



インストールマニュアル (v1.0 用)

開発者：小林茂+遠藤孝則+増田一太郎

1 動作環境

- OS
 - Windows XP SP2/3、Vista または 7
 - Mac OS X 10.4、10.5 または 10.6
- ソフトウェア
 - Java 実行環境 (J2SE 5.0 以上)
 - Flash Professional CS3/4/5・Flex Builder 3・Flash Builder 4・Flex 3/4 SDK^{*1}
 - Processing (1.1)^{*2}
 - Ruby (1.8.6 系列・1.9 系列での動作は未確認)^{*3}+ OSC ライブラリ^{*4}
- ハードウェア
 - Gainer I/O モジュール^{*5}^{*6}
 - Arduino ボード^{*7}または Boarduino などの互換ボード
 - XBee 802.15.4 OEM RF モデム^{*8}または ZB ZigBee PRO RF モデム^{*9}
 - Arduino Fio ボード^{*10}または FIO ボード

2 配布パッケージのディレクトリ構成

- **examples/** : Arduino や Processing 用のサンプル
- **documents/** : 仕様書やマニュアルなど
- **hardware/** : ハードウェア及びファームウェア
- **libraries/** : 各ソフトウェアライブラリ
 - **actionscrip3/** : AS3 用ライブラリ・サンプル・ソースコード
 - **processing/** : Processing 用ライブラリ・サンプル・ソースコード
 - **ruby/** : Ruby 用ライブラリ・サンプル・ソースコード
- **LICENSE.txt** : ライセンス条項
- **README.en.txt** : 概要説明 (英語版)
- **README.ja.txt** : 概要説明 (日本語版)
- **server/** : Funnel Server (Windows 用および Mac OS X 用)
- **tools/** : XBeeConfigTool などの各種ツール

^{*1} <http://www.adobe.com/jp/products/flex/sdk/>

^{*2} <http://processing.org/download/index.html>

^{*3} <http://www.ruby-lang.org/ja/downloads/>

^{*4} <http://raa.ruby-lang.org/project/osc/>

^{*5} <http://www.triggerdevice.com/items/>

^{*6} http://www.sparkfun.com/commerce/product_info.php?products_id=8480

^{*7} <http://www.arduino.cc/en/Main/Hardware>

^{*8} <http://www.digi-intl.co.jp/digi/wireless/zigbee/xbee-series1-module.html>

^{*9} <http://www.digi-intl.co.jp/digi/wireless/zigbee-mesh/xbee-zb-module.html>

^{*10} <http://arduino.cc/en/Main/ArduinoBoardFio>

3 ドライバのインストール

Gainer I/O モジュール、Arduino ボード、および XBee 無線モデムや Arduino Fio ボードとの接続に使用する USB ⇔ XBee 変換モジュール^{*11}のほとんどは FTDI 社の USB-to-UART 変換ブリッジチップ FT232R シリーズを搭載しています。このため、最初にこのドライバをセットアップすればほとんどの全てのハードウェアを利用できます^{*12}。

3.1 Windows XP/Vista の場合

まず最初にドライバをダウンロードします。ウェブブラウザで次の URL にアクセスし、Operating System（オペレーティング・システム）の欄に Windows XP や Vista などと書かれているドライバをダウンロードします。

<http://www.ftdichip.com/Drivers/VCP.htm>

ドライバのダウンロードが終わったら、ZIP ファイルを展開（右クリックして表示されるメニューから「全て展開…」を実行）します。展開が終わったら、I/O モジュールに USB ケーブルの片方のコネクタを接続し、もう片方のコネクタを PC 側に接続します。数秒後、ドライバのインストールを要求するダイアログが表示されますので、ダイアログにしたがってドライバのインストールを行います。

- ソフトウェア検索のため、Windows Update に接続しますか？
→ 「いいえ、今回は接続しません。」選択します
- インストール方法を選んでください
→ 「一覧または特定の場所からインストールする（詳細）」を選択します
- 次の場所で最適のドライバを検索する
→ 「次の場所を含める」をチェックし、先ほどドライバを展開したフォルダを指定します

