

パート3：3GIMを使ったツイッター連携

1. ツイッター連携とは

パート3では、前回パート2で紹介した **IoT デバイス**となる Arduino UNO や Arduino M0、Leonardo、その他 Genuino101（Arduino 製品）+ 3GIM シールド+ 3GIM を使って、温度センサ値をツイッター連携したり、スマホから3GIM上のLEDを点滅させる方法をご紹介します。

ここでのツイッター連携として、以下の2つの事例を紹介します。

(1) 事例1：IoT デバイスから温度センサ値をツイートする方法



図1. 事例1：温度センサ値をツイートする方法

こちらは、IoT デバイスからツイート送信で、遠隔モニタリングすることに使えるものです。

(2) 事例2：スマホからツイートされた値を読み取り LED を点灯・消灯させる方法



図2. 事例2：スマホからツイートし IoT デバイスを制御する方法

こちらは、スマホなどから IoT デバイスに向けて制御する方法で、遠隔制御として使える方法となります。

ツイッターを利用する点では、無償ということで気軽に利用できるメリットはあります。また公開することになり誰もが閲覧できるメリットもあります。このことを理解した上で活用すべきで、何にでも使えるものではありませんので注意してご利用ください。

2. ツイッター連携のための準備

3 GIM を使ったツイッター連携は、以下の手順で、あらかじめ twitter.com の①**アカウント取得**と、Arduino との接続のための②**トークンの取得**を行っておく必要があります。ここではこれらの準備についてご紹介しておきます。

① ツイッターアカウントの取得

まずは、ツイッターのアカウントを取得します。こちらは、以下のところで自分のメールアドレスを使って取得してください。アカウント取得先は、「twitter.com」の「アカウント作成」から行います。Web サイトの手順に沿っていけば簡単に取得できます。



図3. ツイッターのアカウント登録サイト

② トークンの取得

つぎにツイッターを Arduino から連携するためのトークン（ツイッター認証キー）を取得します。こちらは、あらかじめ以下のサイトにアクセスして取得します。

<http://arduino-tweet.appspot.com>

手順としては、上記 URL サイトの取得メニュー画面（図4参照）に入って、図中の「Step1」のトークン取得を選択します。

このトークンは、変更されることはありませんので、忘れた場合には、再度上記の手順で入ってトークンを取得することができます。



図4. Twitterのトークン (Arduino 認証キー) 取得画面



図5. Twitterのトークン (Arduino 認証キー) 取得画面

つぎに図3. で取得 (登録) したアカウントを使って、「連携アプリを認証」ボタンをヒ

ットすることで、図6の画面でトークンを取得することができます。



図6. Twitterのトークン (Arduino 認証キー)

ここで得られたトークンは、プログラム (Arduino のスケッチ) に書き込みますので、必要なときにコピー/ペーストして利用します。(後述)

3. IoTデバイスからの温度センサ値ツイート (事例1)

既にパート2で紹介しましたIoTデバイスを使って、温度センサの値を取得し、それを3分ごとにツイートするスケッチを紹介しましょう。

その前に Web ブラウザから、直接ツイートする例をご紹介します。この場合、以下の行をキー入力して、直接ツイートします。

```
http://arduino-tweet.appspot.com:80/update?token=YOUR_TOKEN&status=MESSAGE
```

ただし、ここでの **YOUR_TOKEN** は、上記で取得したトークンを、**MESSAGE** は、ツイートするメッセージを入れます。この結果は、正しくツイートされた場合には、画面上に「OK」のみが表示されます。

例えば、**MESSAGE** に「temp=25.6%20C」とすると、ツイッター画面には、「temp=25.6 C」と表示されます。ここで「%20」はアスキーコードのスペース (空白文字) となります。

