

教育訓練用組込みマイコン教材の試作

Development of Training Materials for Embedded Microcomputer Systems

菊池達也，大村光徳，渡辺忠史，安原雅彦

KIKUCHI Tatsuya, OMURA Kotoku,

WATANABE Tadashi, YASUHARA Masahiko

1. はじめに

自動車、家電はもとより、携帯電話や電子カードなどにも、マイクロコンピュータ^(注1)を活用した組込み技術が普及しつつある。今や、“ものづくり分野”では、高性能・多機能な組込み用マイクロコンピュータ（以下、組込みマイコン）の利用により、商品の付加価値や国際競争力を高めている。

マイクロコンピュータのプログラム方式の特徴から、ソフトウェアの拡張や修正が容易であり、多品種少量のものづくりやメンテナンスに適合している。さらに、マイクロコンピュータを中心とした基本設計の共通化により、開発期間の短縮、開発費用と製造のコストを下げる事が可能である。組込みマイコン技術は製造業の競争力の根幹を支えている中核技術になっている。

しかし、今日の組込みシステム開発のマーケット規模と比較して、技術者は大幅に不足している⁽¹⁾。技術者教育の現状について経済産業省は次のように報告している⁽²⁾。

現在は、高等教育機関における組込みソフトウェア技術に関する教育カリキュラムや教材が不足しており、十分な教育を受けないまま就職し、企業で組込みソフトウェア開発の業務をしている実態がある。また、企業内においても、体系だった研修カリキュラムや教材が無く、技術者の育成はもっぱら OJT (On the Job Training) に依存しているという例も多数あり、組込み技術者教育に対する期待が高い。

当校の情報系専門課程では 8bit 系 Z80 マイクロプロセッサを使用した Kentac 800Z でアセンブリ言語を使用して実施している。また、応用課程では、16bit 系 M16C マイコン・キットを C 言語で利用している。主に、8bit 系と 16bit 系マイコンで教育訓練を実施している^(注2)。

しかし、最近の組込みソフトウェア産業の動向として、大規模化・高度化・複雑化への対応として、32bit 系組込みマイコンの利用が増加している⁽¹⁾。ディジタル家電、携帯電話、カーナビゲーションなど、メモリを大量に使用してデータや信号を処理する製品は 32bit 系マイコンに移行してきている。当校でも、従来のマイコン教育の上位に相当する 32bit 系の教育訓練の実施が必要とされ、適切な組込みマイコン教材が求められている。

現在、市販されている 32bit 系組込みマイコン実習教材の多くは、「ハードウェアの拡張性に乏しく、自由に改造できない」、「ソフトウェアや電子回路がブラックボックス化されて、内部の仕組みが分からない」、「学習キットとして完成されて、学生の学習意欲がわからない」などの問題点がしばしば話題になっている。また、実習教材の整備金額を、学生一人に 1 台で試算すると、高額である。また、マイコン実習教材に実装されている LED やタクトスイッチなどの最小限の入出力インタフェースだけでは、学生の学習意欲を向上させるには乏しく、学生がプログラムに夢中になれるような制御対

象（教材）も必要とされる。

限られたリソースの中で検討した結果、市販の組込みマイコン・キットを利用して、東京校に既に整備された実習教材を有効活用することにした。組込みマイコン・キットには、32bit 系組込みマイコンを採用した AKI-SH2/7045F（以下、AKI-SH2）を使用した⁽³⁾。理由は、安価で入手性が良いためである。次に、制御対象となる実習教材は、Kentac シリーズのセンサユニット、交通信号ユニット、そして自作したライントレースロボットを利用する。なお、AKI-SH2 は、Kentac のような教育目的に製作されていない。そのため、実習に必要な LED やスイッチなどの入出力部品は実装されておらず、信号線のコネクタだけである。そこで、AKI-SH2 と Kentac シリーズのインタフェースの入出力信号の配列とインタフェース・コネクタの形状を整合させるためにメインボードを製作した。また、学生が、C 言語で制御プログラムのメインプログラムを容易に記述できるように、関数ライブラリも制作した。

