

# 在VxWorks5.5下tffs的创建

硬件平台：基于MPC7145的mvme6100VME板卡。

软件：VxWorks5.5。

TFFS, True Flash File System的简写，是Vxworks在系统flash设备上构造的一个基于DOS的文件系统，用于存放交换及应用程序，bootrom程序等很少需要修改的信息，便于程序的更新和升级。

首先，在板卡上烧录支持tffs的bootrom，并在ftp目录下放置相应的image。在shell中使用以下命令查看tffs是否存在。

```
1  -> tffsShowAll
2  TFFS Version 2.2
3  0: socket=SIMM: type=0x0, unitSize=0x40000, mediaSize=0x2000000
4  1: socket=SIMM: type=0x0, unitSize=0x40000, mediaSize=0x2000000
5  value = 48 = 0x30 = '0'
6  ->
```

shell返回以上字符后说明目前的bootrom和image支持tffs系统的创建。接下来需要格式化flash。

```
1  -> sp sysTffsFormat,0
2  task spawned: id = 1df3bc00, name = s1u0
3  value = 502512640 = 0x1df3bc00
4  ->
```

等待格式化完成后，再创建tffs盘符。

```
1  -> usrTffsConfig(0,0,"/tffs0")
2  value = 0 = 0x0
3  ->
```

此时，我们就可以在VxWorks系统设备中查找到 tffs0 目录。

```
1  -> devs
2  drv name
3    0 /null
4    1 /tyCo/0
5    1 /tyCo/1
6    5 host:
7    8 /vio
8    3 /tffs0
9  value = 0 = 0x0
10 ->
```

```
1  -> cd "/tffs0"
2  value = 0 = 0x0
3  -> ls
4  value = 0 = 0x0
5  ->
```

通过 cd 命令进入到“tffs0”盘里，可以通过 ls 命令列出目前盘里存在的文件。

在前面列出的设备里，“host:”代表ftp指向主机的目录，在“host:”目录下我们就能查看到ftp目录里的文件。

```
1  -> cd "host:"
2  value = 0 = 0x0
3  -> ls
4  LICENSE.TXT
5  LaserData.txt
6  N1225.out
7  README.TXT
8  SETUP
9  bootrom.bin
10 docs
11 host
12 lmPhoneNL.txt
13 setup.log
14 setup.log.abort
15 share
16 target
17 vxworks
18 value = 0 = 0x0
```

这时可以从ftp目录拷贝文件到“tffs0”目录里，这样就可以实现主机个板卡之间文件的交互、程序的更新等。

```
1  -> pwd
2  host:
3  value = 6 = 0x6
4  -> copy "vxworks","/tffs0/vxworks"
5  Copy OK: 2503526 bytes copied
6  value = 0 = 0x0
7  -> ls "/tffs0"
8  /tffs0/vxworks
9  value = 0 = 0x0
10 ->
```

