

# 無線9軸IMUモジュール マニュアル

1.3版  
2015年11月04日  
株式会社アールティ

## マニュアル改定歴

改定日	バージョン	変更内容	担当
2015/11/04	1.3版	Linux環境用のサンプルプログラムに関する記述を追加  「4.3充電時間」の記述を修正 充電時間 4時間 x 充電時間 1時間 o	高橋
2015/10/20	1.2版	ファームウェアバージョンを更新 センサ出力値のバイアスについての記述を追加	高橋
2015/09/16	1.1版	センサの記述をmpu9150からmpu9250に変更	高橋
2015/08/10	1.0版	リリース版	高橋
2015/08/10	0.9版	内容加筆	高橋
2015/08/03	0.1版	作成	高橋

## ファームウェア改定歴

改定日	バージョン	変更内容	担当
2015/10/20	2	ACアダプタ経由でUSBケーブルを刺したときの挙動を修正	高橋
2015/09/16	1	mpu9250用ファームウェアを作成	高橋

## 目次

---

- [1.注意事項](#)
- [2.概要](#)
- [3.使用環境](#)
- [4.仕様](#)
  - [4.1 製品仕様](#)
  - [4.2 モジュールの軸の取り方](#)
  - [4.3 充電機能](#)
  - [4.4 モジュールとの接続方法](#)
  - [4.5 通信仕様](#)
  - [4.6 通信プロトコル](#)
  - [4.7 センサデータの物理量への変換](#)
  - [4.8 センサ出力値のバイアス](#)
  - [4.9 Bluetoothモジュール](#)
  - [5.0 モジュール用ケース](#)
- [5.動作確認](#)
  - [5.1 準備するもの](#)
  - [5.2 USBドライバインストール手順](#)
  - [5.3 Bluetoothペアリング手順](#)
  - [5.4 Processing サンプルプログラムの動作確認](#)
- [6.開発環境の構築](#)
- [7.LPCXpressoのサンプルプログラムについて](#)
- [8.Linuxサンプルプログラムについて](#)
- [9.ファームウェアの書き込み](#)
- [10.お問い合わせ](#)

## 1. 注意事項

本製品をご使用頂く前に本マニュアルを熟読下さい。使用者および周囲の人に対する安全のため、内容をよく理解してから製品をお使い下さい。本製品をご使用したことによる、損害・損失について弊社は一切補償できません。

## 2. 概要

本製品はInvenSense社(インベンセンス)の9軸センサ(3軸加速度、3軸ジャイロ、3軸コンパス)MPU9250を使用したセンサです。本モジュール上には9軸センサとマイコン(LPC1343)が実装されており、センサデータの取得とデータの処理を一枚の基板で実現できます。

無線9軸IMUセンサモジュール(以下、9軸センサモジュール)はロボット、マルチコプタ、ゲーム・アミューズメント、モーションセンサ、ポータブルナビゲーション、ラジコン、模型ヘリ、ヘッドマウントディスプレイなどの製品の開発段階での使用を想定して設計されています。インターフェースはBluetoothとUSBがあり、すでにマイコン側にファームウェアが書き込まれているので、接続するだけですぐにセンサ値を取得することができます。また、マイコンのプログラムを自分で書き換えることでボード側にてレンジの変更やフィルタ等の作成も可能です。

USB接続の場合PCと接続した際にはVirtual COM Portとして認識されます。また、Bluetooth接続の場合はSPP(Serial Port Profile)での通信となります。USB、Bluetoothどちらの接続でも、アプリケーション側からは通常のシリアルポートでのアクセスとまったく同様に使用することができます。出力されるデータは3軸加速度、3軸地磁気、3軸角速度、温度、電池電圧になります。表1にデータの測定レンジと分解能を示します。

	加速度	角速度	地磁気
測定レンジ	$\pm 16[g]$	$\pm 2000[\text{deg/sec}]$	$\pm 4800[\mu\text{T}]$
分解能	16ビット	16ビット	16ビット

表1.データの測定レンジと分解能

## 内容物

- 無線9軸IMUセンサモジュール 1台
- リチウムポリマーバッテリー(400mAh)
- マニュアル 1枚

## サンプルプログラム

- Processing用プログラム sketch\_BT9axisIMU\_test
- Firmware RT-BT-9AXIS-sample.bin
- LPCXpresso用サンプルプロジェクト RT-BT-9AXIS-00-sample
- ケース3Dプリント用データ
- Linux用サンプルプログラム usb9axis\_test.c

### 3. 使用環境

OS: Windows XP SP2以上 / Vista / 7 / 8 / 8.1(32/64bit)

CPU: 800MHz以上の32bit(x86) or 64bit(x64)のプロセッサ

Memory: 512MB以上

Storage: 500MB以上

USB: USB2.0 1ポート

参考: 一部のLinux, Mac環境にて動作を確認.

動作確認済みOSは以下になります.

Rasbian(Raspberry Pi 2), Ubuntu14.04, Mac OS X Marvericks 10.9.4

### 4. 仕様

#### 4.1. 製品仕様

- サイズ: 42mm×30mm×11mm
- 重量: 4g
- センサ: MPU9250
- USBバスパワー動作
- リチウムポリマーバッテリー(400mAh)用充電回路搭載
- マイコン:LPC1343 72MHz 3.0V動作 センサ値取得用ファームウェア書き込み済み
- ローパスフィルタ 100Hz以上カット
- 各センサの測定レンジはMPU9250の最大レンジに設定

#### 4.2. モジュールの軸の取り方

加速度, ジャイロ, 地磁気センサの出力の軸の取り方は図1のようになります. ジャイロの角速度は軸の方向に沿って反時計回りが正方向です. (右ねじを回して進む向きを軸の方向としたときに, ネジを進める回転方向が正)

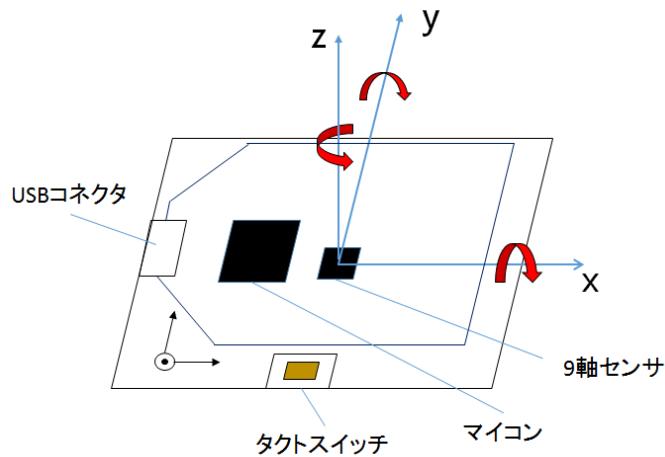


図1. フームウェアの加速度、角速度、地磁気の正方向

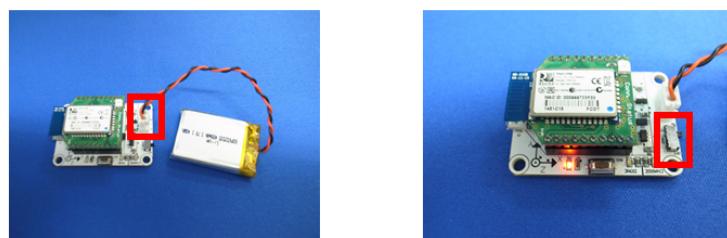
注) 本製品のファームウェアでは地磁気センサの軸の取り方がMPU9250と異なります。

