

移植 MPlayer 到 SBC2410 開發版

作者：[孫善傑](#)

網址：<http://www.ur-solution.com/>

MPlayer 在 Linux 中算是一個相當老牌的撥媒體播放程式了，它是 Linux 作業系統中使用率最高、支援格式最多、執行最穩定、資源耗費最小的影音撥放軟體。這次我要將 MPlayer 移植到 SBC2410 開發版，這片板子是[凱思科技](#)所代理，使用的是 S3C2410 的 CPU，作業系統是 ARM Linux，原本以為要移植上去，只要編譯的時候指定 cross compiler 就好，但是實際上確不是那麼簡單，在開始說明之前，首先要很感謝凱思科技的 kaise 站長大力協助，我在[凱思討論論壇](#)中與 kaise 站長有問答的過程，各位也可以去參考看看，步驟大致如下表。

1. 編譯 zlib
2. 編譯 libid3tag
3. 編譯 libmad
4. 編譯 Mplayer

1. 編譯 & 安裝 zlib

首先請先到[zlib 官方網站](#)中下此此一 lib，下載完成之後我們執行：

```
[petersun@centos madplay] tar zxvf zlib-1.2.3.tar.gz
.. 略 ..
[petersun@centos madplay]$ cd zlib-1.2.3
[petersun@centos zlib-1.2.3]$ CC=arm-linux-gcc \
> ./configure --prefix=/usr/local/arm/2.95.3
Building static library libz.a version 1.2.3 with arm-linux-gcc.
Checking for unistd.h... Yes.
Checking whether to use vs[n]printf() or s[n]printf()... using vs[n]
printf()
Checking for vsnprintf() in stdio.h... Yes.
Checking for return value of vsnprintf()... Yes.
Checking for errno.h... Yes.
Checking for mmap support... Yes.
[petersun@centos zlib-1.2.3]$ make
.. 略 ..
[petersun@centos zlib-1.2.3]$ su
Password:
```

```
[root@centos zlib-1.2.3]# make install <-- install 一定要有 root 權限
cp zlib.h zconf.h /usr/local/arm/2.95.3/include
chmod 644 /usr/local/arm/2.95.3/include/zlib.h /usr/local/arm/2.95.3/
include/zconf.h
cp libz.a /usr/local/arm/2.95.3/lib
cd /usr/local/arm/2.95.3/lib; chmod 755 libz.a
cd /usr/local/arm/2.95.3/lib; if test -f libz.so.1.2.3; then \
    rm -f libz.so libz.so.1; \
    ln -s libz.so.1.2.3 libz.so; \
    ln -s libz.so.1.2.3 libz.so.1; \
    (ldconfig || true) >/dev/null 2>&1; \
fi
cp zlib.3 /usr/local/arm/2.95.3/share/man/man3
chmod 644 /usr/local/arm/2.95.3/share/man/man3/zlib.3
[root@centos zlib-1.2.3]# (安裝完成)
```

2. 編譯 & 安裝 libid3tag

請先到 [madplay 官方網站](#) 將 libid3tag 及 libmad 一併下載回來，然後我們執行：

```
[petersun@centos madplay]$ tar zxvf libid3tag-0.15.1b.tar.gz
.. 略 ..
[petersun@centos madplay]$ cd libid3tag-0.15.1b
[petersun@centos libid3tag-0.15.1b]$ export LDFLAGS='-L/usr/local/
arm/2.95.3/lib/ -L/usr/local/arm/2.95.3/arm-linux/lib/'
[petersun@centos libid3tag-0.15.1b]$ export CPPFLAGS='-I/usr/local/
arm/2.95.3/include/'
./configure CC=/usr/local/arm/2.95.3/bin/arm-linux-gcc --host=arm-
linux --prefix=/usr/local/arm/2.95.3
.. 略 ..
[petersun@centos libid3tag-0.15.1b]$ make
.. 略 ..
[petersun@centos libid3tag-0.15.1b]$ su
Password:
[root@centos libid3tag-0.15.1b]# make install
```

