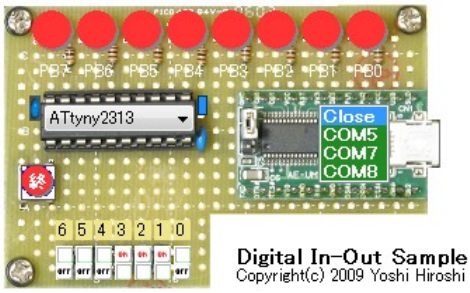
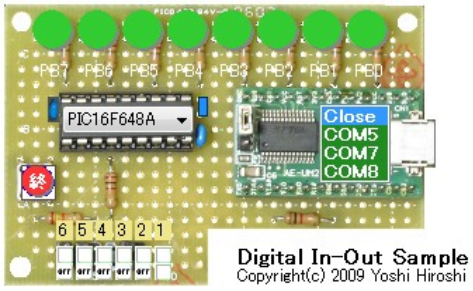
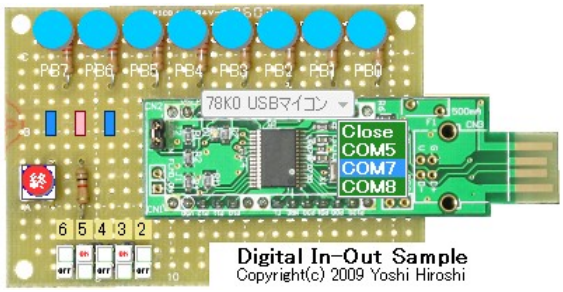
Visual Basic 2008 Express Edition で作る

超簡単リアルタイム・アプリケーション







2009 年 4 月

よし　ひろし

目次

1 はじめに....3
1.1 使用マイコン....3
1.2 ボーレート....3
2 画面と操作....4
3 パターン・エディタ....5
4 ファームウェアの機能....6
4.1 LED 点灯....6
4.2 スィッチ読込....6
4.3 スィッチ監視....6
4.4 起動時の LED チェック機能....6
4.5 各ファームウェア概要....7
4.5.1 ATtiny2313....7
4.5.1.1 回路図....7
4.5.1.2 基板イメージ....7
4.5.2 PIC16F648A....8
4.5.2.1 回路図....8
4.5.2.2 基板イメージ....8
4.5.3 USB マイコン 78K0....9
4.5.3.1 回路図....9
4.5.3.2 基板イメージ....9
4.6 ソース・コード....10
4.6.1 Visual Basic ソースコード....10
4.6.2 ファームウェア・ソースコードと開発環境....11
4.6.2.1 AVR ATtiny2313....11
4.6.2.1.1 開発環境....11
4.6.2.1.2 ソースコード....11
4.6.2.2 PIC16F648A....11
4.6.2.2.1 開発環境....11
4.6.2.2.2 ソースコード....11
4.6.2.3 USB マイコン 78K0....12
4.6.2.3.1 開発環境....12
4.6.2.3.2 ソースコード....12

1 はじめに

Visual Basic 2008 を学習される方のために、超簡単リアルタイム・アプリケーションを開発しました。
本マニュアルでは、完成したアプリケーションの機能についてまとめています。
ソースコードも公開しますのでご参照ください。

