Booting RTEMS

T.S, 9/20/2007

Overview

- Concepts
- Different ways of booting
- 2-Stage booting with NETBOOT
- Using BOOTP/DHCP
- DHCPd config file

Concepts

- Boot Parameters: Parameters that 'personalize' a board and describe its software and networking environment:
  - Were to get parms: NVRAM, BOOTP, usr input
  - Ethernet Port configuration (NVRAM only!)
  - IP environment (address, netmask, gway, ...)
  - Software to execute
    - Boot file
    - Additional parameters ('cmdline') to pass (e.g., 'init script')

Concepts (cont.)

- NETBOOT: a specialized RTEMS application that deals with
  - assembling Boot Parameters from NVRAM, user input, BOOTP
  - downloading 'real' application
  - starting 'real' app. and pass cmdline + other boot parms.
- *Boot method is arch/BSP dependent*
- Described method avail. for 'beatnik' and 'uC5282'.

Different Ways for Booting

1. Where do parameters reside
   a) Hardcoded in app.
   b) NVRAM
   c) User input
   d) BOOTP
   e) Combination of the above
2. How is app. loaded and started
   1. From flash or net by firmware
   2. Two stage approach:
      firmware -> NETBOOT -> application

Where to Store Parameters

- BOOTP: good for production; centralized management. Changes are easily propagated to all IOCs
- NVRAM: bad for production; update of all units required
- NVRAM: good for development + temporary work-arounds
- USER INPUT: good for testing; do something differently ONCE w/o changing NVRAM or BOOTP server setup.