#### 4.3. 充電機能

本モジュールにはリチウムポリマーバッテリーの充電機能が搭載されています。充電を行う場合はリチウムポリマー電池をモジュールに接続し、電源スイッチをONにした状態でUSBケーブルを接続することで開始されます。充電中は赤色のLEDが点灯し、充電が終了すると緑色のLEDが点灯します。充電時間はおよそ1時間です。充電中は必ず目を離さないようにしてください。

注) 尚、バッテリが接続されてない、またはバッテリ電源スイッチがOFFの状態でUSBケーブルを接続した場合は赤緑両方のLEDが点灯します。

1. バッテリーをモジュールに接続
2. 電源をONにする



3. USBケーブルから給電

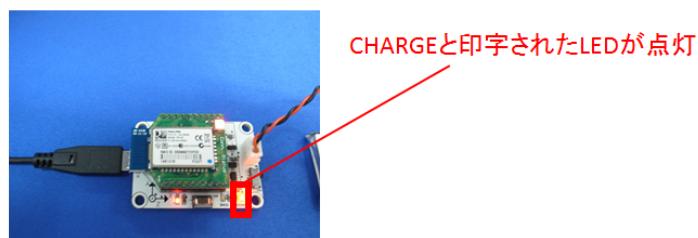


図2. 充電の手順

## 4.4. モジュールとの接続方法

本モジュールはBluetoothおよびUSB経由でのデータ出力が可能です。

### Bluetooth接続

Bluetooth接続を使用するときは、モジュール上のlpc1343マイコンは3V動作なのでレギュレータを通して3Vの電源がマイコンに供給されます。このときTX, RXの信号は3V系であることに注意してください。また、UART接続の際は基板上のLEDが高速で点滅します。Bluetoothのペアリング方法については5章動作確認を参照してください。

### USB接続

USB接続の際はUSBケーブルでPCと本モジュールを接続してください。(USBドライバのインストールは必要、手順は5章を参照) USB接続が確立されると基板上のLEDが低速で点滅します。また、USB接続の際にはBluetooth経由からのセンサデータ出力もされています。(USBケーブルを給電のみの目的で接続しBluetoothの出力を用いることも可能)

## 4.5. 通信仕様

USB接続の場合PCと接続した際にはVirtual COM Portとして認識されます。また、Bluetooth接続の場合はSPP(Serial Port Profile)での通信となります。USB、Bluetoothどちらの接続でも、アプリケーション側からは通常のシリアルポートでのアクセスとまったく同様に使用することができます。USBでPCと接続しデータ送信調歩同期式シリアル通信で行われます。モジュールからはBluetoothの場合50msec毎に、USBの場合10msec毎に30byteのセンサデータが送信され続けます。センサデータの構成については4.6節に記述します。

- USB接続 :CDCクラスを使用して通信
- Bluetooth接続 :SPPプロファイルを使用して通信
- Bluetooth接続時ボーレート :115200 bps
- USB接続時ボーレート :115200 bps
- データ :8bit
- パリティ :なし
- ストップビット :1bit
- フロー制御 :なし
- Bluetooth接続時データ送信周期 :20Hz
- USB接続時データ送信周期 :100Hz

## 4.6. 通信プロトコル

センサ測定値の通信データは表2のような構成になります。0byte目から6byte目までの出力は常に固定なので、受信データの先頭を見つけるのに用いてください。受信したデータをバッファリングし、0byte目から6byte目までの固定パターンを探索し、そこから24byteを取り出せば有効なセンサデータが得られます。

7byte目のタイムスタンプはデータが送信されるたびに0,1,2, … ,255となり、255の次は再び0になります。通信データの抜けを確認するために使用してください。

Byte	内容	Byte	内容
0	0xff	15	AD値 (TEMP)上位8ビット
1	0xff	16	AD値 (GYRO X)下位8ビット
2	0x52(ASCIIコードのR)	17	AD値 (GYRO X)上位8ビット
3	0x54(ASCIIコードのT)	18	AD値 (GYRO Y)下位8ビット
4	0x34(製品識別子下位8bit)	19	AD値 (GYRO Y)上位8ビット
5	0x57(製品識別子上位8bit)	20	AD値 (GYRO Z)下位8ビット
6	ファームウェアバージョン	21	AD値 (GYRO Z)上位8ビット
7	タイムスタンプ(0x00 - 0xff)	22	AD値 (MAG X)下位8ビット
8	AD値 (ACC X)下位8ビット	23	AD値 (MAG X)上位8ビット
9	AD値 (ACC X)上位8ビット	24	AD値 (MAG Y)下位8ビット
10	AD値 (ACC Y)下位8ビット	25	AD値 (MAG Y)上位8ビット
11	AD値 (ACC Y)上位8ビット	26	AD値 (MAG Z)下位8ビット
12	AD値 (ACC Z)下位8ビット	27	AD値 (MAG Z)上位8ビット
13	AD値 (ACC Z)上位8ビット	28	バッテリ電圧(BAT_V)下位8ビット
14	AD値 (TEMP)下位8ビット	29	バッテリ電圧 (BAT_V)上位8ビット

表2. 送信データの割り当て

## 4.7. センサデータの物理量への変換式

センサデータは30byteのデータになっているので、有効なセンサ値に変換するには各byteを結合し以下の計算式を適用します。

### 受信データの結合の例

concatenation\_int [16bit 符号付き整数]  
concatenation\_uint [16bit 符号なし整数]

data\_H [8bit符号なし整数]  
data\_L [8bit符号なし整数]

concatenation\_uint = data\_L + (data\_H<<8)

concatenation\_int = data\_L + (data\_H<<8)  
if(concatenatin\_int > 32767) concatenation\_int = concatenation\_int - 65535

### 加速度

加速度センサ値 : acc [16bit 符号付整数]  
計算式 : acc / 2048 [g]

### ジャイロセンサ

ジャイロセンサ値 : omega [16bit符号付整数]  
計算式: (omega) / 16.4 [deg/sec]

### 地磁気センサ

地磁気センサ値 : mag [16bit符号付整数]  
計算式 : mag \*0.15 [ $\mu\text{T}$  ]

### 温度センサ

温度センサ値 : temp [16bit符号付整数]  
計算式 : temp / 333.87 + 21.0 [ $^{\circ}\text{C}$ ]

### バッテリー電圧値

電圧値 : bat\_v [16bit符号なし整数]  
計算式 : bat\_v // 13107 [V]

## 4.8. センサ出力値のバイアス

9軸センサモジュールから出力されるデータにはバイアス(センサの出力値の原点ずれ)が存在します。これはMEMS9軸センサ特有の現象です。バイアスを補正することより精度の高い測定データが得られます。