以下、試作したマイコン教材の構成と本教材を利用した課題、そして、最後にまとめと今後の課題を述べる。

2. 試作したマイコン教材の構成

試作した組込みマイコン教材のハードウェアの構成、関数ライブラリ、開発環境について述べる。

2. 1 ハードウェアの構成

図 1 に示すように、試作したマイコン教材は、AKI-SH2 とメインボードから構成される。両者は、ピンソケットとピンヘッドで接続されているため着脱が容易で、AKI-SH2 を単体で使用することもできる。

AKI-SH2 は、32bit マイコン SH7045F⁽⁴⁾を実装している。内蔵フラッシュ ROM 256KB、

内蔵 RAM 4KB、拡張 RAM 1MB を実装し、32bit RISC アーキテクチャ搭載⁽⁵⁾して 28.64MHz で高速動作する。AKI-SH2 に実装している D-SUB 型コネクタは RS-232C シリアル通信ポートで、パソコンからのプログラムを転送するために使用する。拡張 RAM を使用する場合は、汎用 I/O ポートは 33 ポート、10bit A/D 変換器は 8 チャンネル、RS-232C シリアル通信ポートは 1 チャンネルが開放されている。AKI-SH2 から、メインボードに、A/D 変換器入力端子、RS-232C、汎用 I/O ポートなどの信号線の端子を配置した。また、外部機器との接続を考慮して、GND と +5V の端子も用意した。

- A/D 変換器の入力は 3 チャンネルを用意した。ノイズの影響を最小限にするシールド線を使用して、アナログ信号を扱えるように、各チャンネルに GND 端子を配置した。これらに加えて、A/D の 2 チャンネルに可変抵抗器を取り付けて A/D 変換を簡便に確認できるようにした。
- RS-232C シリアルポートは、D-SUB コネクタでメインボードに実装した。主に、RS-232C の通信端子を備える計測器や機械との通信や制御の目的に使用する。
- 汎用 I/O ポートは、Kentac シリーズに使用する 32 ポートを 40 ピンのフラットケーブルコネクタに接続した。その内訳を表 1 に示す。各ポートが 8bit であるのは、Kentac の仕様に合わせたためである。信号線の配列については、付録 1 に示す。F9 ポートについては、ユーザが入出力の方向を決定する。なお、F9 ポートに、16 文字 2 行を表示する LCD を接続した。LCD の用途は、ソースコードの変数の値や A/D 変換した値などを表示させて、プログラムのデバックを効率的に作業するためである。F9 ポートを入力として使用する場合には、LCD をコネクタから取り外して使用する。

表 1 Kentac 800Z のポート

ポート番号	ポート数(bit)	入出力の方向
FE ポート	8	出力
FD ポート	8	出力
FC ポート	8	入力
F9 ポート	8	双方向

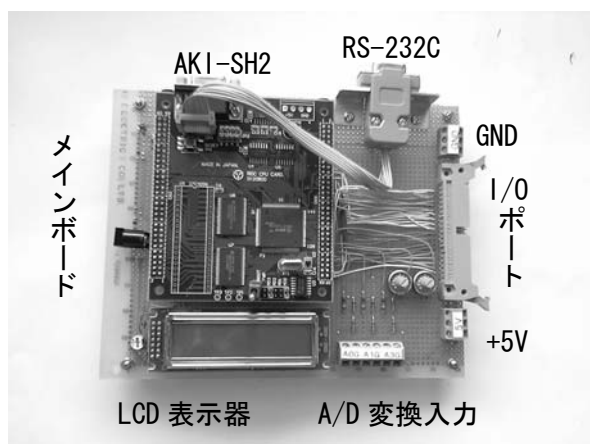


図 1 試作したマイコン教材

表 2 関数の一覧 (一例)

