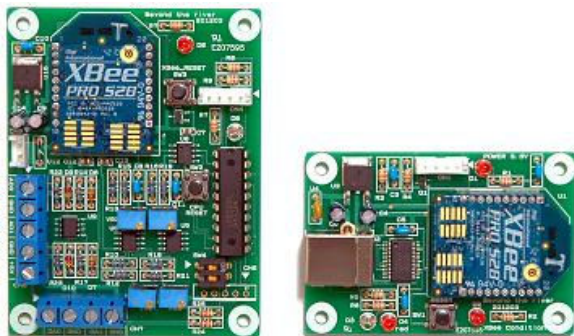


R8C_2ADDA_XBeeセット 取扱説明書

有限会社ビーリバーエレクトロニクス

第1版 2012. 3. 29



第1章 ご使用になる前に

◆セットの内容	3
◆特徴	4
◆各部の機能と名称	5

第2章 使い方

◆ハードウェア接続	6
■USB接続、デバイスドライバインストール	
■A/Dの接続方法	9
■D/Aの接続方法	10
■通信状態LED	12
■配置上のご注意	12
◆コマンド	13
◆ターミナルプログラムからの制御例	14
■コマンドの書き込み、データの読み込み	
■データのロギング、ファイル化、エクセルへの取り込み	
◆調整	20

第3章 使用例

◆温度センサとSSR（ソリッドステートリレー）交流制御、直流制御の接続例	21
--------------------------------------	----

第4章 アプリケーションの開発

◆Windows APIを使用した開発例	22
----------------------	----

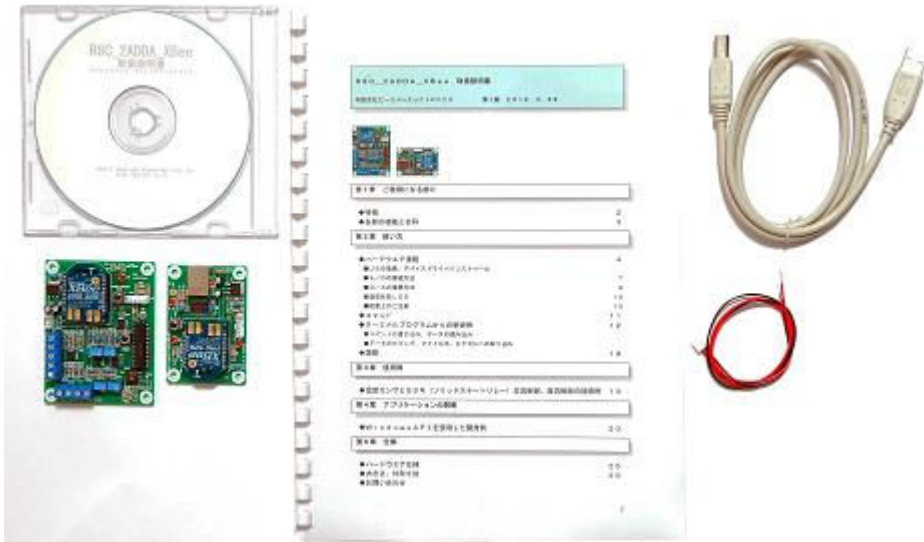
第5章 仕様

◆ハードウェア仕様	27
◆大きさ、外形寸法	28
◆お問い合わせ	

第1章 ご使用になる前に

◆セットの内容

ご確認下さい。



- デバイスドライバ、ウィンドウズソフト等CD 1
- コーディネータ基板、エンドデバイス基板 各1
- マニュアル
- USBケーブル
- 電源ケーブル

◆特徴

■ 本製品は12ビット2chのA/D入力と、12ビット2chのD/A出力をもつ**エンドデバイス基板（子機）**と、パソコンとUSBでつながり、エンドデバイス基板と無線で送受信を行う**コーディネータ基板（親機）**から構成されます。

■ エンドデバイス基板の A/Dは0~5V入力、D/Aは0~5V出力です。+5V 100mA以上の電源が必要です。

■ 送受信ユニットはXBeePRO 52Bで、ZigBee規格に対応する無線通信用です。出力10mW (+10dBm) でエンドデバイス基板とコーディネータ基板間を最大1.5Km離して操作することができます。(屋外最良条件時)。電波法技術基準適合品 (TELEC認証取得済み)。使用に当たって特に申請等は不要です。

■ 大きさはエンドデバイス基板が79×61×15mmです。重さは約35g、コーディネータ基板が65×44×15mmです。重さは約20g。

■ 送受信ユニットはそれぞれ1台のエンドデバイス、コーディネータ用に弊社で設定済みです。背面に貼ってあるPAN ID 数値が同じもの同士で動作します。ユーザーは無線ユニットの存在を意識せず使用でき

