

TWE-Liteを使って、温度情報を収集する簡単なセンサネットワークを作りあげていきます。
複数のTWE-Liteを使い、実践的なセンサネットワークを構築していきます。

3日目 午後
(9/16 am)

VI. TWE-Lite演習 センサネットワークの構築

1. 簡単なリモートコントロール①
2. 簡単なリモートコントロール②
3. 温度センサネットワーク
4. 演習課題

組込み基礎 I

1. 簡単なリモートコントロール①

- 1対1（または1対2）で、相手方のTWE-Liteをリモートで制御する
 - スwitchのON/OFFで、相手方のLEDを点灯・消灯させる
 - 演習の手順
 - a. TWE-Lite RをPCに接続する（あらかじめTWE-LiteのドライバとプログラマをPCにインストールしておくこと）
 - b. 2台のTWE-Lite(AとBと呼ぶ)に、「超簡単!TWEアプリ」をそれぞれ書き込む。1対2の時はAが1台、Bが2台とする
 - c. グループ毎に独立して動作させるため、AとBの設定を変更する
 - d. ブレッドボード上に回路を組む（Aは電池とスイッチ、Bは電池とLEDをTWE-Liteとつなぐ）
 - e. 動かしてみる



1. 簡単なリモートコントロール①

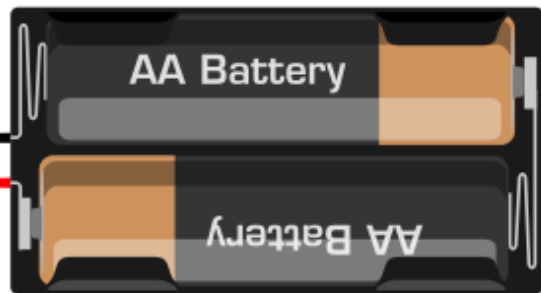
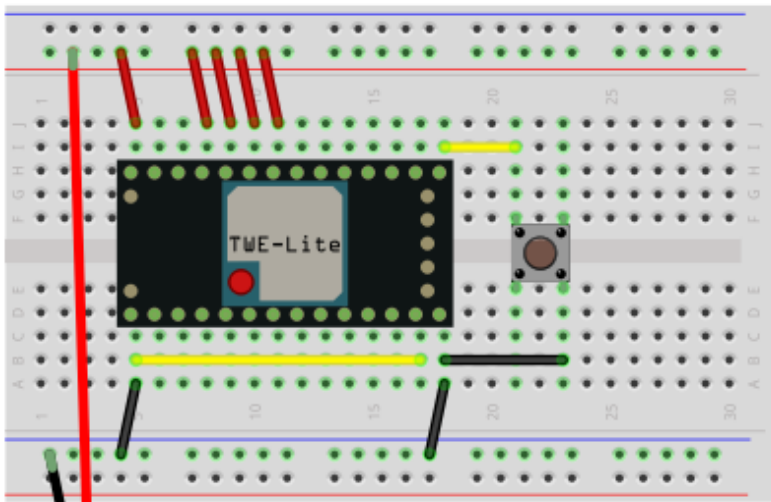
- a. 前述の通り
- b. 前述の通り。ただし、アプリは下記のページにある「ダウンロードページ」のリンク先からダウンロードする
 - http://tocos-wireless.com/jp/products/TWE-ZERO/App_Twelite/index.html
- c. 超簡単!TWEアプリの設定方法
 - 設定には、ターミナルソフトが必要となる。任意のソフトウェアを使えるが、下記のソフトウェア(Tera Term)の利用を推奨する:
<https://osdn.jp/projects/ttssh2/>
 - TWE-Liteは、インタラクティブモードに入ってから設定を行う。詳細は下記参照。
<http://tocos-wireless.com/jp/products/TWE-Lite-DIP/TWE-Lite-DIP-step3-interactive.html>
 - 設定を変更するのは、「アプリケーションID (aコマンド)」のみ。グループ番号を**3**とすると、「**10000003**」を設定して、**S**コマンドで保存する