以下にジャイロセンサと加速度センサの簡易的なバイアスキャリビュレーション方法を示します。ただし、この方法はあくまでも簡易的な方法なので注意してください。

### 1) ジャイロセンサ

センサを完全に静止させた状態でジャイロセンサ値を100-1000回程取得し平均をとります。これを`omega_ref`とします。これ以後のジャイロセンサの測定値`omega`から`omega_ref`をオフセットしてください。

ジャイロリファレンス値 : `omega_ref` [16bit符号付整数]

ジャイロセンサ値 : `omega` [16bit符号付整数]

計算式:  $(\text{omega} - \text{omega\_ref}) / 16.4$  [deg/sec]

### 2) 加速度センサ

基本的にはジャイロセンサと同様な手順でキャリビュレーションが可能ですが、加速度センサの場合静止状態で重力加速度を検出しますので、z軸のみ調整方法が変わります。

センサ基板のz軸が重力の方向と平行になるように置き、完全に静止させます。

静止状態で加速度センサ値を100-1000回程取得し平均を取ります。これを各軸`acc_ref_x`, `acc_ref_y`, `acc_ref_z`とします。これ以後の加速度センサの測定値を`acc_x`, `acc_y`, `acc_z`とします。

加速度リファレンス値: `acc_x`, `acc_y`, `acc_z`[16bit符号付整数]

加速度センサ値: `acc_x`, `acc_y`, `acc_z`[16bit符号付整数]

計算式  $\text{acc}_x: (\text{acc}_x - \text{acc\_ref}_x) / 2048$ [g]

計算式  $\text{acc}_y: (\text{acc}_y - \text{acc\_ref}_y) / 2048$ [g]

計算式  $\text{acc}_z: (\text{acc}_z - (\text{acc\_ref}_z - 1)) / 2048$  [g]

(測定場所の重力の大きさが1gと仮定)

## 4.9. Bluetoothモジュール

9軸センサモジュールは標準でMicrochip社のBluetoothモジュールのRN-42を使用しています。弊社出荷時のRN-42の主な設定は以下の通りです。

- モード : スレーブモード

- ポーレート : 115200bps
- デバイス名 : RT-BT-9AXIS-XXXX  
(XXXXにはRN-42表面に印刷されたMAC ID の下4桁が入ります。(図4赤線部))



図3.デバイス名下4桁の表記表示位置

#### 設定変更手順

RN-42ではTeratarm等のターミナルソフトから設定コマンドを送信することで通信モード、デバイス名、通信ポーレート等の設定を変更できます。

Bluetoothモジュールの設定を出荷状態から変更したいときに行ってください。(9軸モジュールのファームウェアをユーザーが書き換え通信ポーレートを変更する場合にはRN-42のポーレートを変更する必要があります。)

手順:

1. RN-42とBluetoothで接続する
2. \$\$\$ と入力し設定モードに移行(RN-42のLEDが高速点滅し、CMDと表示されれば設定モード移行成功)
3. +を入力後エンターを押しlocal echoをonにする。
4. 設定コマンド入力後エンターを押し設定変更。(OK等のメッセージが表示されればコマンド入力成功)
5. --- と入力後エンターを押しコマンドモードを終了
6. 電源を入れ直し設定反映

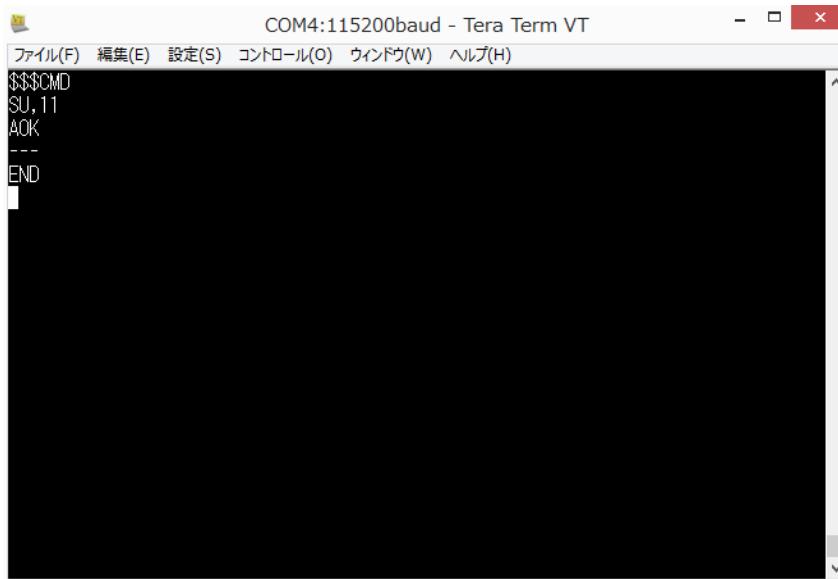


図4.設定画面例

コマンド

ここでは例としてデバイス状態表示、動作モード変更、デバイス名変更、ポーレート変更のコマンドを記載します。

他のコマンドについてはRN-42のデータシートを参照してください。

- デバイス状態表示

コマンド: **D**

内容 : デバイスの基本的な情報を表示します。

- 通信モード変更

コマンド: **SM,[Mode]**

内容 : デバイスの通信モードを変更します。

SM,0	Slave mode
SM,1	Master mode
SM,2	Trigger mode
SM,3	Auto-Connect Master mode
SM,4	Auto-Connect DTR mode
SM,5	Auto-Connect ANY mode
SM,6	Pairing mode

(各モードの動作についてはRN-42のデータシートを参照してください)

- デバイス名変更  
コマンド: **SN,[Device\_Name]**  
内容 : デバイス名を変更します。  
例: デバイス名HOGEに設定 → SN,HOGE

- ポーレート変更  
コマンド: **SU,[baudrate]**  
[baudrate]には、指定したいポーレートの上位2桁を指定。  
内容 : 通信ポーレートを変更します。  
例 : 115200bps → SU,11  
57200bps → SU,57  
9600bps → SU,96