同じような操作を 2 回要求されますので、2 回目も 1 回目と同様にドライバを展開したフォルダを指定します。



図 1 ドライバインストール時の画面例（Windows XP SP2）

^{*11} SparkFun Electronics の XBee Explorer USB（パーツナンバー WRL-08687）など

^{*12} Arduino を使用する場合には配布パッケージにも含まれています。

3.2 Windows 7 場合

まず最初にドライバをダウンロードします。ウェブブラウザで次の URL にアクセスし、Operating System（オペレーティング・システム）の欄に Windows XP や Vista などと書かれているドライバをダウンロードします。

<http://www.ftdichip.com/Drivers/VCP.htm>

ドライバのダウンロードが終わったら、ZIP ファイルを展開（右クリックして表示されるメニューから「全て展開…」を実行）します。展開が終わったら、デバイスマネージャーを開き、USB ケーブルの片方のコネクタをボードに接続し、もう片方のコネクタを PC 側に接続します。すると、デバイスマネージャーで「USB Serial Converter」として表示されますので、以下の手順でドライバのインストールを行います。

- 「USB Serial Converter」を右クリックして、「ドライバー ソフトウェアの更新」を選択します。
- 「どのような方法でドライバー ソフトウェアを検索しますか？」
→ 「コンピューターを参照してドライバー ソフトウェアを検索します」を選択します。
- 「次の場所で最適のドライバを検索する」
→ 「参照」ボタンをクリックし、先ほどドライバを展開したフォルダを指定します。

これで「USB Serial Port」として表示されますので、再度デバイスマネージャーで右クリックし、「どのような方法でドライバー ソフトウェアを検索しますか？」に対して同じ操作を繰り返します。2 回目が終了すると、「ポート (COM と LPT)」に表示されるようになります。

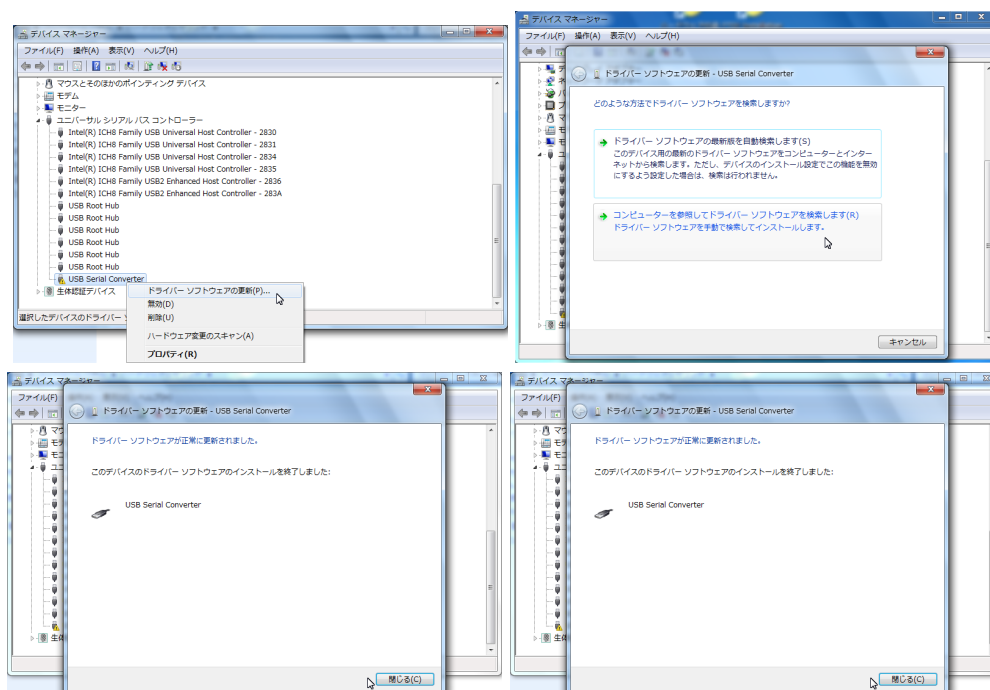


図2 ドライバーインストール時の画面例（Windows 7）

3.3 Mac OS X の場合

まず最初にドライバをダウンロードします。ウェブブラウザで次の URL にアクセスし、Operating System（オペレーティング・システム）の欄が Mac OS X と書かれているドライバをダウンロードします。

