

1 WSPR の常時運用と省エネ

WSPR(Weak Signal Propagation Reporter) は、その名前に示されているとおり、現時刻における伝搬状態を知るのが主目的です。

WSPR 送信局が存在しかつオンエアするかどうかは不確定ですので、受信側ではなるべく常時受信状態を保ちたいところです。長時間運用とすると、受信機と PC の消費電力はできるだけ節減が望まれますが、仮にノートパソコンを使うとしても、20Wh 以上のエネルギーを消費します。エコ対策として、受信回路の簡素化や PC ディスプレーの切断などで対処している現状です。

他には、PC の省エネを図るのも良いでしょう。幸いにも、カード PC と呼ばれる低価格・低電力・小型の PC が開発されています。これが、音声処理や画像表示を含む WSPR ソフトウェアを動作させることができるかが、Windows からのシステム移行を検討する上での懸念です。

以下に、筆者の体験したカード PC と WSPR のコラボレーションを紹介します。

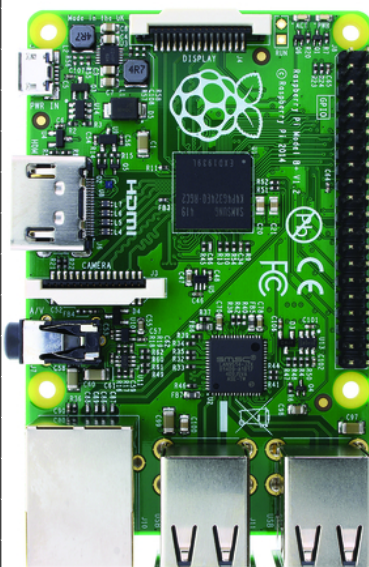
2 Raspberry Pi にできること

2.1 Raspberry Pi model B+の概要

RaspberryPi は、英国の Raspberry Pi Foundation が学校での基本的なコンピュータ科学教育の促進を意図して開発した、シングルボードコンピュータです。2015 年 2 月 18 までに累計 500 万台が販売されたそうです。名称の Pi は開発言語 Python に由来しています。次に示す Raspberry Pi B+ の規格から特徴をつかんでください。

表 1: Raspberry Pi B+のハードウェアと主要性能

SoC	Broadcom BCM2835
CPU	700 MHz / ARM1176JZF-S 2 コア
GPU	250 MHz / Broadcom VideoCore IV
メモリ	512 MB / SDRAM
USB 2.0 ポート	(LAN9512 内蔵ハブ) 4
映像入力	15 ピン MIPI カメラ (CSI)
映像出力	RCA (PAL/NTSC), HDMI 1.3/1.4
音声出力	3.5mm ジャック, HDMI, I2S
ストレージ	microSD カード
ネットワーク (RJ45)	LAN9514(10/100Mbps イーサネット)
低レベル周辺機器	GPIO ヘッダーピン 40 ピン
電源	600mA (3.0W)
電源ソース	5V / microUSB または GPIO
質量	45g
ボードの大きさ	85.60mm x 56.5mm (3.370in x 2.22in)
公式に提供される OS	Debian, Fedora, Arch Linux, RISC OS



最新モデルは、“**Raspberry Pi 2 model B**”です。“Raspberry Pi model B+”より進化していますので、**新規購入には前者**をお奨めします。

Raspberry Pi の製造・販売は RS Components Ltd. と Premier Farnell plc です。(株)秋月電子通商などの販売店から購入もできます。

Raspberry Pi ではストレージメディアにハードディスクではなく microSD カードを使用しますので、騒音と寿命を気にしなくて済みます。

WSPR を使用目的とする上での必要 IO を考えますと、Raspberry Pi B+には (1)4 個の USB ポート、(2)HDMI のビデオおよび音声ポート、(3)17 個 (内 6 個のユーザ定義) の IO 端子が備わっていますから、キーボードおよびマウス操作による入力、市販のデジタルテレビへの出力が可能です。ただし、音声入力端子がありません。これは、小さいボード上での音声信号への雑音対策ができなかったからでしょう。そこで、USB オーディオなどの外付けサウンド機器を用意します。

