェッチ, 実行)	1
試験	1
プログラミング (ハンドアセンブル)	1
転送命令 (LD, ST)	1
加減算命令 (ADD, SUB)	1
分岐命令 (JMP)	1
フラグと条件分岐命令 (JZ, JM, JC)	1
繰り返し処理	2
シフト命令 (SHRA, SHRL, SHLA, SHLL)	2
論理演算命令 (AND, OR, XOR)	1
アドレッシングモード	1
入出力	3
試験	1
ガイダンス	1
クロス開発とデバッグ機能	2
文字入出力	1
スタックとサブルーチン	2
2 進 16 進 / 10 進変換	2
メモリダンプ/クリア/コピー	2
マシンステートとディレイルーチン	1
電子オルゴール (マシンステートの応用)	1
割込み処理	2
試験	1

## 8. 授業への応用例

### 8.1 授業内容

高専の 1 年から 2 年前期の 45 回 (100 分/回) のカリキュラム例を表 2 に示す。

1 年前期では, まず, 情報を 2 進数で表現する方法や論理演算等を学ぶ。次に, マイコンを各自がハンダ付けして組み立てる。最後に, 組み立てたマイコンの操作方法を学習しながら, コンソールパネルから観測できるコンピュータの構成要素や, 命令実行の流れをブロック図を用いて学び, ノイマン型コンピュータの動作の概略を知る。後期では, マイコンを用いて機械語のプログラミングを学ぶ。毎回, 授業の前半で新しい機械語命令が説明され, 後半ではそれを用いるプログラミング演習がマイコンを用いて行われる。

1 年の機械語演習では, メモリ内のビットイメージを体得することに重点を置き, ハンドアセンブルとコンソールパネルだけ用いる。2 年では多くのプログラミング経験を積むことに重点を置き, 時間的に効率良くプログラムを作成することができるクロス開発環境を用いる。2 年の演習にも, 文字端末を外部に接続し

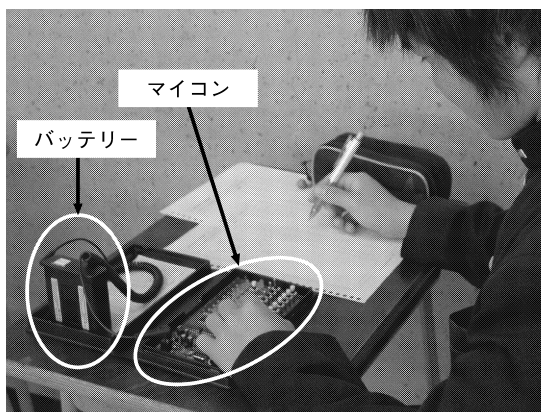


図 7 バッテリーを使用した演習  
Fig. 7 Practice with battery.

て行う文字入出力, マシンステートを応用するディレイルーチンと電子オルゴール, 割込み等, 実機を用いるメリットを生かせるものがある。

### 8.2 演習方法

一般教室での演習ができるように, 別途, バッテリーを 50 個準備した。図 7 に, 一般教室でバッテリーを使用し演習を行っている写真を示す。これによって, 教室移動の時間ロスを防ぐことができるし, 講義の途中でマイコンを使用した動作確認をすることもできる。

1 年後期の演習課題 (宿題含む) は, 掛算プログラムや割算プログラム等, 20 ステップ以内で記述できる機械語プログラムの作成である。このような問題を毎回 2 問程度与え, フローチャートの作成から実機上での動作確認までを演習させる。自宅での演習が可能なので, やり残した演習課題や宿題を持ち帰らせ, ハンドアセンブル結果を次回の授業開始時に提出させる。また, 長期休業に課題を与えることもある。

このように, 多くのプログラミング課題を与えることで, 機械語を通したコンピュータの観察を十分に行うことができる。要件 1~要件 3 を満たす安価で小さな実機は, 学校でも家庭でも利用でき, 演習の機会を増し, ノイマン型コンピュータの原理を実感することに役立つと考えられる。

## 9. 評価

1 年終了時点で, 受講生全員 (約 40 名) を対象にアンケートを行った。アンケートの回答は, 「よく分かった (4)」「分かった (3)」「分からなかった (2)」「習ったことを覚えていない (1)」の 4 段階である。

### 9.1 授業項目

平成 15 年度から 18 年度に実施した, 授業項目と演習項目について質問したアンケートの結果 (一部)

表 3 アンケート結果 (平均値)  
Table 3 Result of the questionnaire.

