

## パート2 : Arduino+ 3G シールドの技術例紹介

### 1. Arduino+ 3G シールドの技術可能性

前回において Arduino および 3G シールドの概要をご紹介しました。ここでは、具体的な Arduino と 3G シールドを使った技術的な可能性を紹介していきます。

Arduino 側では、多くのセンサ類やアクチュエータ類を簡単に、しかも短時間で制御できます。例えば新しいセンサなどを購入した場合でも、アナログ接続かデジタル接続、もしくはシリアル通信 (I2C や SPI) による接続のいずれかであれば、ネット上に多くの事例紹介とともに、そのサンプルスケッチ (Arduino ではプログラムのことをスケッチと呼びます) によるヒントが頂けるようになっていきます。

また 3G シールドを使うことで、インターネット上の Web アプリケーションやクラウドサービス、ツイッター連携、それにメール送信などが利用できるようになります。

つまり Arduino と 3G シールドを使うことで、ワイヤレスセンサネットワークや、そのデータを収集し分析するビッグデータのビジネス、それに前回紹介した M2M ビジネスなどへと繋げることができます。また、その他の 3G シールドの機能としては、GPS (位置情報取得) 機能や、時刻の取得機能、SMS 送受信機能などを持ち合わせています。

以下では、Arduino と 3G シールドを使う上での技術ハードルとして、どのようなものがあるか、またそれがどれだけ簡単かをご紹介していきたいと思えます。

### 2. Arduino と 3G シールドの利用環境の準備

Arduino は、ハードウェアのマイコンボードと、統合開発環境 (IDE と呼ばれる) の 2 つのことを指して言います。マイコンボードは、ネット上で簡単に購入でき、1-2 日間ほどですぐに入手できるようになっています。(スイッチサイエンス社のサイトなど参照)



図 1. Arduino とはマイコンボード (左) と統合開発環境 (右)

また、統合開発環境は、<http://arduino.cc/> のサイトから、PC にダウンロードするようになっています。PIC マイコンや H8 マイコンなどに比べ、Arduino といくつかの電子部品、それに USB ケーブルさえあれば、直ぐに動かすことができるようになっています。ただ、Arduino のドライバを一度だけ、PC にインストールしておく必要はあります。

つぎに 3G シールドですが、こちらもネットから購入できるようになっていますが、利用するための SIM カードは、手持ちのものか、別途手配する必要があります。(現在 3G シールドで利用できる SIM カードは、NTT ドコモの FOMA の SIM カードに限定)

また 3G シールドのソフトウェアの最新版は、ホームページの <http://3gsa.org/> からか、もしくは Wiki ページの <http://a3gs.wiki.fc2.com/> に掲載されているところから、ダウンロードできるようになっています。

この Arduino と 3G シールドを繋いで使う上では、3G シールド側に出ている複数のピンを、Arduino 側に差し込むだけで利用できるようになっています。特にソフトウェアの設定などは不要となっています。

