

OpenIAPr カーネル構築

1 . 概要

クロス・コンパイルを前提にします。

大まかな流れは、

- ・クロス開発環境を構築
- ・OpenIAPr のコンパイル・リンクの方法
- ・ターゲット・マシンへのカーネル・インストール環境の構築
- ・ターゲットへのカーネルのインストール方法

2 . クロス開発環境を構築

2 . 1 . LINUX (Red Hat 7 . 3) を開発機にインストール

・特に Red Hat 7 . 3 でなくても良いと思うが、試していないので誰か挑戦してみてください。

・`/etc/redhat - release` の内容を 7 . 3 から 7 . 0 に編集

(注 . 7.3 はデフォルト・インストールをするとファイアウォールが有効で、`tftp` や `dhcp` が通らないので、ファイアウォールなしでインストールしてください。もし、インストール済みなら何とか `tftp` や `dhcp` が有効になるようにファイアウォールの設定を変更してください。ちなみに私は変更失敗したのでファイアウォール無しで再インストールしました。なお RedHat 9 はデフォルト・インストールはファイアウォール無しのようです。)

2 . 2 . クロス環境をインストール

・Hard Hat Linux 2 . 0 の CD をマウント

```
# cd /mnt/cdrom/bin
```

```
# ./hhl-host-install -hhlimage /mnt/cdrom
```

target には `ibm-walnut` を入力

・サーチパスを、`.bashrc` に追加

```
export PATH = $ PATH : /opt/hardhat/devkit/ppc/405/bin : /opt/hardhat/host/bin
```

```
# cd /root
```

```
# source .bashrc
```

・Hard Hat Linux 2 . 1 の CD をマウント

・クロス環境をアップデート

```
# cd /mnt/cdrom/bin
```

```
# ./hhl-host-install -upgrade /mnt/cdrom
```

target には、`ibm-walnut-ppc_405` を入力

しばらくして、CD を入れるように促されたら、MontaVista Linux 2.1 ppc_405 の CD をドライブに入れる。

3 . OpenIAPr のコンパイル・リンクの方法

3 . 1 . ソースを用意

- ・ openiapr と書かれた CD をドライブに入れる。
- ・ カーネルソースを CD からコピーする。
カーネルを構築する適切なディレクトリに移動してください。
cp /mnt/cdrom/openiapr20040614-2.tar.gz .
- ・ ソースファイルを解凍する。
tar xvzf openiapr20040614-2.tar.gz

3 . 2 . コンフィグレーション

- ・ カーネルソースのディレクトリへ移動
cd openiapr
- ・ コンフィグレーションを変更する場合はコンフィグレーションを行う。
必要なければここはスキップしてください。
make xconfig
CONFIG_HDA1 が元コンフィグレーションファイルです。
コンフィグレーションが終わったら “ Save and Exit ” でコンフィグレーションウィンドウを抜けます。

3 . 3 . make dep

- ・ include ファイル等のソースが依拠する情報を予め生成しておきます。
make dep

3 . 4 . コンパイル

- ・ コンパイル・リンクを行う
make vmlinux

3 . 5 . バイナリへ変換

- ・ バイナリファイルへ変換
ppc_405-objcopy -o binary vmlinux vmlinux.bin
- ・ ターゲットへ tftp で渡す為に/tftpboot へコピーする
cp vmlinux.bin /tftpboot/

4 . カーネル・インストール環境の構築

- ・ターゲットのローダーは DHCP で IP アドレスを取得しますので、DHCPD をセットアップします。
- ・カーネルのローダーへのロードは tftp を使いますので tftpd をセットアップします。

4 . 1 . DHCPD のセットアップ

ターゲットのローダーが相手できるのはこの DHCPD だけらしいので必ず下記手順に従ってください。

- ・ HardHatLinux2 . O CD をマウントし、DHCP のプロダクトをコピーする。

```
# mount -t iso9660 /dev/cdrom /mnt
# cp /mnt/cdrom/other/dhcp - 2.Op15.tar.gz /root/products
# umount /mnt
# cd /root/products
```

- ・アーカイブを解凍しインストールする。

```
# tar xzvf dhcp - 2 . Op15.tar.gZ
# cd dhcp - 2.Op15
# ./configure
# make
# make install
```

