

需要准备以下文件：

zlib 源码包

mtd-snapshot-20050519.tar.bz2 源码包

一个可用的根文件系统 target 文件夹

1. 编译 zlib 库

解压缩 zlib 库，配置其安装目录，使—prefix 参数指定 arm-none-linux-gnueabi 交叉编译器的安装目录

```
>./configure --prefix=/opt/arm-none-linux-gnueabi/ arm-none-linux-gnueabi --shared
```

配置结束后修改 zlib 目录下的 Makefile 文件

指定 CC LD 等参数均使用 arm-none-linux-gnueabi 交叉编译器。然后开始编译

```
>make
```

```
>make install
```

安装结束后会在/opt/arm-none-linux-gnueabi/ arm-none-linux-gnueabi 目录下 include 目录和 lib 目录中增加关于 zlib 的文件。

2. 编译 mtd 工具

解压缩 mtd-snapshot-20050519.tar.bz2，首先编译 ARM 环境下可用的版本。

进入 mtd 的 util 目录中，修改 Makefile 文件中指定的编译工具，安装路径。

```
DESTDIR=/home/target
```

```
CROSS=arm-none-linux-gnueabi-
```

开始编译

```
>make
```

```
>make install
```

安装结束后在/home/target/usr/sbin 目录下多出一系列 mtd 工具。

编译 PC 端制作 jffs2 文件工具

修改 util 目录下的 Makefile 文件，指定用 PC 端工具编译，只要注释掉下面两行即可。

```
#DESTDIR=/home/target
```

```
#CROSS=arm-none-linux-gnueabi-
```

```
>make
```

生成 PC 端使用的 mkfs.jffs2 工具，可以利用该工具将 target 根文件系统制作成 jffs2 格式的映像文件。

```
mkfs.jffs2 -r /home/target -o /home/rootfs.jffs2 -e 0x20000 -s 0x800 -n
```

-e 参数表示擦除大小为 128K -s 参数表示 page 大小为 2048

将 rootfs.jffs2 复制到了 home 目录下

首先使用 NFS 挂载/home/target 根文件系统

```
>mtd_debug info /dev/mtd1
```

确认 mtd.size 必须大于先前生成的 rootfs.jffs2 文件

擦除/dev/mtd1，在 ARM 上执行

```
>flash_eraseall /dev/mtd1
```

然后将 Linux 平台下/home/rootfs.jffs2 文件复制到根文件系统的 tmp 目录下

```
cp /home/rootfs.jffs2 /home/target/tmp
```

将 rootfs.jffs2 烧写到 flash 中，在 ARM 上执行

```
nandwrite -p /dev/mtd1 rootfs.jffs2
```

烧写成功后，重启开发板，在 Uboot 下修改启动参数

```
setenv bootargs mem=64M console=ttyS0,115200 root=/dev/mtdblock1 rw rootfstype=jffs2
```