

# 中高生のためのマイコン制御学習用プログラム言語の 設計と実装

谷岡 政宏<sup>1,a)</sup>

概要：今年度より中学校の技術・家庭科ではこれまで選択であった「D 情報に関する技術(3)プログラムによる計測・制御」が必修となる。しかしながら、人的、予算的、時間的、及び動作環境的といった様々な制約の中で行わなければならないという現実がある。これらを克服するために開発された教材に三田市立長坂中学校教諭浅田寿展氏の考案によるヒダピオシステムがある。このシステムでは、マイコン制御を行う際に BASIC の文法で書かれたプログラムを機械語に翻訳する必要があり、これまでは市販の BASIC コンパイラを使用していた。しかし、市販のコンパイラには(1)インストールする必要がある、(2)中学生にとって煩雑なコンパイル作業が必要である、(3)購入する予算が必要である等の問題点がある。そこで、中学生がマイコンを使って制御を学習するためのプログラミング言語を設計し処理系をスクリプティング言語 AWK で実装した。コンパイラはヒダピオシステムの簡易プログラミングでの文法を受理し、Atmel 社の AVR マイコンのアセンブリ言語を出力する。また、市販の BASIC コンパイラの文法のサブセットにも対応した。開発したコンパイラは実際に中学校の授業で利用されるほか、小中学生を対象にしたものづくり教室での電子オルゴール作成に使用されており、その有効性、実用性について実証されている。

## 1. はじめに

平成 24 年度から中学校の技術・家庭科で、これまで選択であった「D 情報に関する技術(3)プログラムによる計測・制御」(以下 D(3))が必修となる [1]。しかし、技術科の教員不足、教具の購入予算がないなどの理由で現在の実施率は十数パーセントに留まっている [2]。そのような状況の中、D(3)の教具としてヒダピオシステムが開発された [3]。パソコンの USB インターフェイスに接続するだけで動作する安価なハードウェアとして HIDaspx を採用し、インストール不要の専用ソフトウェアにより導入が簡単にできるシステムとなっており、指導用の教材も作成されている。しかし、マイコン制御を行うためには市販のコンパイラをインストールしなければならない。そこで、ヒダピオシステムから利用可能なコンパイラを開発した。

## 2. ヒダピオシステム

ヒダピオシステムとは、ハードに HIDaspx を、ソフトに「JA 制御ヒダピオ」を採用し、さらに「被制御器」や「学習ノート」などの補助教材も準備し、簡単に USB-IO 制御

やマイコン制御が楽しめる学習システムの総称である [4]。JA 制御ヒダピオ (以下ヒダピオと呼ぶ) はパソコン制御とマイコン制御を学習するためのソフトウェアである。ここではマイコン制御に関連する簡易プログラミング、電光掲示板、電子オルゴールについて説明する。

### 2.1 簡易プログラミング

簡易プログラミングはヒダピオでマイコン制御を行うためのプログラムを作成するサブシステムである。プログラムは簡単な BASIC 言語で記述する。使える命令は表 1 の通りである。プログラムの入力はマウスの操作で行える。

命令の選択肢は初級モード、中級モード、上級モード、LCD モード、Sound モードに分類されておりモードを選択することで使用できる命令の範囲が変わるようになっている。

#### 2.1.1 初級モード

Do-Loop 文を使った無限ループと portB=文を使ったパラレルポートへの出力ができる。

#### 2.1.2 中級モード

初級モードに加え、For 文による回数を指定した繰り返しができる。

#### 2.1.3 上級モード

中級モードに加え If 文を使った条件による分岐、Do-

<sup>1</sup> 北海道職業能力開発大学校  
Hokkaido Polytechnic College, Otaru, Hokkaido 047-0292,  
Japan

<sup>a)</sup> ml1tanioka@hokkaido-pc.ac.jp

Loop 文からの条件による抜け出し, 表 2 の代入文, サブルーチンが使用できる.

