

# 小型マイコンを使用した 流星電波観測システム

(ARMマイコン流星電波観測器)

## 概要書 付属

### 目次

|                | (ページ) |
|----------------|-------|
| 付属 A. 主な処理の説明  | 2     |
| 付属 B. 設定ファイル   | 6     |
| 付属 C. エコー計数の方法 | 8     |

初版 : 2022/01/28 : ソフトVer 1.20 対応

付属 A. 主な処理の説明

観測ソフトは、オープンソースで提供されている各種ソフトウェアを利用して処理を構成し、これらのソフトウェアの実行を処理フローに従い Linuxのシェルに記述（プログラミング）することで、自動観測を実現しています。

以下に、各種ソフトウェアをどのように利用しているかを、具体例を用いて示します。

(1) 各種ソフトウェアの使用例

- ・ (a) ~ (e) の項番は、処理フローの項番に対応しています。
- ・ 説明に使用しているファイル名は、便宜上の名称であって、実際の名前は、202201250320. ogg などの年月日時分で作成されます。

(a) RTL-SDRを指定周波数で動作させて受信信号をAudio出力する。

- ソフト仕様
- ・ rtl\_sdr (rtl\_fm) , aplay
  - ・ rtl\_fm を使用して受信信号を復調。
  - ・ 復調した信号を音声としてオーディオ出力デバイスへ出力。
  - ・ この処理を常時稼働させるため、バックグラウンドで実行。

使用例

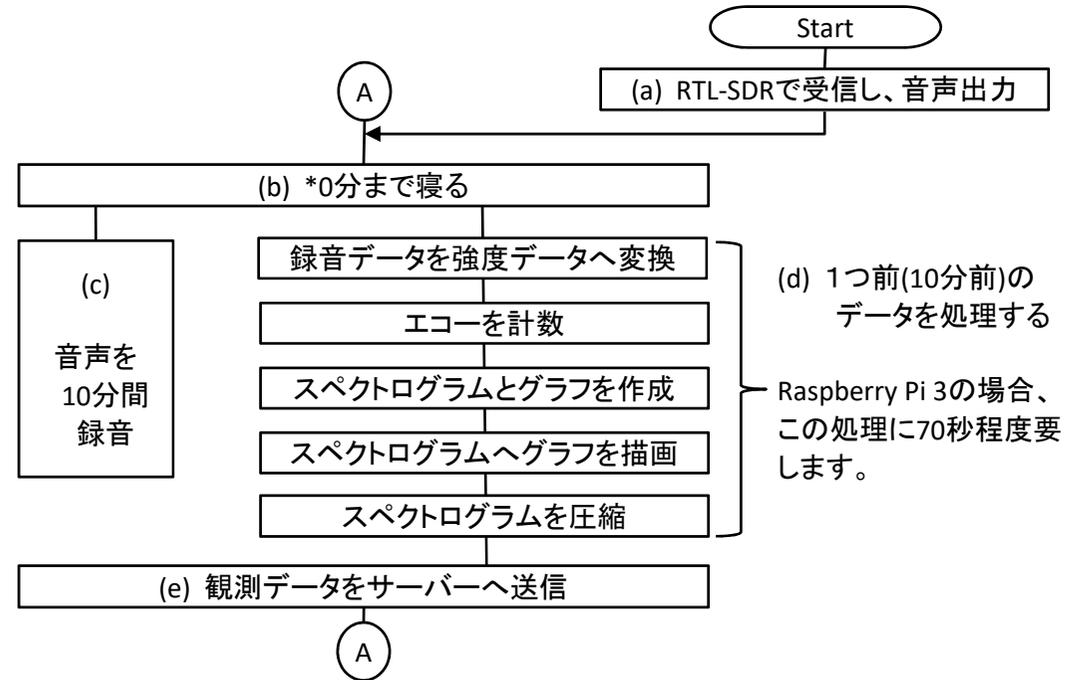
```
rtl_fm -f 53.7540M -M usb -s 6000 -g 49.6 | /usr/bin/aplay -r 6000 -f S16_LE &
```

