

## MS104-SH4

## apLinux USB マスストレージスレーブドライバの使用

3版 2005年1月20日

## 1. 概要

## 1.1 概要

本アプリケーションノートは MS104-SH4 ボードと MS104-USB H/S ボードを使用し、USB マスストレージスレーブデバイスとして動作させます。

FAT 形式でフォーマットされた CompactFlash を MS104-SH4 ボードに挿入し、Windows マシンから CompactFlash の中身を読み取ります。

今回使用している LinuxUSB スレーブデバイスドライバは Lineo（現在 metroworks）社が提案した Lineo USB デバイスフレームワークを用いています。このフレームワークは LinuxUSB ホストドライバと同様に三階層で構成されています。最上位層は USB スレーブデバイスを実装したファンクションドライバ、中間層は最上位層のファンクションドライバと最下位層のバスインターフェースドライバの仲介を行う USB デバイスコアドライバ、最下位層は物理的な USB 通信を司るバスインターフェースドライバです。

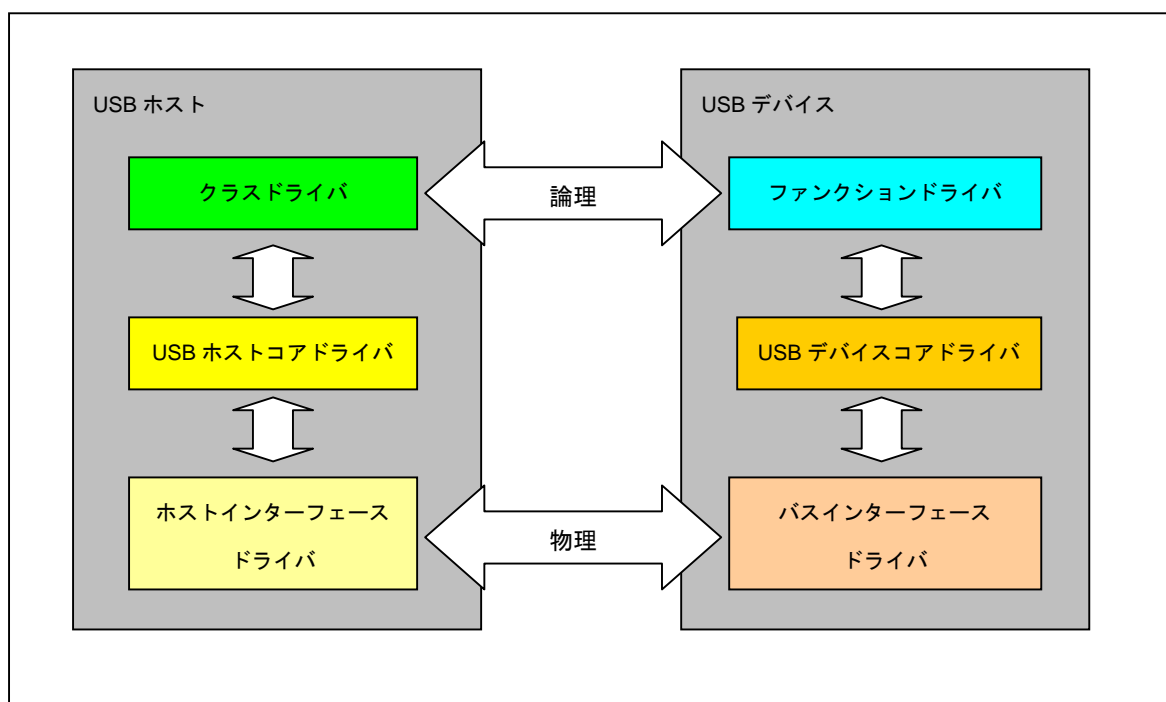


Fig1.1-1 USB デバイスアーキテクチャ

チャンネル	アドレス	割込み番号
CH1	0x1000	9
	0x1001	
CH2	0x1002	7
	0x1003	

Table 1.1-1 USB スレーブデバイス

## 1.2 動作環境

MS104-SH4 Linux 開発環境として『Linux-KIT-A02』がインストールされている必要があります。

『Linux-KIT-A01』をお使いの方はアプリケーションノート『AN321 apLinux バージョン 2.x への移行 AN321.pdf』を参考に、Linux 開発環境のアップデートを行ってください。