- ・以下のファイルがあるか確認する。

```
/var/state/dhcp/dhcpd.leases
```

もし上記ファイルが無い場合、touch コマンドにて新規作成する。

```
# touch /var/state/dhcp/dhcpd.leases
```

- ・DHCP デーモンのコンフィギュレーションファイルを編集する。

```
/etc/dhcpd.conf
```

もし、上記ファイルが無い場合、エディタにて新規作成する。

```
+-----+
```

```
# Sample /etc/dhcpd.conf
# (add your comments here)
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.0.255;
option routers 192.168.0.1;
option domain-name-servers 192.168.0.1;
```

```

option domain-name "madlabo.who";

allow bootp;
subnet 192.168.0.0 netmask 255.255.255.0 {
    range 192.168.0.200 192.168.0.209;
    range 192.168.0.210 192.168.0.219;
}
host Open-IAPr {
    hardware ethernet 00:04:AC:2D:42:03;
    fixed-address 192.168.0.200;
    filename "zImage.embedded";
    option root-path "/opt/hardhat/devkit/ppc/405/target";
#    option root-path "/tftpboot";
}
host Open-IAPr {
    hardware ethernet C1:08:B8:F0:86:4C;
    fixed-address 192.168.0.201;
    filename "zImage.embedded";
    option root-path "/opt/hardhat/devkit/ppc/405/target";
#    option root-path "/tftpboot";
}

```

+-----+

- * 下線の部分を環境に合わせて編集する。
- * Mac アドレスはターゲット起動直後に見ることができます。
詳細は次項を参照。
- ・ dhcpd をリスタートする。
dhcpd restart
chkconfig dhcpd on

4 . 2 . TFTP のセットアップ

- ・ /etc/xinetd.d/tftp ファイルを編集（無ければ新規作成）する。

+-----+

```

service tftp
{
    disable                = no
    socket_type             = dgram

```

```

        protocol          = udp
        wait              = yes
        user              = root
#       server            = /usr/sbin/in.tftpd
        server            = /opt/hardhat/host/bin/in.tftpd
        server_args       = -s /tftpboot
        per_source        = 11
        cps               = 100 2
    }

```

```

+-----+

```

コンフィギュレーションファイルの再読み込みを行う。

```
# service xinetd restart
```

5 . ターゲットへのカーネルのインストール方法

5 . 1 . ターゲットをシリアルで接続

- ・ターゲットのシリアルコネクタと適当なホストをRS232Cケーブルで接続する。ホストで TeraTerm や Minicom を起動し、9 6 0 0 bps、ノンパリティ、1ストップビットに設定する。

5 . 2 . カーネルのインストール

- ・ターゲットの電源を ON する。
- ・直後にホストのシリアルモニタからキーを一つ（何でも良い）叩く。

```

-----
BIOS for Open-IAPr Ver.1.1 (2003/04/18)

```

```

Copyright (c) 2003 Sanyo Electric Co., Ltd.
-----

```

```

===== BIOS Main Menu =====

```

- 1 - Change Boot Device
- 2 - Change Console Port
- 3 - External Bus Configuration
- 4 - Set Kernel Command String
- 5 - Save Changes
- n - Network Devices Configuration
- 0 - Exit

```

->n          n をモニタで入力し Network Device Configuration へ
             移動する

```

=====
===== Network Device Configuration =====

Select Device : eth0(00:04:ac:2d:42:03)

- 1 - Device Change
- 2 - Set Mac Address
- 3 - Ping Test
- 4 - Load Kernel
- 5 - Load Initrd
- 6 - Store Kernel & Initrd to Device
- 7 - Boot Network Kernel
- 0 - Return Main Menu

->4 4 をモニタから入力

Kernel file name :vmlinux.bin vmlinux.bin を入力

ENET: RX OVERRUN.

と表示されたら再度電源 ON からやり直してください。

Sending bootp request ...

Loading file "vmlinux.bin" ...

Sending tftp boot request ...

 block 615 block 2474Transfer Complete ...

Loaded successfully ...

と表示されたら成功です。

成功すると再度 Network Device Configuration が表示されます。

=====
===== Network Device Configuration =====

Select Device : eth0(00:04:ac:2d:42:03)

- 1 - Device Change
- 2 - Set Mac Address
- 3 - Ping Test
- 4 - Load Kernel
- 5 - Load Initrd
- 6 - Store Kernel & Initrd to Device
- 7 - Boot Network Kernel
- 0 - Return Main Menu

->6 6 と入力する
Store Kernel to NOR FLASH (offset:00000000) (y/n) ---->y y を入力
4 コマンドでロードしたカーネルをフラッシュ ROM へ書き込みます。
Kernel has been stored.
書き込みが成功したら上記メッセージが表示されますので後は
->0 0 と入力し Main Menu へ戻る

Press any key to display Menu !!