(b) \*0分まで寝る。

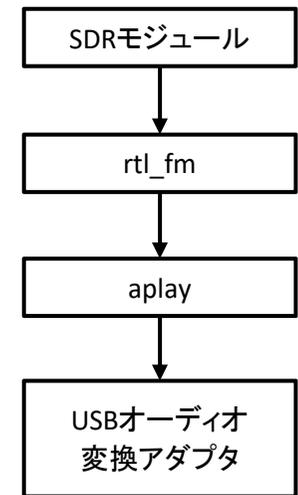
- ソフト仕様
- ・ sleep (コマンド)
  - ・ 起床予定の時刻まで寝る。

使用例

```
sleep WakeupTime - CurrentTime
```



【データの流れ】



(c) 起床後、Audio入力からの音声データの録音を開始する。

- ソフト ・ SOX  
 仕様 ・ pulseaudioを使用して入力デバイス “DEVICE” から音声を 10分（600秒）間、入力し、“FILE.ogg” ファイルへ保存。  
 ・ 並行して1つ前（10分前）に取得した録音データの解析を行うため、この処理はバックグラウンドで実行。

使用例

```
/usr/bin/sox -q -t pulseaudio DEVICE FILE.ogg channels 1 gain 10 trim 0 600 &
```

- 備考 ・ 音量の基本設定は amixer などを使用して事前に設定し、観測状況に応じた微調整を gain で行う。

(d) 一つ前(10分前)の録音データを使用して各種の処理を行う。

(i) 録音データを強度データへ変換

- ソフト ・ SOX  
 仕様 ・ 録音した音声データ “FILE.ogg” をテキストデータにサンプリング 2KHzで変換し “FILE.dat” ファイルへ保存。

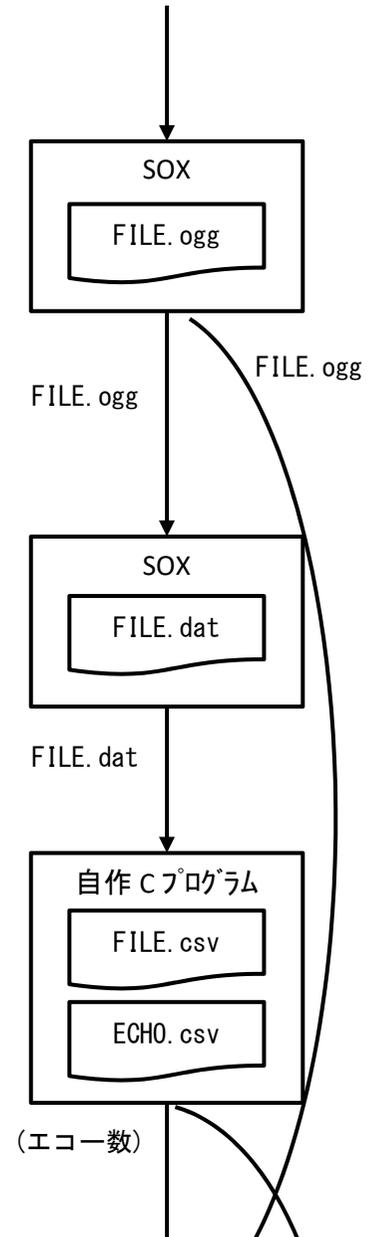
使用例

```
/usr/bin/sox FILE.ogg -r 2k FILE.dat
```

(ii) エコー計数

- ソフト ・ 自作 C プログラム . . . . 頒布パッケージに、ソースコードを同梱。自由に改造して使用できます。
- 仕様 ・ 音声データをテキストへ変換した “FILE.dat” を解析して、エコーを計数。  
 ・ データの扱いを容易にするために、音声データ “FILE.dat” から RMSを算出。  
 ・ 算出したRMSのデータは、グラフ化の為に “FILE.csv” として保存。  
 ・ 計数した結果は、“ECHO.csv” として保存。（1日/ファイル）
- 計数 ・ 10分間の平均値 + 一定値 以上の信号をエコーとする。  
 ・ 10分間の標準偏差の外側の信号をエコーとする。

RMS : 二乗平均平方根 : Root Mean Square



## (iii) スペクトログラム作成

- ソフト ・ SOX  
 仕様 ・ 1 sec/pixel  
 ・ サンプル周波数を 1.8 kHz。  
 ・ HEADER . . . 【例】 観測時刻、エコー数、観測者の情報などを記載。  
 ・ COMMENT . . . 【例】 観測環境などを記載。