ます。

■コーディネータ基板の電源はUSBより供給されます。

■簡単なコマンドで動作します。ターミナルプログラムでロギング、ファイル化、エクセル読み込みなど
ができます。

■1組（エンドデバイス+コーディネータ）単位で同時複数動作可能（混信しません）。

■動作温度範囲 -20～85℃

◆用途

■測定対象が移動、回転するもの 例：自動車の走行中データサンプリング

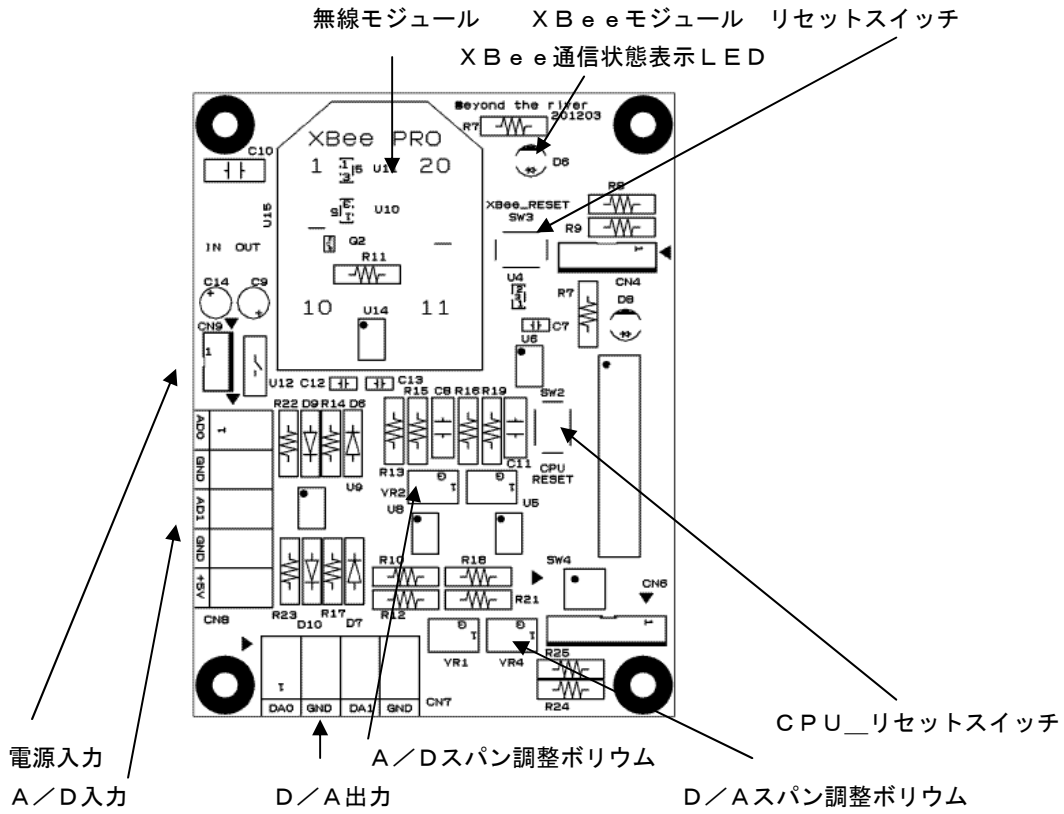
■遠隔地の測定、設定 例：自宅でビニールハウスの温度測定

■高温、低温、低酸素、異臭、大騒音、環境での無人測定、設定。

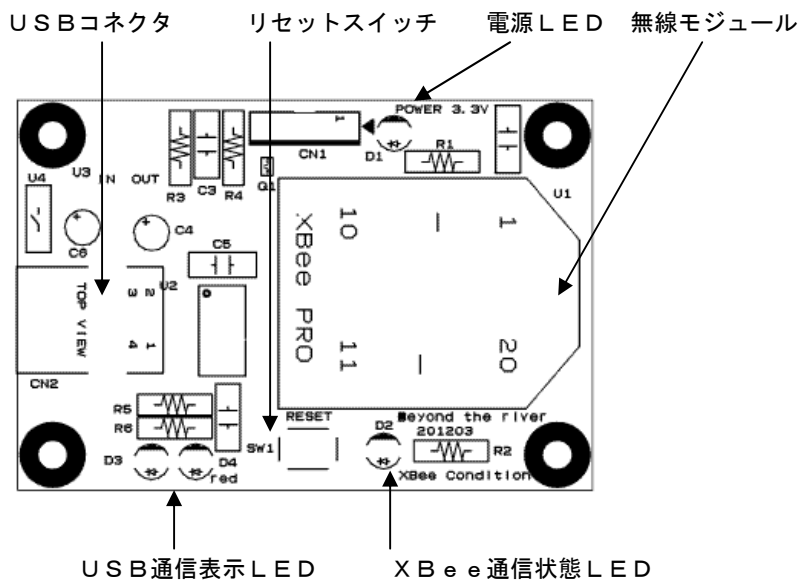
例：-20℃環境でのデータサンプル

◆各部の機能と名称

エンドデバイス基板



コーディネータ基板



第2章 使い方

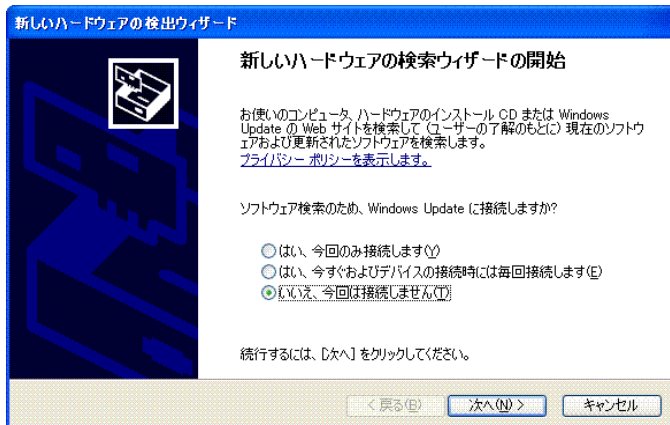
◆ハードウェア接続

■USB接続、デバイスドライバインストール

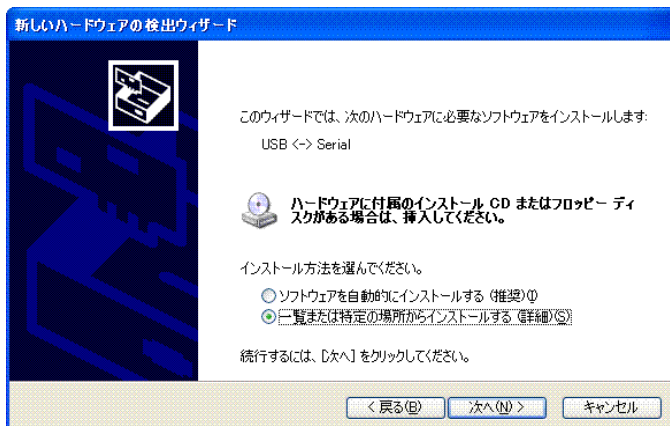
本基板はUSBインターフェイスにFTDI社のFT232RLを使用しています。このICは仮想COM、ダイレクトドライバの2種類の使い方が可能ですが、添付CDのデバイスドライバをインストールすることにより、2種類共使用可能になります。

初めて本基板をパソコンに接続すると「新しいハードウェアが検出されました」と表示され、「新しいハードウェアの検出ウィザードの開始」が表示されます。デバイスドライバの設定を行います。(下記例はWindowsXPのウィザード例)

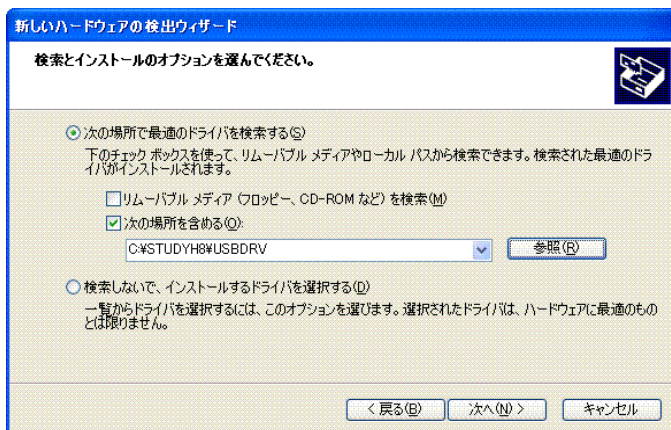
WindowsUpdateへの接続は「いいえ、今回は接続しません」を選択し、「次へ(N) >」をクリックしてください。



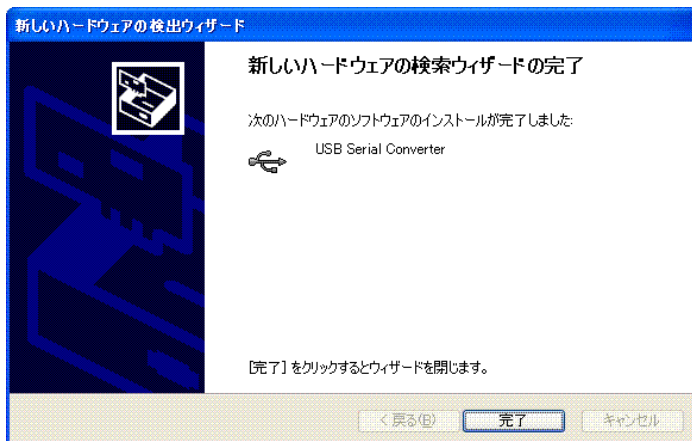
「一覧または特定の場所からインストールする(詳細)(S)」を選択し、「次へ(N) >」をクリックしてください。



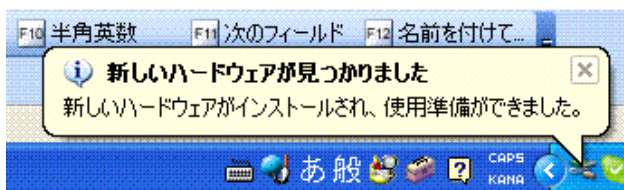
通常のインストールでは「参照(R)」をクリックし添付CDのX:\USBDRVを選択します。「次へ(N) >」をクリックしてください。



インストールが正常に終了したら「新しいハードウェアの検索ウィザードの完了」が表示されますので、「完了」をクリックしてください。その後、再びウィザードが立ち上がりますが、同じように繰り返してください。(仮想 COM ドライバおよびダイレクトドライバ D2XX インストゥールで 2 回行います)

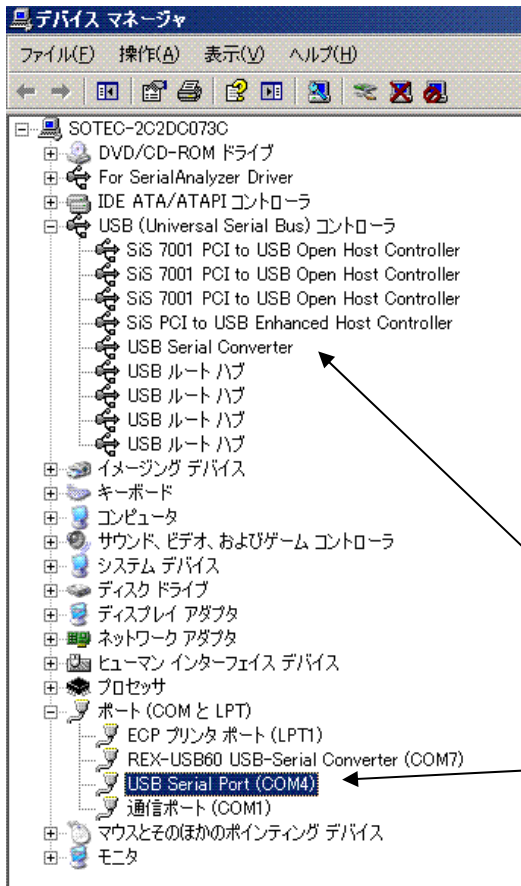


「新しいハードウェアがインストールされ、使用準備ができました」と表示されたら OK です。



これで USB の初期設定は終わりです。次回からは USB ケーブルを挿入すれば USB として認識され動作します。

コントロールパネル→システム→ハードウェア→デバイスマネージャを見ることにより、2 種類のデバイスドライバがインストールされたことが確認できます。



2種類のデバイスドライバ。USBコネクタを抜くとこれらは消えます。

USB Serial Port (COM4) のCOM番号は変更可能です。変更方法は別紙、「COMポートの使い方」をご参照下さい。

なお、デバイスドライバのアンインストールは「USBDRV」の中にあるFTDIUNIN. exeを実行します。



実行するとハードディスク内の xxx. inf ファイルが削除されてしまいますので、再インストールする場合、元のCDから再コピーするか、xxx. inf ファイルを別ディレクトリに退避してから実行してください。

