

AP-RZG-0A

無線 LAN モジュールの使用方法

Rev1.0 2017/10/23

目次

1. 概要	3
1.1 はじめに.....	3
1.2 開発環境.....	3
2. Linux カーネルへの対応方法	4
2.1 ファームウェアのコピー.....	4
2.2 Linux カーネルへのドライバ追加.....	5
2.3 Linux カーネルの作成.....	8
3. 動作確認	9
3.1 Linux の起動.....	9
3.2 動作確認.....	9

表記

●バージョンに関する表記

弊社提供のソース等に関しては、弊社の管理するバージョン番号がファイル名やフォルダ名に付いている場合があります。そのバージョン番号に関しては、本ドキュメントでは、『X』を使用して表現しております。そのため、以下のような表記になりますので、その部分は読み替えてください。

例：

以下の表記がある場合

helloworld-X.X.tar.bz2

Ver1.0 での実際のファイル名は、以下になります。

helloworld-1.0.tar.bz2

●コマンドラインの表記

本ドキュメントには、コマンドラインで入力する操作手順が記載されております。操作は PC 及び XG ボードで行います。それぞれの記述について以下に記載します。

ゲスト OS(Ubuntu)での操作


プロンプトは、『\$』で記載します。

実際のプロンプトには、カレントディレクトリ等が表示されますが、本ドキュメントでは省略します。

AP-RZG-0A ボード上の Linux での操作

プロンプトは、『#』で記載します。


実際のプロンプトには、カレントディレクトリ等が表示されますが、本ドキュメントでは省略します。

本ドキュメント中での入力では、以下のように表現し、入力の最後には、 があります。

例：ゲスト OS(Ubuntu)上で make コマンドを実行する場合の表記

```
$ make 
```

コマンドによっては 1 つのコマンドが複数行で記載されている場合もあります。

その場合には、2 行目以降の入力では ENTER キーを押さずに続けて入力し、 の表記がある行の最後で ENTER キーを入力してそのコマンドを実行してください。

例：2 行続いてコマンド入力がある表記

```
$ cd ~/build/tmp/work/aprzg0a-poky-linux-gnueabi/linux-renesas/3.10+git34547b2a5032ce6dc  
a24b745d608d2f3baac187f-r0/git 
```

1. 概要

1.1 はじめに

本ドキュメントでは、AP-RZG-0A に無線 LAN モジュールを接続して動作を行う方法を説明します。

- ・カーネル対応方法
- ・無線 LAN 設定、動作確認

1.2 開発環境

本ドキュメントでは、Yocto/Poky 開発環境が Ubuntu にインストールされていることが前提となっています。

タッチパネル LCD キットのドライバを Linux カーネルに組み込むには、カーネルのビルド環境が必要となります。カーネルのビルド環境設定は以下のドキュメントを参照してください。

- ・ Install Manual for RZ/G Series
- ・ LK-RZG-A01 Software Manual

2. Linux カーネルへの対応方法

Linux カーネルのデフォルトでは、無線 LAN モジュールを使用する設定になっておりませんので、Linux カーネルを再作成する必要があります。

その再作成する手順を説明します。



本手順では、開発環境がコマンド『`bitbake core-image-base`』または『`bitbake core-image-x11`』にて作成されていることを前提で説明します。

コマンドに関する詳細は、AP-RZG-0A の Linux 開発キット (LK-RZG-A01) のソフトウェアマニュアルでご確認ください。

2.1 ファームウェアのコピー

無線 LAN モジュールを使用するには、以下の 4 つのファイル (ファームウェア) が別途必要となります。

sbinst1, sbinst2, sbdata1, sbdata2

ファイルの入手方法に関しては、WM-RP シリーズの『ハードウェアマニュアル』にコンテンツのダウンロード方法が記載されていますので、そちらでご確認ください。

なお、以下の手順では、~/wm-rp-firm にダウンロードされていることを前提で説明します。

① AP-RZG-A01 ソフトウェアマニュアルにて作成した Linux カーネルのフォルダに移動します。

```
$ cd ~/aprzg0a/build/tmp/work/aprzg0a-poky-linux-gnueabi/linux-renesas/3.10+git34547b2a5032ce6dca24b745d608d2f3baac187f-r0/git ←入力
```

② ディレクトリを作成します。

```
$ mkdir ./drivers/net/wireless/wmrp0xs/Firmware ←入力
```

③ 無線 LAN モジュールのファームウェアをコピーします。

```
$ cp ~/wm-rp-firm/sbinst1 ./drivers/net/wireless/wmrp0xs/Firmware/ ←入力
$ cp ~/wm-rp-firm/sbinst2 ./drivers/net/wireless/wmrp0xs/Firmware/ ←入力
$ cp ~/wm-rp-firm/sbdata1 ./drivers/net/wireless/wmrp0xs/Firmware/ ←入力
$ cp ~/wm-rp-firm/sbdata2 ./drivers/net/wireless/wmrp0xs/Firmware/ ←入力
```