<http://www.ftdichip.com/Drivers/VCP.htm>

ドライバのダウンロードが終わると、自動的にディスクイメージファイル（拡張子は dmg）がマウントされます。もし自動的にマウントされない場合には、ファイルをダブルクリックしてマウントします。マウントが終わると、自動的にドライバのインストーラが起動します。インストーラが起動したら、インストーラの指示に従ってドライバをインストールします。途中で管理者のパスワードを入力するように求められますので、パスワードを入力します。インストール終了時には、指示に従って再起動してください。

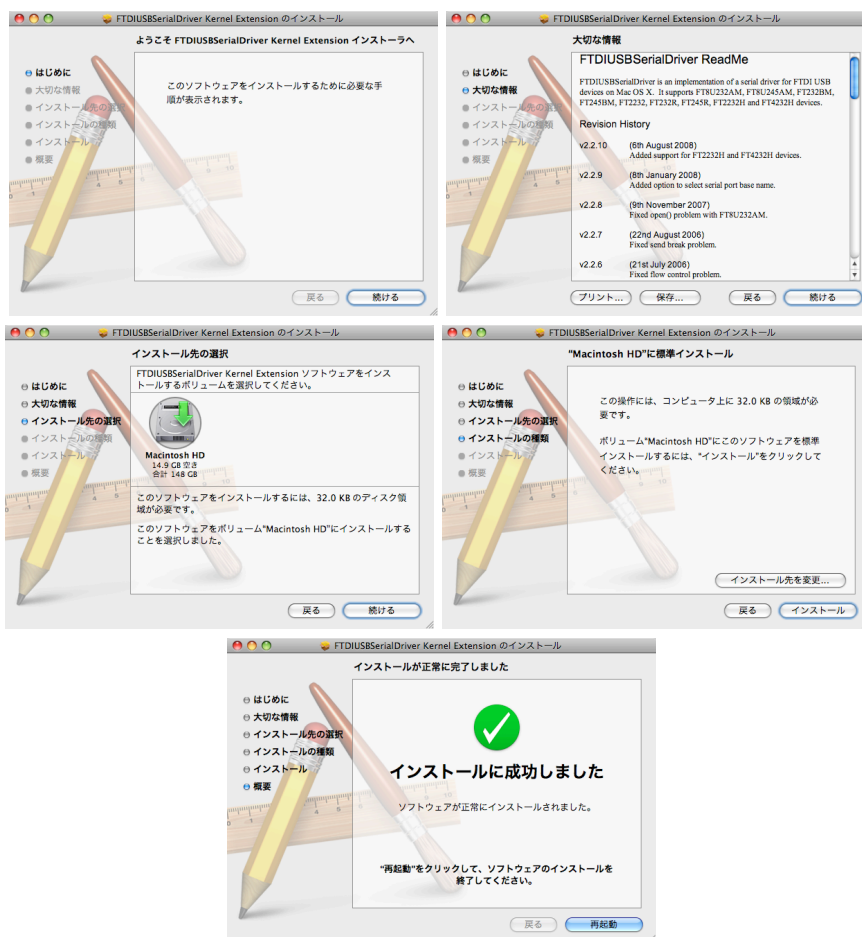


図3 ドライバーインストール時の画面例（Mac OS X 10.5.6）

4 ハードウェアの準備

4.1 Gainer I/O モジュール

Gainer I/O モジュールの場合には、特にハードウェア側の準備は必要ありません。USB ケーブルで PC と接続し、後述する説明に従って Funnel Server を起動するだけですぐに利用できます。

