

無線 LAN 機能付き SD カードを用いた観測機器のネットワーク接続

防災研究所技術室

山崎友也

1. 目的

ネットワーク接続機能を有していない機器で野外観測を行う場合、データは SD カード等のメディアに収録されるため、定期的に現地に赴き回収する必要がある。観測点が遠方の場合は、地震等のイベントが発生してからデータを確認できるまでのタイムラグが大きくなる。こういったスタンドアローン製品のうち SD カードにデータ収録する機器を対象として、無線 LAN 機能付き SD カードを利用しネットワーク接続するシステムを構築した。以降に設定方法や運用例を紹介する。

2. システムの構成

今回、広帯域電磁場観測装置 ELOG-MT (NT システムデザイン社)の保存ファイルをネットワーク伝送するシステムを構築した。図 1 に構成を示す。ELOG-MT のデータ保存先として無線 LAN 機能付き SD カードを用いた。この SD カードが無線 LAN 親機となり、子機として Raspberry Pi を接続することで保存データにアクセスが可能となる。Raspberry Pi には 4G(LTE)通信モジュール 4GPi(メカトラックス社)を搭載し、インターネット経由でのログインを可能とした。

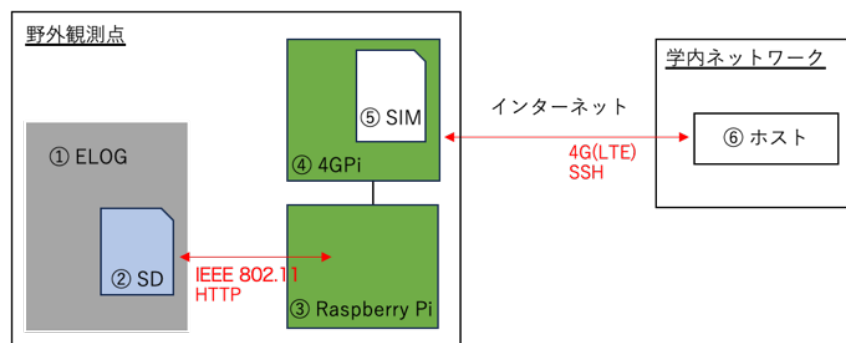


図 1. システム構成

① ELOG-MT

電場、磁場の観測データが SD カードに保存される。データファイルは 1 時間毎に出力され、SD カード内の年月日フォルダに保存される。

② 無線 LAN 機能付き SD カード

FlashAir32GB(東芝社)を使用した。SD カードが無線 LAN アクセスポイントとウェブサーバーの機能を有しており、無線 LAN 接続した子機からブラウザ経由(HTTP)で SD カード内のファイルにアクセスすることができる。ファイル一覧表示などの CGI スクリプトも用意されている。類似製品には、画像データしか扱えない、設定ファイルをカスタマイズできない、といった点で本システムに適用できない製品もあるため注意が必要である。

③ Raspberry Pi

FlashAir およびインターネット接続の制御ホストとして Raspberry Pi を使用した。Raspberry Pi で FlashAir の SSID を指定し無線 LAN 接続する。FlashAir に接続後は curl コマンドなどを用いて SD カード内のファイルを取得することができる。

④ 4GPi

Raspberry Pi 用の 4G(LTE)通信モジュールで、SIM カードを挿入することでインターネット接続が可能となる。Raspberry Pi に搭載した状態を写真 1 に示す。

⑤ SIM カード

通信 SIM としてイプシムのプリペイド SIM を利用した。学内サーバへデータ送信するのみであれば、プライベート IP 用 SIM でも問題ないが、今回は Raspberry Pi の保守性を考慮し、外部から接続できるようグローバル IP 用 SIM を用いた。

⑥ 学内ネットワークホスト

Raspberry Pi がグローバル IP アドレスを持つため、通常のサーバと同じ様に ssh ログインが可能となる。Raspberry Pi に接続後は scp や rsync 等のコマンドを用いて、保存されたデータファイルをダウンロードすることができる。



