

センサーネットワークにおける無線モジュールの利用について

三重大学 工学部・工学研究科 技術部

○山本好弘

yamamoto@elec.mie-u.ac.jp

1. はじめに

報告者が所属する研究室では、無線通信システム、ユビキタスネットワーク（通信）等の通信、ネットワーク系の研究を行っている。その研究テーマの一つとして、フィールドセンサーネットワークがある。この研究に行うにあたり、小電力の無線モジュールとしてサン マイクロシステムズ（現オラクル）の SunSpot（無線センサーネットワークデバイス）を用いてきた。

SunSpot は、プロセッサボード、センサーボードおよび電源（バッテリー）から構成されており、IEEE802.15.4 に準拠した無線ネットワーク機能、各種センサー（3 軸加速度、照度、温度等）機能を有する。なお、ホスト PC との通信は、センサーボードを省いたベースステーションと呼ばれるモジュールが行う（通常のモジュールはリモートと呼ぶ）。

SunSpot には、小型デバイス向けの Squawk Java VM が搭載されており、Java プログラムが直接実行可能となっている。開発は、NetBeans（統合開発環境）に SunSpot API の拡張を行い Java プログラムの作成、および USB 経由によるプログラムの書き込みを行うことにより行う。

高級言語である Java によって、無線ネットワーク、センサーアプリケーションの作成などが容易で使い易い優れたデバイスであったが、現在は入手が困難となっており、これに代わる新たなデバイスとして、情報が豊富で入手し易い XBee（ディジ インターナショナル製）の使用を考えて見る事にした。

2. XBee について

XBee はディジ インターナショナルが開発している小電力無線モジュールで、これまで様々なタイプのものであったが、現在の主なものとしては無線ネットワークに IEEE802.15.4 を用いたシリーズ 1、シリーズ 1 に ZigBee プロトコル（ZigBee アライアンス）を構築したシリーズ 2 がある。また、各シリーズに無線出力が 10mW の PRO タイプと 2mW の標準タイプがある。なお、シリーズ 2 にはディジ インターナショナル独自の ZNET プロトコルを搭載したものもあるが、ZigBee プロトコルの使用を推奨している。

なお、通常 XBee には SunSpot のように制御プログラムを書き込むことが出来ないため、外部の制御用マイコン（PC）からシリアル通信による制御が必要となる。

2. 1. ネットワーク構成

XBee のシリーズ 1 では、SunSpot と同じようにメッシュネットワークをサポートしていないので、メッシュネットワークとして使用するためには、外部の制御プログラムを作成するか、ファームウェアを書き換えて使用する必要がある（Digi-Mesh：ディジ インターナショナル独自のメッシュネットワーク）。

外部の制御プログラムの開発（特にネットワーク関連）については、マイコン関連の開発環境等が整っていないため困難が予想される。そこで、今後ネットワーク関連については、XBee の機能を利用することとし、関連情報が多い ZigBee プロトコルを選択した。

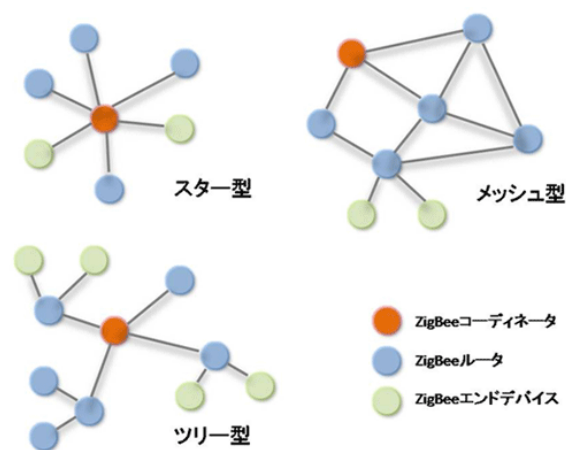


図1 ZigBee のネットワーク構成

ZigBee のネットワークには、ネットワークの形成、制御を担うコーディネータ、ルーティング情報に基づくメッセージ伝送を行うルータ、および任意に配置するエンドデバイスから構成される。エンドデバイスは、他のデバイスに対し直接メッセージの伝送を行うことが出来ないため、ルータを経由して行う必要がある。

図 1 に ZigBee のスター型、ツリー型、およびメッシュ型のネットワークの構成例を示す。ルータからルーティングの機能を除いたものがエンドデバイスであり、従ってエンドデバイスとルータのみの構成も可能であるが、ルータとして使用する際には常時稼働させる必要があるため、省電力を考慮するのであれば極力ルータの使用を控えるようにネットワークを構成する必要がある。