4.2 Arduino ボード

Arduino ボードとの接続には「Firmata」^{*13}を使います。Firmata は Hans-Christoph Steiner が MIDI を参考に考案したプロトコルです。Arduino 0018 にはこの Firmata ライブラリが標準で含まれているため、ライブラリのサンプルを書き込むだけで簡単に I/O モジュールとして利用できるようになります。

Arduino 0018 をウェブサイト^{*14}からダウンロードし、以下の順で操作してください。

1. [Tools] → [Board] から適切なモデルを選択
2. [Tools] → [Serial Port] から該当するシリアルポートを選択
3. [File] → [Examples] → [Firmata] → [StandardFirmata] を選択
4. Upload ボタンを押してアップロード^{*15}

アップロードは 10 数秒で終了するはずです。もしエラーメッセージが表示された場合には、再度アップロードからやり直してみてください。

^{*13} <http://firmata.org/>

^{*14} <http://arduino.cc/>

^{*15} Arduino NG などでは必要に応じてアップロード前にリセットボタンを押してください。

4.3 XBee 無線モジュール

4.3.1 802.15.4 シリーズの場合

Windows の場合には、Digi が提供する XBee 用の設定ツール X-CTU^{*16}が利用できます。`hardware/xbbe/multipoint`に入っている `coordinator.pro` を PC に接続するコーディネータに、`enddevice.pro` をセンサなどに接続するエンドデバイスに（必要に応じて設定を変更した上で）それぞれ書き込んでください。

Mac OS X の場合には、`tools/XBeeConfigTerminal` を利用して必要なコマンドを手動で入力することにより、設定を行うことができます^{*17}。表 1 にコーディネータとエンドデバイスの設定例を示します。なお、出力側も利用するためにはファームウェア 1.0.C.D が必要になります。

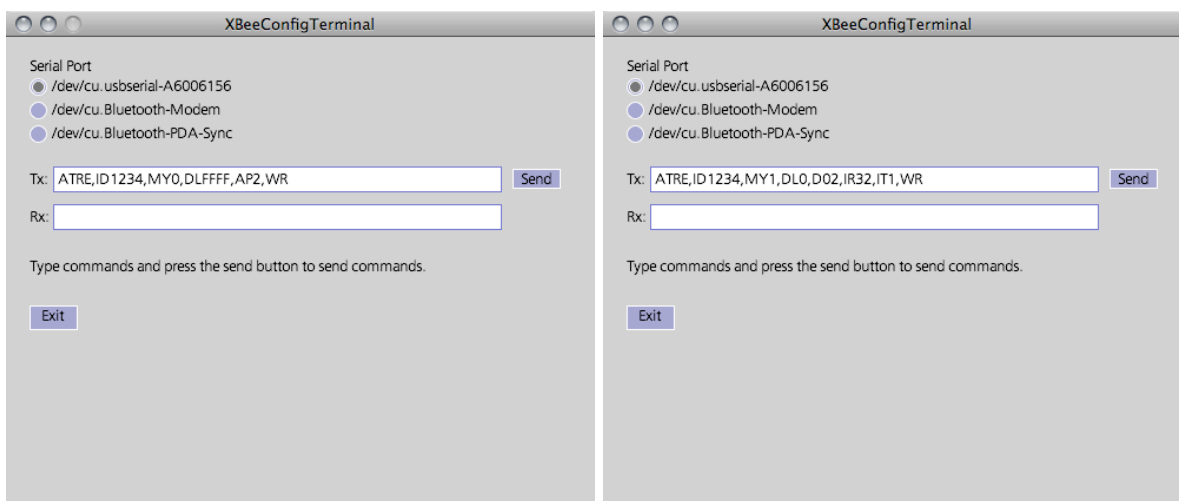


図 4 XBeeConfigTerminal での設定例：コーディネータ側（左）とエンドデバイス側（右）

	コーディネータ	エンド・デバイス	備考
ATRE	-	-	設定をリセット
ATID	1234	1234	PAN ID
ATMY	0	1	
ATDL	FFFF	0	相手側の ID
ATDO	-	2	AD0/DIO0 をアナログ入力に
ATIR	-	32	サンプリング間隔 (0x32 = 50ms)
ATIT	-	1	ここで指定したサンプル数ごとに送信
ATAP	2	-	API モードの設定
ATWR	-	-	設定を書き込む
ATCN	-	-	コマンドモードを抜ける