写真 1. FlashAir と 4GPi を搭載した Raspberry Pi

3. デバイス設定内容

FlashAir と Raspberry Pi でネットワーク確立に必要な設定を以下に示す。

① FlashAir 設定

最初にメーカー配布ツール”FlashAir_tool”を用いて SSID とパスワードを設定する。その後、SD カード内/Volumes/xxxxxxx(デバイス名)/SD_WLAN/CONFIG ファイルを下記の通り編集する。SD カードの IP アドレスは 192.168.1.2 とした。自動切断しないよう APPAUTOTIME を 0 にし、TIMEZONE は ELOG—MT の UTC と合わせるため 0 とした。

■ /Volumes/xxxxxxx/SD_WLAN/CONFIG

```
[Vendor]

CIPATH=/DCIM/100__TSB/FA000001.JPG
VERSION=F15DBW3BW4.00.00
CID=*****
PRODUCT=FlashAir
VENDOR=TOSHIBA
APPSSID=SSID 名
APPNETWORKKEY=*****
APPMODE=4
APPAUTOTIME=300000 ← 0 に変更
DNSMODE=0
LOCK=1

----- 以下追記 -----
DHCP_Enabled=NO
IP_Address= 192.168.1.2
Subnet_Mask=255.255.255.0
Default_Gateway=192.168.1.1
Preferred_DNS_Server=8.8.8.8
Alternate_DNS_Server=192.168.1.1
Proxy_Server_Enabled=NO
TIMEZONE=0
WEBDAV=0
APPCH=1
```

② Raspberry Pi 設定

最初にメカトラック社サイトより 4G Pi 用の Raspberry Pi OS イメージをダウンロードし、Raspberry Pi Imager を用いて OS 書き込みを行う。4G 通信は SIM のユーザ名、パスワードを用いて下記コマンドで設定する。

```
$ nmcli con add type gsm ifname "" con-name "connection name" apn rcmvne.jp user "username" password "password"
```

無線 LAN 接続については、まず raspi-config コマンドを用いて FlashAir への接続を設定す

る。その後固定 IP アドレスを使用する設定を永続化するため、下記の 2 ファイルを編集する。ここでは例としてデバイスの IP アドレスを下記の設定としている。

FlashAir(無線親機) : 192.168.1.2 Raspberry Pi (無線子機) : 192.168.1.3

■ /etc/network/interfaces に追記

```
auto wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface flashair32 inet static
address 192.168.1.3
netmask 255.255.255.0
gateway 192.168.1.2
dns-nameservers 192.168.1.2
```

■ /etc/wpa_supplicant/wpa_supplicant.conf に追記

```
network={
    ssid="FlashAir の SSID 名"
    psk=*****
    key_mgmt=WPA-PSK
    proto=WPA2
    id_str="flashair32"
}
```

なお wpa_supplicant.conf に記載する無線 LAN パスワード(psk)には、下記コマンドより出力される暗号化されたパスワードを記載すると、比較的セキュリティが向上する。

```
$ wpa_passphrase "SSID 名" "パスワード"
```

4G 通信ネットワークについては、4G Pi に SIM を挿入し起動するだけで Raspberry Pi に IP アドレスが割り振られ、インターネット接続が可能となる。しかしながら、ここまでの設定では、インターネット側から来たパケットが FlashAir 側へ出ていき行方不明となるため、外部ホストからの通信に応答しない状態となっている。そこで応答パケットが適切なインターフェースから送信されるよう経路情報を設定する必要がある。ただしこの設定は無線 LAN および 4G 通信の各接続が確立した後に発行しないとエラーとなるため、起動時スクリプトに記述すると処理に失敗することが多い。そこで下記のようなシェルスクリプトを作成し、Raspberry Pi 起動後に crontab で 5 分毎に実行することとした。スクリプトの主な内容は以下の通りである。

- ・インターフェースの状態を確認し、ルーティングが反映済みならスクリプトを終了する。
- ・無線 LAN/ 4 G 通信の状況を確認し、通信確立であればルーティング設定を行う。
- ・無線 LAN/ 4 G 通信が確立していなければリブートする。