質問項目	H.15	H.16	H.17	H.18
「ビット」の意味	3.3	3.3	3.4	3.6
「バイト」の意味	3.3	3.2	3.4	3.6
2進数	3.6	3.5	3.6	3.9
16進数	3.6	3.5	3.6	3.8
2の補数表現	2.6	2.6	2.8	3.1
NO 命令	3.2	3.0	3.4	3.2
HALT 命令	3.8	3.8	3.8	3.9
LD 命令	3.6	3.8	3.8	3.9
ST 命令	3.7	3.8	3.8	3.9
ADD 命令	3.7	3.8	3.7	3.8
SUB 命令	3.6	3.8	3.7	3.8
JMP 命令	3.6	3.8	3.7	3.8
JZ 命令	3.4	3.7	3.5	3.7
JC 命令	3.2	3.7	3.4	3.6
JM 命令	3.3	3.7	3.4	3.7
CMP 命令	3.4	3.7	3.3	3.7
AND 命令	3.3	3.6	3.2	3.5
OR 命令	3.2	3.5	3.2	3.4
XOR 命令	3.0	3.4	3.1	3.3
SHLA 命令	3.0	3.4	3.2	3.1
SHLL 命令	3.0	3.5	3.2	3.2
SHRA 命令	2.9	3.4	3.1	3.0
SHRL 命令	2.9	3.4	3.1	3.1
IN 命令	3.0	3.1	3.0	2.8
OUT 命令	3.0	2.4	3.0	2.7
平均	3.29	3.46	3.38	3.48

を表 3 に示す．ほとんどの項目で平均が「よく分かった」と「分かった」の間になっており，良好な結果といえる．また，表 4 に示す定期試験の正解率も，アンケート結果を裏付けるものになった．

しかし，平成 17 年度と 18 年度の SHLA 命令の結果を比較すると，この部分だけは，アンケートの結果と試験の結果が食い違っているように見える．これは試験で誤答が非常に少なく，誤答の主な原因が単純なミスであるためと考えられる．

アンケート結果から，高専の 1 年生 (16 歳) の学生でも，半年でひとおりの機械語命令を網羅的に学習し，このような高い理解度を示すことが分かった．これは，要件 4 を満たす単純な構成のマイコンを用いたことで，機械語命令の学習に集中できたためと考えられる．

## 9.2 ノイマン型コンピュータの原理

平成 18 年度は，アンケートに表 5 の質問を追加し

表 4 シフトに関する問題の正解率  
Table 4 Correct answer rate of problem concerning shift operation.

命令	H.15	H.16	H.17	H.18
SHLA	74%	-	91%	95%
SHLL	-	95%	-	-
SHRA	45%	85%	75%	68%
SHRL	74%	-	84%	83%
平均	64%	90%	83%	82%

表 5 ノイマン型コンピュータの原理に関する質問  
Table 5 Question concerning principle of Neumann type computer.