表 1 802.15.4 シリーズの設定例

^{*16} <http://www.digi.com/support/productdetl.jsp?pid=3352&osvid=57&tp=4&s=316>

^{*17} 現時点ではファームウェアのアップデートには X-CTU が必要となるため、Boot Camp などを利用して Windows を利用できる状態にしておくことをおすすめします。

4.3.2 ZB ZigBee PRO シリーズの場合

hardware/xbec/zpに入っている coordinator.pro を PC に接続するコーディネータに、router.pro をセンサなどに接続するルータに（必要に応じて設定を変更した上で）それぞれ書き込んでください。ファームウェアは、コーディネータ側には 21xx（Coordinator - API Operation）を、ルータ側には 22xx（Router - AT/Transparent Operation）を書き込みます。表 2 に XBeeConfigTerminal を利用して手動で設定を書き込む場合の参考となる設定例を示します。なお、これは最小限の実験用の構成ですので、実際にメッシュネットワークを構成する際にはルータとエンドデバイスを用意してそれぞれ適切に設定してください。

	コーディネータ	ルータ	備考
ATRE	-	-	設定をリセット
ATID	1234	1234	PAN ID
ATJV	-	1	コーディネータのチャンネルを確認
ATDL	FFFF	0	相手側の ID
ATD1	-	2	AD1/DIO1 をアナログ入力に
ATIR	-	32	サンプリング間隔（0x32 = 50ms）
ATAP	2	-	API モードの設定
ATWR	-	-	設定を書き込む
ATCN	-	-	コマンドモードを抜ける

表 2 ZB ZigBee PRO シリーズの設定例

4.4 Arduino Fio および FIO ボード

FIO は LilyPad Arduino Main Board v1.6^{*18} をベースに XBee を搭載できるようにしたものです。Arduino Fio は FIO のマイコンを ATmega328P に変更し、Arduino の公式ラインナップに加えるにあたっていくつかの変更を加えたものです。これらのボードを入出力ボードとして使うには、Arduino IDE によるファームウェアの書き込みと XBee 無線モデムの設定を行う必要があります。XBee は設定がシンプルな 802.15.4 を推奨します。

4.4.1 ハードウェアの設定

1. XBee Explorer USB または XBee starter kit : RTS と D3 の間をジャンパで接続する
2. 電源スイッチ近くのはんだジャンパを閉じる (FIO ボードの場合のみ、Arduino Fio ボードでは不要)

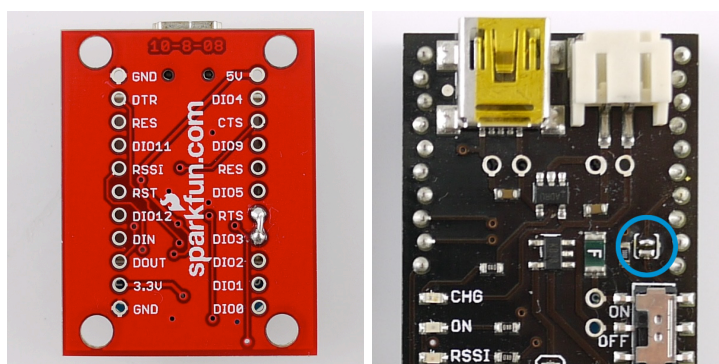


図5 XBee Explorer USB のジャンパ設定 (左) と FIO ボードでののはんだジャンパの設定 (右)

4.4.2 COM ポートの設定 (Windows のみ)

1. デバイスマネージャから COM ポートのプロパティを表示する
2. 「詳細設定」ボタンを押して設定画面を開く
3. 「その他のオプション」で「クローズ時の RTS 設定」をチェックする