#### 2.1.4 LCD モード

LCD 関係の命令文の入力に使用する.

#### 2.1.5 Sound モード

Sound 文の入力に使用する.

### 2.2 電光掲示板

電光掲示板は LCD 表示装置を接続したマイコン上にメッセージを表示させるプログラムを作成するサブシステムである. メッセージと表示の仕方を指定することでマイコン用のプログラムが生成される. そのプログラムをコンパイルし, マイコンにプログラムする.

### 2.3 電子オルゴール

電子オルゴールはマイコンに音楽を演奏させるプログラムを作成するためのサブシステムである. 楽譜の入力はマウスだけで行える. 演奏データは BASIC の Data 文の形で作成される. ヒダピオは, そのデータを読み出して演奏する 1 本のプログラムを出力する. さらにこのプログラムをコンパイラによってマシン語に変換する.

## 3. 設計

ヒダピオでのマイコン制御には MCS Electronics 社の BASCOM-AVR[5] を使い ATtiny2313 用のマシン語コードを出力する必要があった. しかし, 中学校の教室で使うバ

表 1 主な命令文  
Table 1 Statements

文	説明	モード
ddrB=255	入出力方向レジスタの設定	初級
portB=0,300	ポートへの出力	初級
Do	無限ループの先頭	初級
Loop	無限ループの末尾	初級
End	プログラムを終了させる	初級
For i=0 To 7	指定された回数の繰り返し	中級
Next i	For による繰り返しの末尾	中級
portB=2 <sup>i</sup> ,300	ポートへの出力	中級
portB=2 <sup>(7-i)</sup> ,300	ポートへの出力	中級
Exit Do	Do-Loop のループを抜ける	上級
If 条件式 Then	条件による分岐	上級
Else	条件が成立しない場合の処理	上級
End If	If 文の末尾	上級
Sub	サブルーチンのラベル指定	上級
Gosub	サブルーチンの呼び出し	上級
Return	サブルーチンからの復帰	上級
Cls	LCD 表示器の画面を消去	LCD
Lcd "Hello"	LCD 表示器への文字列表示	LCD
Locate 1,1	LCD 表示器の出力位置指定	LCD
Waitms 300	待ち時間の指定 (ミリ秒単位)	LCD
Sound portB.0,282,189	音を鳴らす	Sound

ソコンは管理上, 任意のソフトウェアをインストールすることはできない.

本コンパイラは Windows 上で動作する Atmel 社の 8 ビット RISC 型のマイコンである ATiny2313 用のクロスコンパイラである. 初級モード, 中級モード, 上級モードについては, パソコン制御, マイコン制御で共通のプログラムが使えるようになっているが, LCD モード, Sound モードでは, プログラムの作成はヒダピオから行うものの, 実行にはコンパイルしてマイコンにプログラムする必要がある. また, ヒダピオにはデータを作成するだけでマイコン

表 2 代入文  
Table 2 Assignment Statements

文	説明
a=0	変数 a に 0 を代入
a=i	変数 a に変数 b の値を代入
a=a+1	変数 a の値を 1 増やす
a=a-1	変数 a の値を 1 減らす
a=255-a	変数 a のビット反転
a=a*2	変数 a を左シフト
a=2 <sup>a</sup>	2 の冪を求める
a=int(a/2)	変数 a を右シフト
a=pinB	ポート B の入力値
a=portB	ポート B の出力値
(以下 j, k も同じ)	
i=0	
i=a	
i=i+1	変数 i の値を 1 増やす
i=i-1	変数 i の値を 1 減らす
i=255-i	変数 i のビット反転
i=i*2	変数 i を左シフト
i=2 <sup>i</sup>	2 の冪を求める
i=pinB	ポート B の入力値
i=portB	ポート B の出力値
i=int(i/2)	変数 i を右シフト
(portB= で変数 i が使える)	
portB=a,300	
wt1=300	待ち時間用変数
portB=0,wt1	待ち時間を変数で指定
portB=a,wt2	出力値と待ち時間を変数で指定

