Ronetix Application Note AN015: Writing a YAFFS Image to NAND Flash

Often times it is useful to write the file system for you device into a permanent storage media such as NAND flash. Below you will find steps on how to write a YAFFS/YAFFS2 file system into NAND Flash on the DVEVM boards.

Contents

- 1 Prerequisites
- 2 Writing Flash from Linux
  - 2.1 Writing a Tarball Image
- 3 Setting the Kernel to Boot Using YAFFS/YAFFS2
  - 3.1 Requirements
  - 3.2 Command Line Parameters
    - 3.2.1 For YAFFS
    - 3.2.2 For YAFFS2
- 4 Conclusion

Prerequisites

- A file system image saved as a tarball file
- A terminal application for communicating with the DVEVM. i.e. minicom or hyperterminal
- A copy of the U-Boot bootloader running on the target
  - If you do not have U-Boot running on the target please see the RBL, UBL and host program article for how to get the U-Boot bootloader installed on your board first
- A kernel with YAFFS/YAFFS2 and Flash support enabled
- A root file system mounted from somewhere other than Flash. For this example we will use NFS.
  - For more information on setting up an NFS file system please refer to the “Exporting a Shared File System for Target Access” section of the Getting Started Guide for your board
- A version of the MTD Utilities for the target board to be used in writing the file system image to Flash

Writing Flash from Linux

It is possible to use the Linux MTD subsystem to write the YAFFS/YAFFS2 file system. While this method requires the use of an intermediate file system it has the benefit of not requiring calculation of the start of the file system partition and allows the DaVinci NAND driver to place the contents of the file system in the desired format. This example can be used with a .tar.gz file.

When using Linux, interaction with the Flash device is controlled through the MTD subsystem. The MTD subsystem exports two device nodes per Flash partition. These are:

1. /dev/mtd# – A character device that is used to access the raw Flash device
2. /dev/mtdblock# – A block device used to access the disk/block established in the raw Flash

In the above device nodes the '#' sign represents the partition number. For example partition 0 would have the device nodes /dev/mtd0 and /dev/mtdblock0

Writing a Tarball Image

To write a YAFFS/YAFFS2 image you will use the MTD utilities to erase and copy the file system image into the Flash. The utility that will be used is:

- flash_eraseall – Erases an entire MTD device, which in this case is a partition

The steps to write the YAFFS/YAFFS2 image are:

NOTE: The Sample output below is for the DM6446 EVM. You should see similar output when using other EVMs.

- Determine the MTD device for the file system partition of you Flash device

```
target$ cat /proc/mtd
```

You should see output similar to:

```
target$ cat /proc/mtd
dev:    size   erasesize  name
mtd0: 00040000 00004000 "bootloader"
mtd1: 00020000 00004000 "params"
mtd2: 00400000 00004000 "kernel"
mtd3: 03ba0000 00004000 "filesystem"
```

In this example the file system partition of the flash device is /dev/mtd3

NOTE: For some devices such as the DM355 you may have more than 1 file system partition. This is done to help speed boot times by mounting the root file system from a smaller Flash partition and then allowing the larger second Flash partition to be mounted after boot.

- Erase the file system partition

```
target$ flash_eraseall /dev/mtd3
```

You should see output similar to:

```
target$ flash_eraseall /dev/mtd3
Erasing 16 Kibyte @ 3b9c000 -- 99 % complete.
```

- Mount the Flash file system partition using the block device node.

NOTE: For small block NAND devices (i.e. DM6446) you must mount the file system as a YAFFS file system. For large block NAND devices (i.e. DM355) the file system should be mounted as YAFFS2. Using the wrong version of YAFFS will result in error messages similar to:

```
target$ mount -t yaffs2 /dev/mtdblock3 /mnt
yaffs: dev is 32505859 name is "mtdblock3"
yaffs: Attempting MTD mount on 31.3, "mtdblock3"
yaffs: MTD device does not support have the right page sizes
```

```
mount: wrong fs type, bad option, bad superblock on /dev/mtdblock3,
       or too many mounted file systems
```

- To mount the file system as YAFFS use the following command

```
target$ mount -t yaffs /dev/mtdblock3 /mnt
```

- To mount the file system as YAFFS2 use the following command

```
target$ mount -t yaffs2 /dev/mtdblock3 /mnt
```

- Untar the contents of the tarball file system image to the Flash device

```
target$ cd /mnt
target$ date MMDDhhmmCCYY (this is month,day,hour,minute,century,year format.  i.e. for May 2
2 10:15 2008 this would be "date 052210152008")
target$ tar xzf <imagedir>/rootfs.tar.gz
```

NOTE: The date command is used to avoid warnings about time differences

- Unmount the Flash file system partition

```
target$ cd /
target$ umount /mnt
```

Setting the Kernel to Boot Using YAFFS/YAFFS2

This section covers setting the kernel bootargs in U-Boot to boot from a Flash partition populated with a YAFFS/YAFFS2 file system image.

Requirements

- A kernel with Flash support and YAFFS2 support enabled
  - For information on enabling Flash support in the TI kernel please refer to the Flash configuration in the Kernel page
  - For information on enabling YAFFS2 support in the TI kernel please refer to the YAFFS2 kernel configuration page
  - For devices with 512b block NAND the HW ECC support must be disabled in the kernel for YAFFS to work properly. For information on disabling HW ECC support please the the Disabling NAND HW ECC support page
- A Flash device with a partition populated with a YAFFS/YAFFS2 file system

Command Line Parameters

In order to boot the YAFFS/YAFFS2 file system in Flash you must add the following kernel parameters to the kernel boot arguments defined in the “bootargs” environment variable in U-Boot:

For YAFFS

```
"... root=/dev/mtdblock3 rw rootfstype=yaffs ..."
```

The above bootargs tell the kernel that the root file system is on MTD partition 3 and that the file system type is YAFFS.

For YAFFS2

```
"... root=/dev/mtdblock3 rw rootfstype=yaffs2 ..."
```

The above bootargs tell the kernel that the root file system is on MTD partition 3 and that the file system type is YAFFS.

Conclusion

You should now have a kernel which boot using a YAFFS/YAFFS2 root file system in Flash.

Share on LinkedinShare on FacebookShare on Reddit on TwitterShare Via EmailShare on Whatsapp

Share