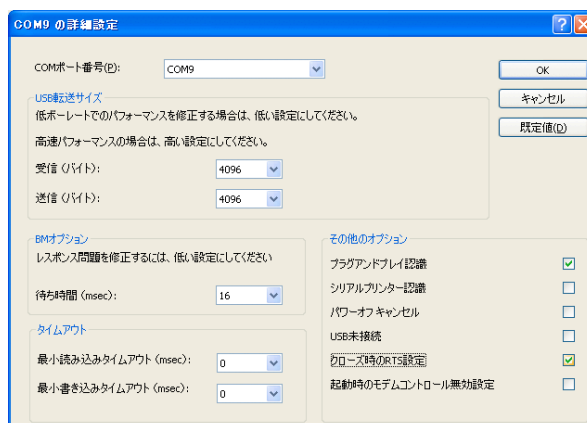


図6 Windows での COM ポートの設定

^{*18} http://www.sparkfun.com/commerce/product.info.php?products_id=8465

4.4.3 XBee の設定

無線でスケッチをアップロードするには、XBee のファームウェアのバージョンが 1.0.A.5 以降である必要があります。XBee は 1.0.8.4 などの古いバージョンで販売されていることが多いため、その場合には X-CTU を使用して 1.0.A.5 以降にアップデートしてください。次に、XBee に設定を書込みます。設定を書き込みには、`tools/XBeeConfigTool` をを使うとよいでしょう。このツールの使用 방법은以下の通りです。

1. Serial Port で XBee USB Explorer などが接続されているシリアルポートを選択します。
2. Mode として Programming Radio (PC に接続する側) か Arduino Fio radio (Fio ボードに搭載する側) を選択します。
3. Baud rate として適切な値を選択します (Arduino Fio ボードであれば 57600bps、FIO ボードであれば 19200bps)。
4. PAN ID を適切な値に設定します。
5. MY ID を適切な値に設定します (Programming radio 側は通常は 0000 で固定)。
6. Configure ボタンを押すと数秒間で書き込みが完了します。