1.1 使用マイコン

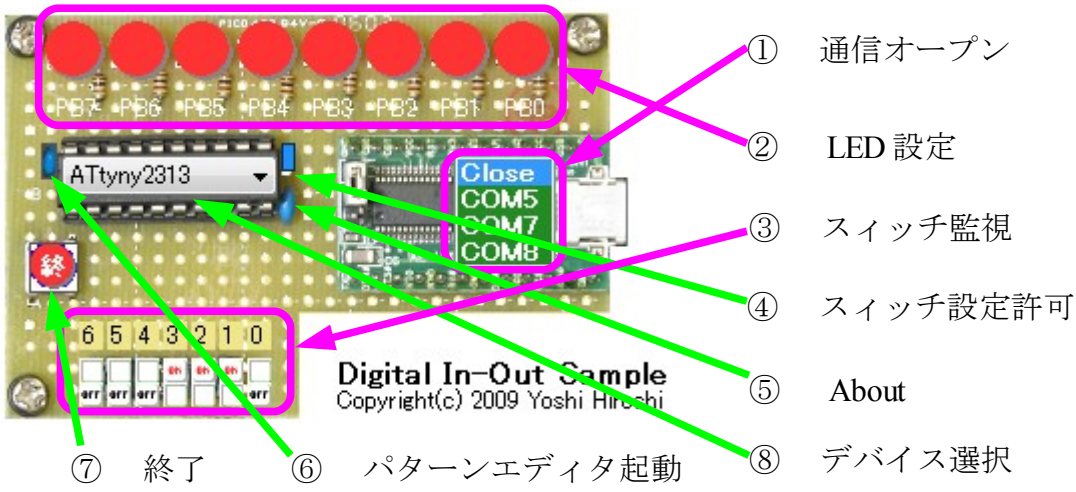
ファームウェアに ATtiny2313/PIC16F648A/USB マイコン(78K0)の 3 種類をターゲットにしています。オリジナルは ATtiny2313 なので、仕様の一部にその名残が出ていますが、ご容赦ください。
もちろん、他のマイコンでも、シリアル通信を用いたメッセージ交換が基本となっていますから、特別な制約はありません。