- ・各パケットは入ってきたインターフェースから応答する。
- ・時刻設定(NTP)のため FlashAir 側ネットワークの優先度を低くする。

■ ルーティング設定スクリプト例

```
#!/bin/sh
. /etc/profile

rule_wlan0='/sbin/ip rule | grep 192.168.1.3'
rule_wwan0='/sbin/ip rule | grep xxx.xxx.xxx.xxx'

if [ -n "$rule_wlan0" ] && [ -n "$rule_wwan0" ]; then
    exit 0
fi

i_wlan0='/sbin/ip addr show wlan0|grep inet'
i_wwan0='/sbin/ip addr show wwan0|grep inet'

if [ -z "$i_wlan0" ] || [ -z "$i_wwan0" ]; then
    /sbin/reboot
fi

if [ -n "$i_wlan0" ] && [ -n "$i_wwan0" ]; then
    /usr/sbin/route del default gw 192.168.1.2
    /usr/sbin/route add default gw 192.168.1.2 metric 800
    /sbin/ip rule add from 192.168.1.3 table 10
    /sbin/ip rule add from xxx.xxx.xxx.xxx table 20
    /sbin/ip route add default dev wlan0 via 192.168.1.2 table 10
    /sbin/ip route add default dev wwan0 via xxx.xxx.xxx.yyy table 20
    /sbin/ip route add table 10 192.168.1.0/24 dev wlan0 proto kernel scope link src 192.168.1.3
    /sbin/ip route add table 20 192.168.1.0/24 dev wlan0 proto kernel scope link src 192.168.1.3
    /sbin/ip route add table 20 xxx.xxx.xxx.0/24 dev wwan0 proto kernel scope link src xxx.xxx.xxx.xxx
    /sbin/ip route add table 10 xxx.xxx.xxx.0/24 dev wwan0 proto kernel scope link src xxx.xxx.xxx.xxx
fi
```

ここでは各デバイスの IP アドレスを下記とした。

FlashAir(無線親機) : 192.168.1.2

Raspberry Pi(無線子機) : 192.168.1.3

Raspberry Pi(4G SIM): xxx.xxx.xxx.xxx

4G SIM ネットワークアドレス: xxx.xxx.xxx.0/24

4G SIM デフォルトゲートウェイ : xxx.xxx.xxx.yyy

4. 運用例

実際の運用では、1 日に一度 Raspberry Pi でスクリプトを実行し、FlashAir 内の前日分データを取得することとした。FlashAir の CGI コマンドで SD カード内のファイル一覧を取得し時刻情報より前日分のファイルを curl によりダウンロードするスクリプトを製作した。FlashAir 経由で 100MB のテストデータを Raspberry Pi にダウンロードしたところ、転送速度は約 40Mbps ほどとなった。また 4 G の電波強度が十分得られる環境で Raspberry Pi から学内ホストへデータ転送したところ、転送速度は約 18Mbps であった。転送速度は野外観測点の電波状況により異なる可能性があるため、今後の運用状況を確認したい。

今回の運用ではデータ転送は日に一度まとめて行うため、転送を行う時間帯以外はRaspberry Pi が起動している必要はない。そこでデジタルタイムスイッチ(Wal front 社)を用いてバッテリー電源をタイマー制御し、消費電力の節約を企図した(写真 2)。Raspberry Pi を1日あたり3時間だけ起動し稼働テストしたところ、2週間の消費電流量は約0.7A程度であった。また今回のシステム構築に際して調達した物品経費一覧を表1に示す。



写真 2. 観測用ボックス収納状況

購入物	購入価格(円)
Raspberry Pi 4 4GB	11,980
Raspberry Pi用マイクロSDカード	1,450
4GPi	27,500
FlashAir 32GB	27,800
イプシムプリペイドSIM(上り高速 12ヶ月)	38,500
デジタルタイムスイッチ	2,023
収納ボックス+ケーブルコネクタ	2,000
合計	111,253

表 1. データ伝送システム購入物一覧