最新のデバイスドライバ、OS別のデバイスドライバ等はFTDI社のホームページよりダウンロード

できませんが、WindowsXP、WindowsVista、Windows7 はこのデバイスドライバで大丈夫です。※2

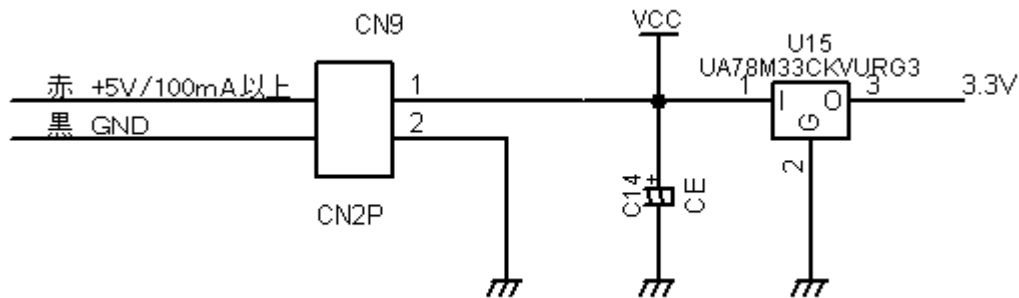
※1 Windows 7では自動検索されて、正常にインストールできない場合があります。その場合、手動でコントロールパネル→システム→デバイスマネージャーで個別のデバイスマネージャーを開き、上記ドライバを再設定して下さい。

※2 FTDI社のデバイスドライバダウンロードサイトではLinuxやWindowsCE等のデバイスドライバもダウンロードできます。



■電源の接続

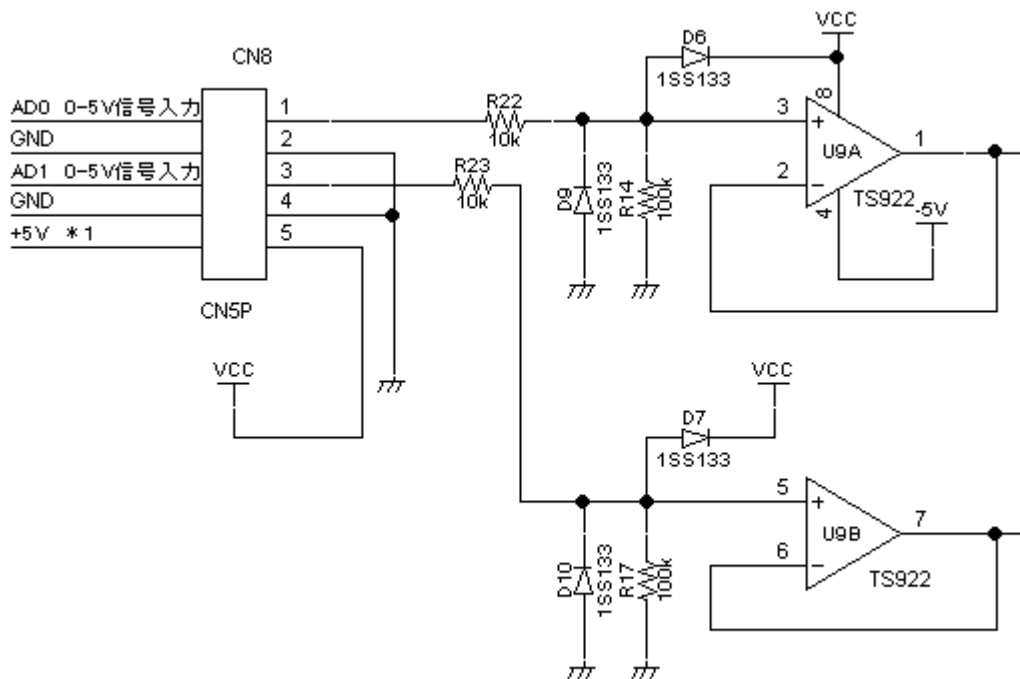
電源は5V、100mA以上の安定したノイズのないものをCN9に添付のケーブルで接続してください。D/Aコンバータはこの電圧以上の電圧を出力することはできません。



■A/Dの接続方法

図は入力端子CN8と内部の回路を示しています。

- 1) 信号は0~5Vの範囲として下さい。マイナスの電圧や、過大な電圧は入力しないで下さい。
- 2) 受け側の入力インピーダンスは110KΩです。



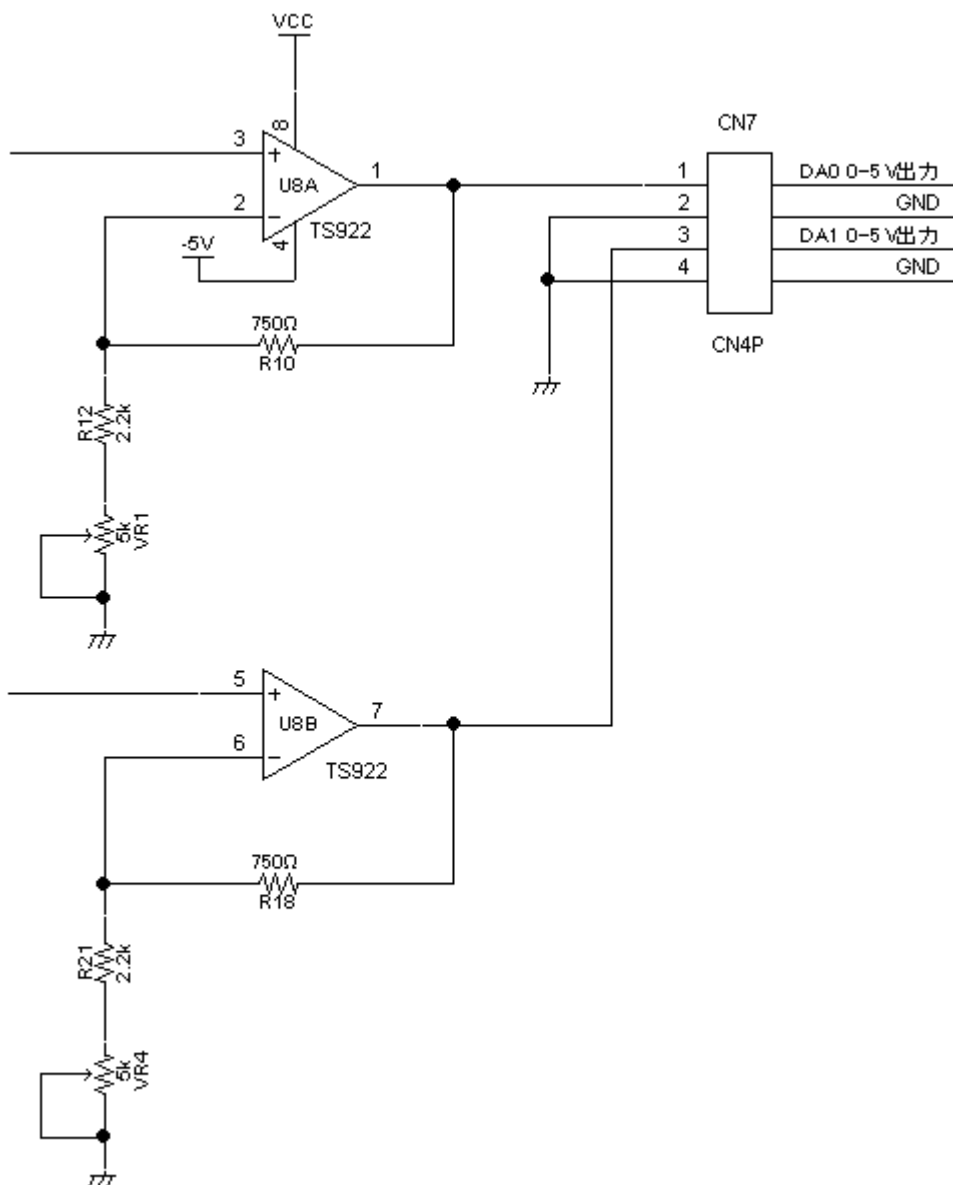
※1 +5V入出力

ここはCN9 1番とつながっています。CN9を使用せず、ここから電源5Vを供給する（入力）ことも可能ですし、CN9に電源を入れ、ここからセンサ等の電源に使用（出力）することも可能です。