1.2 ボーレート

通信プログラムで次の固定値を設定していますので、ファームウェア側もそれにあわせてあります。

ATtiny2313　9,600 ボー
PIC16F648A　9,600 ボー
USB マイコン 78K0　38,400 ボー

2 画面と操作

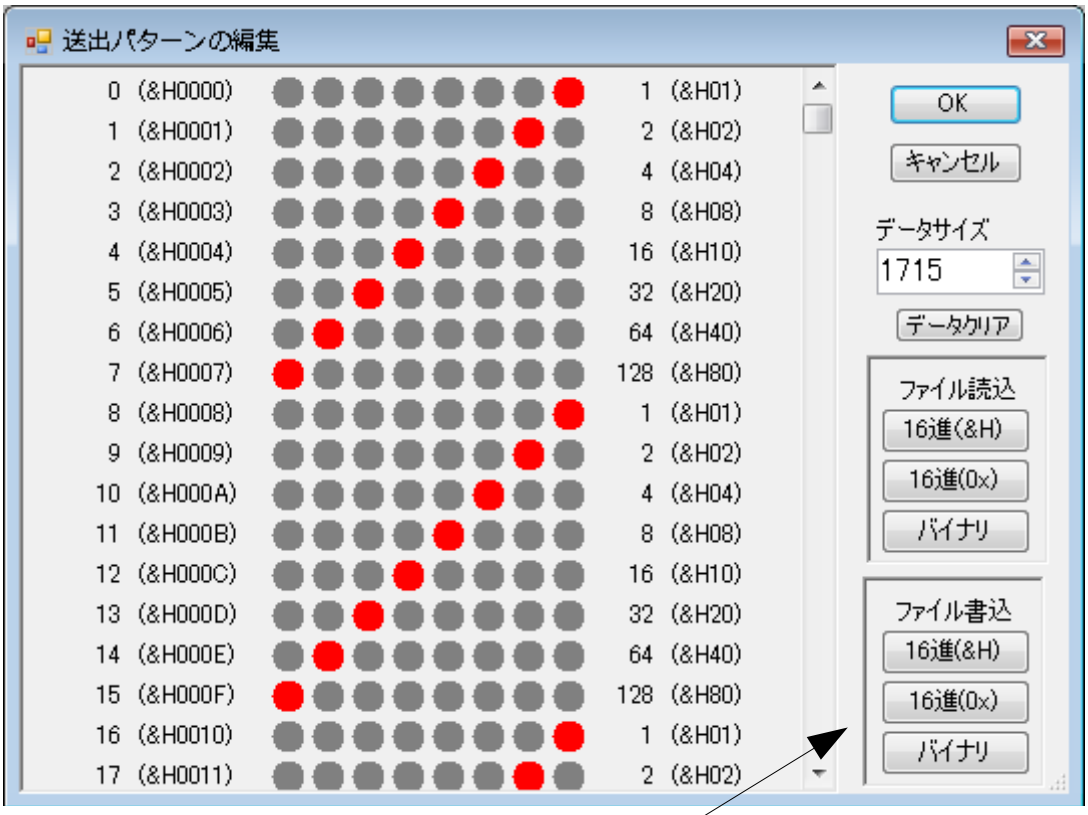


番号	機能	概要
①	通信オープン	一覧表示されたポート名をクリックして、通信ポートをオープンします。 RS232C でも USB 接続でも使用可能です。 オープンしたとき、ファームウェア側のポート B 状態を取得し、表示します。 オープンできないときは、Close 選択状態になります。
②	LED 設定	LED 画像部分をクリックして、LED のビット表示を反転します。
③	スィッチ監視	マイコンに接続されたスィッチの状態を表示します。 通信ポートがオープンしたとき、スィッチの状態を取得して表示します。 その後、スィッチの状態が変更されたときに、表示を変更します。 スィッチの状態により、自動パターン送出を開始します。 BIT6:ON 自動送出開始 / OFF 自動送出停止 BIT5:ON ビット順反転 BIT4:ON ビットパターン反転 BIT3,BIT2:パターン変更スピード　00：高速　～　11：低速
④	スィッチ設定許可	通信オープン時に、パソコン上でのスィッチ値をマウスで変更できるようにします。実際のスィッチには変化はありませんが、スィッチ操作によらず、自動送出のコントロールができます。 ボタンの背景色を変更しています。
⑤	About	About メッセージを表示します。 透明のボタンです。基板画像のパスコン部分がボタンです。
⑥	パターンエディタ起動	パターンエディタを起動します。パターン数は最大 20000 です。 このボタンは、背景に積層セラミックコンデンサのビットマップを貼っただけの、基板には実装されていない、偽装パスコンです。
⑦	終了	プログラムを終了します。
⑧	デバイス選択	デバイスを選択します。

3 パターン・エディタ

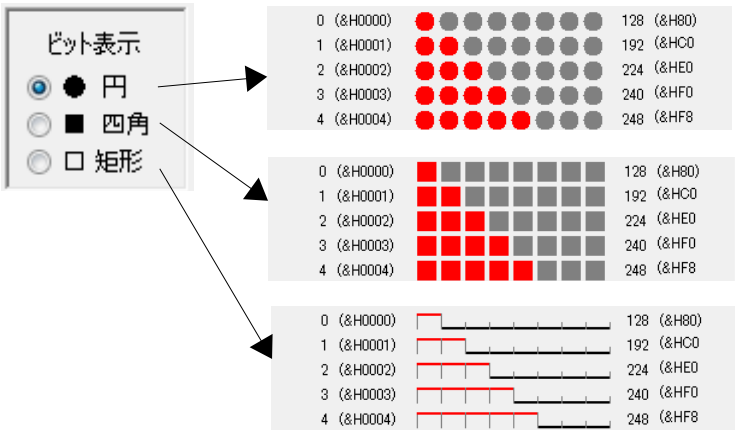
自動送出パターンを編集します。
自動送出パターンは最大 20000 パターンで、繰り返し送出します。
繰り返しのパターン数は、データサイズで設定します。
メモリー上には、データサイズ以降のデータも保存されます。

編集操作は、表示された LED 画像をマウスでクリックして、反転させてください。



ウィンドウはサイズ変更可能です。このような隙間を選んでダブルクリックすると、縦方向に最大化されます。

ビット表示は、3 種類から
選択できます。
編集画面を少し大きくすると、
選択ボタンが表示されます。



4 ファームウェアの機能

ファームウェアの機能は、４種類あります。

項番	機能	概要	AVR	PIC	78K0
1	LED 点灯	パソコンからの指示に従い、LED を点灯する。	○	○	○
2	スィッチ読込	パソコンからの指示で、スィッチの値をパソコンに送出する。	○	○	○
3	スィッチ監視	ファームウェアでスィッチを監視し、値が変化したとき、スィッチの値をパソコンに送出する。	○	○	○
4	LED チェック	電源投入時に、ファームウェアに登録された点滅パターンで LED を点灯する。	○	○	×