### 3. スマホなどのクライアント側から Arduino + 3G シールド側の操作

それでは Arduino と 3G シールドを使ったプログラミングがどれだけ簡単かをご紹介します。いきましょう。

まずご紹介する事例の全体像を説明していきましょう。

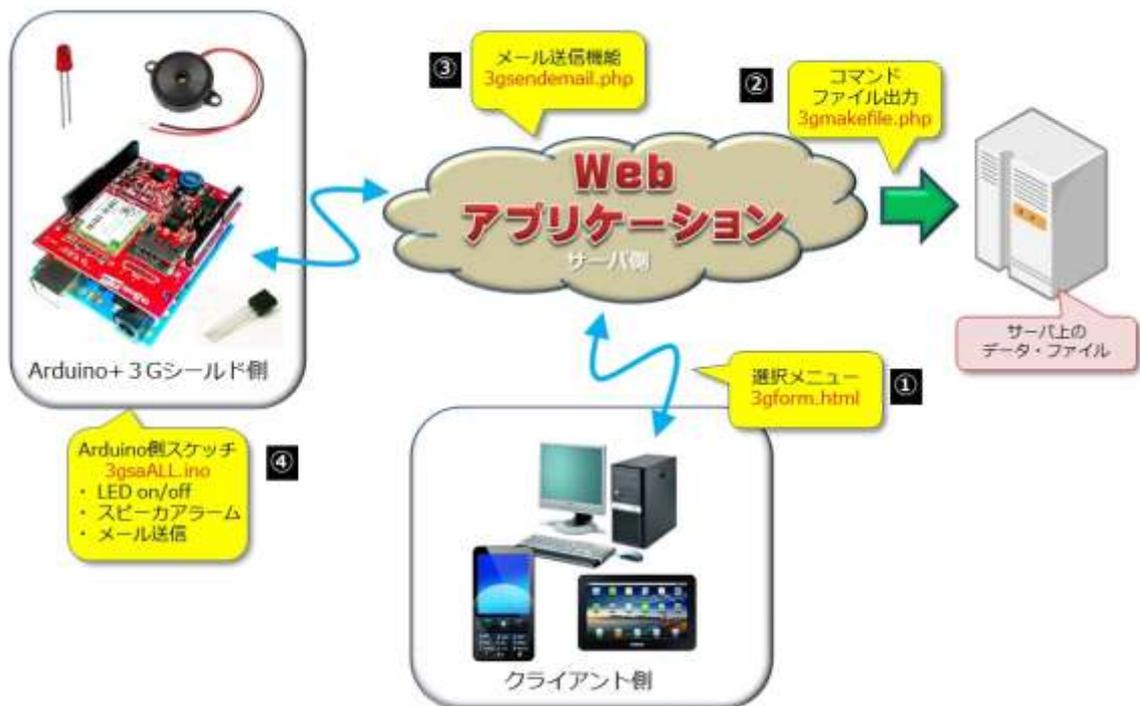


図 2. Arduino + 3 G シールドによる開発事例

この図においては、まず①のクライアント側からサーバ上にある HTML (3gform.html) を起動します。この起動によって、以下のようなメニューがスマホや PC などに表示されます。ここでは単純に、4つのメニューの選択ができるようになっています。

- 1) LED 点灯
- 2) LED 消灯
- 3) 温度センサ値をメールで受信
- 4) スピーカのアラームを発生 (5 秒間)

つまり、Arduino と 3 G シールド上に付けた LED や圧電スピーカ、それに温度センサを使って、遠隔でのクライアント側 (スマホや PC) から、LED を点滅させたり、スピーカを鳴らしたり、また温度センサの値を取り出し自分のメールアドレスに受信したりするものです。

ここでは、サーバ側に、HTML プログラムが 1 つと、PHP プログラムの 2 つの計 3 つを用意しておきます。また、Arduino + 3 G シールドを使う上では、Arduino 上に 1 つのスケッチ (制御プログラム) を用意しておきます。

それでは、以下にそれぞれのプログラムとスケッチをご紹介します。

#### 4. サーバ側の 3 つのプログラム

サーバ側には、上の図で示した番号のとおり、①クラウドから直接呼び出される HTML の「選択メニュー」プログラム(3gsaform.html)と、②「選択メニュー」から自動的に起動する「コマンドファイル出力」プログラム(3gmakefile.php)、それに③ 3 G シールド側から起動される「メール送信機能」が用意されます。

それでは、それぞれのプログラムについて紹介していきましょう。

① 「選択メニュー」プログラム : 3gform.html (HTMLプログラム)

このプログラムはとても簡単で、上記選択肢メニュー (LED点灯やメール送信、アラーム発生) を表示させ、「送信」ボタンが押されると、②の「コマンドファイル出力」プログラム (3gmakefile.php) が、引数渡しで起動されます。