電源は、ミニ USB 端子から 5V 0.7A を供給します。筆者の場合は、USB ケーブルよりもボードに電源線を直接ハンダ付けの方が簡単でした。GPIO の 4 番ピン (+5V) と 6 番ピン (アース) を使っても良いでしょう。ただし、GPIO ピンには CPU に直結しているピンがありますから、これらに +5V が当たらないように注意します。

Raspberry Pi に関する沢山の情報が Raspberry Pi Foundation の HP(<http://www.raspberrypi.org/>) から得られます。

日本語での解説本が多数発行されています。Raspberry Pi の仕組みや OS とアプリケーションのインストールおよびシステムの設定については、例えば、「Raspberry Pi ユーザーガイド」Eben Upton+Gareth Halfacree 著 (株)クイープ訳 (株)インプレス発行が大変参考になります。

2.2 Debian パッケージ

前述のとおり、Raspberry Pi は ARM アーキテクチャーに対応した Linux OS が使えます。筆者は一般に普及しているディストリビューションである Debian を採用しました。この組み合わせは、例えば Intel i386 アーキテクチャーと Microsoft Windows のセットと相対しています。Raspberry Pi + Debian なので、Raspbian と呼称することがあります。

Debian の総合的な解説は、青木 修著「Debian リファレンス」(<https://www.debian.org/doc/manuals/debian-reference/>) が優れています。

Linux でのアプリケーションのインストールは、ユーザにとって煩わしいファイル間の依存関係やリンクの設定を含めたパッケージという形で、フリーなソフトウェアが提供されています。Debian でも、Debian Packages(<https://www.debian.org/distrib/packages>) から数多くのアプリケーションがダウンロード可能です。内容は、開発レベルに応じて Wheezy(stable)・Jessie(test)・Sid(unstable)・Experimental に分類収録されています。

2.3 Raspbian の構築

では具体的にワンボード PC となる Raspbian を構築する手順を述べます。Raspberry Pi B+には正常に電源が供給され、USB ソケットには市販の 109 キーボードおよび USB マウスが接続され、HDMI 端子からディスプレイ (一般の地上デジタルテレビで良い) に HDMI 信号が出力されていることを前提とします。また、Ether 端子を介してインターネット接続があることが必須です。

microSD カードを用意します。OS と基本ファイルだけで 4GB 近くのサイズを要しますから、サ

イズ 16GB 以上でクラス 10 の microSD カードを入手します。家電量販店で 950 円程度で購入できるカメラ用が使えます。

未使用の microSD カードへの NOOBS による Raspbian のインストール方法は次のとおりです。

- ・手順 1...動作中の PC を用いて、NOOBS イメージファイルをダウンロードする。
http://www.raspberrypi.org/downloads/にアクセスして、
NOOBS の「Download ZIP」をクリックするとダウンロードが開始される。
- ・手順 2...ダウンロードされた NOOBS_v1_4_0.zip(現バージョン)を解凍する。
- ・手順 3...解凍されたフォルダとファイルの全てを microSD カードに書き込む。
- ・手順 4...microSD カードを Raspberry Pi のカードスロットに挿入し、電源をオンする。OS 候補の中から「Raspbian」を選択し、他にはチェックを入れない。
- ・手順 5...NOOBS の指示に従い、リブートしてインストールは完了する。

以上で Debian OS を含む幾種類かのアプリケーションがインストールされるはずですが、

Raspbian を最初に立ち上げると、`raspi-config` というハードウェアの設定画面が現れます。内容はキーボードの設定、ロケールの変更、外部入出力のオンオフなど重要な機能が含まれていますので、マニュアルを参考に編集するのが良いでしょう。この内容は後で、`sudo raspi-config` コマンドによって起動して変更することができます。