各ネットワーク構成を識別するためのコード (PAN ID) が用意されており、デバイス間の通信を行うには、PAN ID、チャンネル (無線周波数) を同一にする必要がある。これらの情報を XBee に設定するためのツール (X-CUT) がディジ インターナショナルより無償で提供されている (図 2)。

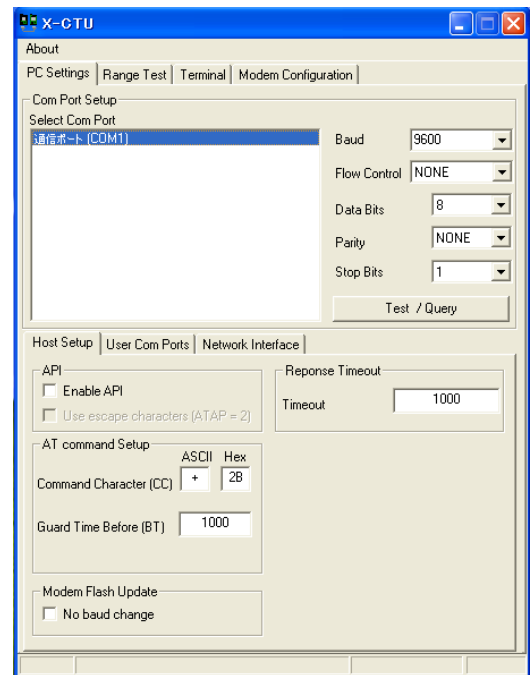


図 2 X-CUT の操作画面

2. 2. XBee によるメッセージの伝送

XBee 間でのメッセージの伝送、および制御を行うためには、XBee とマイコン間でシリアル通信 (RS-232C) による制御コマンド、データの通信を行う必要がある。制御コマンドとして AT コマンドが用意されているが、1 対 n のネットワーク構成時のメッセージ伝送を行うには、複数の AT コマンドを使用する必要があるため、煩雑な操作が必要となる。そのため 1 対 n の通信を行うためのモードとして API モードが用意されている。

API モードは、AT コマンド、ZigBee プロトコル関連のコマンド、およびデータをフレームの送受信を行うだけで実現するモードである。

3. Arduino について

XBee の動作させるためには、マイコンによる制御が基本となるため、簡単な動作検証を行うにはマイコン (ボード) の準備、および制御プログラムの開発がかなりの負荷となる。そこで、制御マイコンとして、オープンソースハードウェアである Arduino を使用することとした。

Arduino はイタリアで生まれたオープンソースハードウェアで、AVR マイコン (アトメル) にブートローダを実装し、統合開発環境 (Arduino IDE) からプログラムを書き込むことにより動作させるマイコンボードである。現在 Arduino には様々な種類があるが、標準的な Arduino UNO では 16 本のデジタル I/O (内 6 本が PWM 出力可)、6 本のアナログ入力を用意されている。また、シールドと呼ばれる様々な拡張インターフェースボードが開発されており、それらシールド専用のライブラリにより容易に制御可能となっている。

プログラミング言語 (Arduino 言語) は、Wiring という言語から派生した構造化言語で、制御を行うための最適化が図られており、マイコン向けの C 言語に比べ容易にプログラム開発が行える。

今回は、Arduino に Xbee 用のシールドを用い動作確認を行った (図 3)。

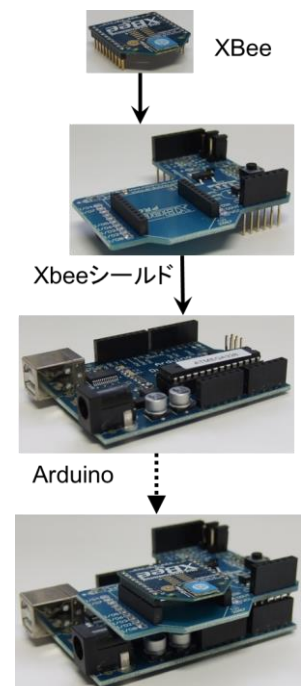


図 3 Arduino ボード

4. おわりに

Arduino のアナログ入力端子を用い、収集したセンサーデータの送受信等の動作確認実験を行った。今後は、センサーネットワークの構築、およびデータの蓄積・表示システムの開発を行う予定である。また、フィールドでの使用を前提とした、省電力での使用の可能性を探る。

参考文献

- 1) 水原文 訳, 「XBee によるワイヤレスセンサーネットワーク」, オライリー・ジャパン
- 2) 田原純一郎, 「Arduino で始める電子工作」, カットシステム
- 3) 浜原和明, 「多点センサで計測・制御するならこれ! XBee」, トランジスタ技術 2011 年 9 月号, pp66-81, CQ 出版