## 使用例

```
/usr/bin/sox -q FILE.ogg -n rate 1.8k trim 0 600 spectrogram -X 1 -y 304 -z 50 -p 1 -a -w Dolph -t "HEADER" -c "COMMENT" -o FILE.png
```

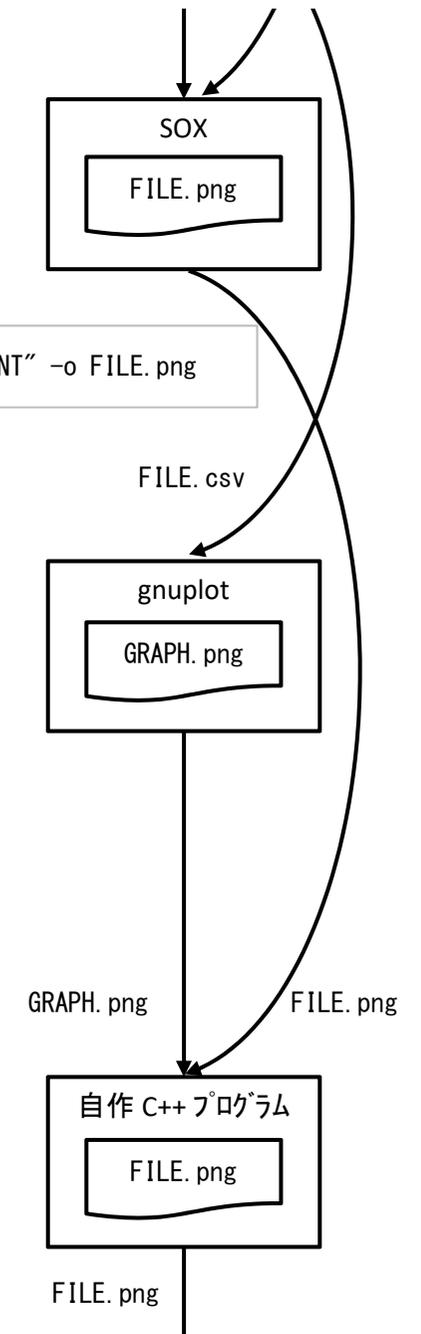
## (iv) グラフ作成

- ソフト ・ gnuplot  
 仕様 ・ 1 sec/pixel  
 ・ RMSデータ "FILE.csv" をグラフ化し、"GRAPH.png" ファイルへ保存。  
 使用例 ・ gnuplot へ設定するパラメータの例。

```
set datafile separator ","
set key off
unset ytics
set yrange [0.03:0.1]
set style fill solid
set border lc rgb "white"
set term png font arial 8 size 635,100 background "black"
set output GRAPH.png
```

## (v) スペクトログラムへグラフを描画

- ソフト ・ 自作 C++ プログラム . . . . 頒布パッケージに、ソースコードを同梱。  
 自由に改造して使用できます。  
 ライブラリ ・ OpenCV  
 仕様 ・ スペクトログラム "FILE.png" の指定の領域へグラフ "GRAPH.png" の指定領域を上書き。

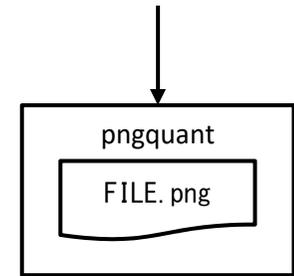


## (vi) スペクトログラムを圧縮

- ソフト ・ pngquant  
仕様 ・ 画質を低下させない程度に圧縮。  
・ 元のファイル "FILE. png" へ上書き。

## 使用例

```
pngquant --ext .png --force --quality=65-80 FILE.png
```



## (e) 観測データをサーバーへ送信

- ソフト ・ sftp , ftp  
expect  
仕様 ・ sftp や ftp を使用して、観測データ (FILE. ogg , FILE. csv など) をサーバーへ送信。  
・ expect は、サーバーとの通信手順を自動化するためのソフトで、サーバーからのデータを受信待ちする用途で使用。



## (2) FTP用設定ファイル (mroftps.txt)

FTPサーバーの情報を設定します。  
ログイン情報を平文で記載しますので、取扱には注意してください。

## 注意点

- ・ ファイル名は、「 mroftps.txt 」から変更できません。
- ・ コメント行は記載できますが、セキュリティリスクを低減するために、書かない方がいいでしょう。
- ・ 各行は、1文字目から記載してください。
- ・ 行の順番を入れ替えることはできません。
- ・ 全角文字の使用はできません。
- ・ 送信先のディレクトリは、FTPサーバーへログインした際のディレクトリからのパスを記載します。