以上のように直接キー入力でツイートすることが理解できたことで、今度は3 GIM で送信する方法を学びます。この3 GIM でツイートする場合には、上記の直接キー入力部分を2つに分けて ([3 GIM V2.0 マニュアル](#)の \$ WP 説明文の URL とボディ)、「\$ WP」コマンドで送信します。この場合には、以下のようにしてシリアル通信 (ここでは「Serial1」) で送信します。

```
$WP URL "token=YOUR_TOKEN&status=MESSAGE"
```

ここで、**URL**= <http://arduino-tweet.appspot.com:80/update> となります。

それでは、3分ごとに、温度センサ値を取得し、ツイートするスケッチをご紹介します。

リスト①：温度センサ値を3分ごとにツイート

```
// リスト①：温度センサ値を3分ごとにツイート
#define BAUDRATE 38400
// #include <SoftwareSerial.h>
// SoftwareSerial Serial1(4, 5);

String URL = "http://arduino-tweet.appspot.com:80/update";
String TOKEN = "token=YOUR_TOKEN";

void setup() {
  while (!Serial);
  Serial.begin(BAUDRATE); Serial.println("Please wait");
  Serial1.begin(BAUDRATE);
  pinMode(7, OUTPUT);
  digitalWrite(7, HIGH); delay(100); digitalWrite(7, LOW);
  String str;
  unsigned long tim = millis();
  do {
    str = Serial1.readStringUntil('\n');
    if (millis() - tim > 14000) {
      Serial.println("conext Error"); while (1);
    }
  } while (str.indexOf("3GIM")<0);
  Serial.println("Ready");
  pinMode(A0, OUTPUT); digitalWrite(A0, LOW); // LM61BIZ: GND
  pinMode(A2, OUTPUT); digitalWrite(A2, HIGH); // LM61BIZ: Vss(3.3V)
}

void loop() {
  static int a1_old = 0;
  int a1_new = analogRead(A1); //温度センサ値
  if ( a1_old != a1_new ) {
    a1_old = a1_new;
    float temp = analogRead(A1) * 0.322 - 60.0; // 温度センサ値変換式
    tweet(String("TIME=" + getTime() + "/temp=" + String(temp) + "C"));
    Serial.println("wait 3 min"); delay(180000); // ツイートは3分毎とする
  }
}

boolean tweet(String message) {
  Serial1.println( "$WP " + URL + " %" + TOKEN + "&status=" + message + "%");
  String str;
  do {
    str = Serial1.readStringUntil('\n');
  } while (str.indexOf("$WP")<0);
  Serial.println(Serial1.readStringUntil('\n'));
  return (str.indexOf("OK")>0);
}

String getTime() {
  Serial1.println("$YT"); //日時取得
  String str;
  do {
    str = Serial1.readStringUntil('\n');
  } while (str.indexOf("$YT") < 0);
  if (str.indexOf("OK") > 0) {
    str = str.substring(7); str.replace(" ", "%20");
  }
  return str;
}
```

取得したトークンに変更

UNOの場合は、0.488に変更

ツイートする関数

日時取得関数

それでは、このスケッチでの実行による結果を確認してみましょう。正しく温度センサ値がツイートされた場合には、図7のようにツイート画面に日時と温度センサ値が表示されます。



図7. ツイートされた画面表示例

上記のスケッチでは、日時と温度センサ値の間には、「/」を入れていますが、これを改行コードに置き換える場合には、「%0D」に変更することで改行されます。

4. スマホからのツイートによる LED 点滅 (事例2)

今度は、スマホ側やPCから「LED on」または「LED off」をツイートして、それをIoTデバイスが読み取って、3 GIMにあるLEDを点滅させる方法を学びましょう。

もちろんスマホ側からツイートできるユーザは限られていて、アカウント取得した際のアカウント名とパスワードが分かる人たちがツイートできるようになります。

スマホからツイートするには、twitter.com サイトからアカウントとパスワードでログインし、ツイートキー入力するだけで簡単に行えます。



図8. ツイート画面から「LED on」と入力

つぎに、ツイートされた文字列を、IoTデバイスから読み込むことを行いますが、その前にWebブラウザから、直接キー入力して、ツイッターの内容を取得する方法を学びましょう。以下が、ツイート先の内容を、最新版から複数回前まで渡ってテキストで取得する方