#### ポーレート、通信モード初期化機能

本モジュールのファームウェアには通信モード、ポーレートを初期化する機能が備わっています。この機能を使うと

- ポーレート 115200bps
- 通信モード Slaveモード

になります。

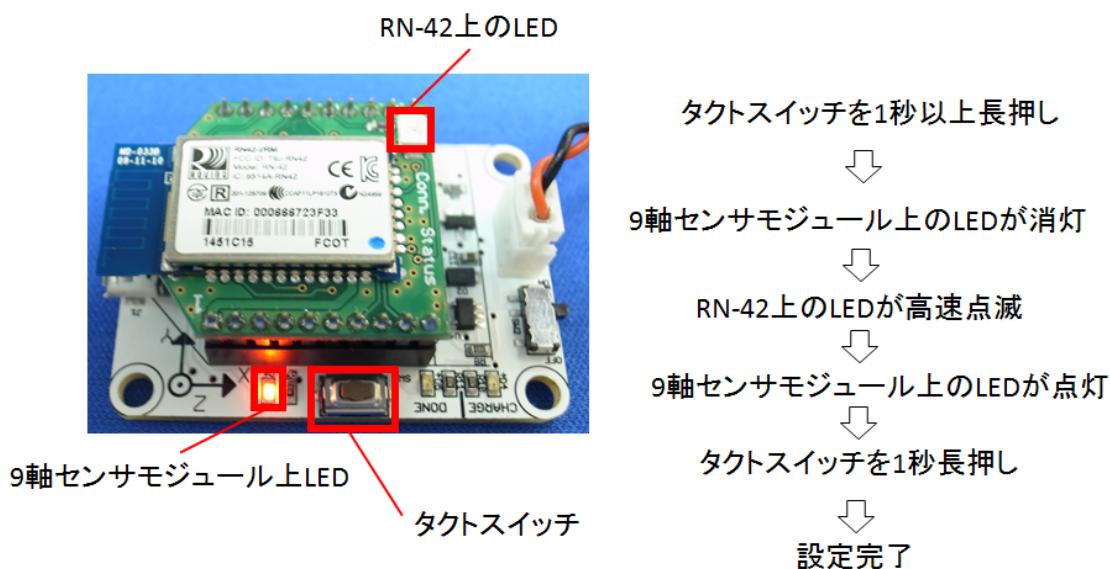


図5. 初期化機能 手順

#### 4.10. モジュール用ケース

本モジュールにはケースは付属しませんが、ケースの3Dデータは公開しております。以下に3Dプリンタを用いて作成したケースの制作例を示します。尚、3Dデータは <http://www.rt-shop.jp/download/RT-BT-9axisIMU/>にてダウンロード可能です。



図6. ケース作例1

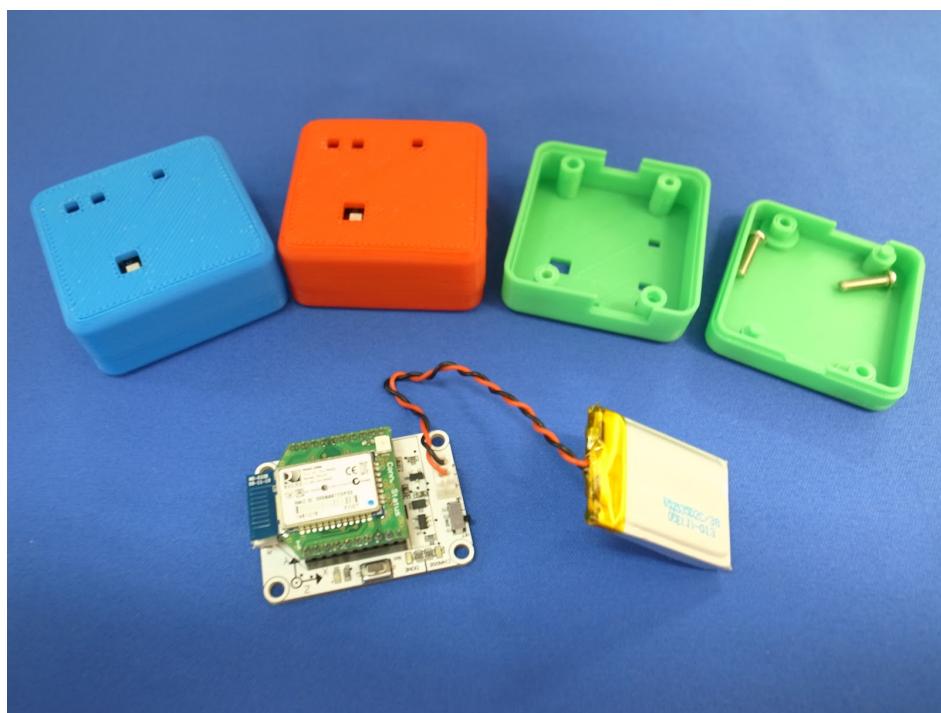


図7. ケース作例2

## 5. 動作確認

### 5.1. 準備するもの

動作確認のために以下を用意してください。

- 無線9軸センサモジュール
- USBケーブル(A to microB)
- Bluetoothで通信可能なPC

## 5.2. USBドライバインストール手順

9軸センサモジュールがパソコンと通信を行うためには、対応したドライバのインストールが必要です。

### 1. 9軸センサモジュールと接続

9軸センサモジュールをPCIにUSBで接続します。その状態でデバイスマネージャーを開いてください。すると、RT-BT-9AXIS-00\_XXXXというデバイスがあるので(ない場合はモジュールを接続したときに新しく表示されたデバイス)選択します。

### 2. ドライバのダウンロード

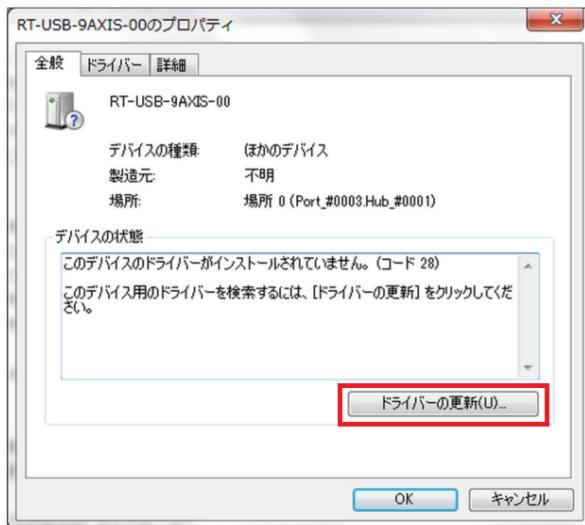
以下の、アールティロボットショップのダウンロードページにアクセスして下さい。

<http://www.rt-shop.jp/download/RT-BT-9axisIMU/>

「RT-BT-9AXIS-00.zip」をダウンロードして、解凍して下さい。

### 3. ドライバの選択

デバイスドライバーがインストールされていないと表示がでます。次にドライバーの更新をクリックします。



ダウンロードしたドライバを使用するため、「コンピュータ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。



解凍したフォルダの中のdriverフォルダを選択し、インストールを開始します。



以上でインストール完了です。

### 5.3. Bluetoothペアリング手順

#### 1. Bluetoothデバイスの検索

PCでBluetoothを有効にした状態でコントロールパネルを開き、コントロールパネルの検索ボックスで「bluetooth」と検索し[bluetoothデバイスの追加]を開きます。



## 2. Bluetoothデバイスの追加

追加するデバイスの検索が行われるので、RT-BT-9AXIS-00\_XXXX(デバイス名)を選択します。

[デバイスを追加します] のウィンドウではいを押すとドライバのインストールが開始され、COMポートとして認識されます。COMポートは発信ポート、着信ポートの二つが追加されますが、本モジュールの通信では発信ポートを使用します。

### ※どちらが発信ポートかを調べる方法

コントロールパネルの検索ボックスでbluetoothと検索し、[bluetooth 設定の変更]を開きます。COMポートタブを開き発信ポートになっているCOMポートを確認します。

(<====デバイス名====>のところにはBluetoothモジュールのデバイス名が入ります。)



#### 5.4. Processing サンプルプログラムの動作確認

アールティロボットショップダウンロードサイトにあるサンプルソフトウェアを用いて、9軸センサモジュールの動作確認を行うことができます。使用するにはJavaがインストールされている必要がありますのでご注意ください。

Javaは次のHPからダウンロードできます。 <http://java.com/ja/download/>

サンプルプログラムの動作確認はProcessingというプログラミング言語を用いて行います。ProcessingはC言語風の文法で図形の描画等の視覚的表現が手軽に実現できます。

##### 1. Processingのダウンロード

次のアドレス<https://processing.org/> からProcessingのHPに飛び画面左のDownloadをクリックします。

**Cover**

**Download**

**Exhibition**

**Reference**

**Libraries**

**Tools**

**Environment**

**Tutorials**

**Examples**

**Books**

**Handbook**

**Overview**

**People**

**Shop**

- » Forum
- » GitHub
- » Issues
- » Wiki
- » FAQ
- » Twitter
- » Facebook

**» Download Processing**

**» Play With Examples**

**» Browse Tutorials**

Processing is a programming language, development environment, and online community. Since 2001, Processing has promoted software literacy within the visual arts and visual literacy within technology. Initially created to serve as a software sketchbook and to teach computer programming fundamentals within a visual context, Processing evolved into a development tool for professionals. Today, there are tens of thousands of students, artists, designers, researchers, and hobbyists who use Processing for learning, prototyping, and production.