## 【 mroftps.txt の例 (2例) 】

```
abcdefg  
xyz123  
xxxx.yyyy.jp  
public_html/data
```

```
qwert@vvvv.ssss.jp  . . . User ID  
xyz123              . . . Password  
111.222.333.444    . . . Host (ドメイン名 または IP アドレス)  
data                . . . 送信先のディレクトリ
```

付属 C. エコー計数の方法

(1) 方法

エコー計数の方法として、以下の2種類の方法を実装し、選択して使用することができます。  
 どちらの方法であっても、エコーが連続する場合は1件にまとめ、この連続する時間はエコー継続時間となります。

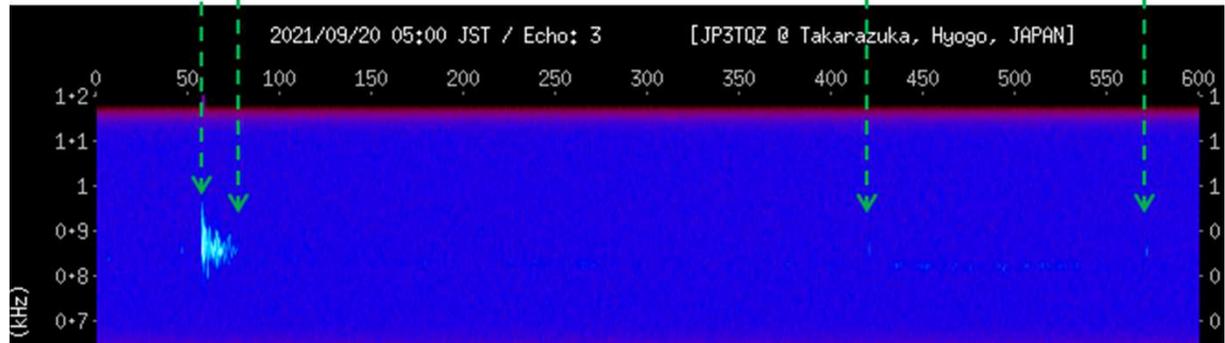
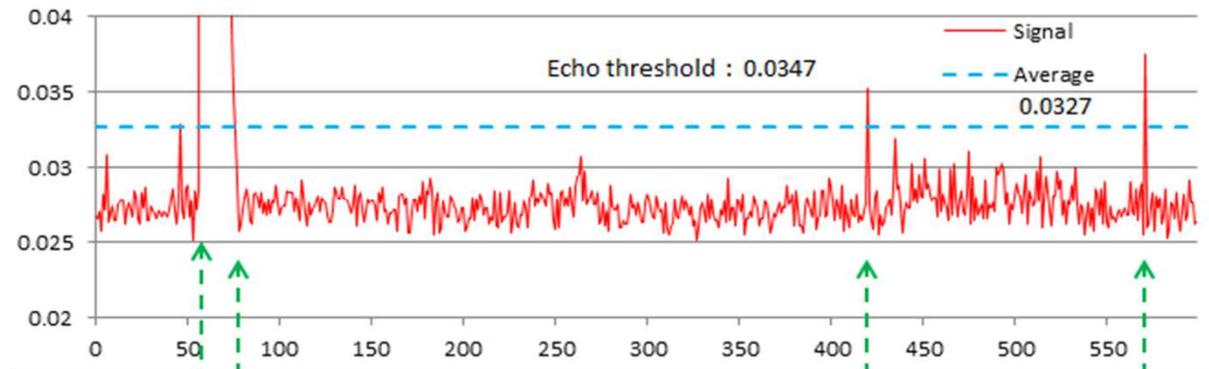
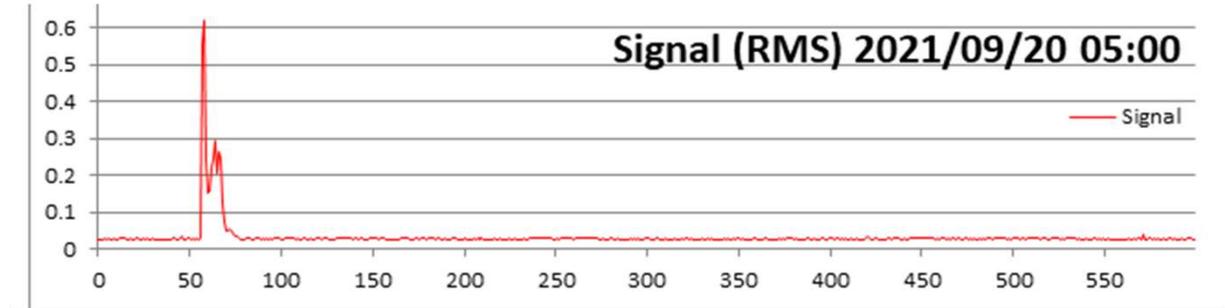
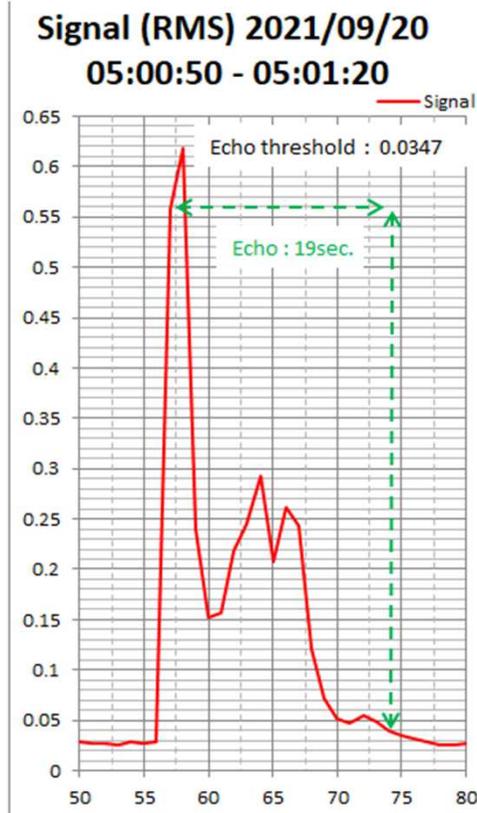
- (a) 10分間の信号強度(振幅)のRMSを算出して、この値が一定値以上の信号をエコーとして検出。
- (b) 10分間の信号強度(振幅)のRMSを算出し、更に標準偏差を算出して、標準偏差の外に存在する信号をエコーとして検出。