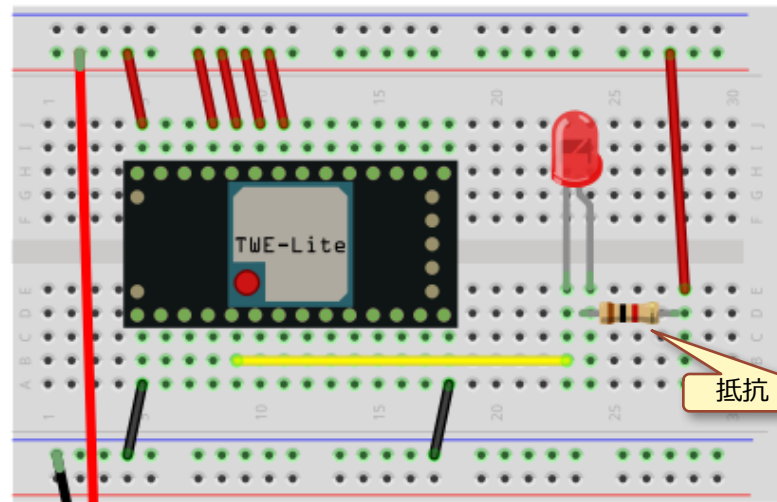
1. 簡単なリモートコントロール①

d. 下記の通りに回路を組む

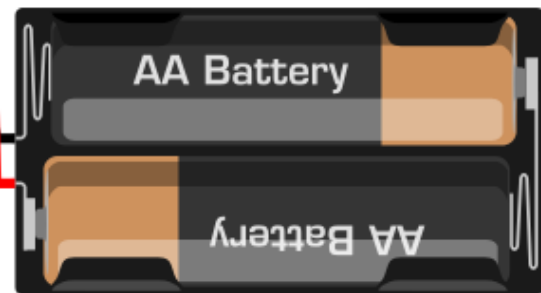
A



B



抵抗 (1kΩ)