- » Free to download and open source
- » Interactive programs with 2D, 3D or PDF output
- » OpenGL integration for accelerated 3D
- » For GNU/Linux, Mac OS X, and Windows
- » Over 100 libraries extend the core software
- » Well [documented](#), with many [books](#) available

**» Exhibition**

**Light Kinetics**  
by Espadaysantacruz

**Pathfinder**  
by Princemio

**Non-Linear Code**  
by Dextro

自分のPCの環境に合わせソフトウェアをダウンロードしてください。  
 また、Processingのバージョンが1.xと2.xでは仕様が異なりサンプルプログラムが動作しないためバージョン2.xをダウンロードしてください。  
 動作確認済みのバージョンは2.2.1(19 May 2014)

The screenshot shows the official Processing website. At the top, there are tabs for "Processing", "p5.js", "Processing.py", and "Processing Foundation". Below the tabs is a large banner with the word "Processing" and a geometric background. To the right is a search bar. On the left side of the main content area, there's a sidebar with links like "Cover", "Download", "Exhibition", "Reference", "Libraries", "Tools", "Environment", "Tutorials", "Examples", "Books", "Handbook", "Overview", "People", and "Shop". The main content area has a large "P" logo. It says "Download Processing. Processing is available for Linux, Mac OS X, and Windows. Select your choice to download the software below." Below this, it shows "2.2.1 (19 May 2014)" and download links for "Windows 64-bit", "Linux 64-bit", "Mac OS X", and "Windows 32-bit". To the right of the download links, it says "The list of revisions covers the differences between releases in detail. Please read the changes if you're new to the 2.0 series." At the bottom of the main content area, there's a section titled "Stable Releases" with links for "2.2.1 (19 May 2014)" and "1.5.1 (15 May 2011)".

ダウンロードが終わったら展開し、processing.exeを起動します。

## 2. プログラムのダウンロード

次のリンク、<http://www.rt-shop.jp/download/RT-BT-9axisIMU/>

から「RT-BT-9AXIS-00.zip」ファイルをダウンロードします。ダウンロードが終わったら展開します。

## 3. 実行する

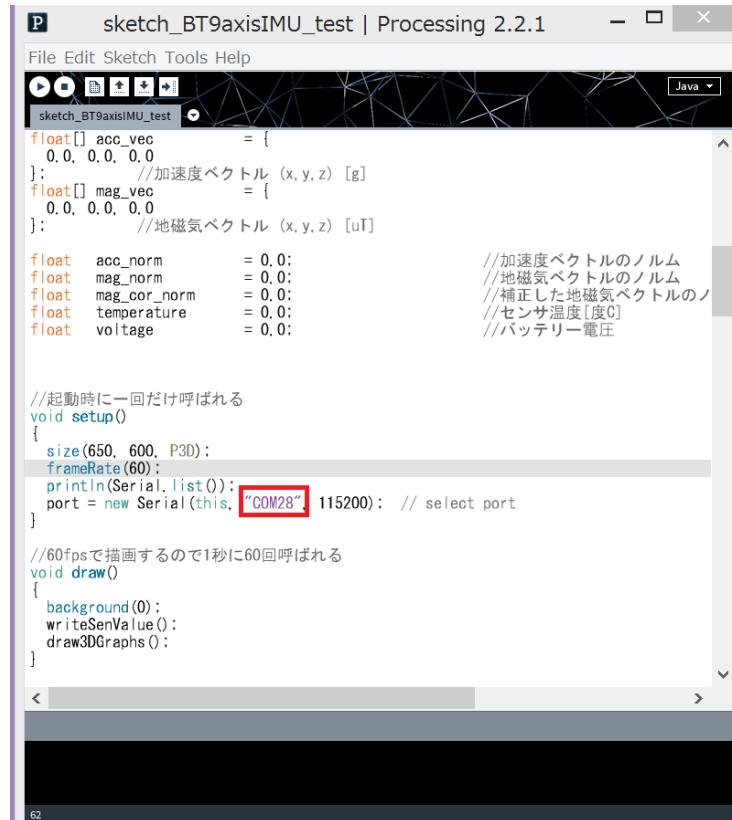
Processingの方にもどり、「File」→「Open...」から、先ほどダウンロードしたプログラムの中にあるsketch\_BT9axisIMU\_test.pdeを選択します。デバイスマネージャにて9軸センサモジュールが接続されているポートを確認したのち、プログラム内の以下の部分を変更してください。

また、サンプルプログラムのコメントが文字化けする場合はProcessingの上部の

File >> Preferences >> Editor and Console font

と進みフォントを日本語表示できるもの(MS ゴシック等)に変更してください。

「COM\*」の番号を設定したのち、左上の再生ボタン(三角のボタン)をクリックしてください。(Linux, Macの環境の場合、COM\*ではなく/dev/tty\*)  
USB, Bluetoothどちらの接続の場合もCOMポートはデバイスマネージャにて調べることができます。



```

P sketch_BT9axisIMU_test | Processing 2.2.1
File Edit Sketch Tools Help
sketch_BT9axisIMU_test Java
float[] acc_vec = { 0.0, 0.0, 0.0 };
// 加速度ベクトル (x, y, z) [g]
float[] mag_vec = { 0.0, 0.0, 0.0 };
// 地磁気ベクトル (x, y, z) [uT]

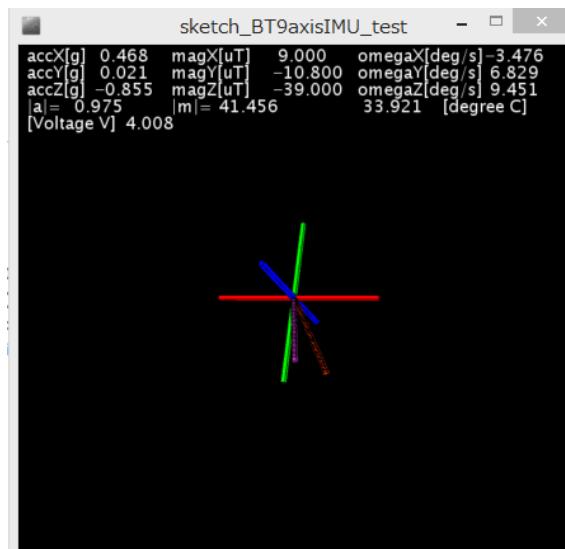
float acc_norm = 0.0; // 加速度ベクトルのノルム
float mag_norm = 0.0; // 地磁気ベクトルのノルム
float mag_cor_norm = 0.0; // 補正した地磁気ベクトルのノルム
float temperature = 0.0; // センサ温度[度C]
float voltage = 0.0; // バッテリー電圧

// 起動時に一回だけ呼ばれる
void setup()
{
    size(650, 600, P3D);
    frameRate(60);
    println(Serial.list());
    port = new Serial(this, "COM28", 115200); // select port
}

// 60fpsで描画するので1秒に60回呼ばれる
void draw()
{
    background(0);
    writeSenValue();
    draw3DGraphs();
}