関数名	内 容
outbyte	1 バイトの出力
getbyte	1 バイトの入力
setbit	ビットに 1 を出力
resetbit	ビットに 0 を出力
inbit	1 ビットの入力
open_kentac	I/O ポートの初期化
delay_ms	ミリ秒単位の時間待ち

2. 2 関数ライブラリ

付録 1 に示すように、Kentac と AKI-SH2 の I/O ポートの信号配列は、必ずしも連番ではない。例えば、Kentac の F9 ポートは AKI-SH2 のポート番号が不連続であり、FC ポートは 2 つのポートからの混合構成になっている。これは、AKI-SH2 の拡張メモリの設計上、汎用ポートをアドレスバス、データバス、メモリ制御の信号線に割り当てているためである。

このような不連続の信号の処理は、bit 操作して、連続なポートとして扱えるよう

に、ソフトウェアで工夫する。初心者が円滑な学習スタートを切れるように、表 2 に示す関数を作成して、ヘッダーファイル kentac.h を用意した。例えば、これらの関数を利用して、FC ポートから入力したデータを、CPU を経由して FD ポートへ出力を繰り返すようなプログラムは図 2 に示すように、メインプログラムの本質を表現したように短くすることができ、理解しやすいと思われる。

```
#include "kentac.h"

main(void) {
    volatile unsigned char i;
    /* レジスタの初期設定 */
    open_kentac();
    while(1) {
        i=getbyte(FC); // FC ポート入力
        outbyte(FD, i); // FD ポート出力
    }
}
```

図 2 入出力のサンプル・ソースコード

2. 3 開発環境

1 章で前述したように、限られたリソース中で教材を整備するという目的から、開発環境は無償のものを検討した。

代表的なものに、無償評価版の HEW (High-performance Embedded Workshop) と非営利団体 FSF (Free Software Foundation) が進めている UNIX 互換のソフトウェア開発プロジェクト GNU (GNU is Not UNIX) の gcc (GNU Compiler Collection) がある。

HEW は、エディタ、コンパイル・リンク、デバックなどのソフトウェア開発に必要な機能を備えた統合開発環境で、WindowsXP、2000 で動作する⁽⁶⁾。ただし、無償評価版は、試用期限を過ぎるとリンクサイズが 256KB に制限される⁽⁷⁾。

GNU の gcc は、UNIX のソフトウェアであるが、Windows 上に UNIX 環境を構築する Cygwin の使用により、Windows をホストとして利用することができる⁽⁸⁾⁻⁽¹⁰⁾。gcc はコンパイラとリンカの機能だけだが、SH 系以外にも、ARM 系、MIPS 系、POWER PC 系 AVR 系、Intel x86 系、M32R 系、H8/300 系、V850 系、TMS320C3 x /C4x 系など数多くのハードウェアに対応し、実務に使われている⁽¹¹⁾。

今回は、AKI-SH2 に付属している gcc を使用し、エディタには無償で提供されている Visual C++ 2005 Express Edition を利用した。

3. 既存の教材を利用した課題

本組込みマイコン教材で、制御対象とした、Kentac800Z USER' S MANUAL⁽¹²⁾の課題や Kentac シリーズのセンサユニット、交通信号ユニット、自作したライントレースロボットを活用した課題事例について述べる。

3. 1 Kentac800Z の課題

Kentac800Z USER' S MANUAL には、Z80 アセンブリ言語用に 3 級から 7 級までの 54 の課題がある。中でも、図 3 に示すステッピングモータを駆動する 6 級の課題(図 4)は、入力したプログラムがメカ的な動きとしてフィードバックされ、視覚的に確認されるので、学習意欲を刺激し、導入教育に有効である。