===== BIOS Main Menu =====

- 1 - Change Boot Device
- 2 - Change Console Port
- 3 - External Bus Configuration
- 4 - Set Kernel Command String
- 5 - Save Changes
- n - Network Devices Configuration
- 0 - Exit

->0 0 と入力し Loader を抜けてカーネルを起動する

5 . 3 . Mac アドレスの参照の仕方

Main Menu で n - Network Devices Configuration を選択すると Select Device 横に表示される **00:04:ac:2d:42:03** が Mac アドレスです。

別の ethernet チャンネルの Mac アドレスを見る場合は 1 - Device Change を一回選択する毎に eth0->eth1->eth0 と表示が変わります。

===== Network Device Configuration =====

Select Device : eth0(**00:04:ac:2d:42:03**)

-
- 1 - Device Change
 - 2 - Set Mac Address
 - 3 - Ping Test
 - 4 - Load Kernel
 - 5 - Load Initrd
 - 6 - Store Kernel & Initrd to Device
 - 7 - Boot Network Kernel

Main Menu

->

5 . カーネル変更点 (参考)

6 . 1 . 2 0 0 4 / 6 / 1 4 現在

オリジナルの OpenIAPr カーネルは `vmlinux-2.4.17` をベースにしている、大容量 HDD をサポートしていなかったのが IDE ドライバのソースファイルは `vmlinux-2.4.19` のファイルに変更しました。

この変更で当初 LBA (論理ブロックアドレス) 2 8 だったのが LBA 4 8 に対応できるようになりました。

1 LBA が 5 1 2 バイト (HDD は通常 5 1 2 です) だとすると

変更前 : $2^{28} = 268,435,456 * 512 = 137,438,953,472$ 137GB (ギガバイト)

変更後 : $2^{48} = 281,474,976,710,656 = 144,115,188,075,855,872$ 144PB (ペタバイト)

となります。

ただしこれはドライバとしては対応しているというだけで、Linux のファイルシステム `ext 2` や `ext 3` は 3 2 ビットで LBA を管理していますのでやはり容量の制限があります。

やはり 1 LBA を 5 1 2 バイトとすると、

$2^{32} = 4,294,967,296 * 512 = 2,199,023,255,552$ 2.2TB (テラバイト)

となります。

HDD の容量は 2 年で倍 (根拠のない数字) とすると、2 . 2 TB が話題になるのは最低でも 6 年先ということになりますので、当分は安心していてください (?) (そのころには Linux ファイルシステム `ext ?` が登場して 2 . 2 TB の壁は超えられているとおもいますから、その時点でカーネルを更新してください。)

(現在でも `mke2fs` でフォーマットする時の 1LBA は最低が 1024 で次は

2048,4096 と選択できるので、

1LBA=1024 なら 4.4TB

1LBA=2048 なら 8.8T

1LBA=4096 なら 17.6TB

となりますので、本当に当分は安心できます。)

現在のカーネルの INITRD (イニシャル・ルート・ディレクトリ) は `/dev/hda1` と固定していますので、変更が必要な場合は `main.c` の `start_kernel()` の該当行を変更してください。あるいは Loader からカーネルへ引数を渡せるように改造してカーネル (あるいは Loader) を再構築してください。

ターゲット環境でカーネルを直接フラッシュ ROM へ書き込む場合は `/dev/mtd` (メモリー・テクノロジー・デバイス) が役に立ちそうですが、書き込み (2004/6/15

現在)は成功しませんでした。mtd デバイスドライバを変更する必要がありそうです。使えるようなら、まずm t dを調べます。

```
# cat /proc/mtd
```

mtd1 にカーネルがあるなら、

```
cat vmlinux.bin > /dev/mtd1
```

とすれば直接ターゲットの Linux 上から書き込めるはずです。

6 . 2 . オリジナルドキュメント

ターゲットでカーネルを構築する場合や NFS あるいは RAM ディスクから起動する場合は別紙 [old-open-iapr-env.pdf](#) を参照してください。