Table1.2-1 使用機器

使用機器等	環境
USB ホスト	PC/AT 互換機
OS	Windows2000 および LinuxPC (RedHatLinux9 または Fedora 等)
USB ホストコントローラ	UHCI
CompactFlash	ハギワラシスコム 64MB FAT 形式によるフォーマット

## 1.3 接続

以下に MS104-SH4 と PC の接続方法について示します。

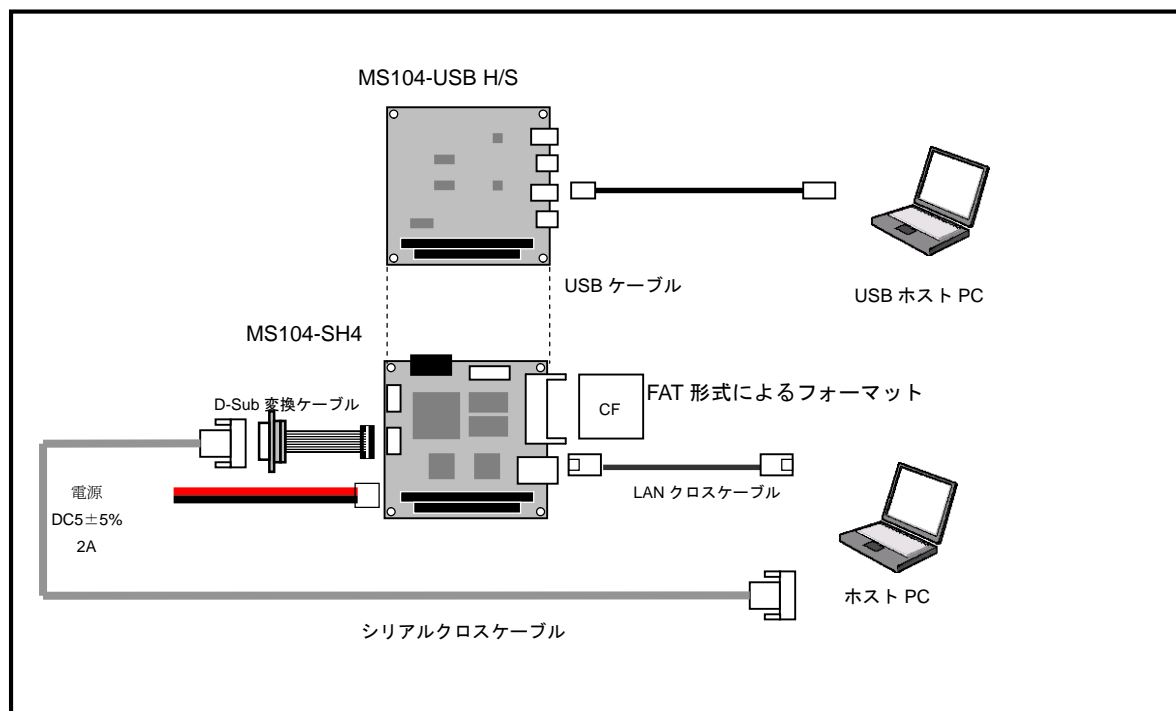


Fig1.3-1 PC との接続

### 1.4 MS104-USB H/S の設定

MS104-USB H/S のチャンネル 2 を USB スレーブデバイスとして使用するための設定を行います。

MS104-USB H/S は 2 チャンネルありますが、USB スレーブデバイスを同時に 2 チャンネル使用することはできません。

- ① I/O アドレスを H'1000 に設定します。SW1 の 4 番のみ OFF にして下さい。

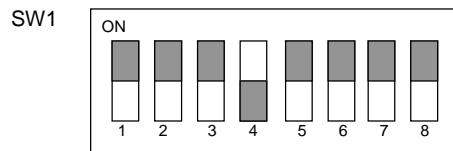


Fig 1.4-1 SW1 の設定

- ② CH1 をホストの設定にします。JP5 の 1 番と 2 番にジャンパピンを接続します。

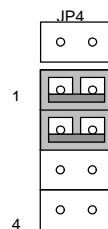


Fig 1.4-2 JP5 の設定

- ③ CH1 の割り込みを PC/104 バス IRQ6 (Linux 上での割り込み番号 9) に設定します。JP3 の CH1 を IRQ6 にジャンパピンで接続します。

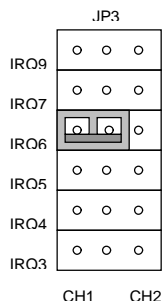


Fig 1.4-3 JP3 の設定 (その 1)

- ④ CH2 をスレーブの設定にします。JP7 の 1 番と 2 番にジャンパピンを接続します。

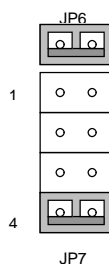


Fig 1.4-4 JP6、JP7 の設定

- ⑤ CH2の割り込みをPC/104バスIRQ7（Linux上での割り込み番号10）に設定します。JP3のCH2をIRQ7にジャンパピンで接続します。

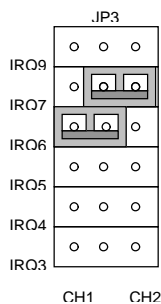


Fig 1.4-5 JP3の設定（その2）

### 1.5 ベンダ・プロダクトIDについて

MS104-USB H/S ボードの試験用として以下のプロダクトIDを開放しております。  
 ベンダIDにつきましては、弊社がUSB Implementers Forumより取得しておりますIDをお使いください。

Table 1.5-1 USBのベンダIDとプロダクトIDの設定

ベンダID	0x0D91
プロダクトID	0xE0D0
	0xE0D1

MS104-USB H/S ボードを製品として組み込む際、必ずUSB Implementers ForumよりベンダIDを取得してください。  