2.2 Linux カーネルへのドライバ追加

Linux カーネルのデフォルトは、無線 LAN モジュールを使用する設定になっておりませんので、以下の手順で Linux カーネルに無線 LAN モジュールのドライバを組み込むように変更します。

- ① ビルド環境の設定をします。

```
$ cd ~/aprzg0a
$ source poky/oe-init-build-env
```

環境設定が終了すると、カレントディレクトリは~/aprzg0a/build に移動します。



カーネルのコンフィギュレーションを初期化する場合は、以下のコマンド実行します。

```
bitbake -c configure linux-renesas --force
```

このコマンドを実行すると以前に行われた menuconfig による設定変更、およびドライバなどのソースの変更は全て初期化されます。

- ② カーネルのカスタマイズをするため、設定画面を開きます。

```
$ bitbake -c menuconfig linux-renesas
```

```
.config - Linux/arm 3.10.31-ltsi Kernel Configuration

Linux/arm 3.10.31-ltsi Kernel Configuration
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module <>

[*] Patch physical to virtual translations at runtime
  General setup --->
  [*] Enable loadable module support --->
  [*] Enable the block layer --->
  System Type --->
  Bus support --->
  Kernel Features --->
  Boot options --->
  CPU Power Management --->
  Floating point emulation --->

v(+)
```

- ③ Wireless LAN メニューに移動します。

[Device Drivers] - [Network device support] - [Wireless LAN]の順に開いていき、『WM-RP-04S/05S』を選択します。

```
.config - Linux/arm 3.10.31-ltsi Kernel Configuration
> Device Drivers > Network device support > Wireless LAN
-----
Wireless LAN
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
-----
< > Intel PRO/Wireless 3945ABG/BG Network Connection (iwl3945)
< > Marvell 8xxx Libertas WLAN driver support
< > Hermes chipset 802.11b support (Orinoco/Prism2/Symbol)
< > Softmac Prism54 support
< > Ralink driver support --->
< > Realtek wireless card support
[ ] TI Wireless LAN support --->
< > ZyDAS ZD1211/ZD1211B USB-wireless support
< > Marvell WiFi-Ex Driver
[*] WM-RP-04S/05S
-----
<Select> < Exit > < Help > < Save > < Load >
```



上記の画面で何も項目が表示されない場合は、一つ上の階層の『Wireless LAN』がチェックされているかご確認ください。

- ④ ESC キーを複数回押し、トップメニューに戻り、設定値を保存します。

```
.config - Linux/arm 3.10.31-ltsi Kernel Configuration
-----
Do you wish to save your new configuration ? <ESC><ESC>
to continue.
-----
< Yes > < No >
```

- ⑤ デバイスツリーファイルを編集します。
以下に編集ファイル名、ファイル場所、編集内容を記載します。

●ファイル名

r8a7745-aprzg0a.dts

●ファイル場所

```

/
|-- home
  |-- guest
    |-- aprzg0a
      |-- build
        |-- tmp
          |-- work
            |-- aprzg0a-poky-linux-gnueabi
              |-- linux-renesas
                |-- 3.10+git34547b2a5032ce6dca24b745d608d2f3baac187f-r0
                  |-- git
                    |-- arch
                      |-- arm
                        |-- boot
                          |-- dts
  
```

●編集内容

ファイル『r8a7745-aprzg0a.dts』のmsiof0 設定部分に無線 LAN モジュールが記述されています。
以下の箇所を『"okay"』とします。

```

&msiof0 {
    pinctrl-0 = <&msiof0_pins>;
    pinctrl-names = "default";

    status = "okay";

    wmrp0x_spi: wmrp0x_spi@0 {
        compatible = "apj, wmrp0x_spi";
        reg = <0>;
        spi-max-frequency = <25000000>;
        //spi-cpha;
        //spi-cpol;

        interrupt-parent = <&irqc0>;
        interrupts = <5 IRQ_TYPE_LEVEL_HIGH>;
        gpio_power = <&gpio2 26 GPIO_ACTIVE_HIGH>;
        gpio_reset = <&gpio4 7 GPIO_ACTIVE_HIGH>;

        status = "okay";
    };
};
  