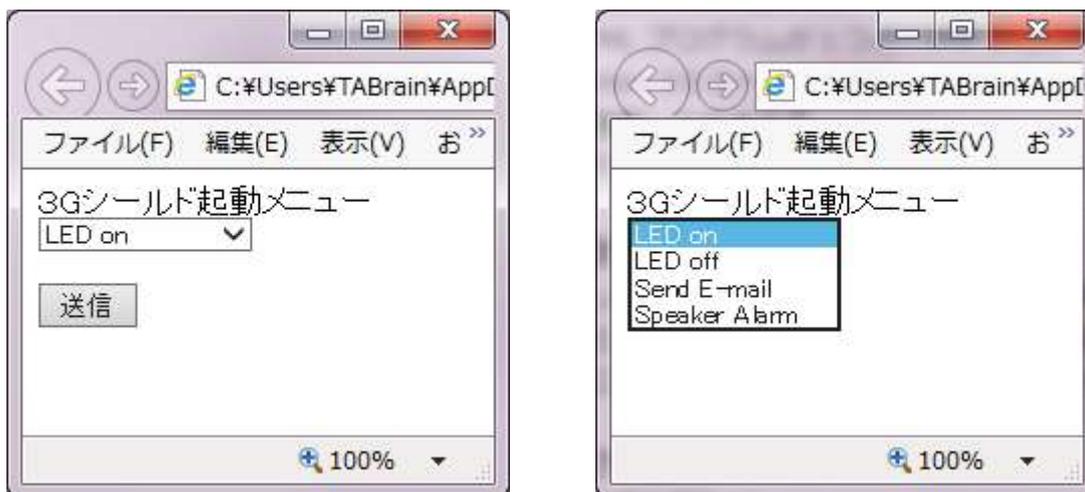


図3. クライアント側で起動した「選択メニュー」(3gform.html) の画面

リスト1に、そのプログラム内容を紹介します。

つまり、このプログラムは、端末側から「[http://\\*\\*\\*\\*/3gfrom.html](http://****/3gfrom.html)」で起動すると、図3の選択メニューが表示され、4つの選択された後に「送信」ボタンが押されると同時に、「3gmakefile.php」のHTTPリクエストが行われます。

```
<form action = "3gmakefile.php" method = "post">
<p> 3Gシールド起動メニュー <br>
<select name="cmd" id="cmd">
<option value="LED on">LED on</option>
<option value="LED off">LED off</option>
<option value="Send E-mail">Send E-mail</option>
<option value="Speaker Alarm">Speaker Alarm</option>
</select></p>
<input type="submit" value="送信">
</form>
```

リスト1. メニュー選択プログラム (3gform.html)

② 「コマンド出力」プログラム : 3gmakefile.php (phpプログラム)

このプログラムは、上記①で選択された値が、「`$_POST["cmd"]`」経由で渡され、それを変数「`$x`」に代入し、一旦「temp.xxx」というファイル名に値を出力します。その

後、ファイル名「3gsa.xxx」にファイルをコピーしています。

```
<meta name="viewport" content="width=device-width,
initial-scale=1.0" />
<?php
// echo $_POST["cmd"];
$x=$_POST["cmd"]."";
$fn1="temp.xxx";
$fp=fopen($fn1,'w');
fputs($fp, $x);
fclose($fp);
$fn2 = "3gsa.xxx";
copy($fn1, $fn2);
echo '実行コマンド = [' . $x . ']';
?>
<br>
<input type="submit" name="buttonName" value=" 戻 る "
onClick="history.back()" />
```

**リスト2. コマンド出力プログラム (3gmakefile.php)**

この中で一旦ファイル名を置き換えているのは、ファイルをオープンした状態で、3G シールド側からのポーリングによる読み込みによって、エラーとなるのを防ぐためのものとなっています。



**図4. クライアント側に表示される「コマンド出力」プログラムの表示例**

このプログラムが終了すると、選択されたコマンドの内容がクライアント画面に、図4のように表示されます。

**③ 「メール送信機能」プログラム : 3gsendmail.php (phpプログラム)**

このプログラムは、Arduino+ 3G シールド側から読み込みされるプログラムで、メール送信先「\$\_GET["email"]」に、引数で渡されてくる温度センサの値「\$\_GET["temp"]」を書き込んで、送信するプログラムとなります。

```

<HTML>
<HEAD><TITLE> 3G send e-mail </TITLE></HEAD>
<BODY>
<H3> 3G shield temp get </H3>
<?php
if(mail($_GET["email"], // to メールアドレス
    'Hi 3G shield' // タイトル
    '3Gshield ALARM ' . "%r%n" . 'DATE = ' . date('Y-m-d') . 'TIME
= ' . date('H:i:s') . "%r%n" . 'TEMP = ' . $_GET["temp"] //本文
    'From: 3GSA<info@***.jp>' . "%n" .
    'X-Mailer: PHP/' . phpVersion()))
echo '<B>SUCCESS TO SEND</B><BR>';
else
echo '<B>faile to mb_send_mail</B><BR>';
?>
</BODY>

```

**リスト3. メール送信機能コマンド出カプログラム (3gmakefile.php)**

このプログラムを起動することで、つぎの図5のようなメールが送られてきます。



**図5. 温度センサ値のメール受信画面**

ここでは、温度センサ値のほかに、日付と時間も同時に添えてメールで送ってきます。

## 5. Arduino+ 3G シールド側のスケッチ

つぎの Arduino+ 3G シールド側には、サーバ側にあるデータ・ファイル（②で書出しされたコマンドの内容）を読み込む機能や、温度センサの値をメール送信する際に、上記③を起動する機能などを備えたスケッチ④が用意されます。（Arduino のプログラムのことをスケッチと呼びます）

このスケッチは、基本的には立ち上がって自動的に無限ループによる Web 上のコマンド出力ファイルを 3 秒おき（スケッチの最後に「delay(3000);」と記述）に見に行き、その中身が変わった時点で、それぞれ 4 つの処理を区別して行います。この Web 上のコマンド出力ファイルを読みに行くのがポーリング作業で、Arduino+ 3G シールド側は、繰り返しコマンド出力ファイルが何に置き換わったかを見に行くことが必要となります。

