

平成 26 年 11 月 6 日

平成 26 年度 都市ガスシンポジウムアネックス 長野会場で発表した GPL 適応のソフトウェアのソースコード公開について

広島ガス株式会社 技術研究所  
所長 沖本 裕

当社より、本展示会に「安価な遠隔通信システムの開発」のテーマで、Raspberry Pi という小型パソコンを用いた簡易ロガーを出展しております。ロガーを構成するためのソフトウェアの一部に GPL 適応のソフトウェアを用いておりますので、ライセンスの規約に基づき、ソースコードを公開します。

1. アナログ値を計測するロガー用のソフトウェア

MCP3208(8 チャンネル 12 ビット A/D コンバータ)を用いて、0~5V あるいは 1~5V のアナログ電圧入力を A/D 変換し、計測値を一定間隔で記録します。ソースコードは次の構文でコンパイルします。

ソースコードを置いたディレクトリにおいて、Raspberry Pi の OS が Raspbian の場合、

```
sudo gcc mcp3208_38.c -o am
```

生成した am というファイルに実行権を付与し、

```
/usr/local/analog_measure/
```

というディレクトリ内に配置します。

また、

```
/usr/local/analog_measure/csv
```

```
/usr/local/analog_measure/tempdata
```

というディレクトリを作成します。

さらに、起動用設定ファイルを作成し、実行ファイルと同一のディレクトリに配置します。具体的には、

(1)

intervalsec.csv というテキストファイルを作成し、

第 1 引数 0:cron による分単位以上の計測間隔、1:秒単位の計測間隔

第 2 引数 第 1 引数が 1 の場合の計測間隔(秒)

第 3 引数 計測プロジェクトの名前

第 4 引数 mcp3208 の電源供給電圧を計測し、mV 単位で記載

それぞれの引数をカンマで区切ります。1 行のみのファイルです。

[例]

### 1,3,試験計測,5120

意味: 3秒間隔で計測し、計測プロジェクト名は「試験計測」、mcp3208の電源供給電圧は、5120mV

#### (2)

measure.csv というテキストファイルを作成し、

第1引数 チャンネル番号(0から7)

第2引数 計測信号レベル 0:0-5V、1:1-5V

第3引数 測定名称

第4引数 測定の単位 kW kg など

第5引数 計測信号下限における実計測値アサイン

第6引数 計測信号上限における実計測値アサイン

それぞれの引数をカンマで区切り、各計測チャンネルについて記載するので、8行のファイルになります。計測使用しないチャンネルがある場合でも、省略できません。

#### [例]

0,1,ポンプ動力,kW,0,10

1,1,圧力,kPa,0.1,0.9

2,1,冷却水流量,L/min,0,200

3,0,〇〇電力,kW,0,5

・  
・  
・

7,1,バーナー温度,K,300,1500

意味: チャンネル0について、計測信号1-5Vに対して、下限0kW、上限10kWをアサインし、計測名称は、「ポンプ動力」

計測の実行は、次のコマンドにて行います。

```
sudo /usr/local/analog_measure/am 0
```

により、計測実行し、計測結果をコンソールに表示します。引数は、Raspberry PiのSPI通信用I/OのCE0を用いる場合が0で、CE1を用いる場合が1です。引数を省略するとエラーとなります。

定期的に計測を実行し、計測結果をファイルに記録させたい場合には、cronを用いて自動実行させます。

1分未満の計測間隔の場合、

```
*/1 * * * * sudo /usr/local/analog_measure/am 0 >>  
/usr/local/analog_measure/tempdata/am.csv
```

で、`am.csv` というファイルに記録できます。実行ファイルにより、計測用ヘッダファイル `mcp3208_header.txt` が自動作成されます。このファイルを `am.csv` のヘッダに配置するために、次のスクリプトを実行します。このスクリプトは、FTP クライアントによる FTP サーバへのデータアップロードと同時に、`am.csv` のデータをリセットします。このスクリプトに名前をつけて、実行権を付与し、`/usr/local/analog_measure` に配置します。また、`cron` により、任意の間隔で、実行させて、データを FTP サーバに集積します。

-----ここから

```
#!/bin/sh

#csv のリネーム、これらのフォルダデータはFTPupload 後削除する
if [ -e /usr/local/analog_measure/tempdata/am.csv ]; then
    cp /usr/local/analog_measure/tempdata/am.csv /usr/local/analog_measure/csv/am$(date
+%Y%m%d%H%M).csv
fi

#ftp サーバへの uplocad
ftp -nv > /usr/local/analog_measure/amftp.log 2>&1 <<EOF
open (FTP サーバの IP アドレス)
user (FTP サーバへログインするための ID とパスワードをスペース区切りで記述)
prompt
passive
ascii
cd (FTP サーバのディレクトリ)
lcd /usr/local/analog_measure/csv
mput *.csv
close
quit
EOF

#FTP uoload でのエラー検知
grep ^226 /usr/local/analog_measure/amftp.log > /dev/null

if [ "$?" -eq "0" ] : then
    #FTP エラーがない場合、/csv のファイル内容を消去する
    rm /usr/local/analog_measure/csv/*.csv
fi

#tempdata の CSV ファイルの内容を削除しタイトル行を作成
if [ -e /usr/local/analog_measure/mcp3208_header.txt ]; then
    cat /usr/local/analog_measure/mcp3208_header.txt > /usr/local/analog_measure/tempdata/am.csv
fi

exit 0
```

-----ここまで

## 2. 温度・湿度を計測するロガー用のソフトウェア

DHT11 あるいは DHT22(AM2303)という温度・湿度センサーにより、温度・湿度を計測し、計測値を一定間隔で記録します。ソースコードは次の構文でコンパイルしま

す。

ソースコードを置いたディレクトリにおいて、Raspberry Pi の OS が Raspbian の場合、

```
sudo gcc -o trhsensor trhread.c -L/usr/local/lib -lwiringPi
```

生成した trhsensor というファイルに実行権を付与し、

```
/usr/local/trhsensor/
```

というディレクトリ内に配置します。

計測の実行は、次のコマンドにて行います。

```
sudo /usr/local/trhsensor/trhsensor 11 4
```

により、計測実行し、計測結果をコンソールに表示します。第 1 引数は、温度・湿度センサーが DHT11 の場合 11、DHT22 の場合 22 となります。第 2 引数は、センサーを接続している GPIO の番号を Raspberry Pi 番号で記載します。Broadcom 番号ではないので、ご注意ください。この例の場合、DHT11 センサーを Raspberry Pi 番号 GPIO4 に接続（Broadcom 番号では GPIO23）し、計測する場合のコマンドを示します。

GPIO の Broadcom 番号と Raspberry Pi 番号は次のように割り付けられています。

Broadcom 番号	Raspberry Pi 番号
4	7
17	0
18	1
27	2
22	3
23	4
24	5
25	6

FTP サーバーへのアップロードの方法は、analog\_measure と同様につき省略します。CSV ファイル用のヘッダファイルはプログラムは自動生成しません。ヘッダファイルは、

```
Date,Humidity(%),Temperature(°C)
```

という一行のファイルを手動で作成してください。

### 3. 技術のご質問は下記へお願いします。

日常の業務を優先しますので、回答に時間がかかる場合、あるいは、回答できない場合があることをご了承ください。

(082)822-0795 広島ガス株式会社 技術研究所 担当：中村