

## 第7章

# シリアル - USB 変換に 使っているマイコン PIC18F14K50

—— USB モジュール搭載のロー・ピン・カウントのお手軽マイコン

付属基板では、A-D コンバータとパソコン間のブリッジ IC として使っています。マイコンに書き込まれているプログラムを変えることで、7本の端子を自由に使って高分解能 A-D 変換機能つきマイコン基板としても使えます。

### センサ信号をパソコンに橋渡しするロー・ピン・カウント・マイコン PIC18F14K50

USB 対応のマイコンを解説すると本が1冊できあがるボリュームになります。実際幾冊もの専門書が出版されていますが、この基板に必要な機能は下記の三つだけです。

- (1) USB から SPI への信号変換
- (2) データ転送の時間差を吸収する程度の少量のバッファリング
- (3) パソコンからのコマンドを A-D コンバータの操作コマンドに変換

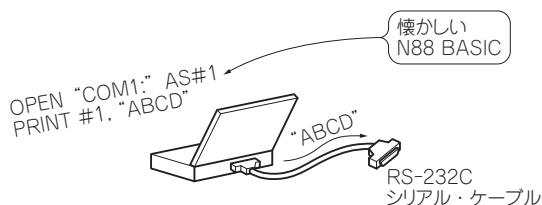


図1 シリアル・ポートなどのレガシー・ポートはプログラムと信号が直感的

### ● パソコンと A-D コンバータをつなぐ USB - SPI 変換 IC が欲しい

USB 接続の機器があふれていますが、その制御は一昔前のシリアル/パラレル・ポート(レガシー・ポートと呼ぶ)のように簡単ではありません。ターゲットと側パソコン側の両方のドライバの知識が必要です。

多くの USB 機器は、USB の動作の一つである「仮想シリアル・ポート」を使っていて、パソコン上では「シリアル・ポート(COM ポート)」のようなふるまいをします。専用のドライバを新たに作成する必要がなく、また、通信ソフトウェアと呼ばれるターミナル・ソフトウェア(Windows の「ハイパーターミナル」など)が使えるため、手操作による制御が可能です(図1)。

また、「シリアル・ポート」時代に蓄積したソフトウェア資産を生かすことができるものです。しかし、それにはターゲット側でも「仮想シリアル・ポート」の準備をしなければなりません。

「仮想シリアル・ポート」を手軽に使うための便利な「USB-シリアル変換 IC」が各社から発売されています。

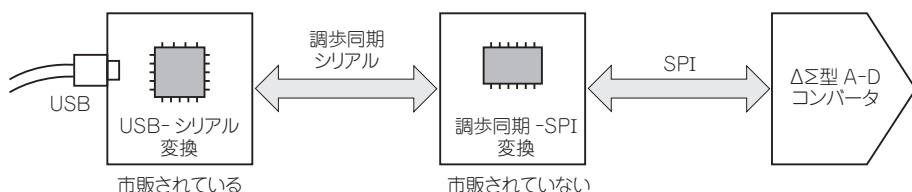


図2 USB-シリアル変換 IC と A-D コンバータの接続

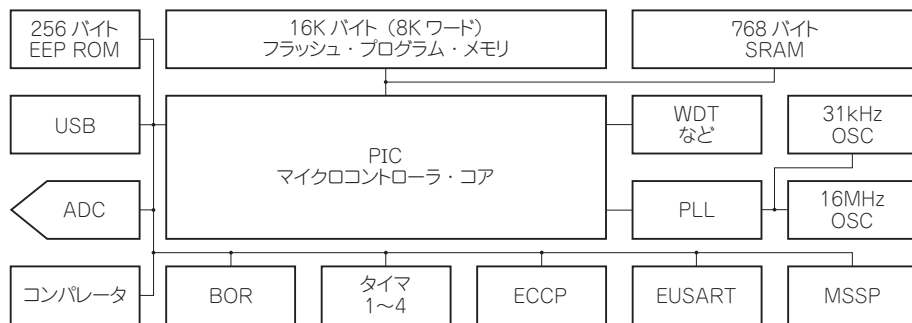


図3 PIC18F14K50のブロック図

A-Dコンバータもシリアル通信なので、そのままつなげられそうですが、同じシリアル・インターフェースでも信号の形式が違うのです。USB-シリアル変換ICのシリアル信号は調歩同期式という形式、一方A-DコンバータはSPI(Serial Peripheral Interface)と呼ばれる形式で、互換性がないために変換作業が必要になります。「USB-SPI変換IC」が世の中にあればよいのですが…。