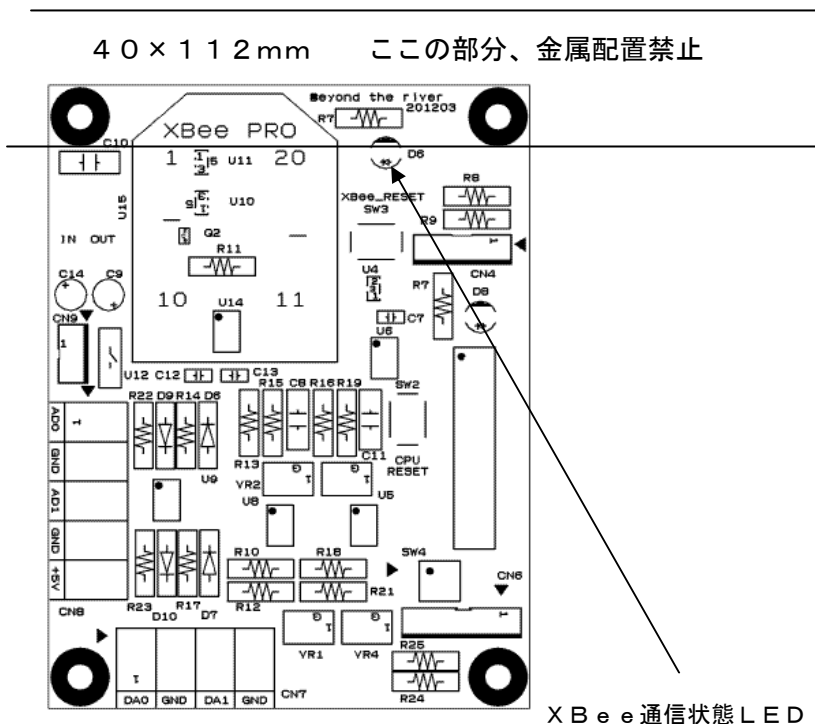
■ D/Aの接続方法

図は出力端子CN7と内部の回路を示しています。

- 1) 出力信号は0～5Vの範囲です。
- 2) 受け側の入力インピーダンスは十分大きい500Ω以上のもので受けて下さい。出力電流が10mA以上流れると出力最大電圧が低下します。
- 3) 出力に10mA以上の電流が必要な場合、外部に電流増幅用の回路を追加して下さい。



■通信状態LED



コーディネータとエンドデバイス間でお互いを認識しているとエンドデバイス側のXBee通信状態LEDが短周期でフリッカします。認識できていないときは長周期のフリッカになります。

認識しているときで、エンドデバイスが省電力モードに入っていないときは約0.6秒周期のデューティ50%のフリッカ、省電力モードに入ったときは約0.3秒周期の90%以上ON周期になります。(ちらちら消えるように見えます)

省電力モードはコーディネータ基板に10秒以上、信号が入力されない場合、エンドデバイス側無線モジュールの消費電力を自動的に減らすモードです。

このように1つのLEDが3つの状態を示しますので、それを見て、無線の状況を確認することが可能です。

電源投入時、コーディネータ基板とエンドデバイス基板の存在認識に数秒必要とします。認識して、接続が確立(XBeeモジュールで勝手に動作)するまでは通信はできませんので、ご注意ください。

■配置上のご注意

■金属ケース等、電波を通さない筐体(導電性物質)に入れて使用することはできません。

■XBeeの三角の部分はアンテナになっています。上図のように40×112mmの部分に、金属を配置して使用することはおやめ下さい。

◆コマンド

コマンドはパソコン側から本ボードに送信し、コマンドに応じた返信があります。

- 1) シリアル設定は38400BPS、8ビットデータ、スタート、ストップ1ビット、パリティビットノンです。
- 2) データは全てアスキーコードです。
- 3) コマンドの終わりはENTER (0x0d) または\$です。表では\$で表示しています。

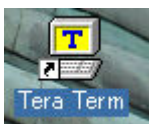
AD入力コマンド		
コマンド	返信	動作
AD0\$	AD0=nnnnmV\$	AD0入力のA/D値を知る nnnn : 0000~5000mV AD0の入力電圧をmV単位で返信します。
AD1\$	AD1=nnnnmV\$	AD1入力のA/D値を知る nnnn : 0000~5000mV AD1の入力電圧をmV単位で返信します。

DA出力コマンド		
コマンド	返信	動作
DA0=nnnn\$	DA0=nnnnmV\$	DA0出力電圧を設定する nnnn : 0000~5000mV DA0に設定した出力電圧値をmV単位で返信します。
DA1=nnnn\$	DA1=nnnnmV\$	DA1出力電圧を設定する nnnn : 0000~5000mV DA1に設定した出力電圧値をmV単位で返信します。

総合入力コマンド		
コマンド	返信	動作
AL\$	AD0、1のデータ	AD値 2つを返信します。
AT=nnnn\$	AD0、1の指定時間連続出力	AD値、2つを返信します。 nnnn : 0.1秒から9999秒(166.65分)までの指定された時間で連続出力します。
ST\$	ST\$	ATコマンド動作停止

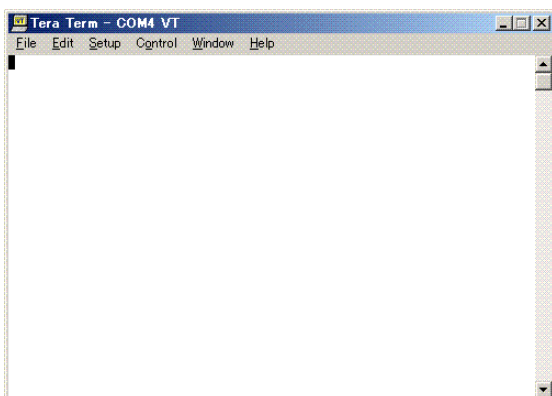
◆ターミナルプログラムからの制御例

無償で使用可能な「Tera Term」を使った本ボードの操作方法を示します。ハイパーターミナル等も同様に使用可能です。

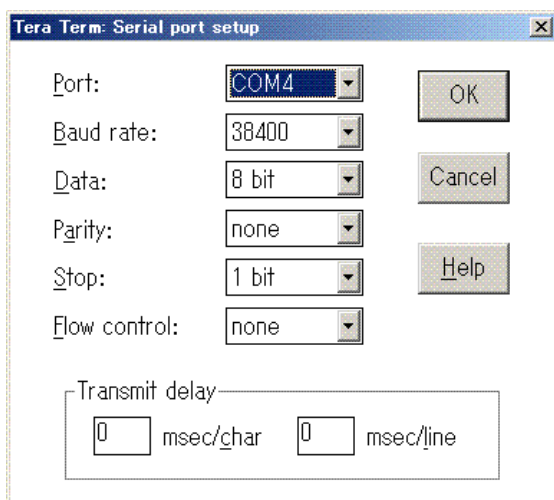


「Tera Term」は窓の杜等からダウンロードできます。

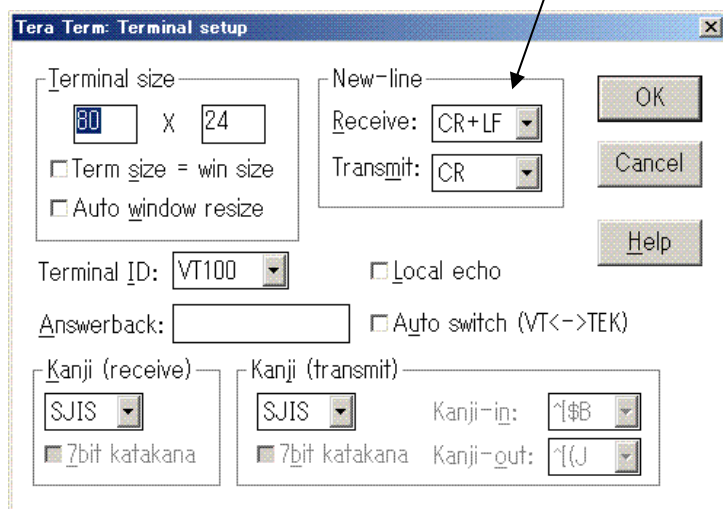
1) 設定



Setup→Serial port setupを以下のように設定します。(COM4の場合)