 ベンダID、プロダクトIDが同一の製品が存在する場合、機器が正常に動作しない恐れがあります。

## 2. USB マスストレージスレーブデバイスの作成

USB スレーブデバイスは Linux カーネルに静的に組み込むことができず、必ずモジュールとして作成しなければなりません。

以下では、Linux カーネルのコンフィグレーション項目、カーネル・スレーブデバイスドライバの作成、ルートファイルシステムの作成について説明します。

### 2.1 Linux コンフィグレーション項目

Table 2.1-1 MS104-USB H/S 用 USB スレーブデバイスコンフィグレーション

設定項目	設定	説明
USB support		USB デバイスに関するカテゴリ
USB Device Support	-	USB スレーブデバイスに関するカテゴリ
Support for USB Devices (as device, not host)	M	USB スレーブデバイスのサポート
VendorID	16 進数	ベンダ ID ※1
ProductID	16 進数	プロダクト ID ※1
Product Name	文字列	プロダクト名
Manufacturer	文字列	製造者名
Self Powered	*	自己電源で動作するか。
USB Device Monitor	M	USB スレーブデバイスモニタのサポート
USB Dev FS	*	USB スレーブデバイス /proc ファイルシステムのサポート
Mass Storage Function	-	マスストレージファンクションに関するカテゴリ
Mass Storage Function Driver	M	マスストレージファンクションのサポート
Override VendorID	16 進数	優先ベンダ ID ※1
Override ProductID	16 進数	優先プロダクト ID ※1
OUT Endpoint	1~15	OUT エンドポイント番号
OUT PacketSize	16 or 32 or 64	OUT エンドポイントのケットサイズ
IN Endpoint	1~15	IN エンドポイント番号
IN PacketSize	16 or 32 or 64	IN エンドポイントのケットサイズ
INT Endpoint	1~15	割込みエンドポイント番号
INT PacketSize	8 or 16	割込みエンドポイントのケットサイズ
Default Mass Storage device name	文字列 (/dev/hda1)	マスストレージデバイスのデバイスファイル名
USB Device Bus Interface Support	-	USB デバイスバスインタフェースに関するカテゴリ
Scanlogic SL11 Device support	M	サイプレス社製 HS811SL (旧スキャンロジック SL11) デバイスのサポート
USB slave support for MS104-USBHS CH?	* ?: 1 or 2	MS104-USB H/S チャンネル?をスレーブデバイスとしてサポート