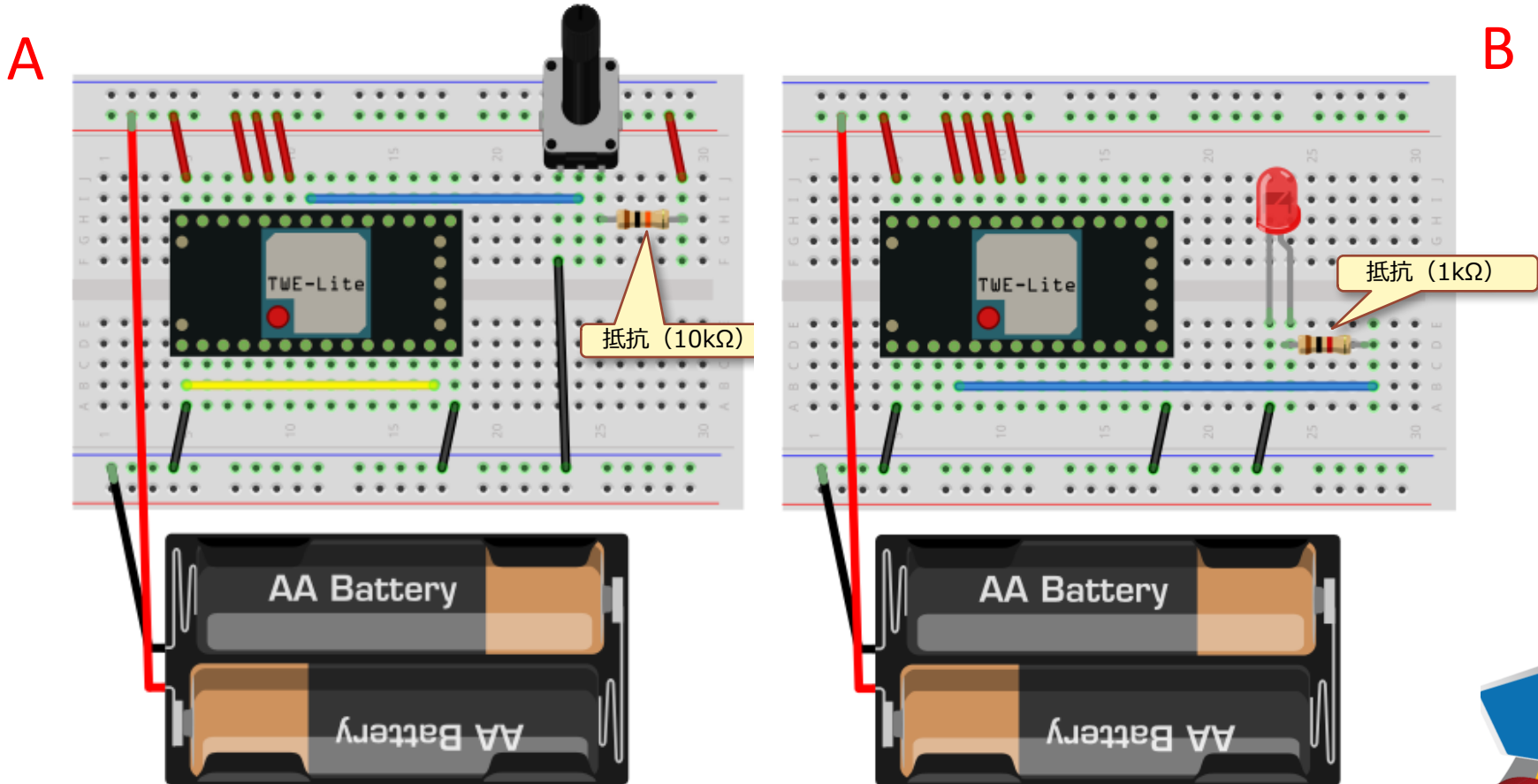
2. 簡単なリモートコントロール②

- 1対1（または1対2）で、相手方のTWE-Liteをリモートで制御する
 - 可変抵抗器を回すと、相手方のLEDの明るさが変化する
 - 演習の手順
 - a. TWE-Lite RをPCに接続する（あらかじめTWE-LiteのドライバとプログラマをPCにインストールしておくこと）
 - b. 2台のTWE-Lite(AとBと呼ぶ)に、「超簡単!TWEアプリ」をそれぞれ書き込む。1対2の時はAが1台、Bが2台とする
 - c. グループ毎に独立して動作させるため、AとBの設定を変更する
 - d. ブレッドボード上に回路を組む（Aは電池と可変抵抗器、Bは電池とLEDをTWE-Liteとつなぐ）
 - e. 動かしてみる

①で実施済みのため、ここでは不要

2. 簡単なリモートコントロール②

d. 下記の通りに回路を組む



3. 温度センサネットワーク

- 子機(1台以上)に接続されている温度センサの情報を親機が収集して、シリアルモニタに表示する

- http://tocos-wireless.com/jp/products/TWE-EH-S/sw_html/mode_LM61.html

– 演習の手順

- a. TWE-Lite RをPCに接続する（あらかじめTWE-LiteのドライバとプログラマをPCにインストールしておくこと）
- b. 2台のTWE-Lite(AとBと呼ぶ)に、「無線タグアプリ」をそれぞれ書き込む。Aには親機用アプリ、B(1台以上)には子機用アプリをそれぞれ書き込む
- c. グループ毎に独立して動作させるため、AとBの設定を変更する
- d. ブレッドボード上に回路を組む（AはArduino、Bは電池と温度センサ等をTWE-Liteとつなぐ）
- e. 親機側のArduino用のスケッチを作成して、書き込む
 - 別途提供するtweliteライブラリを使う
- f. 動かしてみる