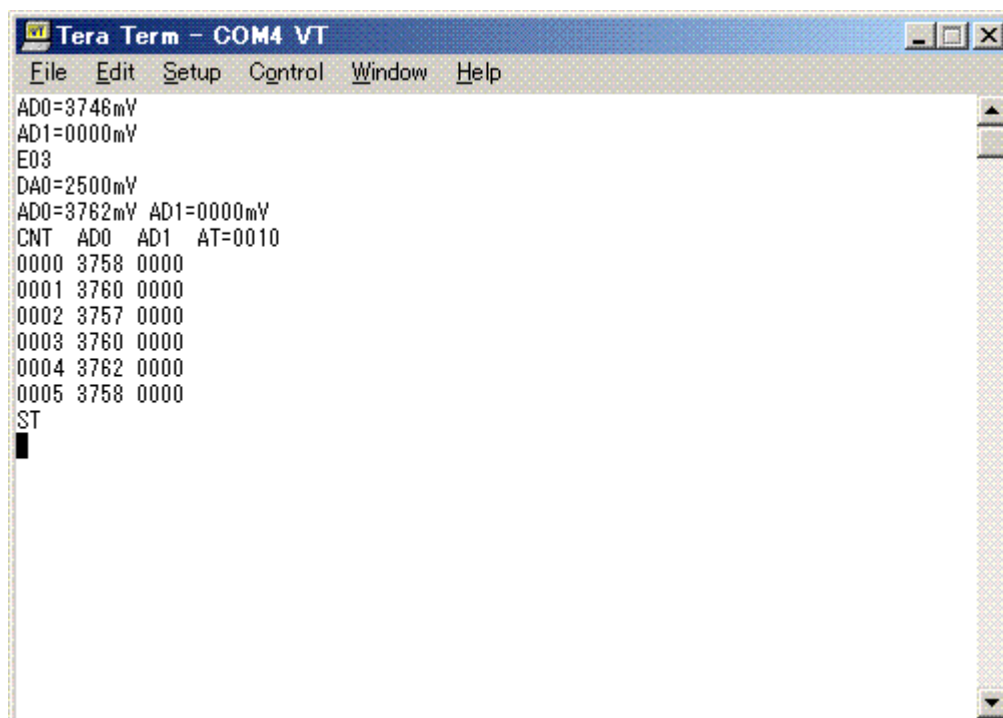


Setup→TerminalでNew-line、ReceiveをCR+LFに変更します。これが無いと表示が改行しません。



■コマンドの書き込み、データの読み込み

以下はいくつかコマンドをキーボードから入力して、本ボードを動作させた例です



- 1) キーボードで ADO「ENTER」と入力。ENTERはアスキーコードでは0x0dです。画面上は表示されません。
- 2) 本ボードから返信 ADO=3746mV (0x0d)。ADOは3.746V入力されています。
- 3) キーボードで AD1「ENTER」と入力
- 4) 本ボードから返信 ADO=0000mV (0x0d)。

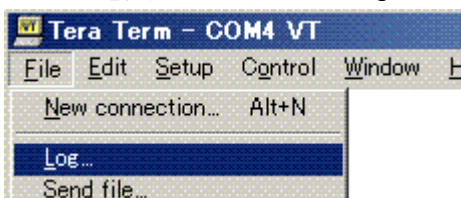
- 5) キーボードで `xxxx E03` エラー 間違ったコマンドでした。
- 6) キーボードで `DA0=2500` 「ENTER」と入力。DA0に2.5Vを出力します。
- 7) 本ボードから返信 `DA0=2500mV`
- 8) キーボードで `AL` 「ENTER」と入力
- 9) 本ボードから返信 `AD0=3762mV AD1=0000mV`
- 10) キーボードで `AT=0010` 「ENTER」と入力
- 11) 本ボードから返信


```
CNT AD0 AD1 AT=0010
0000 3758 0000
0001 3760 0000
```

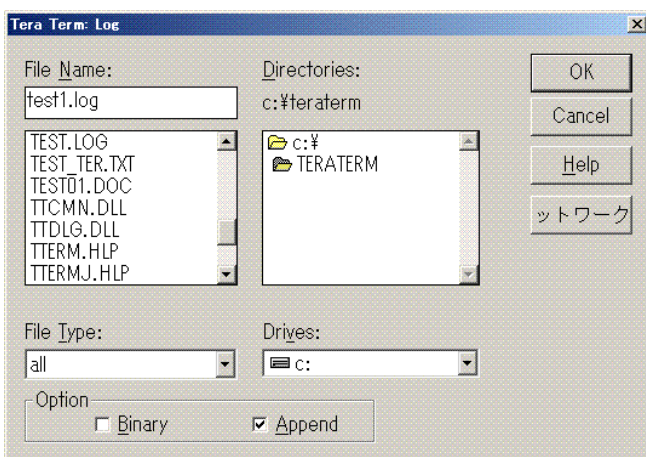
AT=0010=1秒ごとにカウンター値(0~9999)、AD0とAD1の値が送信されます。
- 12) キーボードで `ST` 「ENTER」と入力 ATモードが終了しました。

■データのロギング

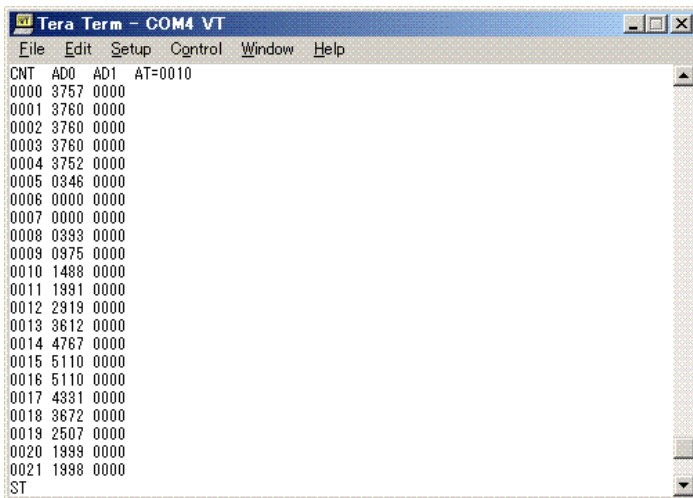
データを受信し、ファイル化して、エクセルで開く方法をご紹介します。初めにデータをセーブしたいファイル名を設定します。File→Log ファイルを選択。