※1 ベンダ ID は弊社が USB Implementers Forum より取得しております「0x0D91」を試験用としてお使いください。  
また、プロダクト ID は MS104-USB H/S 用に「0xE0D0」と「0xE0D1」を試験用としてご使用ください。

※ 製品として MS104-USB H/S を組み込む際は、必ず USB Implementers Forum よりベンダ ID の取得を行ってください。

## 2.2 カーネル・デバイスドライバの作成

### Linux カーネル・デバイスドライバの作成

MS104-USB H/S USB スレーブ対応 Linux カーネルのコンパイル方法について説明します。

- ① LinuxPC に root 権限でログインします。

```
[alpha@ap_linux ms104sh4]$ su -
Password:*****
```

- ② LinuxPC 上で MS104-SH4 用 Linux カーネルソースディレクトリに移動します。

```
[root@ap_linux root]# cd /opt/alpha_project/linux-kit-a02/linux/kernel/celinux-040503-alp/
[root@ap_linux celinux-040503-alp]#
```

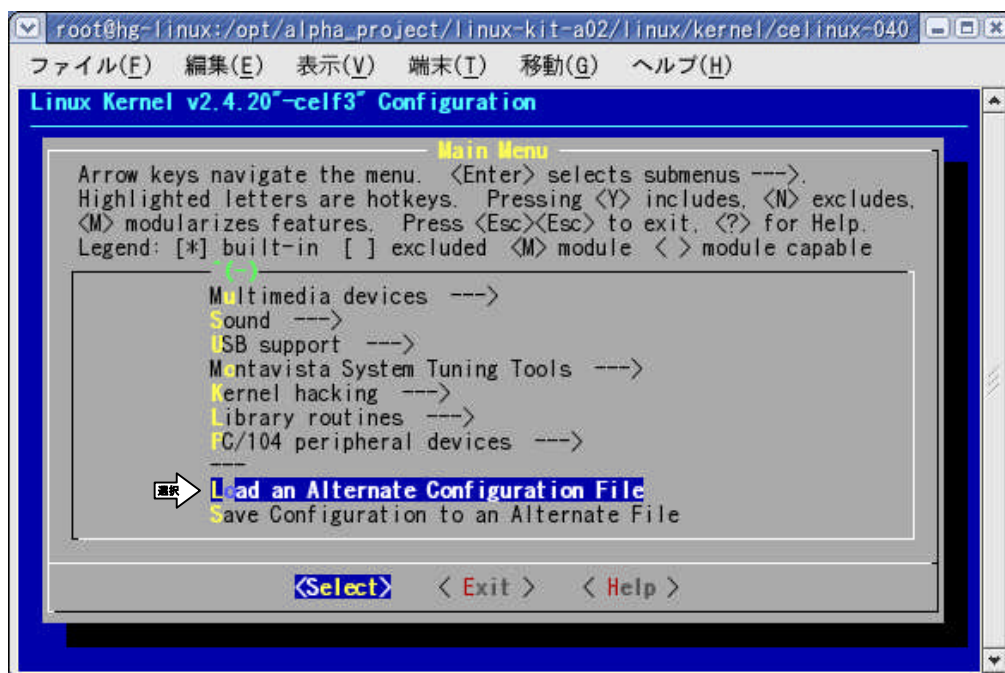
- ③ Linux カーネルのコンフィグレータを起動し、コンフィグレーションを行います。

LinuxPC 上で『**make menuconfig**』を実行してください。

```
[root@ap_linux celinux-040503-alp]# make menuconfig
```