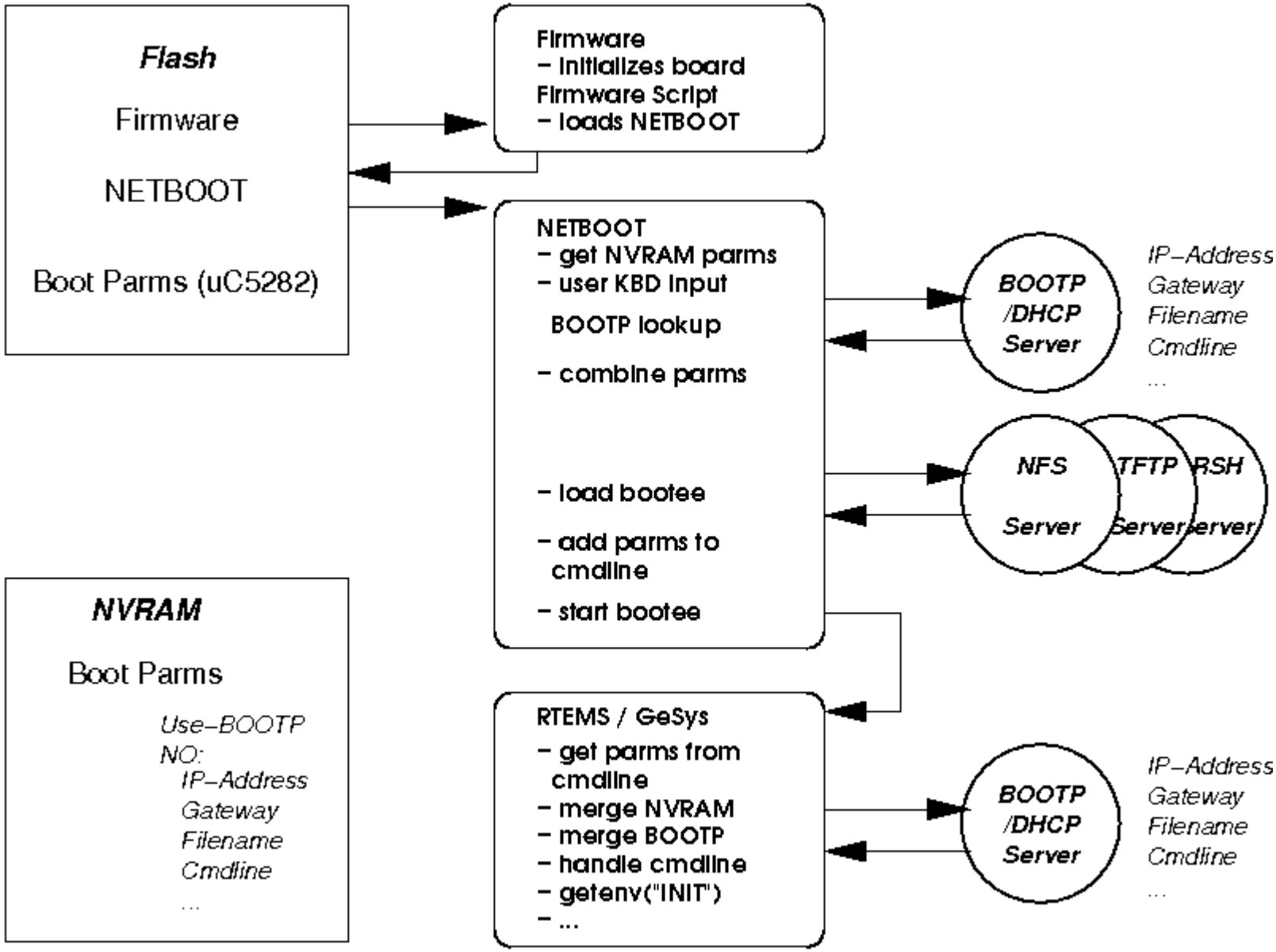
Benefits of Two-Stage Approach

- NETBOOT changes rarely; no need to redeploy app + reprogram flash everywhere when app. changes
- NETBOOT presents same interface on uC5282 and beatnik; no need to 'learn' different firmware.
- NETBOOT supports interactive user input: can change some boot parms w/o modification of NVRAM.
- NETBOOT faster than most firmware
- NETBOOT supports NFS (and RSH)

Two Stage Approach (cont.)

- Booting directly out of flash still has benefits, mostly when traveling.
- Flexible boot parameters always needed.

2-Stage Booting with NETBOOT



More About NETBOOT

- here:

  file:///afs/slac/package/rtems/<version>/src/ssrlApps/netboot/README.netboot

  BSP-specific information is in

  ..../README.<bsp>
- Maybe do another 'How-To' on NETBOOT.

DHCP Config File

- Depends on DHCP (BOOTP superset) server.
- Need to provide
  - IP address, netmask, gateway
  - DNS servers
  - NTP servers
  - Boot file (*Syntax depends on loader; subtle problem when using firmware + BOOTP; NETBOOT is OK*)
  - Option 129 -> command-line string

ISC DHCPd

```
ddns-update-style none;
option domain-name "slac.stanford.edu";
option domain-name-servers ns1.slac.stanford.edu,ns2.slac.stanford.edu;
option ntp-servers ntp1.slac.stanford.edu,ntp2.slac.stanford.edu;
option uarg code 129 = text;

subnet 134.79.xxx.0 netmask 255.255.252.0 {
    option routers 134.79.xxy.1;
    next-server lcls-strauman;
    use-host-decl-names on;

    filename "afsnfs2:/afs/slac:package/rtems/4.7.1/target/ssrlApps/
i386-rtems/pc586/bin/rtems.exe";
   host lclsdev-05 {
        hardware ethernet 00:40:9e:01:1d:ec;
        fixed-address lclsdev-05;
        option uarg  "INIT=/boot/g/lcls/epics/iocBoot/ioctest/st.cmd
IPADDR1=192.168.10.10 MODEL=PP41x";
    }
    }
}
```

PATH Syntax

- Applies to NETBOOT and GeSys:
- file name (NFS only; TFTP / RSH also supported):

```
[<uid>'.'<gid>'@'] <host> ':' <export> ':' <path>
```

- init script (NFS only; TFTP also supported)

```
INIT=<nfspath_as_defined_above>
```

if srv/export identical with boot file use

```
INIT=/boot/<path>
```

quoting available if path contains blanks