

ご購入はこちら **ゲートウェイや高性能端末で差がつく**

# IoT時代の低消費電力Linuxプログラミング入門

第2回

小型LinuxとBLEセンサ・モジュールをつなぐ

平 一平

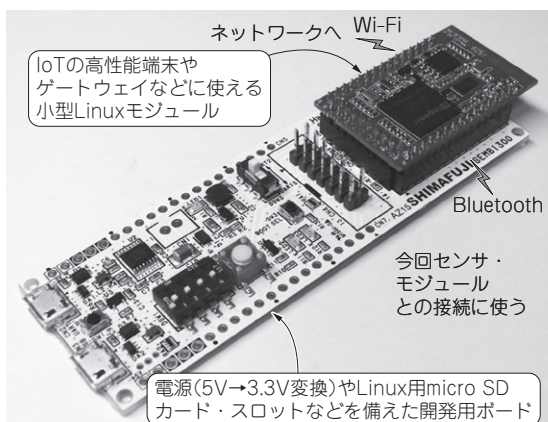


写真1 今回やること…小型LinuxモジュールとBLEセンサ・モジュールをつなぐ

小型Linux Halley IoTモジュールの開発キットT-Kernel2/MIPS-M150ボード  
<http://www.t-engine4u.com/products/tk2mips-m150.html>

前回は、IoT機器向けの小型Linuxモジュール「Halley IoTモジュール」でLinuxを動かし、Wi-Fiでネットワークに接続するところまで準備しました。今回はさらに、センサ・モジュールとつなぎます(写真1、図1)。温度や気圧など10個のセンサを搭載したBLE (Bluetooth Low Energy) 接続のキットSensor Tagを使いました。

## まずはBluetoothの動作確認

まずはSPP (Serial Port Profile) を使用して、Bluetooth機器 (Android 端末) と通信してみます。

### ● Linuxモジュールの初期化手順

Halley IoTモジュールは、初期状態でBluetoothが有効化されていません。Bluetoothを使用するには初期化を行う必要があります。

#### ▶手順1: 初期化コマンドを実行する

Bluetoothの初期化を行うには初期化コマンドを実

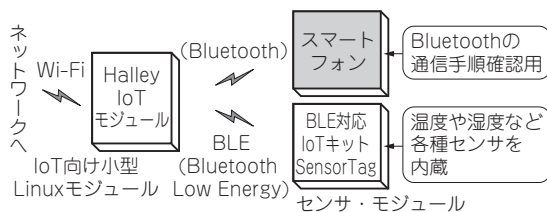


図1 今回は小型Linuxモジュールとセンサ・モジュールをBLEでつなぐ

行します。

```
# /bluetooth/bt_init.sh
```

すると図2(a)のように表示され、Bluetoothが使用できるようになりました。

#### ▶手順2: 設定を確認する

次に設定内容を確認してみます。

```
# hciconfig -a
```

と実行すると、図2(b)のようにBluetoothデバイスの状態が表示されます。hci0が「UP RUNNING」になっているはずですが、Nameはデフォルト状態では「MilkCake」(MIPS-M150の愛称)になっています。この名前はBluetoothの機器検出を行ったときに、相手側に表示される名前です。これを変更したい場合は、bt\_init.shを実行する前に、/etc/bluetooth/main.confの「Name = MilkCake」の部分を変更してください。

#### ▶手順3: 使えるサービスを確認する

次に図2(c)のように、sdptool browse localコマンドでSDP (Service Discovery Protocol) に登録されたサービスを確認します。チャンネル22にシリアル・ポート・サービスが登録されていることを確認します。

### ● 初期化スクリプトbt\_init.shの動作

ここで、bt\_init.sh (リスト1) の処理内容について説明します。

最初に設定ファイルにアクセスするため、カレント・ディレクトリをbt\_init.shのあるディレクトリに変更しています (リスト1の①)。

そして、設定するBDアドレスを読み込んでいます。