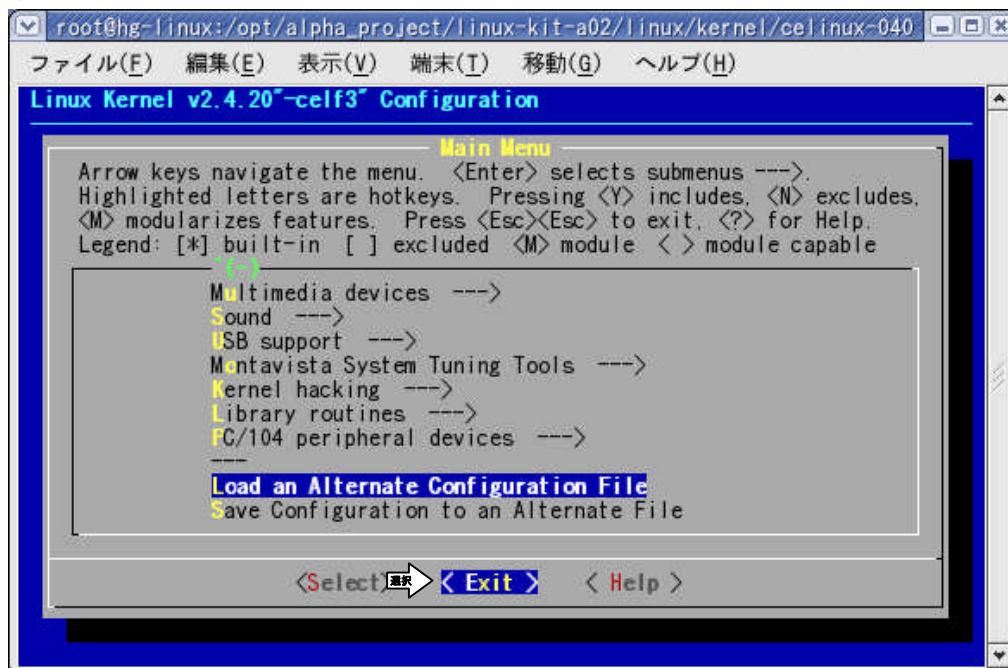
- ④ MS104-USB H/S USB スレーブ対応 Linux カーネルのコンフィグレーションを行います。

『**Load an Alternate Configuration File**』を選択し、『**arch/sh/def-configs/ms104-sh4/ms104usbd.config**』と入力した後、<OK> を選択します。