```

3Dモデルが表示されたら、9軸センサモジュールを好きな方向に傾けてみてください。加速度センサと地磁気センサの値に応じて3Dモデルが動きます。



## 6. 開発環境の構築

9軸センサモジュールに搭載されているマイコンには、あらかじめファームウェアが書き込まれていますが、ユーザーによるプログラムの書き換えも可能です。開発環境としては、NXP社から提供されている統合開発環境(IDE)LPCXpressoを推奨しています。

<http://www.lpcware.com/lpcxpresso>

LPCXpressoのアクティベーションを行うと、256KBまでのコードサイズで開発が可能となります。また、有料版のProバージョンを購入すると、サイズ制限はなくなります。

### 1. ダウンロード

以下のURLよりLPCXpressoのインストーラーをダウンロードします。

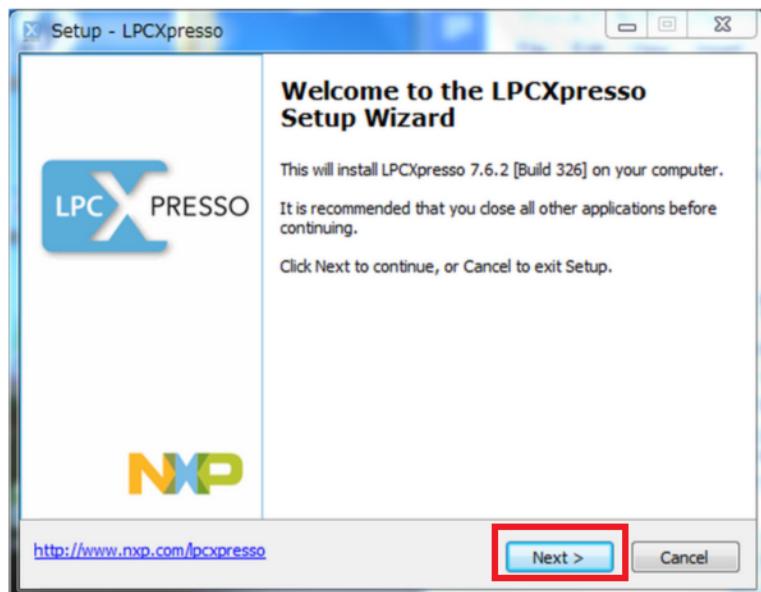
<http://www.lpcware.com/lpcxpresso/download>