質問		4	3	2	1	平均
右を実感	プログラム内蔵方式	16	15	6	3	3.1
していま	逐次実行	23	14	3	0	3.5
すか	2進法	24	13	3	0	3.5

た．回答は，従来のアンケートと同じ 4 段階である．全体的に，平均が 3 を超えており，良好な結果である．「プログラム内蔵方式」と「逐次実行」については，前期の最終週に簡単に説明しただけであること，「実感していますか」と質問したことを合わせると，機械語プログラミングの経験から「実感をともなって」理解されたといえる．特に 2 進法については，従来，高学年になっても「特殊なもの」と感じる学生が少なくなかったが，要件 1，要件 2 を満たし，内部を 2 進数で観察できるコンソールパネルを実装した実機を用いたことにより，自然に「あたりまえのもの」と感じるようになったと考えられる．

以上のように，本マイコンを用いた機械語演習から「実感をともなって」ノイマン型コンピュータの原理が理解されることが明らかになった．本論文で報告したマイコンは，独自に CPU を設計し，コンソールパネルを教育用に工夫し，そのうえで小型に実装した．その結果，本マイコンを使用した演習は，機械語を容易に学習することができ，2 進法が自然に受け入れられ，教室でも自宅でも手軽に行うことができるものになった．これによって，ノイマン型コンピュータの動作原理を理解可能になったと考える．

## 10. おわりに

本論文において，機械語教育用マイコンの開発と授業への応用について述べた．このマイコンは，実機として実現され (要件 1)，教育用に工夫したコンソール

毎年，同様な出題があり，かつ，年度によるアンケート結果の変化が大きいため，シフト命令を比較した．

16 進数と 2 進数の変換ミスと思われる誤答が多い．平成 17 年度の方がミスを誘発しやすいビット並びであった．

実際のアンケートでは，用語を記憶していない場合でも答えられるように「命令もデータもメモリに格納されること」「命令を順番に実行すること」「プログラムもデータも 2 進数で表現されること」の注釈を付加した．



パネル(要件2)を実装している。また、VHSビデオカセットサイズの小型で安価なものである(要件3)。命令セットアーキテクチャ、入出力、割込み、クロス開発環境も教育用に吟味した仕様になっている(要件4)。実際に授業で使用しアンケートや定期試験の結果から、機械語とノイマン型コンピュータの原理について、良好な学習効果が認められた。

現在、本マイコンシステムの設計データ、クロス開発環境、テキスト、ドキュメント等はすべてインターネット上に公開している。また、マイコン本体も一般向けに業者を介して供給中である。

今後の課題としては、マイコンの、より効果的な授業での利用方法の検討、教育効果についてのさらに詳細な調査があげられる。

謝辞 本研究の遂行にご高配を賜った、徳山高専情報電子工学科の山田健仁教授、力規晃助手、新田貴之助教、2年の授業と4年の実験でマイコンを使用いただいている守川和夫准教授、マイコンを授業に採用する決断をしていただいた池田信彦元学科主任、マイコンの実装と学生への供給に尽力していただいた(株)兼清電子山城利勝部長(株)竹上電気商会竹上都雄社長に深謝する。なお、本研究の一部は財団法人理工学振興会平成14年度教育研究助成金の援助による。