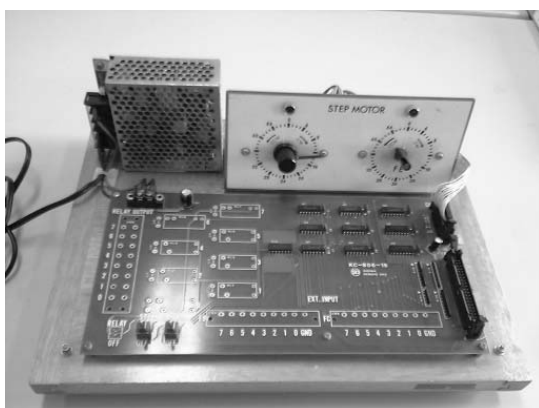


図 3 4 相ステッピングモータ

```

ORG      8000H
LD       A, 11H
LD       E, 0AH
LOOP:    OUT      (0FDH), A
CALL     106H
RLCA
JP       LOOP
END

```

図 4 ステッピングモータの Z80 アセンブリ言語

```

#include "kentac.h"

int main(void) {
    volatile unsigned int nPhase = 1;
    volatile unsigned int nPps = 100;
    open_kentac();
    while(1){
        outbyte(FD, nPhase);
        delay_ms(nPps);
        nPhase<<=1;
        if (nPhase==0x10){
            nPhase =1;
        }
    }
}

```

図 5 ステッピングモータの C 言語ソースコード

図 4 のアセンブリ言語と同じフローで、C 言語で記述したソースコードを図 5 に示す。一般的に、C 言語で記述したソースコードは行数がアセンブリ言語より少なくなるが、図 5 の行数が多い理由は、マクロ定義やライブラリの呼び出すヘッダファイルの設定や変数の宣言を記述する必要があること、さらに C 言語には循環シフト命令がないので、「RLCA」の命令に相当する部分を条件判断文として作成しているためである。また、C 言語の特徴である括弧 {} も、一因になっている。

3. 2 自作ライントレースロボット

図 6 に示すライントレースロボットは、以前、筆者（菊池）が東京校の電子技術科の学生を指導して、Kentac 800Z で制御するために製作したものである。本ライントレースロボットは、2 個の 4 相ステッピングモータで左右の車輪を制御し、3 個の反射形フォトセンサで黒いラインをトレースする。既述した課題の 4 相ステッピングモータの基本的な制御知識に、入力信号処理、条件判断文などを学習すれば、ライントレースロボットの課題にチャレンジできる。過去に、プログラミングが苦手な学生が、ロボットの制御に夢中になっているうちに、必要な知識とスキルを自然に習得した好事例があり、本課題は教育効果が期待される。

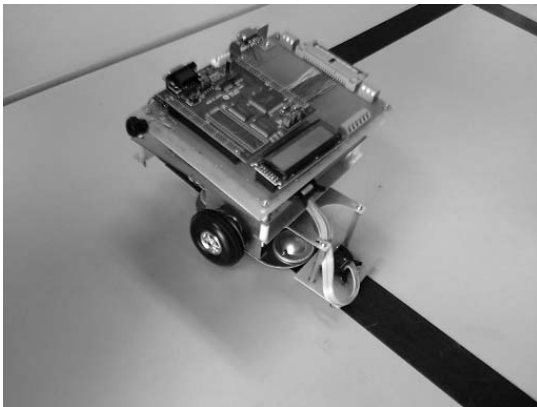


図 6 自作ライントレースロボット

3. 3 センサユニット

図 7 に示すセンサユニット (Kentac-860) は、センサ部に、超音波センサ、フォトインタラプタ、リミットスイッチ、照度センサ、交流 100V ゼロクロス検出回路、駆動部に、ブザー、ソリッドステートリレー (SSR)、ランプ、スピーカーを備えている。また、音声合成 LSI を実装し、「ありがとうございました」などの予め登録された 6 つの音声を再生する。

センサユニットの課題 1 では、お店の来客への挨拶をセンサユニットで代行させるものである。2 個のフォトインタラプタにお

店の出入り朽ちを監視させて、入口ゲートのフォトインタラプタを人が通過すれば「いらっしゃいませ」、出口のゲートを通すれば「ありがとうございました」と音声を出力する。