SPIはマスタ(親機)側がすべての主導権を握る通信方式で、ランダムに返信が返る調歩同期式とはデータ送受信のタイミングも異なります。USB-シリアル変換ICを導入しても、その先には変換用にもう一つマイコンが必要になります(図2)。そのため結局、USB対応のマイコンを使うことにしました。必要な機能は「USB-SPI変換」なので、できるだけ小規模にしたい、という要求にぴったりなのがPIC18F14K50(マイクロチップテクノロジー)というマイコンです。

## ● PIC18F14K50を使う

PIC18F14K50のブロック図を図3に示します。以下に、特徴を列記します。

- (1) PICのなかでは「ハイ・パフォーマンス」に分類する高性能マイコンで、その名のとおりの「少ピンUSBマイコン」です。コンパクトながら、PIC18のペリフェラルはほぼ満載しています。
- (2) 必要最小限のメモリ容量  
USB制御部(USB制御に必要な領域)を除いても数Kワード使えます。
- (3) USBフレームワークというソフトウェア群によりUSB部は最小限の手直しだけで開発できる資産の継承が可能です。
- (4) 豊富なサンプル・プログラム
- (5) 外付け部品はクロックのみ
- (6) 低消費電流
- (7) USB機能を使用しないときは、20ピンPIC18Fとして扱うことができる  
付属基板では、USB、タイマ1chとAD7793との

接続にMSSP(Master Synchronous Serial Port)モジュール、汎用I/Oポートを使っています。

## ▶ 仕様概要

プログラム・メモリ：8Kワード  
SRAM：768バイト(USB用DPRAM 256バイト含む)  
クロック：最高48MHz(フル・スピードUSB時)  
動作電圧：1.8～5.5V

—\*—

標準のPIC18Fシリーズと異なる部分はUSBだけで、タイマやI/Oの制御はいままでのソフトウェア資産が活用できます。

USBに関して必要な回路は、唯一外付け部品となるオシレータです。12MHzを4通倍して48MHzのUSBクロックを作成します。このクロックはUSB規格を満足させるため一般のセラミック発振子は使用できず、USB専用の±0.1%以下の初期精度のものか、水晶発振器が必要です。

USBのフル・スピード/ロー・スピードを切り替えるためのプルアップを内部で設定可能です。

USBフレームワークの「デモプロジェクト」を修正して開発するのが近道です。

開発言語はPIC18Fシリーズ用コンパイラ“C18”の教育目的版が無償で使用できます。

## USBファームの簡単な開発手順

まず、USBフレームワークを入手します。マイクロチップテクノロジー社のウェブ・サイトから“USB Framework for PIC18, PIC24 & PIC32”をたどっていくと“Microchip Application Libraries v2010-08-04”(2010年8月現在)をダウンロードすることができます(図4)。