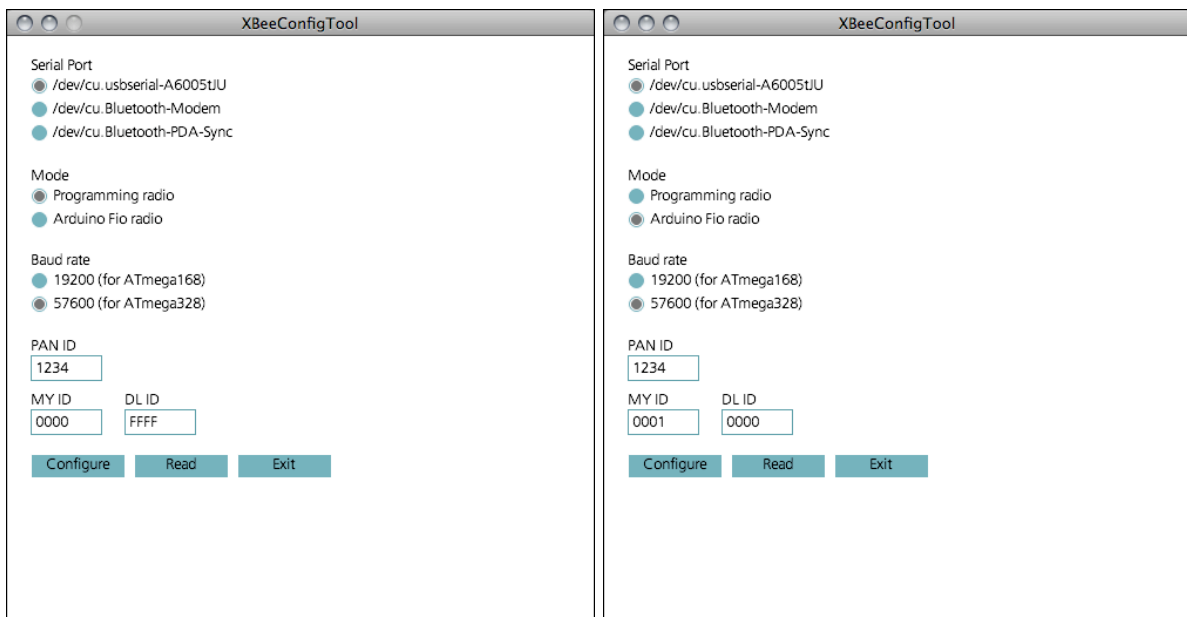


図 7 XBeeConfigTool での設定例：PC に接続したプログラミング側（左）と Fio ボード側（右）

これ以外の方法として、Windows であれば X-CTU を使う方法があります。`hardware/fio/xbec` に入っている `coordinator_auto_reset.pro` (自動リセットあり) または `coordinator.pro` (自動リセットなし) を PC に接続するコーディネータに、`enddevice.pro` を FIO に搭載するエンドデバイスに (必要に応じて設定を変更した上で) それぞれ書き込んでください^{*19}。

以下は参考用として手動で設定を書き込む際の手順です。手動で設定を書き込む場合には、XBeeConfigTerminal を使用すると便利です。

^{*19} 自動リセットの設定方法は <http://www.ladyada.net/make/xbec/arduino.html> で紹介されていた方法を参考に一部改変したものです。

	コーディネータ	エンド・デバイス	備考
ATRE	-	-	設定をリセット
ATBD	6	6	通信速度を 57600bps に
ATID	1234	1234	PAN ID
ATMY	0	1	
ATDL	FFFF	0	相手側の ID
ATD3	3	5	DIO3 を DIN に／DIO3 を HIGH に※
ATIC	8	-	DIO Change Detect で DIO3 の変化を検出※
ATIU	-	0	I/O Output を無効に※
ATIA	-	FFFF	I/O Input Address を設定※
ATWR	-	-	設定を書き込む
ATCN	-	-	コマンドモードを抜ける

表 3 Arduino Fio 用の XBee 802.15.4 シリーズの設定例

4.4.4 ファームウェアの書き込み

Arduino 0018 をウェブサイト^{*20}からダウンロードし、以下の順で操作してください。

1. [Tools] → [Board] から [Arduino Pro or Pro Mini (8MHz) w/ ATmega328] (Arduino Fio の場合) または [Arduino Pro or Pro Mini (8MHz) w/ ATmega168] (FIO の場合) を選択
2. [Tools] → [Serial Port] から該当するシリアルポートを選択
3. [File] → [Sketchbook] → [Open] で hardware/fio/firmware/StandardFirmataForFio を選択
4. Upload ボタンを押す

アップロードは 10 数秒で終了するはずです。もしエラーメッセージが表示された場合には、再度アップロードからやり直してみてください。

5 各ライブラリごとの準備

5.1 ActionScript 3

libraries/actionscrip3/examples/に各ハードウェア用のサンプルがあります。例えば、Gainer I/O モジュール用のサンプルは GainerTest.as、Flash IDE 用のファイルは GainerTest.fla、パブリッシュ済みの Flash Player 用ファイルは GainerTest.swf になります。Flash IDE 以外でパブリッシュする場合には、それぞれの環境に合わせて libraries/actionscrip3/src/をソースパスに設定して下さい。

リスト 1 mxmclc でサンプルをコンパイルする際のオプションの設定例

```
$ mxmclc GainerTest.as -sp ../src
```

次に、ウェブブラウザで <http://tinyurl.com/2rg3lq>^{*21}にアクセスして、「グローバルセキュリティ設定」の「これらのファイルとフォルダを常に信頼する→追加→フォルダを参照...」でサンプルのフォルダを追加してください。この設定を行わないと、Funnel Server と Flash Player の通信ができません。

5.2 Processing

1. 書類フォルダ^{*22}に Processing/libraries/funnel/というフォルダを作成してください。
2. libraries/processing/にある library/と examples/をその中にコピーしてください。