```

2.3 Linux カーネルの作成

Linux カーネルのみ再ビルドする手順を説明します。

① カーネルのビルドをします。

```
$ bitbake -c compile linux-renesas --force ←
Loading cache: 100% |#####|
##| ETA: 00:00:00
Loaded 1846 entries from dependency cache.

:
  途中省略
:

NOTE: Executing RunQueue Tasks
NOTE: Tasks Summary: Attempted 226 tasks of which 225 didn't need to be rerun and all succeeded.

Summary: There was 1 WARNING message shown.
```

② カーネルのビルドが成功したら、作成されたカーネルをデプロイ(配布)します。

```
$ bitbake -c deploy linux-renesas ←
Loading cache: 100% |#####|
##| ETA: 00:00:00
Loaded 1846 entries from dependency cache.

:
  途中省略
:

NOTE: Tasks Summary: Attempted 233 tasks of which 227 didn't need to be rerun and all succeeded.

Summary: There was 1 WARNING message shown.
```


3. 動作確認

本章では、無線 LAN モジュールを動作させる手順を説明します。

3.1 Linux の起動

無線 LAN モジュールドライバを追加した Linux の起動方法を説明します。なお、Linux カーネルを microSD カードに書き込む方法は、AP-RZG-0A の Linux 開発キット (LK-RZG-A01) のソフトウェアマニュアルでご確認ください。

- ① AP-RZG-0A ボードの CN4 に WM-RP-04S もしくは WM-RP-05S を接続します。
接続方法に関しては、無線 LAN モジュールの『ハードウェアマニュアル』でご確認ください。
- ② 『2. Linux カーネルへの対応方法』の Linux カーネルが書き込まれた microSD カードをスロットに挿入した後に、電源を入れます。

3.2 動作確認

無線 LAN モジュールを使用する方法の説明を行います。例としてインフラストラクチャを使用した手順で説明します。なお、アクセスポイントは、以下の表の設定とします。

設定項目	設定値
SSID	WLAN-SSID
パスワード	wlan-pass
セキュリティモード	WPA2

Table3.2-1 アクセスポイントの設定例

- ① root でログインします。

```
aprzg0a login: root
```

- ② セキュリティモードを WPA2 に設定します。

```
# iwpriv wlan0 setsecuritymode 2
```

他のセキュリティモードに設定する場合は、以下の表を参考に値を設定してください。

セキュリティモード	設定値
Open (No Security)	0
WPA1	1
WPA2	2
WEP	3

Table 3.2-1 セキュリティモードの設定値

- ③ SSID 等を設定します。

```
# iwconfig wlan0 essid WLAN-SSID key s:wlan-pass
```

- ④ IP アドレス等を設定します。

```
# ifconfig wlan0 192.168.128.203 netmask 255.255.255.0 ◀入力
```

上記の設定を行うことでアクセスポイントに接続できます。

後は、wlan0 に対して通信をすることで動作できます。例としては、ping を用いた方法を説明します。

- ⑤ アクセスポイントに接続されている機器(IP アドレスは、192.168.128.210 と仮定)に対して ping 送信で通信確認をします。

```
# ping -I wlan0 192.168.128.210 ◀入力
```

ご注意

- ・本文書の著作権は、株式会社アルファプロジェクトが保有します。
- ・本文書の内容を無断で転載することは一切禁止します。
- ・本文書に記載されているサンプルプログラムの著作権は、株式会社アルファプロジェクトが保有します。
- ・本文書に記載されている内容およびサンプルプログラムについての技術サポートは一切受け付けておりません。
- ・本文書の内容およびサンプルプログラムに基づき、アプリケーションを運用した結果、万一損害が発生しても、弊社では一切責任を負いませんのでご了承下さい。
- ・本文書の内容については、万全を期して作成いたしましたが、万一ご不審な点、誤りなどお気付きの点がありましたら弊社までご連絡下さい。
- ・本文書の内容は、将来予告なしに変更されることがあります。

商標について

- ・R8A7745 は、ルネサスエレクトロニクス株式会社の登録商標、商標または商品名称です。
- ・Linux は、Linus Torvalds の米国およびその他の国における登録商標または商標です。
- ・Yocto Project は、Linux Foundation の登録商標です。
- ・U-Boot は、DENX Software Engineering の登録商標、商標または商品名称です。
- ・Windows®の正式名称は、Microsoft®Windows®Operating System です。
- ・Microsoft、Windows は、米国 Microsoft Corporation.の米国およびその他の国における商標または登録商標です。
- ・Windows®10、Windows®7 は、米国 Microsoft Corporation.の商品名称です。
- ・VirtualBox は、OracleCorporation の商品名称です。
本文書では下記のように省略して記載している場合がございます。ご了承下さい。
Windows®10 は、Windows 10 もしくは Win10
Windows®7 は、Windows 7 もしくは Win7
- ・その他の会社名、製品名は、各社の登録商標または商標です。