3. 編譯 & 安裝 libmad

```
[petersun@centos madplay]$ tar zxvf libmad-0.15.1b.tar.gz
.. 略 ..
[petersun@centos madplay]$ cd libmad-0.15.1b
[petersun@centos madplay]$ ./configure CC=arm-linux-gcc --enable-
speed --disable-debugging --enable-fpm=arm --disable-nls --disable-
mmap --host=arm-linux --prefix=/usr/local/arm/2.95.3/arm-linux
.. 略 ..
[petersun@centos libmad-0.15.1b]$ make
.. 略 ..
[petersun@centos libid3tag-0.15.1b]$ su
Password:
[root@centos libid3tag-0.15.1b]# make install
```

3. 編譯 & 安裝 Mplayer

```
[petersun@centos sbc2410]$ tar jxvf MPlayer-1.0pre6.tar.bz2
.. 略 ..
[petersun@centos sbc2410]$ cd MPlayer-1.0pre6
[petersun@centos sbc2410]$ ./configure --cc=/usr/local/arm/2.95.3/
bin/arm-linux-gcc --target=arm-linux --enable-static --disable-win32
--disable-mp3lib --disable-dvdread --enable-fbdev --disable-mencoder
--disable-mpdvdkit --disable-x11 --disable-sdl --enable-mad --with-
madlibdir=/usr/local/arm/2.95.3/arm-linux/lib
.. 略 ..
[petersun@centos MPlayer-1.0pre6]$ make
.. 略 ..
ccl: warning: -g with -fomit-frame-pointer may not give sensible
debugging
ccl: warning: -g with -fomit-frame-pointer may not give sensible
debugging
./codec-cfg ./etc/codecs.conf > codecs.conf.h
/bin/sh: ./codec-cfg: cannot execute binary file
```

結果會出現上面的錯誤訊息，codec-cfg 是在 make 的過程中產生出來的可執行檔，但是我們當初是使用 cross compiler，所以這個執行檔當然不能在 PC(x86)的平台上執行，那怎麼辦呢？沒關係，我們換一種方式執行。

```
[petersun@centos sbc2410]$ ./configure
.. 略 ..
[petersun@centos MPlayer-1.0pre6]$ make
.. 略 ..
```

做完上述的步驟，編譯出來的是 PC(x86)平台的 Mplayer，其中產生的 codec-cfg 也可以執行，我們先將其備份到另一個檔案。

```
[petersun@centos MPlayer-1.0pre6]$ cp codec-cfg codec-cfg.x86
```

現在我們再來編譯一次 ARM 的版本。

```
[petersun@centos sbc2410]$ ./configure --cc=/usr/local/arm/2.95.3/
bin/arm-linux-gcc --target=arm-linux --enable-static --disable-win32
--disable-mp3lib --disable-dvdread --enable-fbdev --disable-mencoder
--disable-mpdvdkit --disable-x11 --disable-sdl --enable-mad --with-
madlibdir=/usr/local/arm/2.95.3/arm-linux/lib
.. 略 ..
[petersun@centos MPlayer-1.0pre6]$ make
.. 略 ..
ccl: warning: -g with -fomit-frame-pointer may not give sensible
debugging
ccl: warning: -g with -fomit-frame-pointer may not give sensible
debugging
./codec-cfg ./etc/codecs.conf > codecs.conf.h
/bin/sh: ./codec-cfg: cannot execute binary file
```

咦！還是有錯，沒關係，我們將剛剛備份的檔案給覆蓋掉新產生的 codec-cfg

```
[petersun@centos MPlayer-1.0pre6]$ cp codec-cfg.x86 codec-cfg
[petersun@centos MPlayer-1.0pre6]$ make <-- 在執行一次 make
.. 略 ..
subreader.o sub_cc.o find_sub.o m_config.o m_option.o parser-cfg.o
m_struct.o edl.o unrarlib.o mixer.o parser-mpcmd.o libvo/libvo.a
libao2/libao2.a libmpcodecs/libmpcodecs.a libaf/libaf.a
libmpdemux/libmpdemux.a input/libinput.a postproc/libswscale.a osdep/
libosdep.a libavcodec/libavcodec.a libavformat/libavformat.a -
lmad -static -lnsl liba52/liba52.a libmpeg2/
libmpeg2.a -static -lnsl -static -lpthread -ldl
-lm
[petersun@centos MPlayer-1.0pre6]$
```