3. 温度センサネットワーク

a. 前述の通り

b. 前述の通り。ただし、アプリは下記のページにある「ダウンロードページ」のリンク先からダウンロードする

- 別途提供

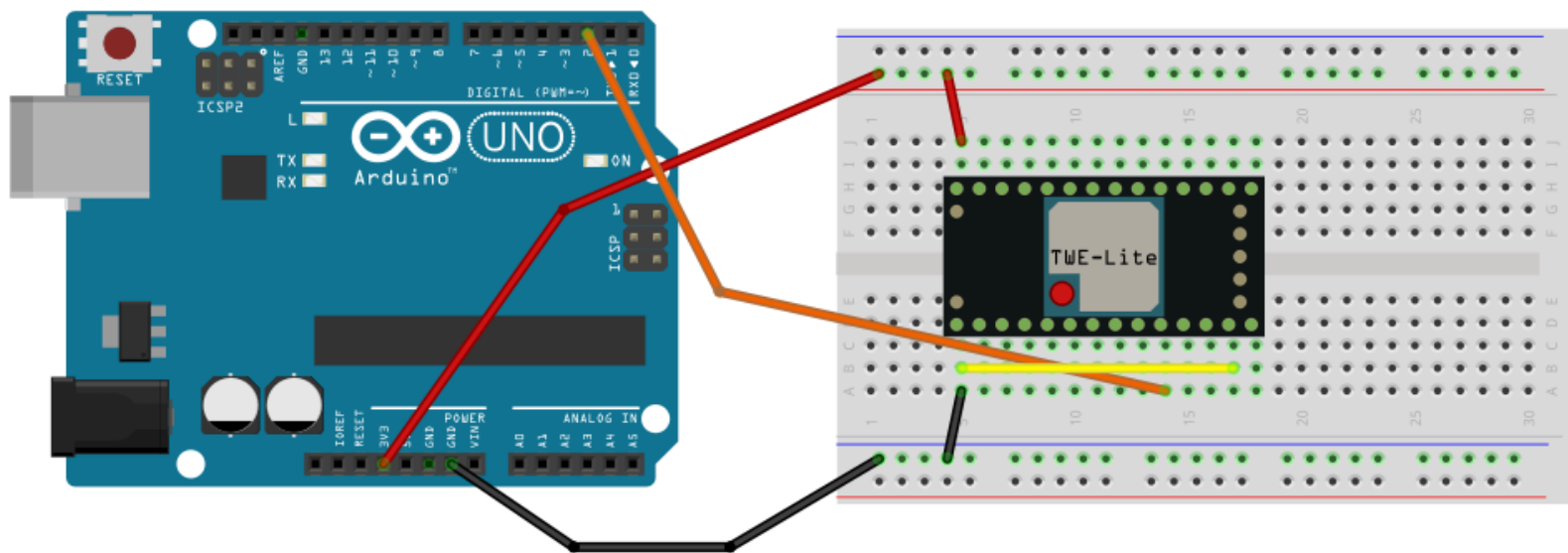
c. 無線タグアプリの設定方法

- 設定には、ターミナルソフトが必要となる。任意のソフトウェアを使えるが、下記のソフトウェア(Tera Term)の利用を推奨する：
<https://osdn.jp/projects/ttssh2/>
- TWE-Liteは、インタラクティブモードに入ってから設定を行う。詳細は下記参照。
http://tocos-wireless.com/jp/products/TWE-EH-S/sw_html/interactive.html
 - 設定を変更するのは、親機は「アプリケーションID (**a**コマンド)」、子機は「アプリケーションID (**a**コマンド)」と「センサ種別の設定 (**m**コマンド)」とする。
 - » グループ番号を**3**とすると、**a**コマンドで「**10000003**」を設定する
 - » 子機の場合に限り、**m**コマンドで「**11**」を設定する
 - » 最後に、**S**コマンドで保存する

