

加速度センサ出力の グラフィカル表示器の製作

小林 茂

本誌 2008 年 5 月号の付属 ARM 基板「CQ-STARMBB」には、USB ケーブルでパソコンと接続するだけですぐに利用できるように、サンプルのファームウェアが書き込まれている。このモードの一つとして、「Gainer」⁽¹⁾というツール・キットのサブセットが実装されている。5 月号では Flash Player 上で動作するコンテンツ「カエルがびよん」が紹介されたが、今月号ではもう少し内容に踏み込んで紹介する。また、Processing⁽²⁾という開発環境でのプログラミングについても紹介する。（筆者）

1. ツール・キット Gainer

Gainer は IAMAS（情報科学芸術大学院大学と岐阜県立国際情報科学芸術アカデミーの総称）で 2005 年度から 3 年間活動した PDP（Programmable Device Project）という学内プロジェクトがきっかけとなって開発がスタートしました。現在では、さまざまなデベロッパによるライブラリが公開されたことにより、ActionScript 2/3、Processing、Max/MSP 以外に、.NET、Perl、PureData、Python、Ruby、Squeak といったプログラミング環境からも利用できるようになっています。

Gainer の特徴は以下の通りです。

- ① プロトタイピングの段階から最終形まで用いることができる
- ② ブレッドボードと組み合わせて試行錯誤しながら進めることができる
- ③ 自分自身のモジュールを部品レベルから組み立てて理解を深めることができる（キットで入手した場合）

- ④ 複数のコンフィグレーションの中から目的にあったものを選択できる

- ⑤ ソフトウェアとハードウェアがオープン・ソースで公開されている

当初のターゲットは、「プログラミングはある程度できてもハードウェアはあまり経験がない。でもパソコンの画面だけで完結しないコンテンツを作りたい」というデザイナーやアーティスト向けのツールでした。最近では情報系の学部などでハードウェアを扱うためのツールとしても普及してきているようです。また、ハードウェアをオープン・ソースにしたことで、日本以外に米国でも独自にデザインし直したものが販売されています（右掲のコラム参照）。

2. ARM 基板のファームウェア

ARM 基板がサポートしているコマンドは表 1 にある 5 種類です。通常の Gainer I/O モジュールがサポートしているコマンドは全部で 25 個ほどありますが、ARM 基板のファームウェアではサブセットを実装しています。

コマンドは ASCII 形式で、「*」（ASCII コード 42）がコマンド・ターミネータです。例えば、I/O モジュールをリセットする場合には「Q*」を送信します。

3. シリアル通信ツールでの動作確認

通常、ソフトウェア・ライブラリが行っている I/O モジュールとの通信を、手動で行ってみましょう。手動で行

Keyword

Gainer, Processing, オープン・ソース, 仮想 COM ポート, 静的加速度, 動的加速度, 加速度センサ

コラム オープン・ソース・ソフトウェアとオープン・ソース・ハードウェア

オープン・ソースという言葉はソフトウェアに関してはかなり普及してきました。ハードウェア開発で利用するツールの中にも、オープン・ソースのソフトウェアは増えてきており、特別に意識することなく利用するケースが多いと思います。

これに対して、オープン・ソース・ハードウェアは比較的新しい概念です。ソフトウェアの場合、オープンにされるのはソフトウェアの設計図であり実装であるソース・コードですが、ハードウェアの場合には回路図とレイアウトで、EAGLE⁽³⁾などのデジタル・データがそのまま公開される場合がほとんどです。

これにより、上級のユーザは既存のハードウェアを自由に改変して自分自身のプロジェクトのために新しいものを作ったり、それを公開することができます。

EAGLEなどの安価で高性能なプリント基板CADが普及したことにより、こうした流れが自然に行えるようになりました。最近では、ARMプロセッサと組み込みLinuxを搭載したChumby (www.chumby.com)のようなオープン・ソース・ハードウェアもリリースされており、今後要注目の流れだと思います。

うことにより、ASCIIベースのインタラクティブな通信が実感できるでしょう。このためのツールにはシリアル通信ツールである アックノリツチ Acknowrich を使います。

Acknowrichはchuchu氏が開発し、無償で利用できるよう公開しているツールです。ASCIIおよびバイナリの両モードで利用できるシリアル通信ツールで、マイコンなどを用いた開発ではとても重宝しています。下記のURLで入手できます。

<http://www.biwa.ne.jp/~chu2/software/ack/>

配布ファイルはLHA形式のアーカイブになっているので、ダウンロード後、適当な解凍ツールで解凍します。インストールに関して特別な手順はないので、解凍が終わったらそのまま起動できます。

起動の前に ARM 基板を USB ケーブルでパソコンに接続しておきます。

起動後、「ファイル」→「シリアルデバイス」でSTM32の仮想COMポートを選択します。仮想COMポートは5月号特集で設定したものを 사용합니다。指定したCOMポートの通信ウィンドウが開いたら、次の順序でコマンドを入力します。それぞれのコマンドはウィンドウ上部の送信データ入力エリアに入力した後、リターン・キーの入力により送信されます。

表1 ARM基板のファームウェアがサポートしているコマンド

コマンド	リブライ	説 明
Q	Q	I/O モジュールをリセットする
?	?1.0.0.00	ファームウェアのバージョンを問い合わせる
KONFIGURATION_1	KONFIGURATION_1	I/O モジュールをモード 1 に設定する
i	ixxxxxxxxx	アナログ・ポート 0～3 までの値を連続で取得する
E	E	値の連続取得を終了する

- ① Q^*
- ② $?^*$
- ③ `KONFIGURATION_1*`
- ④ i^*

図1はこの順序でコマンドを入力した時の表示例です。最後の「i*」を入力すると通信ログ・エリアに「i8287A800*」のような文字列が次々と表示されます。これは、アナログ入力の値であることを示す「i」の後に16進数で4バイト分を続けたもので、先頭からX軸、Y軸、Z軸の値を8ビットで表したものになります。4個目の値はダミーで常に00hになります。値の取得を止めるには、「E*」と入力します。

なお、STM32F103VBT6内蔵のA-Dコンバータは12ビットの分解能がありますが、Gainerのプロトコルではシリアル通信のトラフィックとのバランスなどの理由により8ビットになっています。もともと12ビットのデータは8ビットに切り捨てられ、2けたの16進数として送信されます。

ライブラリはASCII形式でこのようなコマンドを送信し、その結果を受け取って処理しています。

次に、プログラミング環境Processingと、Processing用のライブラリを組み合わせ、加速度センサの出力をモニタする方法を紹介します。

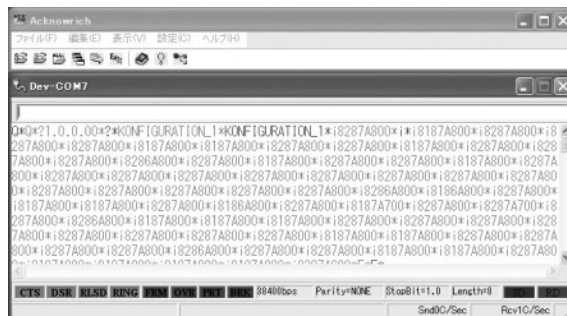


図1 AcknowrichでGainerの初期化コマンドを入力したときの画面例