例として test1.log を新規に作成。OK をクリックします。

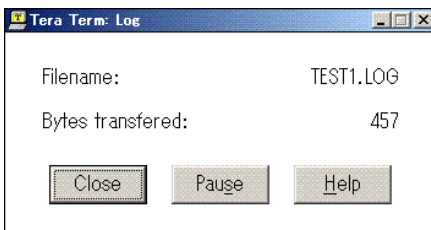


例としてキーボードから AT=0010 「ENTER」と押し、1秒間隔でデータを受信します。

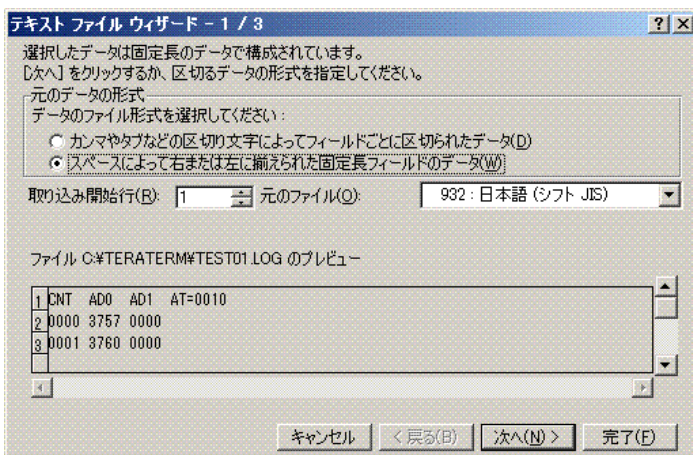


ST「ENTER」で停止させます。

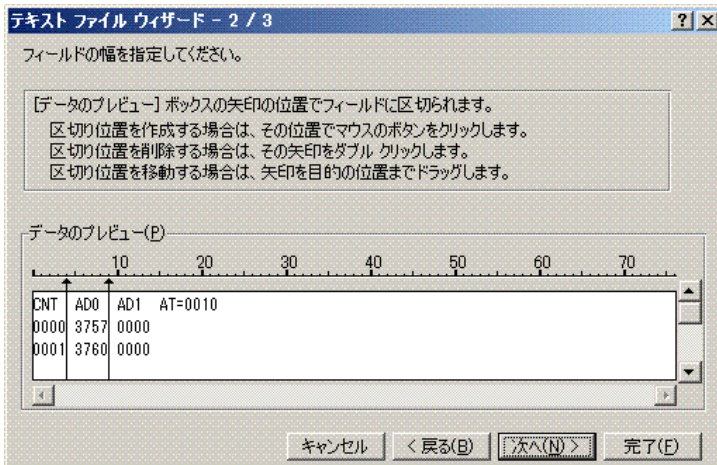
Log ファイルを「Close」させます。右のタスクボタンがログファイルです。



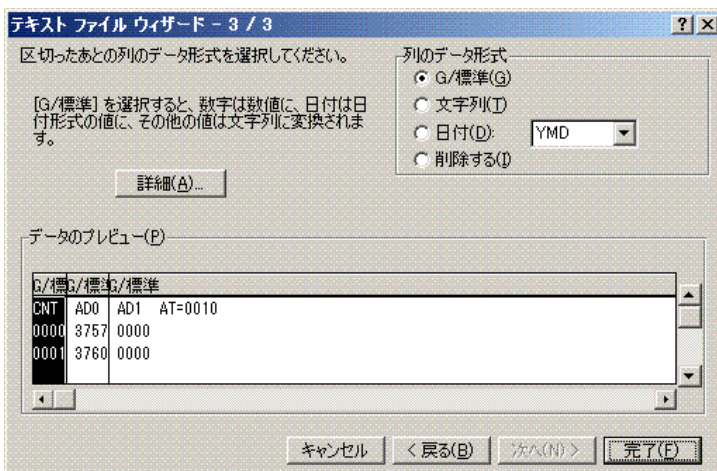
エクセルで「test1.log」を開きます。以下のようなウイザードが開始されます。



次を選択



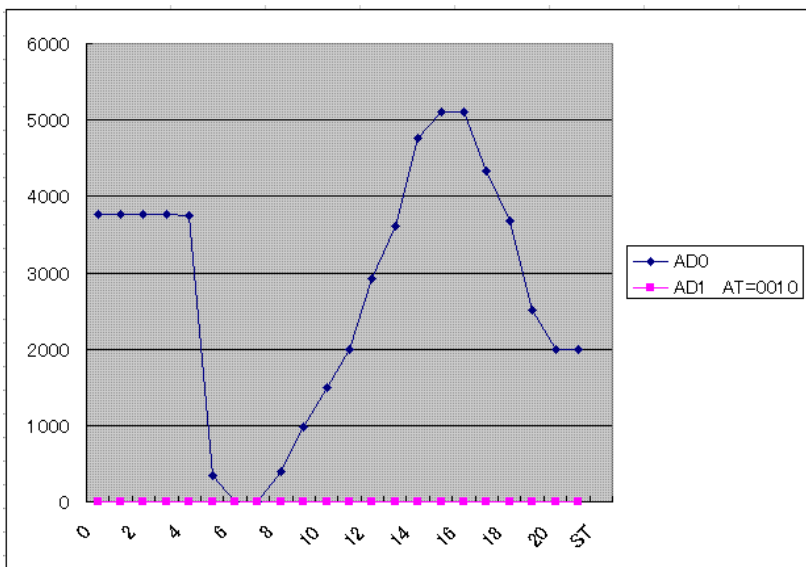
次を選択



「完了」を選択。エクセルに以下のように数値が展開されます。

Microsoft Excel - TEST01.LOG				
ファイル(F) 編集(E) 表示(V) 挿入(I) 書式(O) ?				
R26C1				
	1	2	3	4
1	CNT	AD0	AD1	AT=0010
2		0	3757	0
3		1	3760	0
4		2	3760	0
5		3	3760	0
6		4	3752	0
7		5	346	0
8		6	0	0
9		7	0	0
10		8	393	0
11		9	975	0
12		10	1488	0
13		11	1991	0
14		12	2919	0
15		13	3612	0
16		14	4767	0
17		15	5110	0
18		16	5110	0
19		17	4331	0
20		18	3672	0
21		19	2507	0
22		20	1999	0
23		21	1998	0
24	ST			

グラフにしてみました。

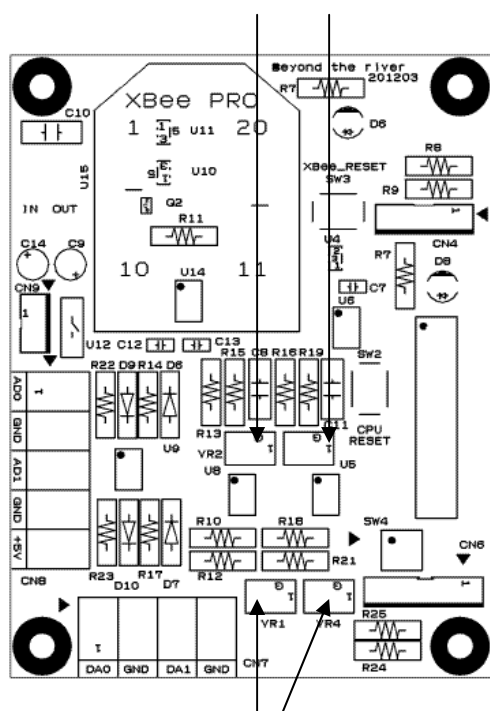


◆調整

A/DやD/Aは分解能12ビット=4095ですので、本来5000mVの1mVを測定する能力はありません。5000/4095=1ビット1.22mVが最小分解能です。演算で1mVを表現しています。また、それ以外にもハードウェアは部品精度、使用温度等の影響を受け、結果として4桁数値の下1桁についてはあまりあてにならない、また詳細な調整はできません。

但し、多少の調整を行いたい場合、方法はあります。VR2はAD0のゲイン、VR3はAD1のゲイン、VR1はDA0のゲイン、VR4はDA1のゲインをそれぞれ少し、調整できます。多回転ポテンシオメータです。回転端まで回すと音がしますので、それ以上回転しても数値は変化しません。

