

第8章

アナログ・データ・ キャプチャの製作

—— USB Mass Storage Class に対応した簡易計測器

河野 崇

ここでは、USB を介してパソコンと接続できる簡易計測器の設計事例を紹介する。ADuC7026 の A-D コンバータからデータを取り込む。USB Mass Storage Class に対応しており、パソコン側のドライバ・ソフトウェアを開発しなくても使用できる。

(編集部)

最近の組み込み機器に要求される機能の一つとして、パソコンとの接続機能があります。

ひと昔前はシリアル(EIA-232)接続が主流でした。この規格は、もともとモデムとパソコンを接続するために策定されたものであり、リモート・ホストとのキャラクタ・ベースのデータ通信に最適化されています。汎用性が高く直感的に理解しやすいため、簡単に制御ソフトウェアを書くことができます。多くの場合、組み込み機器をパソコンから制御するという用途ではこれで十分です。しかし、転送速度が低いため、高い応答性が必要だったり、大量のデータを高速にやりとりするという用途には不向きです。

これに対して、最近のパソコンでよく使われている USB は、制御にも高速データ転送にも適した規格です。しかし、一部の規格化された機能以外については、OS に依存した専用のドライバ・ソフトウェアを用意する必要があり、開発に負担がかかるという問題があります。

本稿では、USB 固有の情報を知らなくても、オープンな規格に準拠した方法で USB を活用する手法を提案します。USB Mass Storage Class へ対応することで、計測器やアクチュエータなどの制御機能とデータ転送機能を持つ機器を USB 経由で接続します。

Mass Storage Class は USB 経由でハード・ディスクやフロッピー・ディスクといった外部記憶装置を接続するための規格で、多くの OS で標準的にサポートされており、

特別なドライバがなくても利用できます。OS は、コマンド/ステータス用およびデータ転送用のパーティションをもつ外部記憶装置としてデバイスを認識します。

制御用ソフトウェアは、POSIX 規格で定義されている入出力システム・コール `open()`、`read()`、`write()`、`close()` を用います。デバイス・ファイルにアクセスすることにより、デバイスの制御やデータの入出力を行います。デバイスは仮想的に、あるいは物理的に制御用やデータ用の記憶領域(パーティション)を用意し、それらに対してパソコンが USB 経由で送ってくる Mass Storage Class のコマンドを解釈・実行します。制御用パーティションに対する書き込みはコマンド発行、読み出しはステータス要求として解釈します。もちろん、データ用パーティションに対する読み書きはデータの読み書きと解釈します。

実装例として、ADuC7026 内蔵の A-D コンバータ(2 チャンネル)からアナログ・データを取り込んでパソコンに転送する簡易計測器を作成しました。

OS は、ストレージ機器に対してキャッシュ機構を持っています。`open()` システム・コールでオープンする限り、オプション・フラグ(`O_DIRECT` と `O_SYNC`)を設定することでキャッシュを回避することができます。しかしここでは、アクセス方法の選択肢を広げ、あらゆる環境下で OS のキャッシュ機構を避けることを意図して、メディア・ロック機構のない USB フロッピー・ディスクをエミュレートしています。こうすることで、ディレイは避けられないものの、ライト・キャッシュを自動的に回避できます。