次に、超音波センサを利用して、センサユニットから対象物までの距離を測定する課題がある。測定結果をメインボードの LCD に表示させる。その他、SSR を利用したランプの明るさの制御など、センサと機器を組み合わせ、幅広い形の応用実習が可能である。

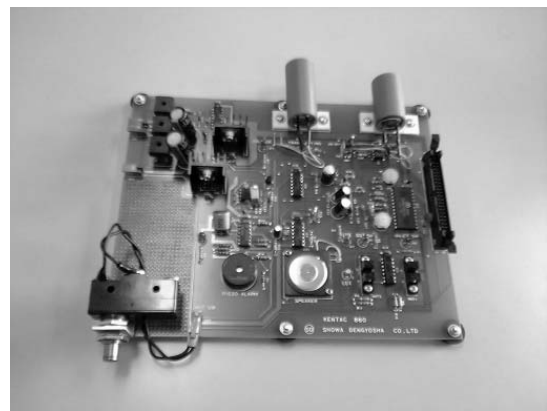


図 7 センサユニット

3. 4 交通信号ユニット

図 8 に示す交通信号ユニット (Kentac-855) は、十字路交差点と横断歩道で構成されている。交差点には、青色、黄色、赤色の LED が信号機を表現し、東西南北の 4 箇所 (計 12 個) に配置してある。交差点の各停止位置には、反射形フォトセンサが実装され、車両を検出することができる。横断歩道には、車両用信号機が 2 箇所、歩行者用信号機が 2 箇所、押しボタンスイッチが 2 箇所ある。課題としては、交差点の東西と南北の信号機で「青、黄」と「赤」を交互に点灯させたり、また深夜を想定して車両を検出したら、信号を「青」に切り替える。さらに、交差点と横断歩道の信号機を連携させる高度な課題への展開も可能である。

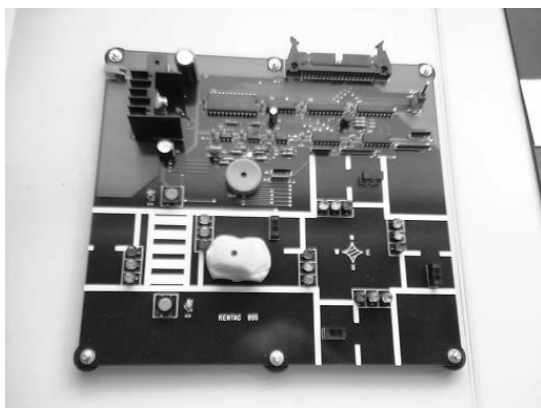


図 8 交通信号ユニット

4. まとめ

安価で入手性がよい 32bit マイコン・キット AKI-SH2/7045F を用いて、教育訓練用組込みマイコン教材を試作した。また、制御対象には、東京校に既に整備された Kentac シリーズのセンサユニット、交通信号ユニットと自作したライントレースロボットを有効活用した。AKI-SH2 と Kentac シリーズの教材を物理的・電氣的にインタフェースするために、メインボードを自作した。また、C 言語で制御プログラムのメインプログラムを容易に記述できるように、関数ライブラリも制作した。

今後は、試作した教材を利用して学生に組込み技術の教育訓練を展開していく予定である。また、信号処理、ネットワーク、リアルタイム OS などの高度な課題に取り組み、さらに改良を加えた後に必要な台数を整備して、組込み教育を充実させて、学生の育成と企業ニーズに応えたい。

謝 辞

職業能力開発総合大学校電気システム工学科の高橋久先生より有益な助言を頂戴した。また、本研究は平成 19 年度職業能力開発総合大学校東京校グループ研究の支援を受けて進めたことを感謝を込めて付記する。

[参考文献]