ここまでの作業で PC としての基本は完了しました。特に指定しなければ、ホスト名は「pi」、パスワードは「raspberrypi」です。独自のホスト名を追加するには、次を入力します。

- ・手順 6...自分 (例えば、「yuki」) を `sudo useradd yuki` と入力する。
- ・手順 7...パスワードの質問に答える。

試しに、コンソール画面からのコマンド (例えば、「ls Enter」) を入力してください。また、「startx」コマンドで GUI 画面を表示させて、操作具合に馴染んでください。

GUI からの離脱には「Control + Alt + BackSpace」キーを押します。システムの正常終了には「`sudo shutdown -h now`」コマンドを使います。また、「`sudo halt`」も使えます。

3 初歩の Linux コマンド

Linux の操作は、コンソール画面からコマンドの実行と GUI 画面からクリックの 2 種類があります。この内コマンド操作は Windows ユーザは未経験ですし、かつての DOS ユーザでも記憶が薄れているので、以下に基本的なものを抜粋しておきます。

- ・ls : カレントディレクトリの内容を一覧表示
- ・cd : ディレクトリの移動
- ・mv : ファイルを別のディレクトリへ移動またはファイル名の変更
- ・rm : ファイルの削除
- ・rmdir : ディレクトリの削除
- ・mkdir : ディレクトリの作成

Debian がインストールされていると、便利な次のコマンドが使えます。

- ・apt-get : Debian パッケージの取得
- ・sudo : 制限つきコマンドの実行

4 Linux で WSPR

Raspbian が正常に動作しているとして、いよいよ Linux で WSPR を構成しましょう。

前述したとおり、Raspberry Pi に不足しているハードウェアを USB サウンドカードで補います。筆者のテストでは、USB Sound Blaster は使えました。IO データなどの USB オーディオも多分大丈夫でしょう。ただし、オンキヨー (株) 製 SE-U55 デジタルオーディオプロセッサは不動作を確認しました。

WSPR ソフトウェアのインストールは、英語版の http://www.dj0abr.de/english/technik/dds/wsprbanana_x_luxus.htm を参考にしました。

最初に、WSPR を納めるフォルダを準備します。

- ・手順 8...“cd ~” でホームディレクトリへ移動する。
- ・手順 9...“mkdir wsprcan-master” でフォルダを作成する。
- ・手順 10...“cd wsprcan-master” でフォルダ移動し、以後ここで作業する。

Kurt Moraw/DJ0ABR 作の WSPR ソフトウェアである“wsprlinux.exe”の動作に必要な次のライブラリを準備します。

- ・ libsndfile1 ・ libsndfile1-dev ・ libfftw3-3 ・ libfftw3-dev ・ mono-complete
- ・ libtool ・ gettext ・ autoconf

これらを取得するには、次のコマンドを入力します。

- ・手順 11...“sudo apt-get install libsndfile1”のように、これら 8 個のファイルを指定する。
- ・手順 12...“mono -V” コマンドを入力し、バージョン情報を知る。

ここで mono がバージョンが 3.10.0 以降であるならば、次の手順は不要です。実行すると数時間を費やします。

- ・ mono 手順 1...“git clone git://github.com/mono/mono.git”
- ・ mono 手順 2...“cd mono”
- ・ mono 手順 3...“git checkout mono-3.12.0-branch”
- ・ mono 手順 4...“./autogen.sh --prefix=/usr/local”
- ・ mono 手順 5...“make”
- ・ mono 手順 6...“sudo make install”
- ・ mono 手順 7...“reboot”

古いバージョンのライブラリのために WSPR がクラッシュする場合には、次の措置でアップデートします。

- ・ gdi 手順 1...“sudo apt-get install libgtk2.0-dev”
- ・ gdi 手順 2...“sudo apt-get install libjpeg8-dev”
- ・ gdi 手順 3...“git clone git://github.com/mono/libgdiplus”
- ・ gdi 手順 4...“cd libgdiplus”
- ・ gdi 手順 5...“./autogen.sh --prefix=/usr/local”
- ・ gdi 手順 6...“make”
- ・ gdi 手順 7...“sudo make install”