接下来，我们可以通过修改VxWorks的启动设置，使系统启动的时候从tffs上引导，不用从ftp引导。

重启之后按照如下步骤修改：

```
1  Press any key to stop auto-boot...
2  7
3
4  [VxWorks Boot]: c
5
6  '.' = clear field; '-' = go to previous field; ^D = quit
7
8  boot device          : geisc0 tffs0=0
9  processor number     : 0
10 host name            : host
11 file name            : vxworks /tffs0/vxworks
12 inet on ethernet (e) : 192.168.2.61
13 inet on backplane (b):
14 host inet (h)        : 192.168.2.202
15 gateway inet (g)     :
16 user (u)             : mvme6100
```

```

17 ftp password (pw) (blank = use rsh): 1234
18 flags (f) : 0x0
19 target name (tn) :
20 startup script (s) :
21 other (o) : geisc0
22
23 [VxWorks Boot]: @
24
25 boot device : tffs=0
26 unit number : 0
27 processor number : 0
28 host name : host
29 file name : /tffs0/vxworks
30 inet on ethernet (e) : 192.168.2.61
31 host inet (h) : 192.168.2.202
32 user (u) : mvme6100
33 ftp password (pw) : 1234
34 flags (f) : 0x0
35 other (o) : geisc0
36
37 Attaching to TFFS... done.
38 Loading /tffs0/vxWorks...1627776
39 Starting at 0x100000...
40
41 Attached TCP/IP interface to geisc unit 0
42 Attaching interface lo0...done
43
44 Adding 4120 symbols for standalone.
45
46
47 [][][][][][][][][][][][][][][][][][][][][][][]
48 [][][][][][][][][][][][][][][][][][][][][][][]
49 [][][][][][][][][][][][][][][][][][][][][][][]
50 [][][][] [] [] [] [] [] [] []
51 (R) [] [] [] [] [] [] [] []
52 [] [] [] [] [] [] [] [] [] []
53 [] [] [] [] [] [] [] [] [] [] [] [] [] []
54 [] [] [] [] [] [] [] [] [] [] [] [] [] []
55 [] [] [] [] [] [] [] [] [] [] [] []
56 [] [] [] [] [] [] [] [] [] [] []
57 [] [] [] [] [] [] [] [] [] [] [] []
58 [] [] [] [] [] [] [] [] [] [] []
59 [][][][][][][][][][][][][][][][][][][]
60 [][][][][][][][][][][][][][][][][] Development System
61 [][][][][][][][][][][][][][][][][]
62 [][][][][][][][][][][][][][][][] VxWorks version 5.5.1
63 [][][][][][][][][][][][][][][][] KERNEL: WIND version 2.6
64 [][][][][][][][][][][][][][][][] Copyright Wind River Systems, Inc., 1984-
65 2003

```

```

66 CPU: Emerson MVME6100-0163 - MPC 7457.
    Processor #0.
67 Memory Size: 0x20000000. BSP version 1.2/8.
68 WDB Comm Type: WDB_COMM_END
69 WDB: Ready.
70 ->

```

从启动信息可以看出，系统是从tffs盘引导启动的。

我们也可以把程序放入tffs盘里，通过启动脚本使板卡在上电后自动引导程序，达到上电后程序立即启动的目的。

向 tffs0 盘中拷入启动脚本 `start`，程序文件 `MotionPPC.out`，这写名字根据自己程序的实际情况填写就可以，然后修改系统启动参数。

```

1 -> ls
2 vxworks
3 MotionPPC.out
4 start
5 value = 0 = 0x0

```

```

1 [Vxworks Boot]: c
2
3 '.' = clear field; '-' = go to previous field; ^D = quit
4
5 boot device      : tffs=00
6 processor number : 0
7 host name       : host
8 file name       : /tffs0/vxworks
9 inet on ethernet (e) : 192.168.2.61
10 inet on backplane (b):
11 host inet (h)    : 192.168.2.202
12 gateway inet (g) :
13 user (u)        : mvme6100
14 ftp password (pw) (blank = use rsh): 1234
15 flags (f)       : 0x0
16 target name (tn) :
17 startup script (s) : /tffs0/start
18 other (o)        : geisc0

```

重点是 `startup script (s)` 和 `other(o)` 将他们修改为 `/ttds0/start` 和 `geisc0`，这样就可以使系统启动后先运行 tffs0 盘里的 `start` 文件，并且在程序里还可以使用 `geisc0` 网络。启动完成如下。

```

1 Executing startup script /tffs0/start ...
2 cd "/tffs0/"
3 value = 0 = 0x0
4 ld< MotionPPC.out
5 value = 502365328 = 0x1df17c90
6 TaskInit
7 Listen socket create successfully!
8 Bind to port successfully!
9 Listening!
10 value = 488246816 = 0x1d1a0e20
11 cd "host:"

```

```
12 | value = 0 = 0x0
13 | 5000
14 |
15 | Done executing startup script /tffs0/start
```

start文件的程序内容如下：

```
1 | cd "/tffs0/"
2 | ld< MotionPPC.out
3 | TaskInit
4 | cd "host:"
```

启动脚本很简单，第一行是切换到 /tffs0 盘符里，然后加载程序文件，再执行程序文件的中的函数 TaskInit ,最后再切换到ftp目录下， host: 指的是ftp目录，也就是我们在用ftp启动时指定的目录下，程序启动的时候默认都是 host: 目录。

通过创建 /tffs 文件系统，使板卡的flash设备操作和我们平时使用PC机一样方便，我们可以对板卡的文件很方便的进行改动。在程序里使用C语言的文件库就可以对文件进行改动。并且可以把程序中需要永久保存的配置文件放入 tffs 中，使程序启动的时候更加方便。