- (1) 経済産業省, 2006 年版組込みソフトウェア産業実態調査 報告書(2006)
- (2) 経済産業省, 組込み技術者教育ベストプラクティス集(2007)
- (3) 秋月電子通商, <http://akizukidenshi.com/>
- (4) ルネサス テクノロジ, ハードウェアマニュアル: SH7045, SH7044, SH7043, SH7042, SH7041, SH7040 グループ, Rev. 6.00 (2003)
- (5) ジョン・L. ヘネシー, デイビッド・A. パターソン, コンピュータ・アーキテクチャ, pp. 180-181, 日経 BP(1994)
- (6) 高橋久, C 言語によるモータ制御入門講座—SH マイコンで学ぶプログラミングと制御技法, 電波新聞社 (2007)
- (7) 評価版ソフトウェア, http://japan.renesas.com/fmwk.jsp?cnt=evaluation_software_mid_level_landing.jsp&fp=/products/tools/evaluation_software/
- (8) GNU, <http://www.gnu.org/>
- (9) gcc, <http://gcc.gnu.org/>
- (10) cygwin, <http://www.cygwin.com/>
- (11) M. Barr, A. Massa, C と GNU 開発ツールによる組み込みシステムプログラミング第 2 版, オライリー・ジャパン(2007)
- (12) 昭和電業社, KENTAC 800Z USER' S MANUAL

[注]

(注 1) 本研究のマイクロコンピュータとは、日本工業規格 情報処理用語 (JIS X 0001) のマイクロコンピュータの定義である「記憶装置及び入出力装置を含み、処理装置が一つ以上のマクロプロセッサから構成されるデジタル計算機」のうち、半導体集積回路から構成されるものである。いわゆる、Windows や Mac などのパソコンのことではない。

(注 2) 情報系では、専門課程の総合製作や応用課程の開発課題において、SH 系や ARM 系の 32bit マイコンを利用することがあるが、一部の学生を対象とし、限定的な内容である。

付録 1 Kentac 800Z と AKI-SH2 の配列表

Kentac 800Z		AKI-SH2	
No.	信号名	No.	信号名
1	GND	2	GND
2	GND	4	GND
3	IR5	59	NMI
4	IR6	38	IRQ0
5	F9 ㊦ ー 0bit	35	PA5
6	F9 ㊦ ー 1bit	24	PA6
7	F9 ㊦ ー 2bit	21	PA7
8	F9 ㊦ ー 3bit	22	PA8
9	F9 ㊦ ー 4bit	19	PA9
10	F9 ㊦ ー 5bit	56	PA15
11	F9 ㊦ ー 6bit	11	PA20
12	F9 ㊦ ー 7bit	14	PA21
13	FC ㊦ ー 0bit	15	PB2
14	FC ㊦ ー 1bit	18	PB3
15	FC ㊦ ー 2bit	17	PB4
16	FC ㊦ ー 3bit	20	PB5
17	FC ㊦ ー 4bit	57	PA16
18	FC ㊦ ー 5bit	58	PA17
19	FC ㊦ ー 6bit	13	PA18
20	FC ㊦ ー 7bit	16	PA19
21	FE ㊦ ー 7bit	23	PE15
22	FE ㊦ ー 6bit	26	PE14
23	FE ㊦ ー 5bit	25	PE13
24	FE ㊦ ー 4bit	27	PE12
25	FE ㊦ ー 3bit	28	PE11
26	FE ㊦ ー 2bit	29	PE10
27	FE ㊦ ー 1bit	30	PE9
28	FE ㊦ ー 0bit	31	PE8
29	FD ㊦ ー 7bit	32	PE7
30	FD ㊦ ー 6bit	33	PE6
31	FD ㊦ ー 5bit	34	PE5
32	FD ㊦ ー 4bit	51	PE4
33	FD ㊦ ー 3bit	52	PE3
34	FD ㊦ ー 2bit	53	PE2
35	FD ㊦ ー 1bit	54	PE1
36	FD ㊦ ー 0bit	55	PE0
37	未使用	60	RESET
38	未使用		未使用
39	+5V	1	+5V
40	+5V	3	+5V