次に wsprlinux.exe 本体とその信号処理を担う Steven J Franke/K9AN 作のデコーダである「k9an-decoder」をダウンロードします。

- ・手順 13...“<http://www.dj0abr.de/files/wspr/runtime/wsprlinux.exe>”をコピーする。
- ・手順 14...“http://www.dj0abr.de/files/wspr/runtime/arm/wsprutil_arm.zip”をコピーし、現フォルダに解凍する。

5 wsprlinux を使う

5.1 WSPR ターミナルの起動

以上で Raspbian で WSPR が動作することになります。さあ、試してみましょう。

まず、“startx” コマンドを打って GUI 画面を立ち上げます。その上部にある LXTerminal を開き、次のコマンドを発します。

- ・手順 15...“mono ./wsprlinux.exe”を入力する。

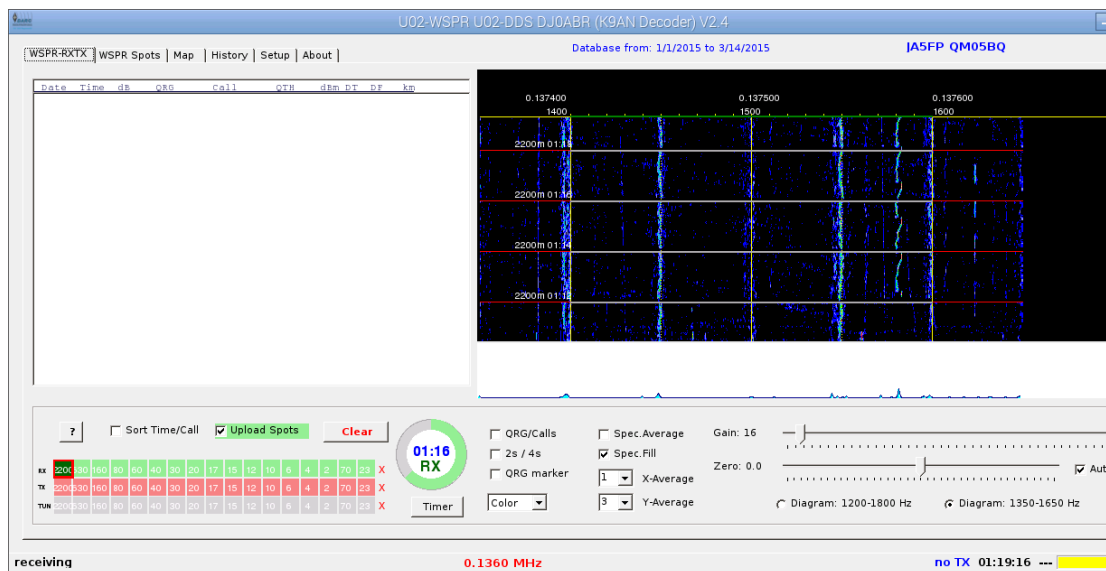


図 1: wsprlinux のトップ画面

wsprlinux の使用方法は、オリジナルの Joe Taylor/K1JT の wspr.exe と基本的には同じ機能ですので、WSPRNet へのデータのアップロードも行われます。ここでは操作方法の詳細な説明は割愛します。

優れている機能では、Raspian がインターネットに接続されていると NTP(Network Time Protocol) を使って現在の時刻を取得し、WSPR は自動的に正偶数分に送信または受信がスタートします。

5.2 wsprlinux による送信——未完項

wsprlinux の送信機能は、(1) 一般の無線機を対象とするオーディオ出力、(2)U02-DDS から FSK 信号を発生するための制御出力の 2 種類があります。また、リグコントロールなどの補助機能がありますが、筆者は未だそれらを試していません。

弱点として、wsprlinux には WSPR-15 モードは現在時点ではカバーしておらず、今後の進化が望まれます。

2015/03/17 記