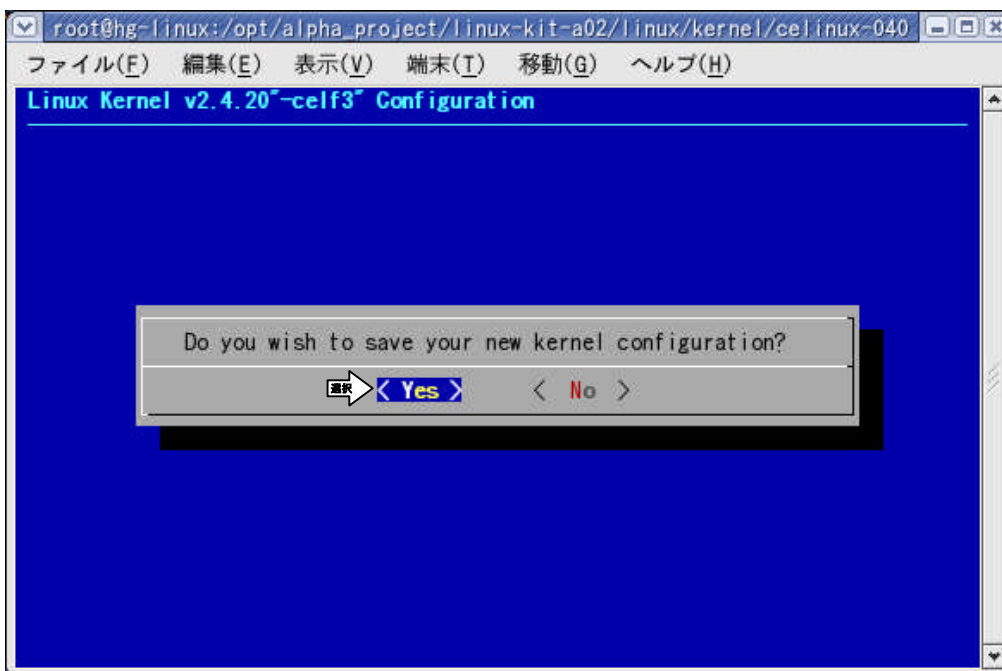




- ⑤ MS104-USB H/S USB スレーブ対応 Linux カーネルのコンフィグレーションが完了したら、< Exit > を選択します。



『Do you wish to save your new kernel configuration?』と表示されるので < Yes > を選択します。



⑥ Linux カーネルの依存関係記述ファイルを更新します。

LinuxPC 上で『make dep』を実行してください。

```
[root@ap_linux celinux-040503-alp]# make dep
```

⑦ 前回コンパイルを実行したときの Linux カーネル中間ファイル等を削除する場合は『make clean』を実行します。

```
[root@ap_linux celinux-040503-alp]# make clean
```

⑧ Linux カーネルをコンパイルします。

LinuxPC 上で『make zImage』を実行してください。

```
[root@ap_linux celinux-040503-alp]# make zImage
```

※ コンパイルが正常終了すれば『arch/sh/boot』ディレクトリ下に Linux 圧縮カーネルイメージ『zImage』が作成されます。

⑨ モジュールのコンパイルを行います。

LinuxPC 上の『make modules』を実行してください。

```
[root@ap_linux celinux-040503-alp]# make modules
```

⑩ モジュールのインストールを行います。

LinuxPC 上の『make modules\_install』を実行してください。

```
[root@ap_linux celinux-040503-alp]# make modules_install
```



- ⑩ インストール先ディレクトリを確認してください。  
下記のようなディレクトリ構成になります。

Fig 2.2-1 モジュールインストールディレクトリ

```
/opt/alpha_project/linux-kit-a02/linux/filesystems/module(インストール先)ディレクトリ
|-- lib
  |-- modules
    |-- 2.4.20-celf3
      |-- build -> /opt/alpha_project/linux-kit-a02/linux/kernel/celinux-040503-alp
      |-- kernel
        |-- drivers
          |-- cdrom
            |-- cdrom.o
          |-- input
            |-- input.o
            |-- keybdev.o
            |-- mousedev.o
          |-- scsi
            |-- osst.o
            |-- scsi_mod.o
            |-- sd_mod.o
            |-- sg.o
            |-- sr_mod.o
          |-- usb
            |-- device-ms104sh4
              |-- bi
                |-- sl11_bi.o
            |-- storage_fd
              |-- storage_fd.o
            |-- usbdcore.o
            |-- usbdmonitor.o
            |-- usbdserial.o
            |-- hc_sl811.o
            |-- serial
              |-- ftdi_sio.o
              |-- usbserial.o
            |-- storage
              |-- usb-storage.o
            |-- usbcore.o
            |-- usbkbd.o
            |-- usbmouse.o
        |-- pcmcia
```

## 2.3 ルートファイルシステムの作成

### ルートファイルシステムの作成

『2.2 カーネル・デバイスドライバの作成』で作成した Linux モジュールを RAM ディスクルートファイルシステムにインストールします。

- ① LinuxPC に root 権限でログインします。

```
[alpha@ap_linux ms104sh4]$ su -
Password:*****
```

パスワードを入力してください

- ② 『2.2 カーネル・デバイスドライバの作成』で作成した Linux カーネルと「Linux-KIT-A02」の添付 CD-ROM、もしくは、「MS104-SH4 ファイルセンター」から MS104-SH4 用 RAM ディスクイメージ (ルートファイルシステム) を取得し、TFTP サーバディレクトリ『/tftpboot』にコピーします。

