

# TWELITE…ファームウェア改造で低電力化

原田 恒平

## 農場や工場のバラマキ・センシングに向くかも

トワイライト (モノワイヤレス) は、低消費電力の中距離無線付きマイコンです。通信能力があまり高くないため、数十m<sup>2</sup>の範囲で、1カ所当たり1～2個のセンサ値を監視するようなIoTデバイスに向いています。送信出力の違いで、標準機種のBLUEと、高出力のREDがあります。今回は、低消費なセンシング・デバイスをテーマに、BLUEを使用します。特徴は、次の通りです<sup>(1)</sup>。

- 通信周波数は2.4GHz帯
- 通信規格はIEEE 802.15.4準拠
- 通信速度は最大250kbps
- 通信プロトコルは独自規格であるTWELITE NETおよびIEEE 802.15.4 MAC
- 消費電流は、RAMの電源もOFFにするスリープ時で約0.1  $\mu$  A
- 送信電流は約15mA、受信電流は約17mA
- 搭載マイコンのコアは32ビット
- 通信距離は、通信に干渉するものがない環境で、見通し1km程度
- 中継機により通信経路の冗長化が可能 (中継器間で最大3ホップ可能なマルチホップ中継機能)

## 本稿で取り組むこと

### ● 問題点：低消費電力で数十台接続するのは難しい

低消費電力なIoTセンシング・デバイスでは、子機間欠モードで動作させる方法があります。子機が1台しかない場合であれば、その方法で問題ないのですが、子機が10台、20台と複数存在した場合は、子機から一斉にデータが送信されると、親機に通信が集中 (輻輳) してしまい、データを正しく受信できなくなります。さらに、子機1台からの受信データの処理に時間がかかってしまった場合は、その間に他の子機から送信されたデータを確実に取得できません。ひど

いときには次の送信タイミングも重なってしまい、相当数のデータを取りこぼしてしまう恐れがあります。「子機ごとのスリープ間隔を重ならないように設定すればよい」という考え方もありますが、それでも子機の電源が入るタイミングや、リトライ処理などによるスリープ開始タイミングのズレなどによって、いつかは通信タイミングが重なってしまいます。

### ● 解決策：親機から子機へデータ送信を要求する

この問題を解決するためには、子機から定期的に親機へデータを送信するのではなく、親機から子機へデータ送信を要求する方法があります。TWELITEの標準アプリケーションや、その他の低消費電力な動作向けのファームウェアでは、親機からのメッセージを受信する仕組みが無効化されているため、代わりに、シリアル通信アプリケーションApp\_Uartのファームウェアを使用します。このファームウェアは、連続動作となるため、電池での駆動が推奨されていません。そこで、少しでも長く稼働できるように、ファームウェアをカスタムしてみます。

### ● 低消費電力で動作するようにファームウェアをカスタムする

「ファームウェアをカスタムする」と聞くと、ハードルが高い作業のように感じるかもしれませんが、TWELITEのファームウェアは、あらかじめ用途別に幾つか用意されており、オープンソースのためわずかな追加・修正で、カスタムが可能です。

## 実験用ハードウェア構成

親機としてUSBタイプの<sup>モノステック</sup>MONOSTICKとPCを使用します。子機には、<sup>トワイライトチップ</sup>TWELITE DIPに、アナログ温度センサを接続し、電池駆動できるようにします (写真1)。

図1にハードウェアの構成を、表1に今回使った部品を示します。

子機のアナログ・センサはMOSFETを使うことで、