なお、78K0 では、LED チェック機能は搭載していません。

4.1 LED 点灯

通信メッセージの指示で、LED を点灯します。

番号	メッセージ	説明
1	“WBxx”	LED の点灯を指示します。点灯パターンはｘｘの 16 進 2 桁で表現します。 各ビットの ON は点灯を、OFF は消灯を意味します。

4.2 スィッチ読込

通信メッセージの指示で、指定されたポートの値をパソコンに送出します。

番号	要求メッセージ	応答メッセージ	説明
1	“RD”	“PDxx”	ポートに接続されているスィッチの値を取得し、その結果を 16 進 2 桁で返します。 スィッチの値は、XX の 16 進 2 桁で表現します。

4.3 スィッチ監視

スィッチが変更されたとき、パソコンにスィッチの値を送出します。

番号	要求メッセージ	応答メッセージ	説明
1	なし	“CDxx”	スィッチの値を 16 進 2 桁で返します。

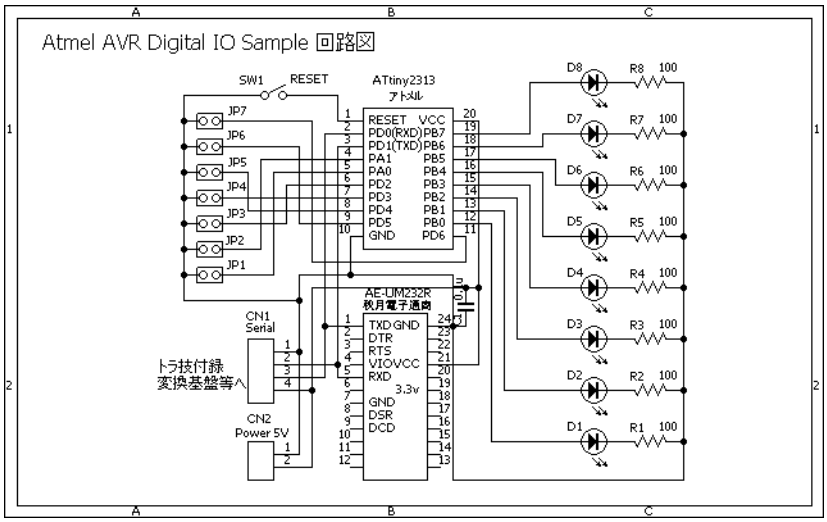
4.4 起動時の LED チェック機能

ファームウェアが起動する時、スィッチの状態をチェックして、LED 点灯チェックを行います。
ファームウェアによって異なります。