表 3 If 文での条件式  
Table 3 Condition in If

文	説明
pinD.0=1	入力ポート D のビット 0 が 1 に等しい
pinB=255	入力ポート B が 255 に等しい
a=0	変数 a が 0 に等しい
pinD.0>1	入力ポート D のビット 0 が 1 より大きい
pinB>255	入力ポート B が 255 より大きい
a>0	変数 a が 0 より大きい
pinD.0<1	入力ポート D のビット 0 が 1 より小さい
pinB<255	入力ポート B が 255 より小さい
a<0	変数 a が 0 より小さい

```
Config PortB=8
Do
  PortB=8,1000
  PortB=0,1000
Loop
```

図 1 サンプルプログラム 1  
Fig. 1 Sample Program 1.

を電光掲示板や電子オルゴールにする機能がある。これらも、コンパイルする必要がある。

そこで、簡易プログラミング、電光掲示板、電子オルゴールの3つでコンパイルできることを目標とした。また、インストールせずに使用できるものとした。

## 4. 実装

### 4.1 記述言語と出力コード

本コンパイラの実装をするにあたり、処理系の記述言語について検討することにした。コンパイラを最も簡単に実装できると考えられる言語として AWK を選んだ。コンパイラのように複雑な処理をするソフトウェアの記述には通常コンパイラ型言語を使うが、大半が文字列処理であることからスクリプト言語が向いていると考えた。処理時間に関してはそれほど大きなプログラムを作ることは無いので問題にならないと考えている。近年様々なスクリプト言語が発表されているが、高機能になるほどパソコンにインストールするのが面倒である。AWK は実行ファイルをコピーするだけで使える。本コンパイラからの出力であるオブジェクトコードは ATiny2313 のアセンブリ言語で、Atmel 社のアセンブラと同じ文法を採用した。アセンブラとしては GPL2 でライセンスされている AVRA を使用することにして、本コンパイラとともに配布できるようにした。アセンブリ言語を出力することで HEX ファイルの出力をアセンブラに任せることができ開発期間を短縮することができた。

#### 4.1.1 字句解析

トークンへの分解は AWK のフィールドに対応させる。代入文や式については空白文字を含まないことにし、正規表現により別途解析する。

#### 4.1.2 構文解析

各文は、1 行の中で完結する。第 1 フィールドからどの文であるかを判別することができる。制御文の場合は対応関係の解決にスタックを使用した。

#### 4.1.3 コード生成

文が識別された段階で対応するアセンブリ言語を出力する。

#### 4.1.4 最適化

最適化は行っていない。

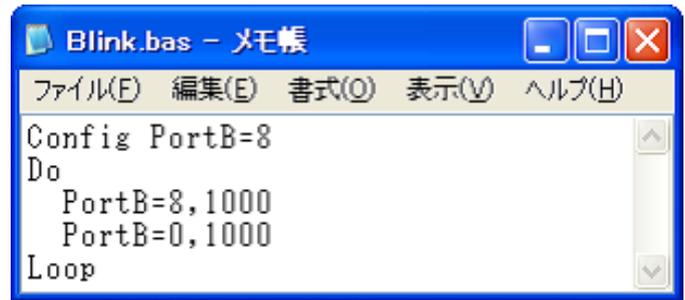


図 2 メモ帳でプログラム作成  
Fig. 2 Edit a Source Code Using Notepad



図 3 ドラッグ&ドロップでコンパイル  
Fig. 3 Compile with Drag and Drop

### 4.2 コンパイル法

本コンパイラは、ヒダピオでの利用を想定して開発したが、ヒダピオとは独立したシステムである。高校生以上の生徒に対応した教材を作成する場合には単独で使用することを想定している。図 1 のプログラムの作成には Windows のメモ帳等が使える (図 2)。コンパイルはソースファイルをパソコンピのショートカットアイコンにドラッグ&ドロップするだけで簡単に行うことができ、アセンブリ言語のソースである「bascom.asm」とアセンブルした結果のマシン語ファイルである「bascom.hex」が出力される (図 3)。図 1 のプログラムをコンパイルした結果のアセンブリ言語のリストを一部抜粋したものを図 4 に示す。