3. 温度センサネットワーク

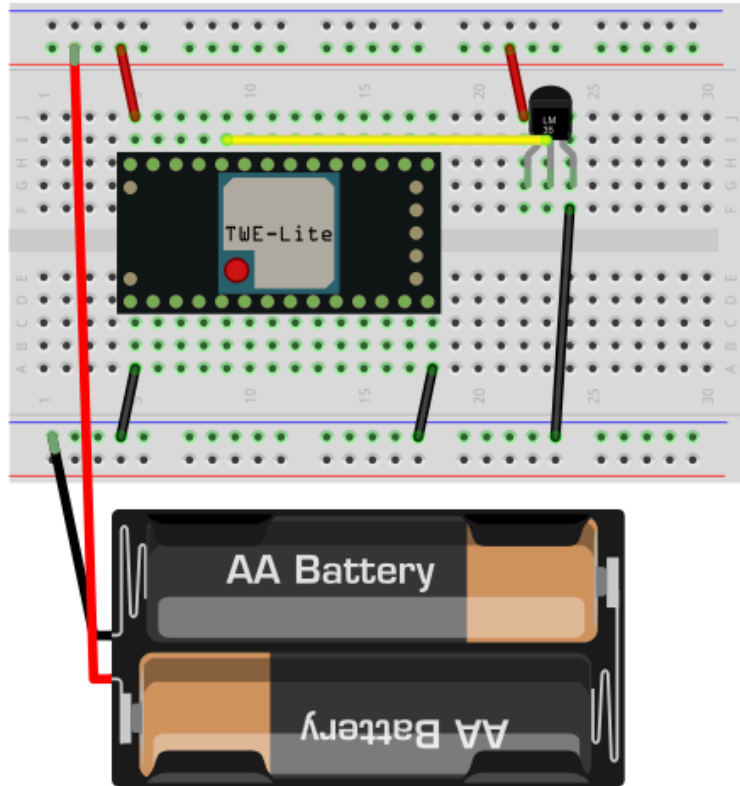
d. 下記の通りに回路を組む

A



3. 温度センサネットワーク

B



【使用パーツ】

ブレッドボード

TWE-Lite DIP

LM61BIZ

単三電池BOX

単三電池

ジャンパ線

3. 温度センサネットワーク

e. 親機側のArduinoのスケッチの例を示す

```
#include <SoftwareSerial.h>
#include <Time.h>
#include <twelite.h>

SoftwareSerial tSerial(2, 3);
TWELite twelite(tSerial);

void setup() {
  setTime(11, 00, 00, 16, 9, 2015);
  Serial.begin(9600);
  twelite.begin(9600);
}

void loop() {
  if (twelite.available() > 0) {
    // Packetが届いたとき
    Serial.print(hour());
    printDigits(minute());
    printDigits(second());
    Serial.print(",");
    twelite.get();
    Serial.print(twelite.id(), HEX);
    Serial.print(",");
    Serial.println(twelite.temperature());
  }
}
```

```
void printDigits(int digits) {
  Serial.print(":");
  if(digits < 10)
    Serial.print('0');
  Serial.print(digits);
}
```

【動作仕様】

子機からのパケットを待ち受けして、パケットが届いたら受信時刻とともに子機のMACアドレス(下位4桁)と温度(単位は℃)を、コンマ区切りで1行ずつ出力する。

演習課題

- 温度センサネットワークをベースとして、下記の改造を行う：
 - 子機の温度が28度を下回っていたら、緑色のLEDを点灯する
 - 子機の温度が28度を超えたら、赤色LEDを点灯する。さらに、30度を超えた場合は、赤色LEDを点滅させる。
- 上記の動作をする回路とスケッチを作成する