またここでは、メール送信と圧電スピーカの連続的な操作を避けるため、制御を見送る処理を取っています。この見送るための変数として、2 つのブーリアン値「swm」と「swa」を使って制御を行っています。

```
#include <SoftwareSerial.h>
#include "a3gs.h"
// LM61BIZ output pin: A1
#define LM61BIZ_Pin 1
int port = 80;
int spk=8;
//char res[15];
char path2[100]; //メールアドレス+温度センサ値バッファ
int getTemp(void)
{
    int mV = analogRead(LM61BIZ_Pin) * 4.88;
    return (mV - 600);
}
void spkAlarm() {
    for(int i=0; i<500; i++) {
        digitalWrite(spk,HIGH);
        delay(3);
        digitalWrite(spk,LOW);
        delay(2);
    }
}
void setup()
{
    pinMode(A0, OUTPUT); // A0(LM61BIZ - GND)
    digitalWrite(A0, LOW);
    pinMode(A2, OUTPUT); // A2(LM61BIZ - VSS+)
    digitalWrite(A2, HIGH);
```

温度センサ値の計算関数

圧電スピーカを 5 秒間鳴らす関数

温度センサのピン接続 (A0,A1,A2 利用)

圧電スピーカを D8 に接続

```

Serial.begin(9600);
pinMode(8,OUTPUT);
pinMode(13,OUTPUT);
while(!(a3gs.start() == 0 && a3gs.begin()==0)) ;
Serial.println("Start..");
a3gs.setLED(true);
}
void loop()
{
static boolean swa=true, swm=true;
if(swa==false || swm==false) a3gs.setLED(false);
const char *server = "www.*****.jp";
const char *path = "/**/3gsa.xxx";
char res[15];
const int len = 15;
if (a3gs.httpGET(server, port, path, res, len) == 0) {
Serial.println(res);
if(strncmp(res,"LED on",6)==0) {
swa=true; swm= true;
digitalWrite(13,HIGH);
}
else if(strncmp(res,"LED off",7)==0) {
swa=true; swm= true;
digitalWrite(13,LOW);
}
else if(strncmp(res,"Send E-mail",11)==0) {
if ( swm ) {
int temp = getTemp();
sprintf(path2,
"/*****/3gsendemail.php?email=*****&temp=%d.%d", temp/10,
temp%10);
if (a3gs.httpGET(server, port, path2, res, len) == 0)
Serial.println( res );
delay(5000);
}
swm = false;
}
else if(strncmp(res,"Speaker Alarm",13)==0) {
if (swa) { spkAlarm(); };
swa = false;
}
}
}

```

LED を D13 に接続

3 G シールドの初期化

3 G シールド上の LED 点灯

3 G シールド上の LED 点滅

サーバ側のサイト

コマンドファイルのフォルダ

コマンド読み込み「res」に設定

LED on の場合

LED off の場合

温度センサ値のメール送信 の場合

送信先メールアドレス

3gsendmail.php 起動

圧電スピーカアラームの場合

```
    delay(100);
  }
  Serial.print(".");
  delay(3000);
}
```

#### リスト4. Arduino+ 3G シールド側のスケッチ (3gsaALL.ino)

このスケッチの中では、3G シールド専用の関数群は、赤字のわずか4つしか使っていません。最初の初期化で「a3gs.start」と「a3gs.begin」を使い、つぎに3G シールド上のLEDを On/Off させるための「a3gs.setLED」、それにインターネットとの接続のために「a3gs.httpGET」の4つとなります。

この「a3gs.httpGET」は、最初のものが「コマンド出力」ファイルの内容を読み取るもので、後のものは温度センサ値をメール送信する「3gsendmail.php」を起動するものとなっています。

このように、httpGETでは、サーバ側のプログラム位置のアドレス（変数の「server」と「path」または「path2」）が設定され、ポート番号「port」それに引き渡す「res」との長さ「len」の引数だけで、インターネットとの送受信ができるようになっています。

如何でしょうか？これだけ簡単に、インターネットとの連携ができることは、自分でもできると思われるのではないのでしょうか？

これらのプログラムを応用するだけで、Arduino+ 3G シールド側の入出力の電子部品を、簡単に遠隔地から操作でき、またセンサ値などを簡単に取得することができるようになります。

つぎの号では、フリーのクラウドサービス Xively.com に、センサ値をアップしたり、ツイッターにセンサ値をアップしたりする事例を紹介しましょう。こちらもとても簡単にできる仕組みとなっています。以前某大学の先生が、3G シールドを手にとって僅か数時間後にクラウドにセンサ値をアップされました。この驚異的な時間は、技術ハードルの低さを物語っていると言えると思います。それでは次号をお楽しみにしてください。

以下は、3G シールド関連の情報が掲載されているサイトです。

1. NPO 法人 3G シールドアライアンスのサイト <http://3gsa.org/>
2. 3G シールド技術情報および保守サイト <http://a3gs.wiki.fc2.com/>
3. Facebook「NPO 法人 3G シールドアライアンス」でも最新情報紹介