この他にも場合に応じていくつかのコンパイル法をとることができる。

#### 4.2.1 スクリプトの直接実行

```
> gawk -f comp35.awk ソースファイル名
```

ソースファイルの拡張子は省略できない。「comp35.awk」のファイル名はバージョンにより変わる。HEX ファイルを得るには、出力した bascom.asm をアセンブルする。

```
> avra bascom.asm
```

```

; Config PortB=8
  ldi R16,0b00001000
  out DDRB,R16
; Do
do_loop_top1:
; PortB=8,1000
  ldi R16,0b00001000
  out PORTB,R16
  ldi R24,low (1000)
  ldi R25,high (1000)
  rcall waitms
; PortB=0,1000
  ldi R16,0b00000000
  out PORTB,R16
  ldi R24,low (1000)
  ldi R25,high (1000)
  rcall waitms
; Loop
  rjmp do_loop_top1

```

図 4 サンプルプログラム 1 のコンパイル結果 (抜粋)

Fig. 4 Compiled Code of Sample Program 1 (fragments).

#### 4.2.2 バッチファイルの使用

> bascompil ソースファイル名

ソースファイルをコンパイルし、HEX ファイルを出力する。

#### 4.2.3 ショートカットの使用

「バスコンピ」はバッチファイル bascompil.bat へのショートカットである。このショートカットにソースファイルをドラッグアンドドロップする。bascompil.bat がある場合はダブルクリックで自動的にコンパイルする。mkshortcut.vbs はショートカットを作成するスクリプトファイルである。これをダブルクリックするとショートカット「バスコンピ」を作成する。

#### 4.2.4 キーボードからの入力

コマンドプロンプトから次のようにソースファイルを指定しないで起動すると、キーボード (標準入力) から入力できる。

```

> gawk -f comp35.awk
Ctrl+Z で入力を終了する。

```

#### 4.3 コンパイル・オプション

本コンパイラでは、次のコンパイル・オプションが使える。コンパイル・オプションは Gawk の -v オプションで与える。

**CPU\_Chip** ターゲットの CPU 名を与える。ATtiny2313 または ATmega168p(ATmega168P) を指定できる。

**CPU\_Clock** CPU の動作クロックを与える。

**Output\_file** 出力するアセンブリ言語のソースファイル名を指定する。

```

,
ddrB=255
Do
  portB=1,300
  portB=0,300
Loop
End

```

図 5 ヒダピオの簡易プログラミングの例

Fig. 5 Sample of Simple Programming.

#### 4.4 コンパイルモードとコンパイルモードスイッチ

本コンパイラは簡易プログラミングでの文法を受理しアセンブリ言語を出力するネイティブコンパイラであるが、ヒダピオからは従来の BASCOM-AVR 用のソースプログラムを出力する。また、電光掲示板や電子オルゴールのプログラムは BASCOM-AVR 用に作成されたものである。従って、いくつかの BASCOM-AVR 用の命令文等も処理できなければならない。図 5 に示す簡単な例から図 6 に示すプログラムが出力される。そこで、表 4 の命令文がコンパイルできるようにした。また、出力されたプログラムから分かる通りポート B に対する出力が反転している。これは、ヒダピオシステムで使用する標準的なハードウェアでは負論理を採用しているが中学生にとって 0 で LED が点灯し 1 で LED が消灯するのは分かりにくいからである。そこで本コンパイラでは HIDapio モードと BASCOM-AVR モードの二つのモードを持ちポート B に対する出力が負論理にするか正論理にするかを切り替えられる。また、この切り替えを表 4 の中の \$REGFILE=, Config PortB=, Dim 等がプログラムの先頭にあることを見て行っている。HIDapio モードでは図 6 のプログラム中で行っている初期化処理と同等の処理をプログラムの先頭に挿入する。