MEMO



参考資料

1. TWE-Programmerのインストールと使い方
2. tweliteライブラリについて
3. xively.comの利用方法について

1. TWE-Programmerのインストールと使い方

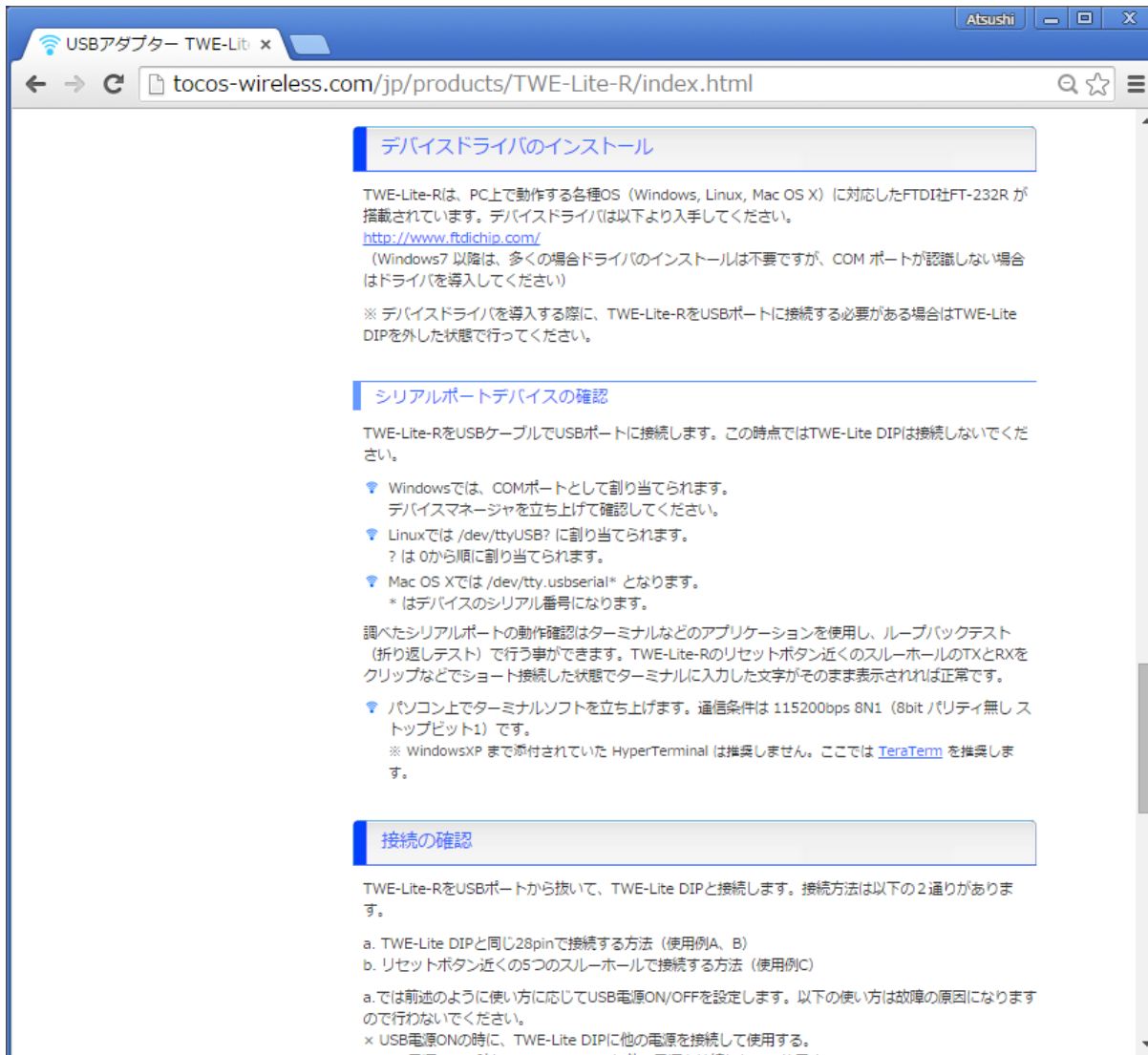
ここにアクセスする → <http://tocos-wireless.com/jp/products/TWE-Lite-R/index.html>

The screenshot shows the website for Tocos Wireless. The main header features the company logo and name. Below it is a navigation bar with links for 'トップページ', 'ワイヤレス製品', '技術情報', 'お問い合わせ', and '会社情報'. The current page is 'USBアダプター TWE-Lite R (トワイライター)'. A sidebar on the left lists various wireless products and software. The main content area displays the product name 'USBアダプター TWE-Lite R' and 'TWE-Lite R トワイライター'. Below this, there is a description of the product and a video player with the title '超簡単！無線マイコンTWE-Lite(トワイライト)インタラク...'. The video player shows a computer screen with a software interface.

TWE-Lite Rの
使い方を詳しく
解説したページ

ソフトウェアや
ドライバのイン
ストール方法は、
下方にある

1. TWE-Programmerのインストールと使い方



USBアダプター TWE-Lite x

tocos-wireless.com/jp/products/TWE-Lite-R/index.html

デバイスドライバのインストール

TWE-Lite-Rは、PC上で動作する各種OS (Windows, Linux, Mac OS X) に対応したFTDI社FT-232R が搭載されています。デバイスドライバは以下より入手してください。
<http://www.ftdichip.com/>
(Windows7 以降は、多くの場合ドライバのインストールは不要ですが、COM ポートが認識しない場合はドライバを導入してください)
※ デバイスドライバを導入する際に、TWE-Lite-RをUSBポートに接続する必要がある場合はTWE-Lite DIPを外した状態で行ってください。