下記のコマンドは「Linux-KIT-A02」の添付 CD-ROM から Linux カーネルと RAM ディスクイメージ (ルートファイルシステム) をコピーする場合です。

```
[alpha@ap_linux root]# cp /opt/alpha_project/linux-kit-a02/linux/kernel/celinux-040503-alp/arch/sh/boot/zImage
/tftpboot/vmlinuz-ms104sh4-x.x-usbd
[alpha@ap_linux root]# mount /dev/cdrom /mnt/cdrom
[alpha@ap_linux root]# cp /mnt/cdrom/binary/ms104-usb/ramdisk-ms104sh4-x.x.gz /tftpboot/ramdisk-ms104s
h4-x.x-usbd.gz
```

- ※ TFTP サーバの起動については MS104-SH4 Linux 導入マニュアル「MS104-SH4 Linuxstart.pdf」をご覧ください。
- ※ 「MS104-SH4 ファイルセンター」のアドレスは巻末に掲載されております。また、アクセスにはユーザ登録が必要となります。
- ※ 「x.x」はバージョン番号を示します。バージョン 2.0 の場合は「2.0」になります。

- ③ LinuxPC 上で『/tftpboot』に移動し、RAM ディスクのマウントポイントを作成します。

```
[root@ap_linux root]# cd /tftpboot
[root@ap_linux tftpboot]# mkdir -p mnt
```

- ④ LinuxPC 上で RAM ディスクイメージを解凍します。

```
[root@ap_linux tftpboot]# gzip -d ramdisk-ms104sh4-x.x-usbd.gz
```

- ※ ファイル名の x.x はバージョン番号を示します。

- ⑤ LinuxPC 上でルートファイルシステム RAM ディスクイメージをマウントします。

```
[root@ap_linux tftpboot]# mount -t ext2 -o loop ramdisk-ms104sh4-x.x-usbd mnt
```

- ⑥ USB スレーブデバイスドライバモジュールを RAM ディスクイメージにコピーします。

LinuxPC 上で『cp -r /opt/alpha\_project/linux-kit-a02/linux/filesystems/module/lib/modules mnt/lib/.』を実行してください。

```
[root@ap_linux tftpboot]# cp -r /opt/alpha_project/linux-kit-a02/linux/filesystems/module/lib/modules/ mnt/lib/.
```

- ⑦ USB スレーブデバイスシステムファイル『`/etc/sysconfig/usbd`』を修正します。

LinuxPC 上で『`vi mnt/etc/sysconfig/usbd`』を実行します。

```
[root@ap_linux tftpboot]# vi mnt/etc/sysconfig/usbd
```

- `/etc/sysconfig/usbd` ファイル

```
USBBD=on ← “on” に変更する。
```

```
STORAGE=yes ← “yes” に変更する。
```

- ⑧ LinuxPC 上で RAM ディスクイメージをアンマウントします。

```
[root@ap_linux tftpboot]# umount ramdisk-ms104sh4-x.x-usbd
```

- ⑨ LinuxPC 上で RAM ディスクイメージを圧縮します。

```
[root@ap_linux tftpboot]# gzip ramdisk-ms104sh4-x.x-usbd
```

## 2.3 WindowPC と USB スレーブデバイスの接続

① 『Fig1.3-1 USB ハブの接続』を参考に MS104-SH4 と MS104-USB H/S を接続します。

② MS104-SH4 の電源を入れ、RedBoot を起動します。

Linux カーネルと RAM ディスクをダウンロードし、apLinux を起動させます。

TFTP サーバの IP アドレスは「192.168.1.201」と仮定します。

```
RedBoot> load -r -b 0x8c210000 -m tftp -h 192.168.1.201 vmlinuz-ms104sh4-x.x-usbd
Raw file loaded 0x8c210000-0x8c30bb05, assumed entry at 0x8c210000
RedBoot> load -r -b 0x8c360000 -m tftp -h 192.168.1.201 ramdisk-ms104sh4-x.x-usbd.gz
Raw file loaded 0x8c360000-0x8c6f4754, assumed entry at 0x8c360000
RedBoot> exec -m 1 -f 0 -r 0x0100 -l 1 -i 0x360000 -j 0xc00000
```