ご自身のOSにあったものを選択してください。

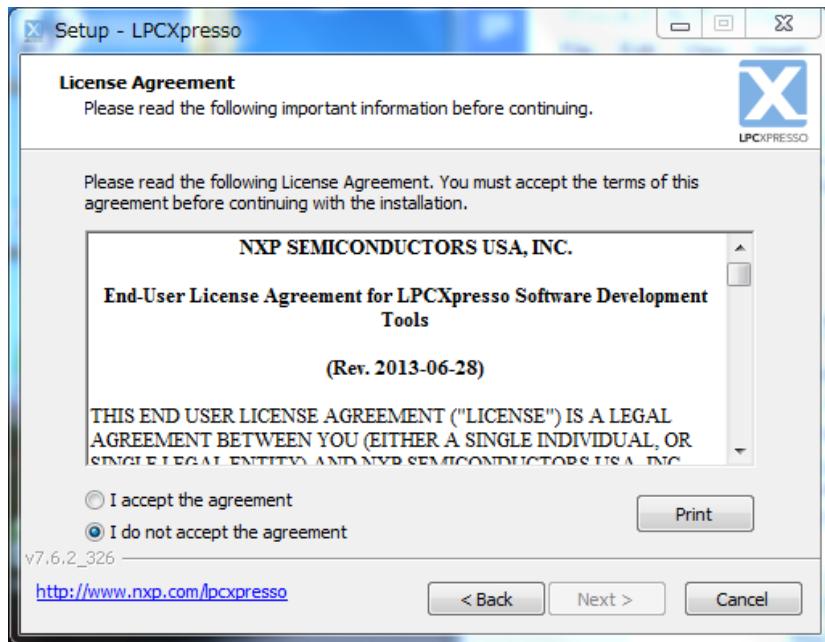
### 2. インストール

ダウンロードが完了したら、インストーラーをダブルクリックし実行します。

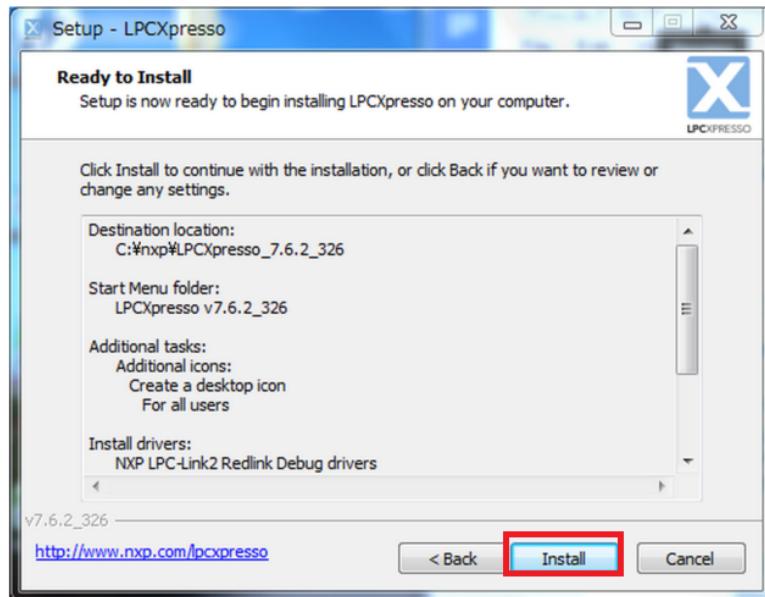
「Next >」をクリックします。



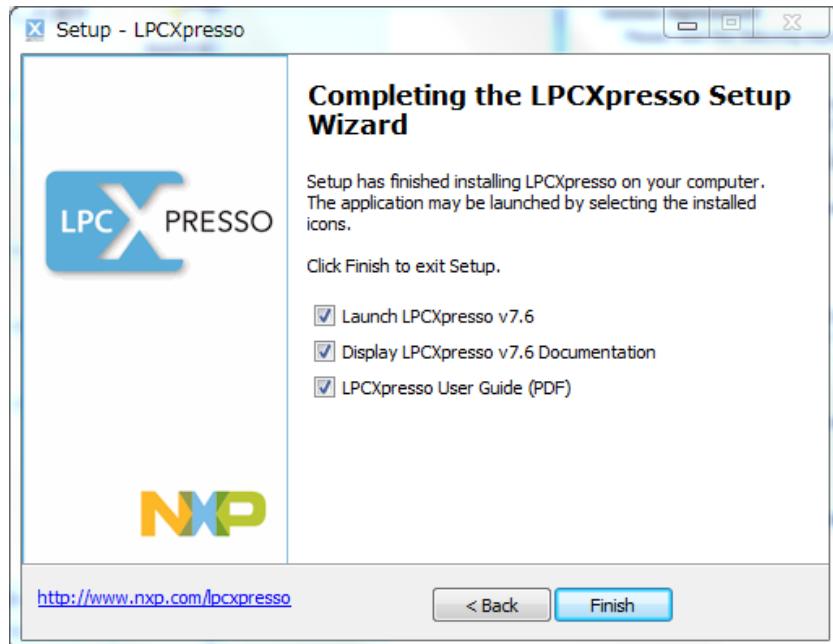
ライセンスの同意が求められますので、説明を読み同意するならば、「I accept the agreement」にチェックを入れて、「Next >」をクリックします。これ以降は変更が必要な点はありませんので、インストール開始まで「Next >」を選択し続けてください。



インストール直前まで設定が完了したら、「Install」をクリックしインストールします。

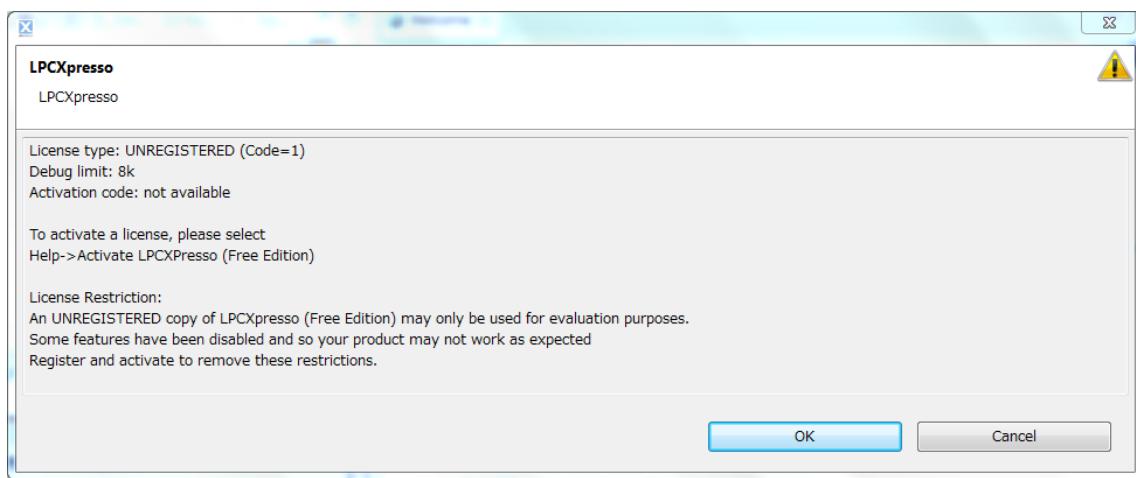


インストールが完了したら、「Finish」をクリックしインストーラを終了します。一番上のチェックボックスにチェックが入っていれば、LPCXpressoが自動で起動します。



### 3. アクティベーション

アクティベーションを行っていない状態では、以下のように8KBまでのサイズ制限が掛かっています。



このサイズ制限を広げるには、LPCware.comにてアカウントを作成し、アクティベーションに必要なシリアルコードを取得する必要があります。

## 7. LPCXpressoのサンプルプログラムについて

LPCXpressoのサンプルプログラムは「RT-BT-9AXIS-00.zip」内の「LPC Xpresso Sample Program」にあります。このファイルをLPCXpressoにインポートして使用することができます。本章ではサンプルプログラムのインポート方法を説明します。