シリアルポートデバイスの確認

TWE-Lite-RをUSBケーブルでUSBポートに接続します。この時点ではTWE-Lite DIPは接続しないでください。

- Windowsでは、COMポートとして割り当てられます。
デバイスマネージャを立ち上げて確認してください。
- Linuxでは /dev/ttyUSB? に割り当てられます。
? は0から順に割り当てられます。
- Mac OS Xでは /dev/tty.usbserial* となります。
* はデバイスのシリアル番号になります。

調べたシリアルポートの動作確認はターミナルなどのアプリケーションを使用し、ループバックテスト (折り返しテスト) で行う事ができます。TWE-Lite-Rのリセットボタン近くのスルーホールのTXとRXをクリップなどでショート接続した状態でターミナルに入力した文字がそのまま表示されれば正常です。

- パソコン上でターミナルソフトを立ち上げます。通信条件は 115200bps 8N1 (8bit /パリティ無し /ストップビット1) です。
※ WindowsXP まで添付されていた HyperTerminal は推奨しません。ここでは [TeraTerm](#) を推奨します。

接続の確認

TWE-Lite-RをUSBポートから抜いて、TWE-Lite DIPと接続します。接続方法は以下の2通りがあります。

- TWE-Lite DIPと同じ28pinで接続する方法 (使用例A、B)
- リセットボタン近くの5つのスルーホールで接続する方法 (使用例C)

a. では前述のように使い方に応じてUSB電源ON/OFFを設定します。以下の使い方は故障の原因になりますので行わないでください。
× USB電源ONの時に、TWE-Lite DIPに他の電源を接続して使用する。

1. TWE-Programmerのインストールと使い方

USBアダプター TWE-Lite-R

tocos-wireless.com/jp/products/TWE-Lite-R/index.html

接続の確認

TWE-Lite-RをUSBポートから抜いて、TWE-Lite DIPと接続します。接続方法は以下の2通りがあります。

- TWE-Lite DIPと同じ28pinで接続する方法（使用例A、B）
- リセットボタン近くの5つのスルーホールで接続する方法（使用例C）

a. では前述のように使い方に応じてUSB電源ON/OFFを設定します。以下の使い方は故障の原因になりますので行わないでください。

- × USB電源ONの時に、TWE-Lite DIPに他の電源を接続して使用する。
- × USB電源OFFの時に、TWE-Lite DIPに他の電源を接続しないで使用する。

b. ではTWE-Lite DIPに他の電源を接続してください。

以下の手順で接続・電源投入を行ってください。

- USBポートへ接続する。
- TWE-Lite DIPに別の電源を供給する場合は、TWE-Lite DIPの電源を切った状態でUSBケーブルを接続し、接続後にTWE-Lite DIPの電源を投入する。

注意事項

- USBに接続した状態で利用してください。
USBに接続しない状態ではモジュールを動作させることはできません。
- 外部のマイコンに UART0 を接続する場合は、TWE-Lite R からモジュールを取り外して利用してください。
UART0 のポートが USB-UART 半導体に接続されているため、外部のマイコンなどに UART0 を接続すると不都合が発生します。

書き込み用ソフトウェア

TWE-Lite DIP にアプリを書き込むために以下のソフトウェアを使用します。

GUIアプリ

[TWE-Liteプログラマ](#)

ファイルをドラッグアンドドロップするだけで簡単に書き込みができます。Window専用です。

コマンドラインアプリ

このリンク先にあるアプリをダウンロードして、インストールする！

1. TWE-Programmerのインストールと使い方

The screenshot shows the website for TWE-Lite Programmer. The browser address bar shows `tocos-wireless.com/jp/tech/misc/LiteProg/index.html`. The website header includes the TOCOS Wireless Engine logo and navigation links for トップページ, ワイヤレス製品, 技術情報, お問い合わせ, and 会社情報. The main content area is titled "TWE-Lite専用プログラマ" and includes a description: "TWE-Lite専用の書き込みツールです。ファイルをドラッグアンドドロップするだけで簡単にアプリの書き込みが可能です。" Below this is a section for "Windows用 TWE-Lite プログラマ" with a description: "Windows 環境 (.NET) で動作する GUI 版 TWE-Lite プログラマです。TWE-Lite-R 以外でも使用できますが、TWE-Lite-R の場合、COMポートの自動認識・プログラムモード(※1)を自動設定するので操作が簡単になります。" An inset image shows the TWE-Lite Programmer (0.3.3.1) software window with fields for COMS, TWE-Lite シリアル番号 (6300001), and a file path. A "再書き込み" button is visible. Below the software window, a note states: "※1 TWE-Lite を自動設定を使用せずにプログラムモードで起動する方法については、[こちら](#) (jenprog ユーティリティの替わりに本ツールを使用します)を参考にしてください。" The "使用方法" section lists: 1. COMポートを選択します。TWE-Lite がプログラムモードになっていない場合は、下部にメッセージが出ますので、もう一度接続・プログラムモードの確認を行ってください。 2. プログラムファイルを選択します。中央のボタンを押して選択するか、ファイルをドラッグアンドドロップしてください。正しいファイルを選択した場合は、自動的にプログラムが始まります。