展開してインストーラを実行すると、フレームワークを構築します(図5)。

PIC18F14K50のUSBサンプルのプロジェクト・ファイルは、リスト1のような長い名前です。このサ

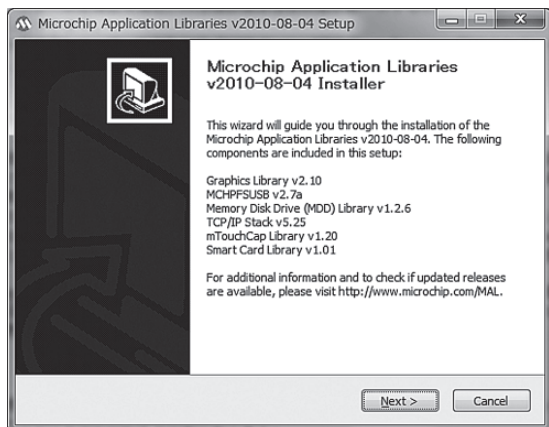


図4 インストーラ開始画面

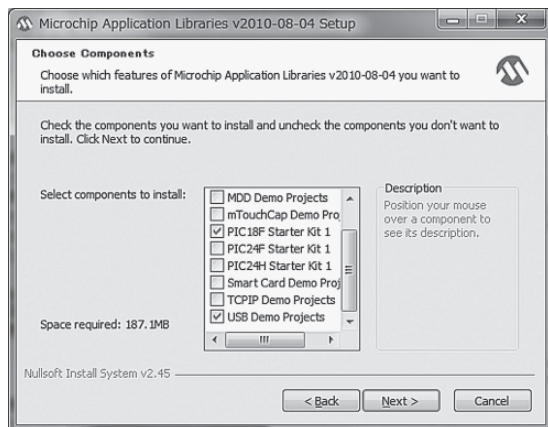
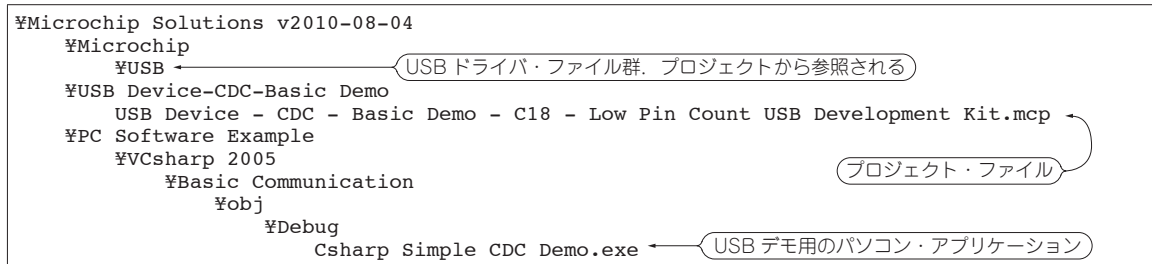


図5 USB と PIC18F スタータ・キットのみにした

# リスト1 構築されたフレームワークの中のおもなディレクトリ構造



ンプル・プログラムは USB 開発ボード用で、スイッチ入力や LED 表示を操作する簡単なものです。回路図は別途入手の“Low Pin Count USB Development Kit” マニュアルに掲載されています。参考になるでしょう。このサンプルの動作は、下記のようなものです。

- ① スイッチを押すとパソコン側アプリケーションにメッセージが出る
- ② USB の動作を LED で表示する
- ③ パソコン側から送信した文字列コードに 1 を足した文字列を返す(“ABC” と入力すると “BCD” と返る)。

画面を図 6 に示します。このパソコン用のアプリケーションが“Csharp Simple CDC Demo.exe”で、改造したサンプルの動作確認にも使えるので、開発当初にパソコンのアプリケーションも同時に開発する負担は軽減できます。

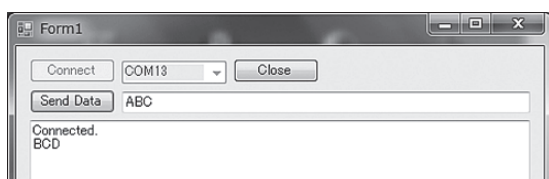


図6 サンプル・アプリケーションの画面

## ● ソースを改造する場合や ID 変更のヒント

変更するところは、main.c と usb\_descriptors.c です。

main.c は追加したい制御部分とハードウェアの初期設定、割り込みの記述などを行います。

usb\_descriptors.c では、ID と機器名を設定しますが、個人や研究室レベルの範囲で変更する必要はありません(筆者も自社 ID は取得していない)。

付属基板では本稿のためにメーカで用意された ID に変更してあります。ここを変更したときは、ドライバ設定ファイル「inf」ファイルも変更してください。リスト 2 の例では「mchpcdc.inf」から「usbadc.inf」へファイル名も変更したので 101 行目の変更となりました。

メイン処理は、USB サンプルをご覧ください。ループの中で USB 制御を数ミリ秒ごとに呼ばなくてはならないので、シーケンス制御をさせるときは、状態変数を使ったステート・マシンにする必要があります。リスト 3 に例を示します。

付属基板のファームウェアも付属 CD-ROM 添付してあります。元のサンプルと比較して、変更箇所などを参考にしてください。