A/Dスパン調整用ポリウム



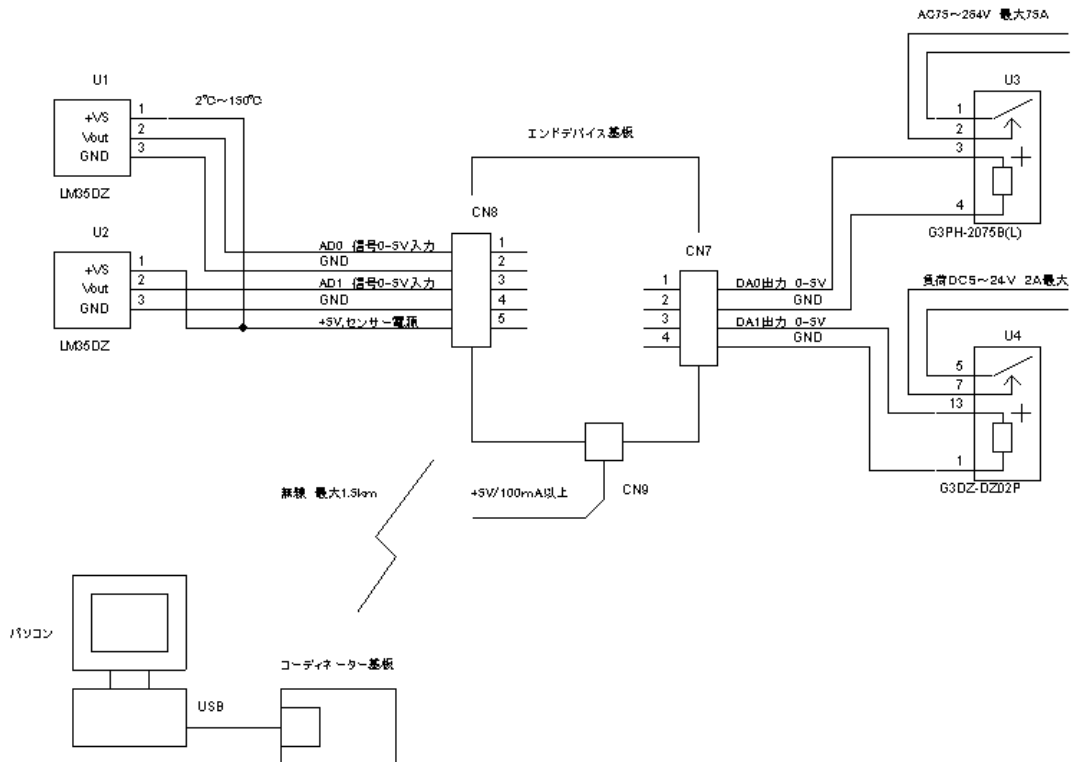
D/Aスパン調整用ポリウム

調整中のAD値の連続読み込み、D/Aの連続出力は4章で説明するパソコンソフトが活用できます。

第3章 使用例

◆温度センサとSSR（ソリッドステートリレー）交流制御、直流制御の接続例

温度センサLM35DZをそれぞれAD0, AD1に接続し、DAでSSR（ソリッドステートリレー）をドライブする例です。



U1、U2は $1^{\circ}\text{C}=10\text{mV}$ 出力の 2°C から 150°C まで計測できる安価な半導体温度センサです。読んだ電圧値がそのまま温度になりますので、変換が不要です。※1

DAはもちろん、 $0-5\text{V}$ にリニアに変化させてモーターの速度を変えたり、ヒーターの熱を変えたりする使い方も出来ませんが、例ではDA0はSSRのU3をON、OFFさせてAC $75\sim 264\text{V}$ 最大 75A までの負荷制御を行っています。制御に必要な電流は 5mA ですので、DAの出力で十分制御できます。 5V を出すとON、 0V を出すとOFFになります。

DA1は直流専用のリレー（MOSFETリレー）、U4を制御してDC $5\sim 24\text{V}$ 最大 2A の負荷をON、OFFさせています。入力インピーダンス 750Ω ですので、 $5\text{V}/750\Omega \approx 6.7\text{mA}$ で 10mA 以内ですので、問題なく制御できます。

接続負荷等で不明な点などありましたら、取説末尾の「お問い合わせ」までお尋ね下さい。

※1 数値が安定しない場合、CN5のAD入力とGND間に $0.1\mu\text{F}$ の積層セラミックコンデンサを挿入してください。

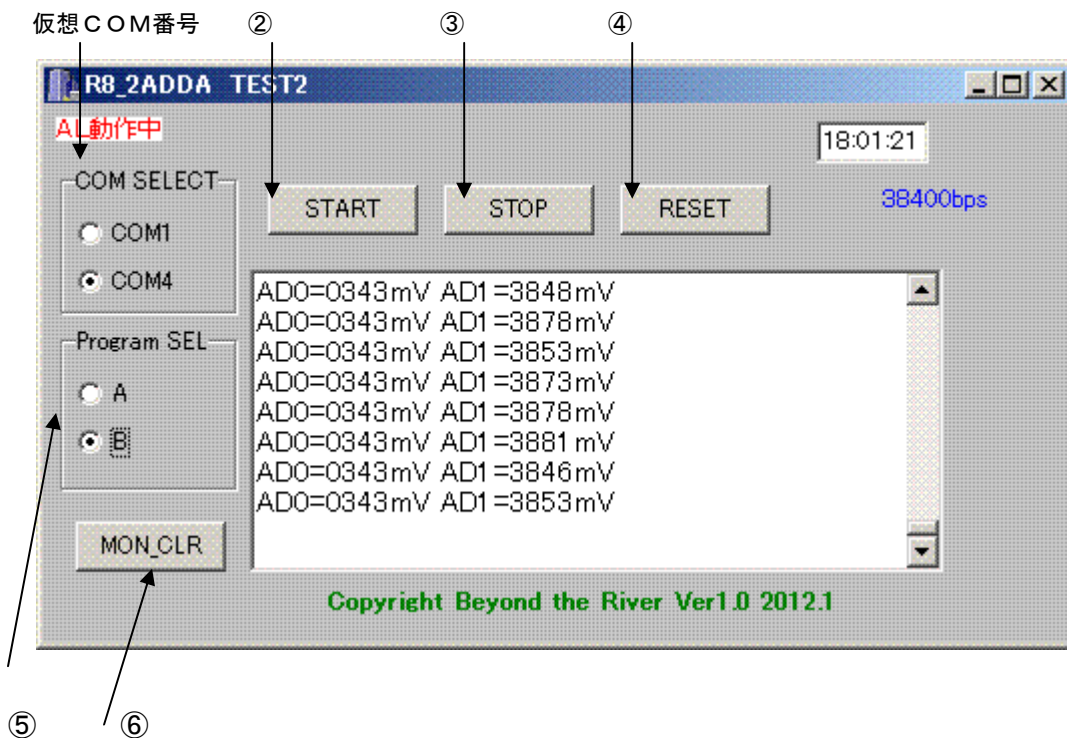
第4章 アプリケーションの開発

◆WindowsAPIを使用した開発例

手動で無く、自動的に本ボードを動かしたい場合、パソコン側アプリケーションプログラムの作成が必要です。

CDに添付しているサンプルソフトは仮想COMを使用した本ボードとの通信例です。動作は2通りで、ProgramSELで切り替えます。SELがAのとき、DA0, 1に0V、5V、2.5Vの電圧を5秒間隔でSTOPがあるまで出力しています。Bのとき、ALコマンドをSTOPがあるまで発行しています。

本ソフトはBolandC++ Builder Ver1.0で作成されていますが、WindowsAPIを使用していますから、他の処理系でもこれを参考に書くことが出来ると思います。



①イニシャル、ボードリセット

ソフトウェア動作開始時にイニシャルでCOM4のボーレート等パラメタを設定して、オープンしています。また本ボードをハードリセットしています。

```
//-----  
__fastcall TForm1::TForm1(TComponent* Owner)  
    : TForm(Owner)  
{  
int cf;
```

```

hCom = CreateFile("COM4",
                 GENERIC_READ | GENERIC_WRITE,
                 0,
                 NULL,
                 OPEN_EXISTING,
                 0,
                 NULL
                );

```

```
fSuccess = GetCommState(hCom, &dcb);
```

```

dcb.BaudRate = 38400;
dcb.ByteSize = 8;
dcb.Parity = NOPARITY;
dcb.StopBits = ONESTOPBIT;

```

```
fSuccess=SetCommState(hCom, &dcb);
```

```

start_flg = 0;
received_flg = 0;

```

```
//-----
```

②START

スタートを押すと、⑤プログラムセルの状態 (RadioButton4) により動作プログラムを選択し、データを送信し、動作させています。

```

char dat[100];
unsigned char cf, loop, c1, c2;
unsigned long WriteNumber;
DWORD dwErrorMask;
COMSTAT comstat;

long ltime;

MessageBeep(0);

start_flg = 1; //送信動作開始フラグ

while(start_flg != 0)
{
    if (RadioButton4->Checked)
    {