### 参 考 文 献

- 1) 中西正和：機械語プログラミング教育の必要性と実習用コンピュータの製作(1), *bit*, Vol.25, No.8, pp.24-31 (1993).
- 2) 中西正和：機械語プログラミング教育の必要性と実習用コンピュータの製作(2), *bit*, Vol.25, No.9, pp.58-63 (1993).
- 3) 清水敬子, 阿刀田央一, 高橋延匡：情報工学系学科の計算機初期教育におけるEDSACの活用の試みと効果, 情報処理学会論文誌, Vol.24, No.3, pp.281-292 (1983).
- 4) 清水敬子, 伊勢正治, 中森真理雄：情報工学系の計算機初期教育のための環境について, 電子情報通信学会論文誌(A), Vol.J71-A, No.9, pp.1734-1741 (1988).
- 5) 西田知博, 矢原潤一, 増澤利光, 松浦敏雄：目的に応じて観察の抽象度を変更可能な計算機シミュレータECASを用いた教育とその評価, 情報処理学会情報教育シンポジウム(SSS2002)論文集, Vol.2002, No.12, pp.245-252 (2002).
- 6) 渡辺博芳, 荒井正之, 武井恵雄：CPUとアセンブラ授業のためのシミュレータWCASLの評価, 情報処理学会第62回全国大会, No.5X-1, pp.225-226 (2001).
- 7) Menczer, F. and Segre, A.M.: OAMulator: A Teaching Resource to Introduce Computer Architecture Concepts, *Journal of Educational Resources in Computing*, Vol.1, No.4, pp.18-30 (2001).
- 8) 重村哲至, 山田健仁, 新田貴之, 力 規晃, 原田徳彦, 三木 幸：コンソールパネルを持つビデオカセットサイズの教育用マイコンの開発, 情報処理学会研究報告, Vol.2004, No.49(CE-74), pp.25-32 (2004).
- 9) 重村哲至, 山田健仁, 新田貴之, 力 規晃, 奥本 幸, 原田徳彦, 守川和夫：コンソールパネルを持つコンピュータ教育用マイコン, 電子情報通信学会大会講演論文集, Vol.2005, No. 情報・システム1, p.155 (2005).
- 10) 中西正和, 佐藤健吾, 山口文彦, 加藤良信：機械語教育システム—PDP-11教育のハードウェアとソフトウェア, 情報処理学会夏のプログラミング・シンポジウム報告集「情報技術とコンピュータ」, pp.95-100 (1998).
- 11) 神原弘之, 安浦寛人：計算機教育用マイクロコンピュータの開発とその応用, 情報処理, Vol.33, No.2, pp.118-127 (1992).
- 12) 神原弘之, 安浦寛人：情報工学のための教育用マイクロプロセッサKUE-CHIP2, 情報処理学会新しい時代の情報処理教育カリキュラム・シンポジウム論文集, Vol.94, No.9, pp.87-94 (1994).
- 13) 末吉敏則, 田中康一郎, 柴村英智：再構成可能な論理LSIを用いた教育用マイクロプロセッサ：KITE, 情報処理学会研究報告, Vol.92, No.82(ARC-96), pp.101-108 (1992).
- 14) 新田貴之, 古川達也, 相知政司：仮想シングルボードコンピュータx80を用いた学習環境の提案, 電気学会論文誌(C), Vol.125, No.1, pp.128-133 (2005).

(平成18年1月19日受付)

(平成19年6月5日採録)



重村 哲至(正会員)

昭和39年生。平成元年豊橋技術科学大学大学院工学研究科情報工学専攻修了。同年より徳山工業高等専門学校助手。現在准教授。平成17年より佐賀大学大学院工学系研究科博士後期課程在学中。オペレーティングシステム、プログラミング、コンピュータネットワークに関する教育と研究に従事。



古川 達也（正会員）

昭和 31 年生。昭和 59 年九州大学大学院工学系研究科博士後期課程単位取得退学。同年より長崎大学工学部助手。昭和 62 年佐賀大学理工学部電気工学科講師。昭和 63 年同助教授。平成 5 年同電子工学科助教授へ配置換え。平成 13 年同電気電子工学科教授。工学博士。電子情報通信学会，電気学会，計測自動制御学会，日本シミュレーション学会，日本 AEM 学会各会員。



相知 政司

昭和 39 年生。平成元年長崎大学大学院工学研究科修士課程修了。同年 NTT に入社。平成 3 年佐賀大学理工学部助手。平成 12 年同電気電子工学科講師。平成 14 年同助教授。博士（工学）。工学教育の研究に従事。電気学会，計測自動制御学会，電子情報通信学会，日本シミュレーション学会，日本 AEM 学会各会員。



林 敏浩（正会員）

昭和 41 年生。平成 3 年徳島大学大学院工学研究科情報工学専攻修士課程修了。平成 6 年同大学院博士後期課程システム工学専攻修了。博士（工学）。同年佐賀大学理工学部講師。平成 9 年同助教授。平成 16 年香川大学総合情報基盤センター助教授。環境型教育システム，日本語教育システム，情報ネットワーク利用の教育システムの研究に従事。電子情報通信学会，教育システム情報学会，日本教育工学会，人工知能学会，ゲーム学会，米国 AACE 各会員。