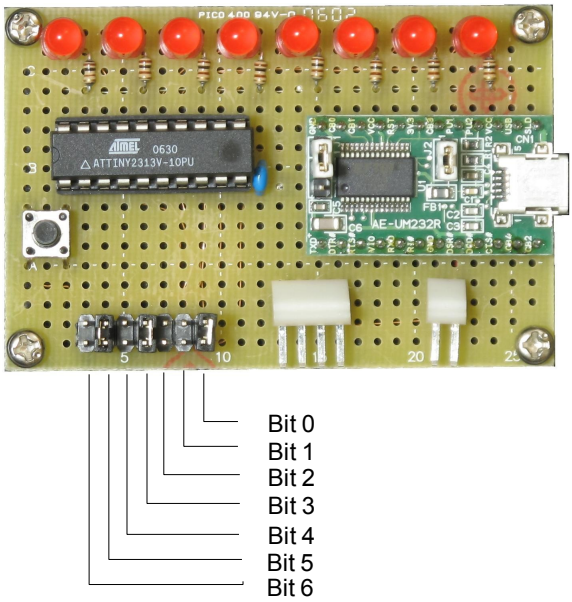
4.5 各ファームウェア概要

4.5.1 ATtiny2313

4.5.1.1 回路図



4.5.1.2 基板イメージ

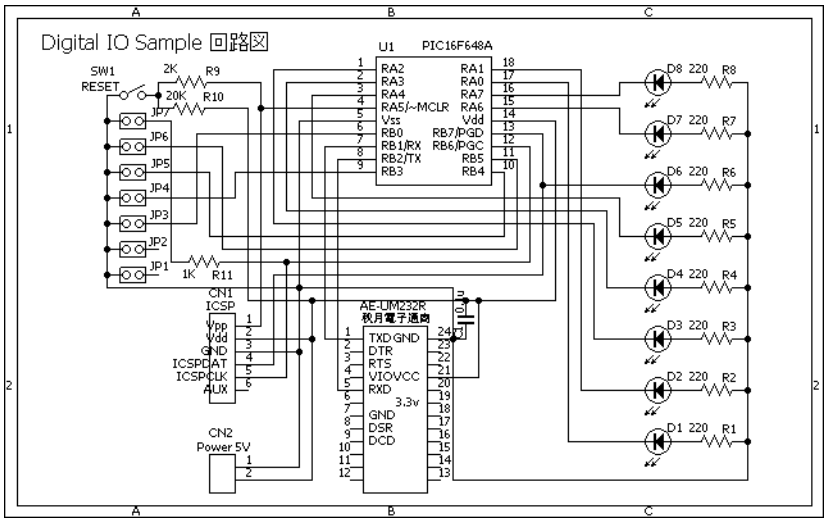


注：Bit 4 を SHORT にしないと、起動時の LED チェックから抜け出しません。

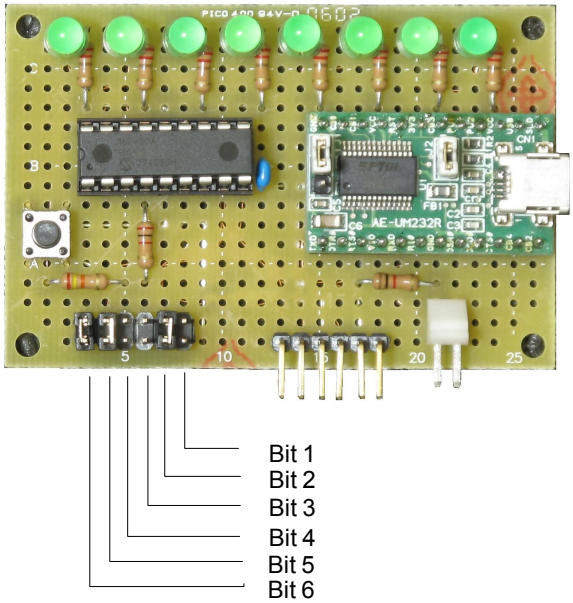
ビット	IO ピン	概要
Bit 6	PD6	パターン自動送出許可　OPEN:自動送出 SHORT:停止
Bit 5	PD5	ビット逆順　OPEN:ビット逆順（上下ビットの入替え）
Bit 4	PD4	パターン反転　OPEN:反転（ON で消灯） SHORT:ON で点灯 リセット時の LED チェック　OPEN:LED チェックする
Bit 3, Bit 2	PD3, PD2	パターン送りスピード　ON,ON:最高速 OFF,OFF:最低速 （LED チェックでも有効）
Bit 1	PA1	
Bit 0	PA0	リセット時の LED チェックパターン OPEN:PROGRAM ROM から SHORT:EEPROM から