```

```

dat[0] = 'A';
dat[1] = 'L';
dat[2] = '$';

WriteFile(hCom, dat, 3, &WriteNumber, NULL);

Label3->Caption = "A L 動作中";

etime = 2;
received_flg = 0;
while(received_flg == 0)
{
    if(etime == 0)
    {
        received_flg = 1;
    }
    Application->ProcessMessages();
    // loop が長いときに必ず入れる
}

etime = 5;
while(etime != 0)
{
    Application->ProcessMessages();
    // loop が長いときに必ず入れる
}

}
else
{
Label3->Caption = "D A 動作中";

dat[0] = 'D';
dat[1] = 'A';
dat[2] = '0';
dat[3] = '=';
dat[4] = '0';
dat[5] = '0';
dat[6] = '0';
dat[7] = '0';
dat[8] = 0x0d;
WriteFile(hCom, dat, 9, &WriteNumber, NULL);

```

以下省略

③STOP

フラグを降ろして、動作を停止しています。

```
void __fastcall TForm1::Button2Click(TObject *Sender)
{
    MessageBeep(0);
    start_flg = 0;
}
```

④RESET

STOPと同じ、フラグクリアです。

```
void __fastcall TForm1::Button4Click(TObject *Sender)
{
    MessageBeep(0);
    start_flg = 0;
}
```

⑤プログラム選択

2通りのプログラムをここで選択します。

⑥モニタ画面クリア

Memo1 をクリアしています。

```
void __fastcall TForm1::Button3Click(TObject *Sender)
{
    Memo1 -> Clear();
}
```

■データの受信、表示

データの受信と表示は定周期割り込みの Timer1Timer の中で行っています。受信バッファが0でない場合、データを読み込み、Memo1 に表示させています。

```
void __fastcall TForm1::Timer1Timer(TObject *Sender)
{
    unsigned long Size;
    DWORD dwErrorMask;
    COMSTAT comstat;
    char dat[50], cf;

    ClearCommError(hCom, &dwErrorMask, &comstat);
        if(comstat.cbInQue > 0) //受信バッファ0?
        {
            for(cf = 0; cf < 50; cf++) {dat[cf]=0;}
                ReadFile(hCom, dat, comstat.cbInQue, &Size, NULL);
            // 受信バッファ0でないので読み込み
                IntToStr(Memo1->Lines->Add(dat)); // データ表示
                received_flg = 1; //受信したフラグ
        }
```

```
        }  
        if(etime != 0) {etime--;}        //  
  
        Edit1->Text=Time();            // 時計表示  
    }
```

詳細はCD中のソースコード拡張子cppを参照下さい。

第5章 仕様

◆ハードウェア仕様

A/D部	仕様
入力点数	2点
分解能	12ビット 1/4095 1ビット 1.22mV
入力抵抗	100KΩ
入力定格	入力0～5V、5V以上、マイナス電圧を印加しないで下さい。
精度	フルスケールの±1%以内
保護回路	ダイオードによる過電圧保護回路
応答速度	4msec平均（コマンドが発行されてから実行されるまで、返信を含む時間） 最高1秒間に250コマンド実行。

D/A部	仕様
出力点数	2点
分解能	12ビット 1/4095 1ビット 1.22mV
出力定格	出力0～5V 電流10mA以内（接続可能負荷インピーダンス500Ω以上） 電源電圧以上の電圧は出力できません
精度	フルスケールの1%以内
応答速度	A/D部と同じ

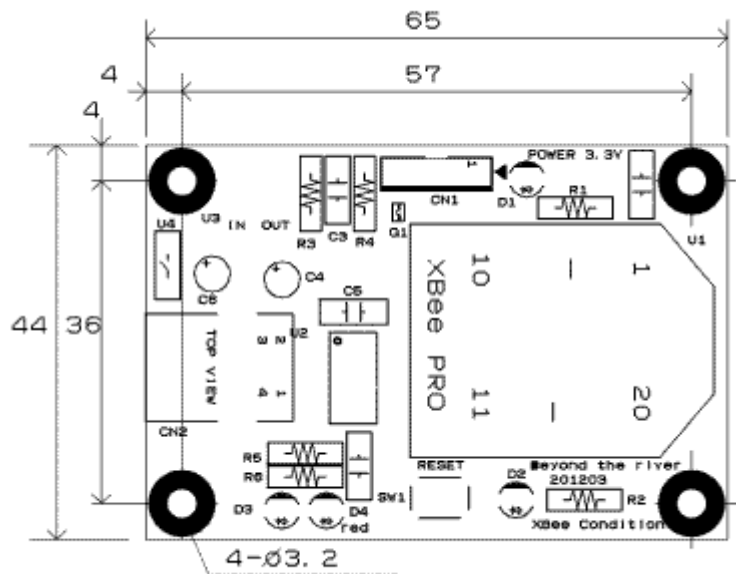
コーディネータ部	仕様
消費電流	本体最大100mA USBから供給
無線モジュール	XBee PRO S2B 通信距離 最大1.5km
使用条件	0～50℃、10～90%RH 但し、結露しないこと ノイズ、導電性粉塵（水を含む）の無いところ
外形寸法	65×44×15mm
質量	約20g
使用USBケーブル	通常サイズ

エンドポイント部	仕様
消費電流	本体最大100mA 外部から供給
無線モジュール	XBee PRO S2B 通信距離 最大1.5km
使用条件	0～50℃、10～90%RH

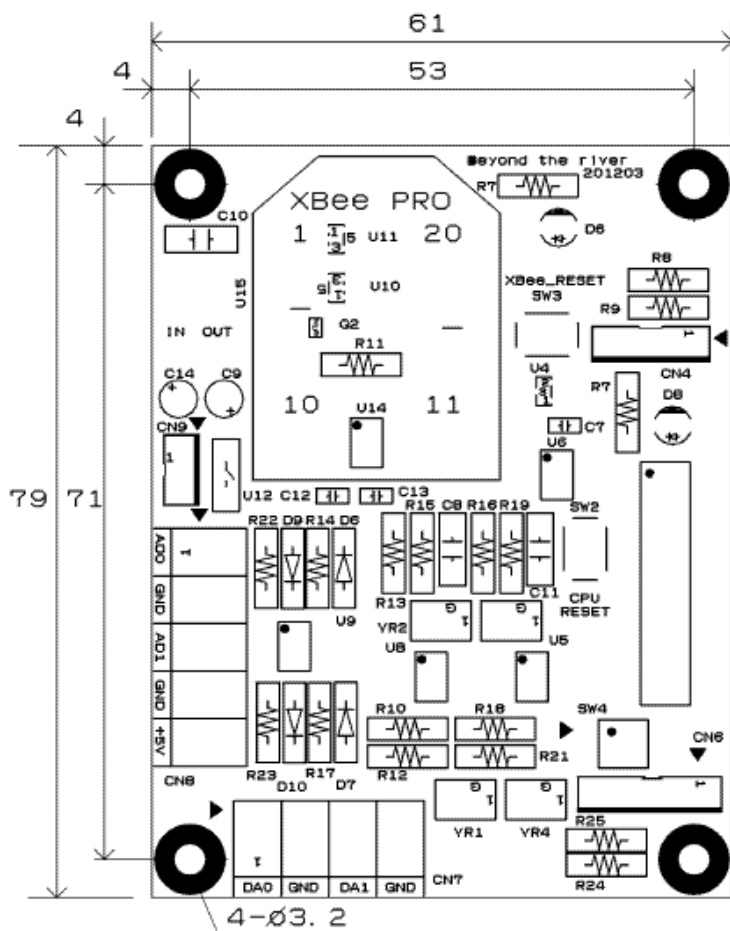
	但し、結露しないこと ノイズ、導電性粉塵（水を含む）の無いところ
外形寸法	79×61×15mm
質量	約35g
端子（4P）、（5P）	マイナスドライバーで線を固定するタイプ。 ED2611-ND ED2612-ND

◆大きさ、外形寸法

コーディネータ基板



エンドデバイス基板



◆お問い合わせ

以下のところにお尋ね下さい。

〒350-1213 埼玉県日高市高萩 1 1 4 1 - 1

TEL 042 (985) 6982 営業時間 9~17時

FAX 042 (985) 6720

e-mail : info@beriver.co.jp

Homepage : <http://beriver.co.jp>

有限会社ビーリバーエレクトロニクス ©Beyond the river Inc. 20120403