^{*20} <http://arduino.cc>

^{*21} http://www.macromedia.com/support/documentation/jp/flashplayer/help/settings_manager04.html

^{*22} Windows XP の場合にはマイ ドキュメント、Windows Vista の場合にはドキュメント、Mac OS X の場合には書類

3. この状態で Processing を起動し、[Sketch] → [Import Library...] のメニュー項目として「funnel」が表示されるのを確認してください。

5.3 Ruby

libraries/ruby/examples に Gainer、Arduino、XBee、Arduino Fio（または FIO）用のサンプルがあります。また、libraries/ruby/examples/action-coding/にある action-coding 用のサンプルを実行する場合には、<http://code.google.com/p/action-coding/wiki/Tutorial> を参照して action-coding の設定を済ませてください。

6 Funnel Server の設定と起動

Funnel Server の設定と起動は ActionScript 3 と Ruby の場合に必要です。Processing の場合には必要ありません。配布パッケージの **server** フォルダの中に入っている Windows 用 (funnel-server-win.zip) または Mac OS X 用 (funnel-server-mac.dmg) の圧縮ファイルを開き、デスクトップなどの適当な場所にコピーしてからダブルクリックして起動します。起動すると、図 8 の左側のようなウィンドウが表示されます。もし、FunnelServer 自体が起動しない場合は、J2SE 5 以上の Java 実行環境がインストールされているかどうか確認してください。無事に起動できたら、ウィンドウの下部、左側のメニューで適切なボードのタイプを選択し、その後で右側のメニューでシリアルポートを選択します。ポートの選択が終わるとハードウェアとの接続を行い、その結果がメッセージとして表示されます。図 8 の右側は Arduino ボードに接続した例です。

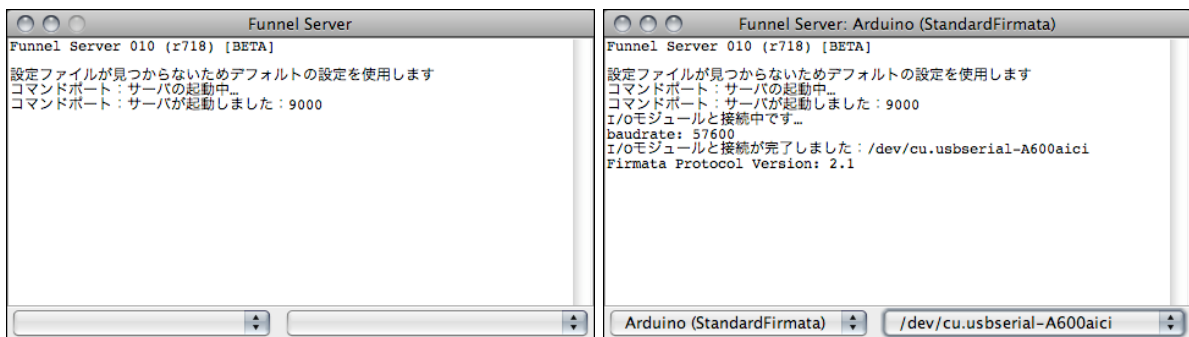


図 8 最初に起動した直後の状態（左）とボードとポートを選択して接続が完了した状態（右）

7 各ライブラリごとの動作確認

7.1 ActionScript 3

`libraries/actionscript3/examples/`にあるサンプルを Flash Player で再生してください。それぞれのサンプルで実装されている内容に関しては、それぞれのサンプル中のコメントを参照してください。

7.2 Processing

もし動作確認のために Funnel Server が起動している場合には終了してください。Open ボタンをクリックし、`examples` フォルダの中からそれぞれのハードウェアのスケッチを開いて実行してください。それぞれのサンプルで実装されている内容に関しては、スケッチ中のコメントを参照してください。

7.3 Ruby

Ruby から `libraries/ruby/examples/` フォルダの中からそれぞれのハードウェアのスク립トを開いて実行してください。それぞれのサンプルで実装されている内容に関しては、スク립ト中のコメントを参照してください。