下方にリンクのある「TWE-Lite プログラマ 0.3.4.1」をダウンロード&インストールする

1. TWE-Programmerのインストールと使い方



- ① TWE-Lite RをPCに接続し、TWE-Lite Rにアプリを書き込みたいTWE-Liteを装着する。
- ② その後、TWE-Liteプログラマを起動して、書き込みたいアプリ(.binファイル)をドラッグ&ドロップでこのウィンドウへ投げ込む。



2. TWE-Lite用ライブラリ twelite

- **概要**

- TWE-Liteを簡単に利用するための簡易ライブラリ
- フレームの詳細を知らなくても、TWE-Liteを利用できる

- **API**

メソッド名	機能概要	引 数
begin(uint32_t baudrate)	ライブラリの初期化を行う	baudrate: 通信速度(デフォルトは9600)
available()	受信したパケット数を返す	-
get()	パケットを受信する	-
id()	MACアドレスの一部(下位4桁)を取得する	-
temperature()	温度(°C)を取得する	-
voltage()	電池電圧(mV)を取得する	-

- **補足**

- TWE-Liteのすべての機能を利用できるわけではなく、無線タグアプリでセンサの種類を0x11 (LM61BIZによる温度計測) に指定した場合のパケットのみを取り扱う。

2. TWE-Lite用ライブラリ twelite

- **void begin(uint32_t baudrate)**
 - 本ライブラリを使用可能にする。引数baudrateには、XBeeに設定しているボーレート (BD)と同じ値を指定すること。
- **int available()**
 - 届いているパケットの数を返す。届いていないときは0を返す。
- **int get()**
 - 届いているパケットを取得する。
 - 成功した場合は、取得したパケットのサイズ(バイト数、1以上)を返す
- **unsigned int id()**
 - get()で取得した最新のパケットのMACアドレスの下位4桁(0001~ffff)を取得する。
 - 取得に失敗した場合は0を返す。
- **int temperature()**
 - get()で取得した最新のパケットの温度(単位は℃)を取得する
- **int voltage()**
 - get()で取得した最新のパケットの電池電圧(単位はmV)を取得する



TWE-Lite設定のまとめ

- リモートコントロール
 - 親機、子機の区別はない
 - Arduinoは使わない

	A(コントロールする側)	B(コントロールされる側)
プログラム	App_TweLite_Master_JN5164_1_6_6.bin	
設定	アプリケーションID (グループ分けのため) チャンネルID (混信を防ぐため)	
回路	スイッチ、電池	LED、抵抗、電池
通信速度 (設定モード)	115,200bps (+++を入力)	

TWE-Lite設定のまとめ

- センサネットワーク
 - 親機、子機の区別がある

	A (親機)	B (子機)
プログラム	Samp_Monitor_Parent_J N5164_1_5_5.bin (改造版)	Samp_Monitor_EndDevice_Input_JN5164_1_5_5.bin
設定	アプリケーションID(a) チャンネルID(c) オプション(o)=20	アプリケーションID(a) チャンネルID(c) センサ種別(m)=11
回路	Arduino UNO	温度センサ、電池
通信速度 (設定モード)	9,600bps (+++を入力)	115,200bps (M2ピンをGNDでリセット)
ライブラリ	twelite	-
	ロジック電圧は3.3Vなので Arduino UNOと接続す	親に温度を送る間隔は 設定(dコマンド)で変更