※ ファイル名の x.x はバージョン番号を示します。

③ apLinux に root 権限でログインします。

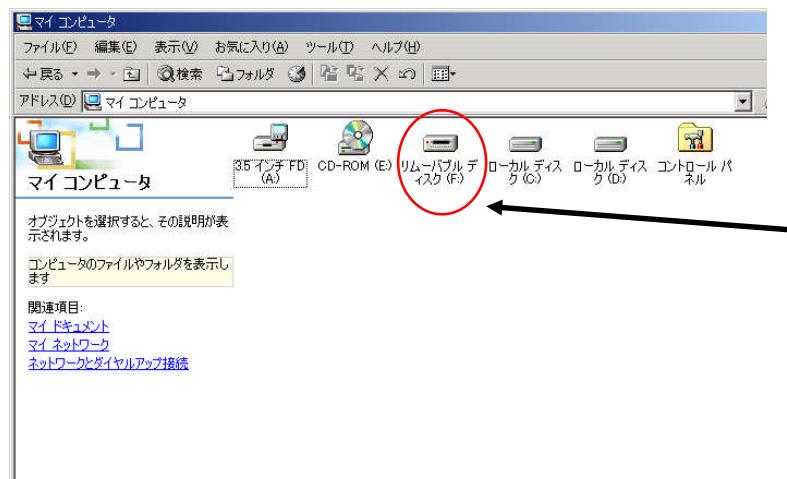
```
MS104SH4 login: root
```

④ WindowsPC と MS104-USB H/S を USB ケーブルで接続します。

・ MS104-SH4

```
[MS104SH4@root]# hda: hda1
```

・ WindowsPC



MS104-SH4 に挿入されている CompactFlash が外部ディスクとして認識される

⑤ WindowsPC より CompactFlash の中身を確認します。

この USB マスストレージスレーブデバイスは読み込み専用です。書き込みはできません。

### 3. USB マスストレージスレーブデバイスの使用上の注意

- ・ USB スレーブデバイスはその特性上高い割込み優先度を設定しています。コンソールとして使用しているシリアルデバイスよりも高い優先度が設定されているため、コンソールの入出力ができなくなってしまうことがあります。
- ・ 動作の検証は Windows2000、EHCI ホストコントロールドライバのみで行っております。
- ・ OS や USB ホストコントローラによっては動作しない場合がございます。

### 4 保証とサポート

弊社では最低限の動作確認をしておりますが、Linux および付属ソフトウェアの性能や動作を保証するものではありません。また、これらのソフトウェアについての個別のお問い合わせ及び技術的な質問は一切受け付けておりませんのでご了承ください。

個別サポートをご希望されるお客様には、別途有償サポートプログラムをご用意しておりますので、弊社営業までご連絡ください。

Linux など、付属する GPL ソフトウェアのソースコードは弊社ホームページより全てダウンロードすることができます。また、これらのソフトウェアは不定期にバージョンアップをおこない、ホームページ上で公開する予定です。

## ご注意

- ・本文書の著作権は（株）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。
- ・本文書の内容は、将来予告なしに変更されることがあります。
- ・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質問等のサポートは一切受け付けておりませんのでご了承ください。
- ・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。万が一不審な点、誤りなどお気づきの点がありましたら弊社までご連絡下さい。
- ・本文書の内容を実行した結果、万が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、万が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。

- ・ SuperH は、（株）ルネサステクノロジの登録商標、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。
- ・ SL11、SL811 は サイプレス社の登録商標、商標または商品名称です。
- ・ コンパクトフラッシュはサンディスク社の商標です。
- ・ Windows2000 はマイクロソフト社の商標です。
- ・ 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。



株式会社アルファプロジェクト  
〒431-3114  
静岡県浜松市東区積志町 834  
<http://www.apnet.co.jp>  
E-MAIL : [query@apnet.co.jp](mailto:query@apnet.co.jp)