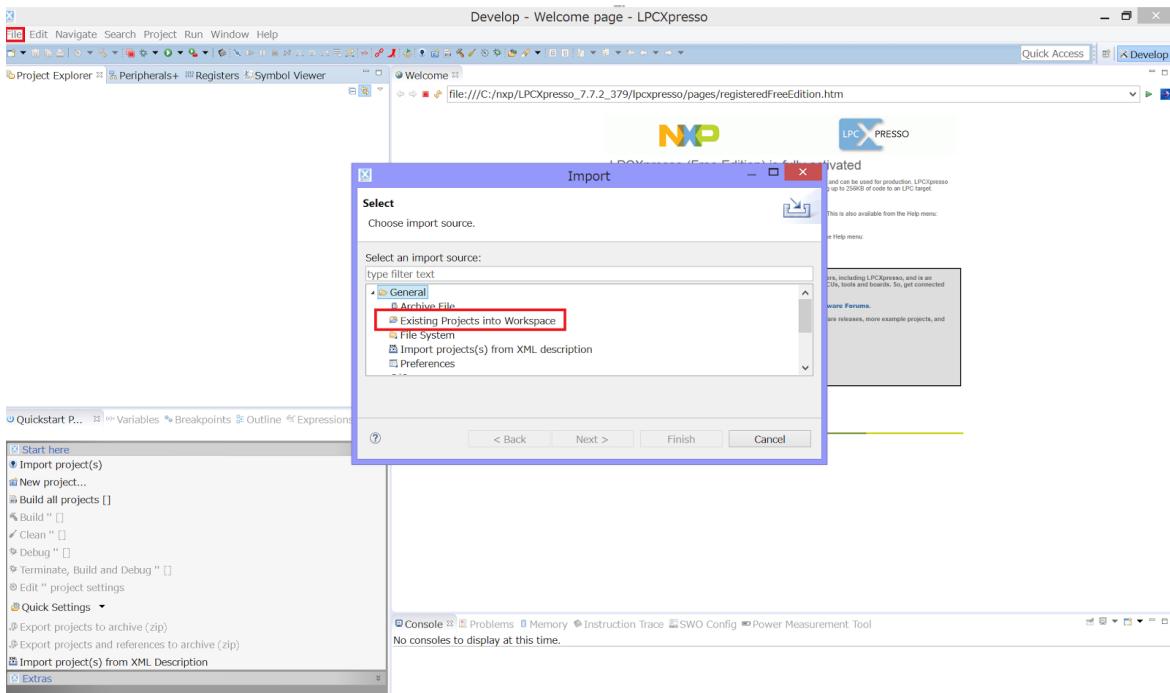
### 1. サンプルプログラムのインポート方法

サンプルプログラムは「LPCXpresso Sample Program」内にある以下の2プロジェクトです。

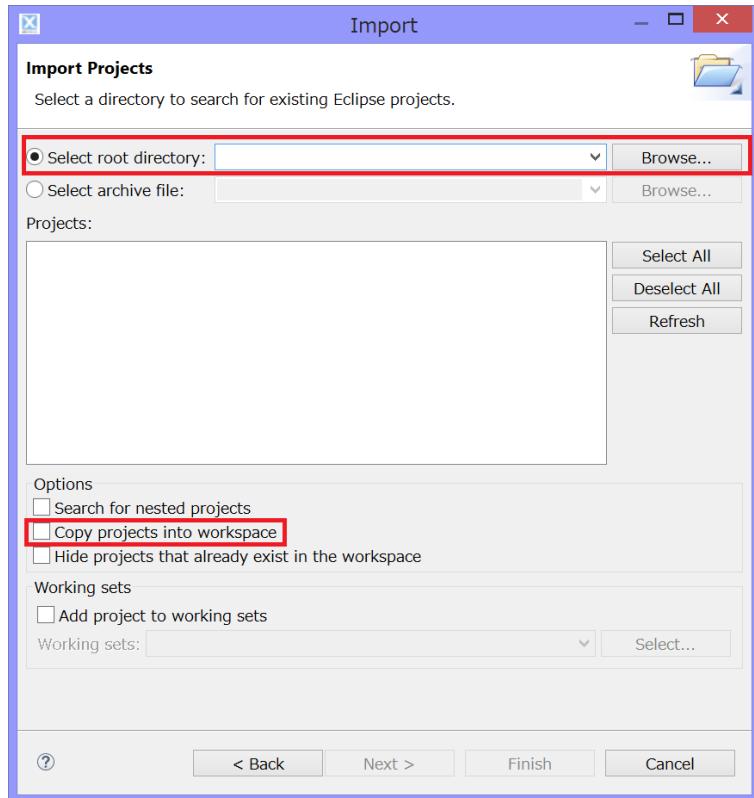
- RT-BT-9AXIS-00-sample
- CMSISv1p30\_LPC13xxプログラム本体は「RT-BT-9AXIS-00-sample」ですが、「CMSISv1p30\_LPC13xx」にあるライブラリファイルを読み込むため両方のプロジェクトがインポートされている必要があるので注意してください。

サンプルプログラムのインポートの手順は以下のようになります。

- LPCXpressoを起動
- File>>Import>>General>>Existing Projects into Workspaceを選択



- Select root directory>>Browse でインポートしたいディレクトリを選択
- LPCXpressoのworkspace内にプロジェクトのコピーを作成する場合は OptionsのCopy projects into workspaceにチェックを入れる
- Finishをクリック



- 2つのプロジェクトをインポートした状態でctrl + B (build all)を押し、エラーが  
出ずコンパイルが通ればインポート完了です。

## 8. Linuxサンプルプログラムについて

Linux環境でのサンプルプログラムについて解説します。動作環境としてはUbuntuを想定します。サンプルプログラムはLinux Sample Program内のusb9axis\_test.cになります。

### 実行手順

1. USBケーブルでセンサモジュールとPCを接続

2. /dev配下のttyから始まるファイルで、センサモジュールを接続して新しく表示されたファイルを記憶

/dev/ttyACM0で認識されることが多い  
(Mac環境などでは他の名前の場合もあり)

コマンド: ls /dev/tty\*

3. usb9axis\_test.c内のstatic char serial\_port[128]という箇所を編集

static char serial\_port[128] = "手順2で記憶したデバイスファイルのパス"  
static char serial\_port[128] = "/dev/ttyACM0"

4. gccでusb9axis\_test.cをコンパイル

usb9axis\_test.cが/hogeというPathにあるとして  
コマンド: gcc /hoge/usb9axis\_test.c  
a.outというファイルができます。

5. 手順4でできた実行ファイルを管理者権限で実行

コマンド: sudo /hoge/a.out

## 実行結果

9軸センサモジュールから出力される30byteのデータを各byteずつ画面上の表示し続けます。プログラムはCtrl + cで終了させてください。

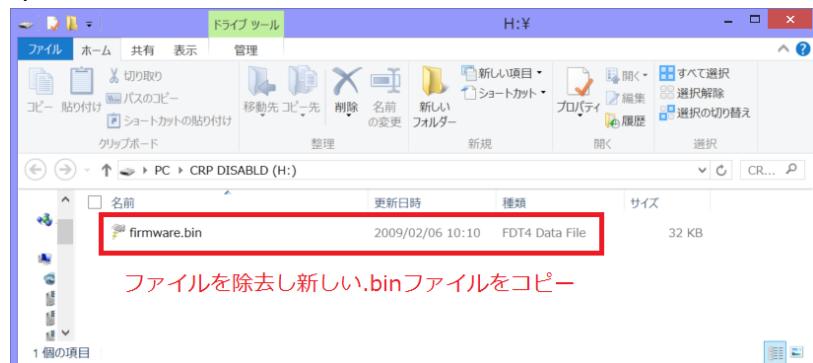
```
|14|85 ||
|15|f5 ||
|16|f ||
|17|fe ||
|18|11 ||
|19|11 ||
|20|51 ||
|21|0 ||
|22|e4 ||
|23|0 ||
|24|de ||
|25|ff ||
|26|86 ||
|27|ff ||
|28|ff ||
|29|da ||
=====
len=30 byte=====
|0|ff ||
|1|f ||
|2|52 ||
|3|54 ||
|4|39 ||
|5|43 ||
|6|f ||
|7|11 ||
|8|2f ||
|9|fd ||
|10|d1 ||
|11|ff ||
|12|ff ||
|13|f6 ||
|14|f6 ||
|15|5 ||
|16|87 ||
|17|fd ||
|18|dd ||
|19|11 ||
|20|ec ||
|21|0 ||
|22|da ||
|23|0 ||
|24|da ||
|25|ff ||
|26|ff ||
|27|ff ||
|28|cb ||
|29|db ||
```

## 9. ファームウェアの書き込み

初期状態では9軸センサモジュール上のLPC1343に既に9軸センサの出力を取得するためのファームウェアが書き込まれています。ここでは自作プログラムの書き込み方法を説明します。

1. 9軸センサモジュール上のタクトスイッチを押したままUSBケーブルを接続  
このとき、モジュール上のLEDが点灯します。

2. タクトスイッチから手を離します.
3. ブートローダーの起動まで待機
4. もともとのfirmware.binを削除
5. LPCXpressoで作成した自作プログラムの.binファイルをコピー



以上でファームウェアの書き込みは完了です。

## 10. お問い合わせ

カスタム等も有料にて承っておりますので、お気軽にお問い合わせ下さい。  
If you have any inquiries upon this product, please contact us at the following.

---

RT Corporation 株式会社アールティ

住所: 〒101-0021 東京都千代田区外神田3-2-13山口ビル3F

Address: 3F, 3-2-13 Sotokanda, Chiyodaku 101-0021, Tokyo, Japan

TEL +81-3-6666-2566 FAX +81-3-5809-5738

E-mail: [shop@rt-net.jp](mailto:shop@rt-net.jp)

Open: 11:00a.m.- 18:00p.m. (JST+9)

Close: weekend, national holiday, summer vacation, new year

---

---

### Copyright

All the company and product names in this document are trademarks or registered trademarks of their respective companies.

All the documents, photos, and illustrations are copyrighted and protected by the copyright law of Japan and overseas. All the contents in this document are not allowed to be uploaded to any public or local area networks such as the Internet without permission from RT Corporation.