現在我們已經成功的編譯一個可以在 SBC2410 上執行的 Mplayer 了，現在我們就可以來測試了嗎？還不行，我們還有一個地方要修改，在 Mplayer 目錄之下有一個 etc 的目錄，我們要修改這個目錄下的一個 codecs.conf 檔案，將所有

audiocodec mp3 及 videocodec ffmpeg 區段都給 mark 起來，只留下 mad 就好。例如：

```
videocodec ffmpeg1
  info "FFmpeg MPEG 1"
  status working
  format 0x10000001 ; mpeg 1
  fourcc mpg1
  fourcc PIM1 ; Pinnacle hardware-mpeg1
  fourcc VCR2
  driver ffmpeg
  dll "mpeg1video"
  out YV12,I420,IYUV
```

請改成

```
;videocodec ffmpeg1
;  info "FFmpeg MPEG 1"
;  status working
;  format 0x10000001 ; mpeg 1
;  fourcc mpg1
;  fourcc PIM1 ; Pinnacle hardware-mpeg1
;  fourcc VCR2
;  driver ffmpeg
;  dll "mpeg1video"
;  out YV12,I420,IYUV
```

修改完成之後，mplayer 就可以在 sbc2410 上執行了。

4. 如何來進行測試

我們可以將 mplay 直接放到板子上面執行，但是在測試（研發）階段不建議這樣做，我們可以利用 NFS 的方式進行，請在 Linux 中修改：

```
[petersun@centos ~]$ su
Password:
[root@centos petersun]# vi /etc/exports
-----以下為 exports 的內容-----
/home/petersun/sbc2410          *(rw, sync, no_root_squash)
/home/petersun/sbc2410/root    *(rw, sync, no_root_squash)
```

/home/petersun/sbc2410 這些就是您要分享出來的目錄，請依照您的環境來設定，*(rw, sync, no_root_squash) 是對分享出來目錄的設定，更多有關 NFS 的設定說明，請參考 [鳥哥的網站](#)。設定完成之後我們執行下列的指令來生效：

```
[root@centos petersun]# /etc/init.d/portmap restart
Stopping portmap: [ OK ]
Starting portmap: [ OK ]
[root@centos petersun]# /etc/init.d/nfs restart
Shutting down NFS mountd: [ OK ]
Shutting down NFS daemon: [ OK ]
Shutting down NFS quotas: [ OK ]
Shutting down NFS services: [ OK ]
Starting NFS services: [ OK ]
Starting NFS quotas: [ OK ]
Starting NFS daemon: [ OK ]
Starting NFS mountd: [ OK ]
[root@centos petersun]#
```

那在板子上要如何來掛載呢？指令如下：

```
mount -t nfs -o nolock 192.168.1.100:/home/petersun/sbc2410 /mnt/nfs
```

其中 192.168.1.100 是 PC 的 IP，/home/petersun/sbc2410 就是在 exports 中分享出來的目錄，/mnt/nfs 是你掛載的地方，請先用 mkdir -p /mnt/nfs 在板子上建立這一個目錄。

每次要開機都要打這麼多的字，實在很麻煩，所以我們將這一段做成 nfs.sh 檔案，如下：

```
#!/sbin/sh
mount -t nfs -o nolock 192.168.1.100:/home/petersun/sbc2410 /mnt/nfs
```

但是我們還是會有一個困擾，就是 PC (Notebook) 每次到別的地方，IP 都會不一樣，例如筆者經常在家中、公司、學校中使用，這三個地方 IP 都不同，怎麼辦呢？我們來修改一下 nfs.sh：

```
#!/sbin/sh
echo "rempte(NFS) ip : "
read remote_ip
mount -t nfs -o nolock $remote_ip:/home/petersun/sbc2410 /mnt/nfs
```

如此，你只要在板子上執行 `nfs.sh`，就會先問你 IP，然後再去掛載了。現在我們可以來試試看摟。