4.5.2 PIC16F648A

4.5.2.1 回路図



4.5.2.2 基板イメージ

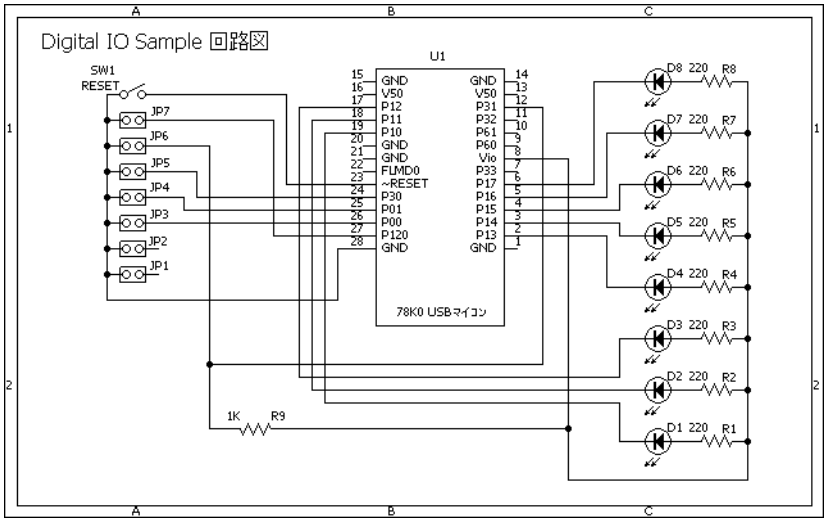


注：Bit 4 を SHORT にしないと、起動時の LED チェックから抜け出しません。

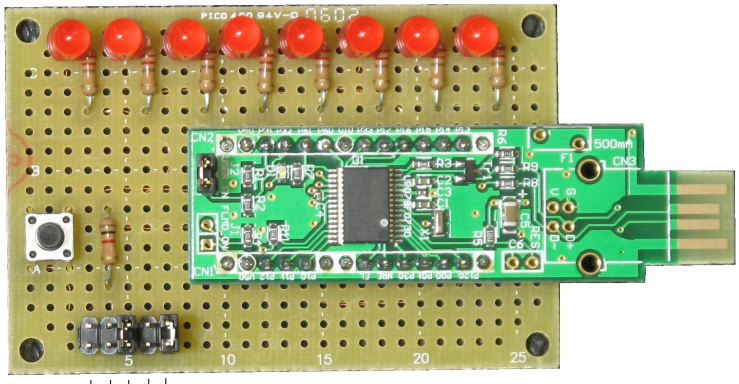
ビット	IO ピン	概要
Bit 6	RB6	パターン自動送出許可　OPEN:自動送出 SHORT:停止
Bit 5	RB5	ビット逆順　OPEN:ビット逆順（上下ビットの入替え）
Bit 4	RB4	パターン反転　OPEN:反転（ON で消灯） SHORT:ON で点灯 リセット時の LED チェック　OPEN:LED チェックする
Bit 3, Bit 2	PB3, PB0	パターン送りスピード　ON,ON:最高速 OFF,OFF:最低速 （LED チェックでも有効）
Bit 1	未接続	未使用

4.5.3 USB マイコン 78K0

4.5.3.1 回路図



4.5.3.2 基板イメージ



Bit 2
Bit 3
Bit 4
Bit 5
Bit 6

ビット	IO ピン	概要
Bit 6	P120	パターン自動送出許可　OPEN:自動送出　SHORT:停止
Bit 5	P31	ビット逆順　OPEN:ビット逆順（上下ビットの入替え）
Bit 4	P30	パターン反転　OPEN:反転（ON で消灯）　SHORT:ON で点灯
Bit 3, Bit 2	P01,P00	パターン送りスピード　ON,ON:最高速　OFF,OFF:最低速

プログラムは ROM に書き込んでください。
J2(ユーザープログラム起動)は OPEN/SHORT どちらでも OK です。
アプリケーションでは、USB マイコンを選択したときに、ROM 実行コマンドを送出しますが、BIOS が動作しているときは、このコマンドを実行してファームウェアを起動します。ファームウェアがすでに起動しているときはこのコマンドを無視します。