法です。

```
http://twansform.appspot.com/ACCOUNT /text/NO
```

ここで、**ACCOUNT** が、ユーザアカウント名となります。上記で登録したアカウント名をここに入れて、**NO** には、出力する行数を入れます。

たとえば、事例 1 で出力したものを (NO=) 5 行だけ出力してみると、以下のような内容が Web ブラウザに表示されます。

```
TIME=2016/04/13 23:07:27/temp=19.39C  
TIME=2016/04/13 23:05:26/temp=19.70C  
TIME=2016/04/13 23:04:25/temp=20.30C  
TIME=2016/04/13 23:01:24/temp=20.61C  
TIME=2016/04/13 23:00:23/temp=20.30C
```

それでは、このことを理解した上で、スマホから「on」または「off」が入った文字列を順次送信したら、IoT デバイス側で 1 行だけ読み、3 GIM 上の緑 LED が点滅するプログラムを開発してみましょう。

すでに、パート 2 からサンプルを見ているとご理解頂けると思いますが、今回の場合には、ツイッター上の 1 行は、「\$WG」を付けて上記のツイート文を読み込みます。この場合には、以下のようになります。

```
$WG http://twansform.appspot.com/ACCOUNT /text/1
```

この行を、あらかじめ URL に設定しておきますので、ACCOUNT のみをご利用アカウント名に変更して、ご利用ください。また、スマホや PC からのツイートでは、同じ文章を 2 度キー入力するとエラーとなりますので、毎回買えるには、カウントアップする番号などを先に書いてから小文字の「on」または「off」を入れて、3 GIM 側の LED の点灯・消灯を行っててください。

リスト② : スマホからツイートして LED on/off 遠隔制御

```
// リスト② : スマホからツイートして LED on/off 遠隔制御  
#define BAUDRATE 38400  
// #include <SoftwareSerial.h> UNO の場合コメント外す  
// SoftwareSerial Serial1(4, 5);  
  
String URL = "http://twansform.appspot.com/ACCOUNT/text/1";  
  
void setup() {  
  while (!Serial);  
  Serial.begin(BAUDRATE); Serial.println("Plsese waite");  
  Serial1.begin(BAUDRATE);  
  pinMode(7, OUTPUT);  
  digitalWrite(7, HIGH); delay(100); digitalWrite(7, LOW);  
  String str;
```

```

unsigned long tim = millis();
do {
  str = Serial1.readStringUntil('\n');
  if (millis() - tim > 14000) {
    Serial.println("conext Error"); while (1);
  }
} while (str.indexOf("3GIM") < 0);
Serial.println("Ready");
}
void loop() {
  String str;
  Serial1.println("$WG " + URL);
  do {
    while (!Serial1.available());
    str = Serial1.readStringUntil('\n');
  } while (str.indexOf("$WG") < 0);
  Serial.println(str);
  if (str.indexOf("OK") > 0) {
    str = Serial1.readStringUntil('\n');
    Serial.println(str);
    if (str.indexOf("on") > 0) {
      Serial1.println("$YL 1");
    }
    else if (str.indexOf("off") > 0) {
      Serial1.println("$YL 0");
    }
  }
  delay(30000); // 30 秒ごとツイート確認
}

```

ツイート文読み込み

LED on/off

これまでの資料は、以下のところからダウンロードできます。

パート1：3GIMを使った開発事例

http://tabrain.jp/3GIM_V2.2/Part1%20Arduino%20and%203GIM.pdf

パート2：3GIMを使った技術資料（遠隔制御・メール送信など）

http://tabrain.jp/3GIM_V2.2/Part2%20Arduino%20and%203GIM.pdf

以上