#### 4.5 コンパイラ・ドライバ

本コンパイラの利用者から直接 AWK が見えないようにするために Windows の CMD.EXE 用バッチスクリプト

表 4 主な BASCOM-AVR 互換命令文

Table 4 BASCOM-AVR Compatible Statements

文	説明
\$REGFILE=	ターゲット CPU の指定
\$CRYSTAL=	動作周波数の設定
Config LCD=	LCD 表示装置に関する設定
Config PortD=	入出力方向レジスタ (DDR) の設定
PortD=	ポート D への出力
Dim a As Integer	変数の宣言
Config PortB=	入出力方向レジスタ (DDRB) の設定
Data 10%	プログラム中のデータ
Read a	データの読み込み
Restore	データ位置指定
a=Lookup(i,label)	データテーブルからデータを変数に読み出す

```
$regfile = "Attiny2313.dat"
$crystal = 1000000
Config Lcd = 16 * 2
Config PortD=Input
PortD=&B11111111
Dim i As Integer
Dim j As Integer
Dim k As Integer
Dim x As Integer
Dim a As Integer
Dim wt1 As Integer
Dim wt2 As Integer
Config PortB=255
Do
  Portb=&B11111110
  Waitms 300
  Portb=&B11111111
  Waitms 300
Loop
portB=255
End
```

図 6 ヒダピオの出力 ( bascompi.bas ) の例

Fig. 6 Sample of Simple Programming(Output).

トとしてコンパイラ・ドライバを作成し、ドラッグ&ドロップによって簡単にコンパイルできるようにショートカットを作成する。ショートカットのリンク先には絶対パスを指定するため利用者が作成しなくてはならないが、WSH(Windows Script Host) で動作する VBScript で記述されたスクリプトにより半自動で作成するようにした。電子オルゴールは AWK の変換スクリプトを書いて対応することにして、コンパイラ・ドライバで自動的に実行するようにした。AWK の処理系としては klabaster gawk 4.0.0[6] を使用した。

#### 4.6 電光掲示板への対応

ヒダピオの簡易プログラミングでは符号無し整数のみを扱う (BASCOSM-AVR の Integer 型にマッピングされるが符号は考慮されていない) ことから本コンパイラはデータ型として 16 ビット符号なし整数のみを考えている。掲示板のプログラムでは文字データを 1 文字ずつデータ文でプログラムメモリ上に配置する。AVRA の制約によりプログラムメモリ上ではワード単位で配置する必要がありこのままでは BASCOM-AVR に比べ 2 倍の容量を必要とし、プログラムによっては 2KB しかない ATtiny2313 のプログラム用 Flash メモリが溢れてしまう。そこで、Data 文のなかに &Hxx の形式で指定されたデータがあればバッファに記録しておき、ラベルが End 文が見つかったところで Data 文の終わりと判断し 2 バイト単位で db 擬似命令を出力することにした。一方、Read 文で指定された変数が Dim 文によって Byte 型と宣言されている場合はメモリから 1 バイトだけ読み出し変数の下位バイトに格納することにした。

#### 4.7 電子オルゴールへの対応

ヒダピオが生成する電子オルゴールのプログラムでは 32bit の除算と文字列処理関数が使われているが本コンパイラはこれらの機能を持っていないのでコンパイルすることはできない。そこで、本コンパイラでコンパイルできるプログラムへ内部で変換することにした。