4.6 ソース・コード

4.6.1 Visual Basic ソースコード

ソース・コード・ファイルの一覧を次に示します。

```
vbsample --- Sample3      +-- Sample3  +-- AboutDialog.vb
                          |            +-- AutoPattern.vb      自動送出
                          |            +-- PatternEdit.vb      パターンエディタ
                          |            +-- Sample3.vb          メイン・プログラム
                          |
                          +-- PortView +-- PortView.vb        コントロール（１）
                                       +-- ByteEdit.vb        コントロール（２）
```

次に示すファイルは、フォーム・デザイナーが生成する内部ファイルです。メモ帳などで簡単に閲覧できますが、決して書き込まないでください。

```
vbsample --- Sample3      +-- Sample3  +-- AboutDialog.Designer.vb
                          |            +-- AboutDialog.resx
                          |            +-- PatternEdit.Designer.vb
                          |            +-- PatternEdit.resx
                          |            +-- Sample3.Designer.vb
                          |            +-- Sample3.resx
                          |            +-- Sample3.vbproj
                          |            +-- Sample3.vbproj.user
                          |
                          +-- PortView +-- PortView.Designer.vb
                                       +-- PortView.resx
                                       +-- PortView.vbproj
                                       +-- ByteEdit.Designer.vb
                                       +-- ByteEdit.resx
```

xxx.Designer.vb ファイルは、VisualBasic が自動的に生成するソースコードです。コンポーネントに設定したプロパティなどをすべて確認できるだけでなく、コンポーネントの使い方の参考になります。

4.6.2 ファームウェア・ソースコードと開発環境

ファームウェアのソースコードは、各デバイスごとのディレクトリに格納しています。

```
firmware  --- +--- avr          ATtiny2313 用ファームウェア
              +---  pic16f648a  PIC16F648A 用ファームウェア
              +--- 78k0         USB マイコン 78K0 用ファームウェア
```

4.6.2.1 AVR ATtiny2313

4.6.2.1.1 開発環境

開発環境は、ほとんど標準ともいえる、AVR Studio4.15 ＋ WinAVR-20081205 です。
AVR Studio は Atmel 社のホームページから無料でダウンロードできます。
WinAVR は SourceForge で公開されています。

4.6.2.1.2 ソースコード

ひとつのソースファイルにすべてまとめています。

```
firmware     +--- avr       +-- DigitalInOut.c
```

4.6.2.2 PIC16F648A

4.6.2.2.1 開発環境

（１）Hi-Tech Software PICC Lite C Compiler + MPLAB IDE
PIC は多くの開発環境がありますが、とりあえず、手に入りやすく、簡単な環境を利用しました。
MPLAB は Microchip 社のホームページから無料でダウンロードできます。
また、MPLAB インストール時に HITECH C コンパイラーの制限つきバージョンをインストールできますので、それを同時にインストールして使用します。

（２）Forest Electronic Development WIZ-C
ビジュアルな開発環境として、Wiz-C も試してみました。
こちらも、制限つきの無料バージョンをダウンロードして使うことができます。

4.6.2.2.2 ソースコード

（１）Hi-Tech Software PICC Lite C Compiler + MPLAB IDE
ひとつのソースファイルにすべてまとめています。

```
firmware     +--- pic16f648a          +--- htc       +-- DigitalInOut.c
```

（２）Forest Electronic Development WIZ-C

開発環境が自動的に多くのファイルを生成します。
今回は、そのうちのひとつのファイルにすべての機能を入れ込みました。

firmware　+--- pic16f648a　+--- wizc　+-- DigitalInOut_User.c

4.6.2.3 USB マイコン 78K0

4.6.2.3.1 開発環境

開発環境は、トラ技 2008 年 8 月号 CD-ROM の PM+6.30 を使用しました。

4.6.2.3.2 ソースコード

ひとつのソースファイルにまとめていますが、ディレクトリ上には多くのファイルが開発環境によって生成されているので、ちょっとわかりにくいかもしれません。
ROM を使うように設定してください。RAM 用のファイルも含まれていますが使用していません。

Firmware　+--- 78k0　+-- DigitalInOut.c