(2) 閾値

(a) 10分間の平均値+設定値で判定。

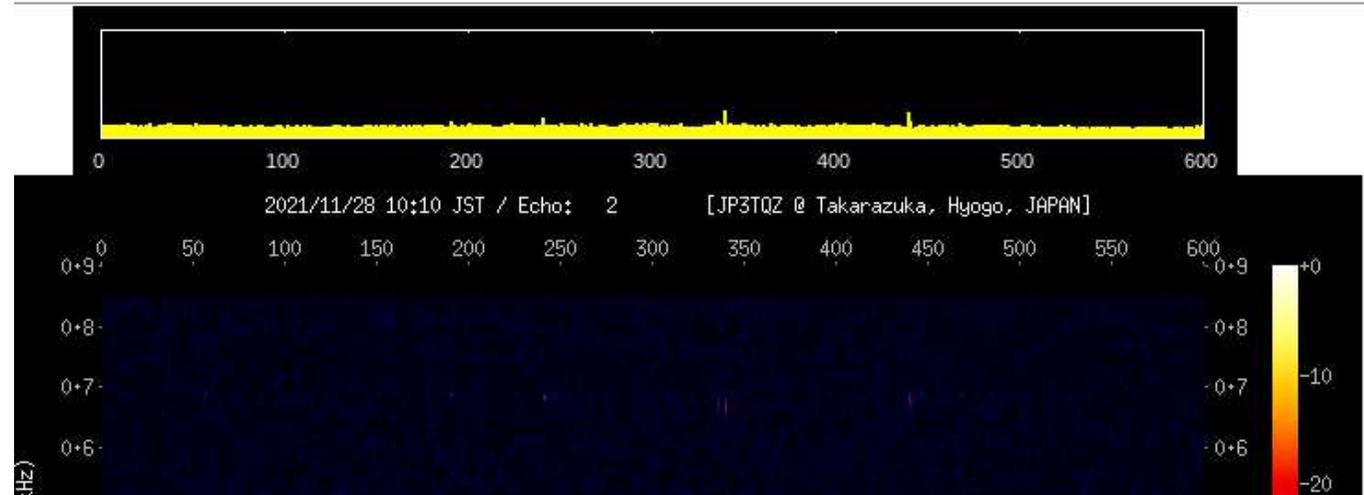
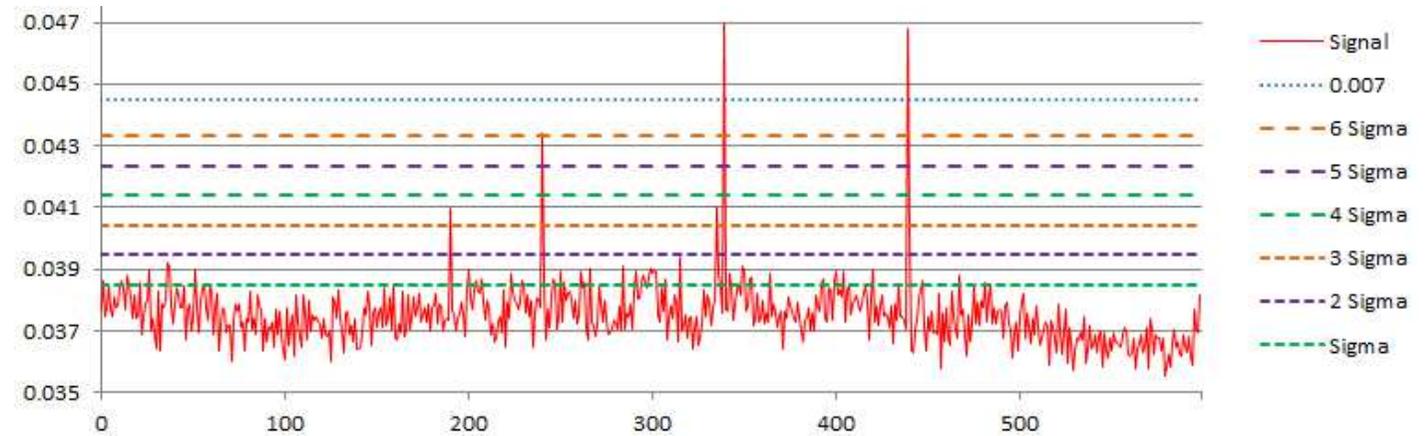
【例】  
 閾値 = 平均値(0.0327) + 設定値(0.002)  
 = 0.0347

|    |          |
|----|----------|
| 51 | 0.026733 |
| 52 | 0.027816 |
| 53 | 0.025185 |
| 54 | 0.028418 |
| 55 | 0.027104 |
| 56 | 0.02858  |
| 57 | 0.557854 |
| 58 | 0.61872  |
| 59 | 0.240024 |
| 60 | 0.152291 |
| 61 | 0.156572 |
| 62 | 0.218419 |
| 63 | 0.24468  |
| 64 | 0.292189 |
| 65 | 0.207426 |
| 66 | 0.262458 |
| 67 | 0.242779 |
| 68 | 0.121956 |
| 69 | 0.071212 |
| 70 | 0.051622 |
| 71 | 0.047549 |
| 72 | 0.054686 |
| 73 | 0.048066 |
| 74 | 0.039343 |
| 75 | 0.034909 |
| 76 | 0.03221  |
| 77 | 0.029115 |
| 78 | 0.025708 |
| 79 | 0.026264 |



(b) 10分間の標準偏差で判定。

- ・ スペクトログラムに表示されている Echo: 2 は、(a)項の閾値で観測した結果で、この例の閾値は、  
閾値 = 平均値 + 設定値 (0.007) となっています。
- ・ 図の上部に記載のグラフは、(a)項の方法と標準偏差の方法を評価したものです。



(c) 閾値の変更方法

観測シェル「mrospec.sh」のヘッダーの少し下に記載している変数「STD\_SGM」の設定値を編集して変更します。

```
##
# Set SD(standard deviation) to judg echo.
#
STD_SGM=5
```

- ・ STD\_SGM = 0 : 10分間の平均値 + 設定値 . . . (a)の処理
- ・ STD\_SGM = 0 以外 : 標準偏差 (初期値 5σ)