変換用のプログラムは AWK のスクリプトとして作成した。

およそ次の動作をする。

- (1) コンパイラ・ドライバから変換用スクリプトを呼び出す
- (2) 変換スクリプトは、
  - (a) プログラム中に Sound 文があれば出力ピンを取り出す
  - (b) Data 文があれば演奏データであるか確認する
  - (c) 演奏データであれば Sound 文の引数として与えるデータに変換する
  - (d) Sound 文を含みかつ Data 文に含まれるデータが全て演奏データであれば電子オルゴールのプログラムとみなし、プログラムを生成する ( bascompi\_orz.bas )
- (3) コンパイラ・ドライバは bascompi\_orz.bas があれば、それをソースプログラムとしてコンパイラを呼び出す
- (4) コンパイラ・ドライバは bascompi\_orz.bas を消去する

#### 4.8 ヒダピオとの統合

本コンパイラはヒダピオの簡易プログラミングの文法を受理しアセンブリ言語のソースプログラムを出力する。ヒダピオの出力する BASCOM-AVR 用のソースプログラムもコンパイルすることが出来る。コンパイラ・ドライバは起動時に bascom.bas または bascompi.bas という名前のファイルが在ればソースプログラムとして処理する。bascom.bas に対しては bascom.asm が bascompi.bas に対しては bascompi.asm が出力ファイルの名前となる。ヒダピオは本コンパイラでコンパイルするために bascompi.bas というファイルを出力するので、これをコンパイルする。さらに、浅田氏により bascompi.exe が開発され、ダブルクリックすることで bascompi.bat へのショートカットを使って bascompi.bas をコンパイルすることができる。これは bascompi.bat のショートカットを直接起動するとコンソールが表示され中学生に余計な物を見せることになるのを防いでいる。

#### 5. おわりに

中学生でも使えるマイコン制御学習用プログラム言語バスコンピの設計及び実装を行った。簡易プログラミングで生徒が作成したプログラム、電光掲示板で生成されたプログラム及び電子オルゴールで生成されたプログラムのすべ

てをコンパイルできる。本コンパイラはヒダピオに取り込まれており、実際の授業で使用されている。また、電子オルゴールを題材にしたものづくりイベントでも使用しており、実用性の面でも問題ないことが確認されている。本コンパイラがより多くの中学生に利用され計測・制御の理解に資することを望んでいる。

謝辞 本研究を進めるに当たりヒダピオの作者である浅田氏に協力頂き、成果である本コンパイラをヒダピオシステムに取り込んで頂いた。赤松武史氏には AVR マイコン用の演算ライブラリの使用を許可していただいた。ここに感謝の意を表する。

#### 参考文献

- [1] 文部科学省：中学校学習指導要領, 文部科学省( オンライン ), 入手先 ( [http://www.mext.go.jp/a\\_menu/shotou/new-cs/youryou/chu/index.htm](http://www.mext.go.jp/a_menu/shotou/new-cs/youryou/chu/index.htm) ) ( 参照 2012-5-15 )
- [2] 開隆堂出版株式会社：地域の実態と特性を活かした教科経営の工夫, 開隆堂出版株式会社 ( オンライン ), 入手先 ( <http://www.kairyudo.co.jp/general/data/contents/05-data/02-chu/gijutsu/jissen/e-shimane-01.pdf> ) ( 参照 2012-5-15 )
- [3] 浅田寿展：JA 制御 ヒダピオシステム ( 中学校, 技術・家庭科, コンピュータ制御, USB-IO, プログラミング ), ( オンライン ), 入手先 ( <http://hidapio.jp/> ) ( 参照 2012-5-15 )
- [4] 浅田寿展：初心者にも取り組むことが可能な「プログラムによる計測・制御」の教材開発をめざして, 開隆堂出版株式会社 ( オンライン ), 入手先 ( <http://www.kairyudo.co.jp/general/data/contents/05-data/magazine/kgk-journal/vol45-1.pdf> ) ( 参照 2012-5-15 )
- [5] Electronics, M.: Home – MCS Electronics, MCS Electronics (online), available from ( <http://www.mcselec.com/> ) (accessed 2012-9-14).
- [6] Klabaster: Freeware Downloads, (online), available from ( <http://www.klabaster.com/freeware.htm#dl> ) (accessed 2012-5-15).