1. 簡易計測器のシステム構成

設計した簡易計測器のブロック・ダイアグラムを図 1 に

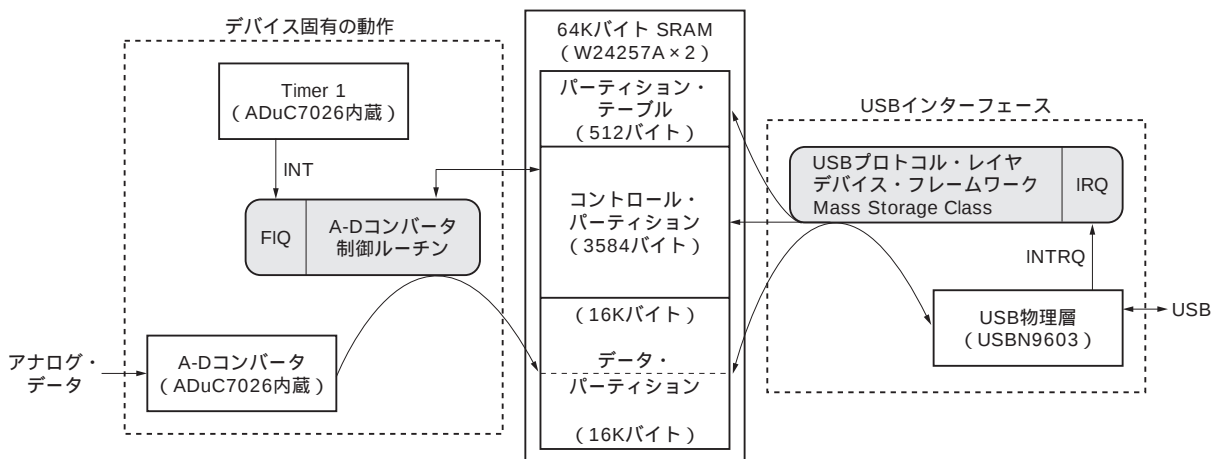


図1 設計した簡易計測器のブロック・ダイアグラム

USB インターフェースとデバイス固有の動作(今回はA-Dコンバータからのデータ取り込み)の独立性を高め、機能追加(例えばD-Aコンバータへのデータ出力など)をUSB インターフェースに手を加えることなく実現できるようにするために、64Kバイトの外付けSRAMを用いて物理的にメモリ・パーティションを用意している。

示します。

USB インターフェースとデバイス固有の動作(今回はA-Dコンバータからのデータ取り込み)の独立性を高め、機能追加(例えばD-Aコンバータへのデータ出力など)をUSB インターフェースに手を加えることなく実現できるようにするために、64Kバイトの外付けSRAMを用いて物理的にメモリ・パーティションを用意しています。

● USB インターフェース部

図1の右側にUSBインターフェースの構成が示されています。

USB物理層については、米国National Semiconductor社の「USBN9603」に任せ、プロトコル層(USBトランザクションの処理)、デバイス・フレームワーク(USBデバイス・リクエストの処理)、Mass Storage Classの処理をソフトウェアで行います。

ソフトウェアでは、USBN9603の割り込み要求信号(INTRQ)によって、ADuC7026のIRQ割り込みとして呼び出されます。

● アナログ・データ取り込み部

図1の左側にデバイス固有の動作(A-Dコンバータからのデータ取り込み)が示されています。

このソフトウェアは、ADuC7026内蔵のタイマ(Timer 1)の割り込みによって呼び出されます。Timer 1にはサンプリング周波数がセットされており、呼び出されるたびに

制御パーティションをチェックし、コマンドの解釈を行います。

データの取り込みが指示されれば、呼び出されるたびにA-Dコンバータからデータを読み出して、SRAMのデータ・パーティションに格納します。また、必要に応じてステータスを制御パーティションに書き込みます。この処理はサンプリング周波数ごとに確実に行わなければならないため、USBN9603の割り込み信号よりも優先度の高いFIQ割り込みとして呼び出しています。

具体的には、制御パーティションは0バイト目がCH0へのコマンド、1バイト目がCH1へのコマンド、2バイト目がコマンド・キューとなっています。

今回は説明を簡単にするため、データ取り込み開始コマンド('G')と中止コマンド('N')だけを用意しました。コマンド・キューは、CH0、CH1へのコマンド・バイトに有効なコマンドが書き込まれたことを示すためのものです。デバイス側は、このバイトに'*'が書き込まれたことを認識すると、これを' ' (スペース文字)でクリアすると同時に、CH0、CH1のコマンド・バイトを解釈・実行します。

データ・パーティションとしては、CH0、CH1用にそれぞれ16Kバイトずつのパーティションを用意しました。'G'コマンドが与えられると、対応するチャネルのパーティションの先頭から順次、A-Dコンバータからのデータを格納していき、末尾に到達したら対応するコマンド・バイトに動作完了のマークとして'#'を書き込み、終了します。

パソコン側はコマンド発行後、定期的に制御パーティシ