Hands-on Workshop: SDK Bring-Up for QorIQ Communications Processors

AMF-ENT-T1032

Haiying Wang
Software Engineer

October 2013

Session 1

Getting Started

Session 1 - Getting Started

The Things You Will Do

- Walk through the steps to use the Freescale open-source SDK for QorIQ processors
- Prepare the board so it will reset into u-boot
- Deploy and run software scenarios for cases :
  - SMP Linux
  - DPAA Debug

Session 1 - Getting Started

The Equipment You Will Work With

- P2041RDB QorIQ DPAA Development System
- A host Windows laptop

Session 1 - Getting Started

Host Basics

- Host : Windows7
  - Teraterm /* start the terminal for serial output */
  - Tftpd32 /* provide tftp server to download images */

Session 1 - Getting Started

Networking

- IP addresses :
  - Host IP : `192.168.1.199 host FSL-QorIQ`
  - P2041RDB IP : `192.168.1.100 target`
- Host runs **`tftpd`** service to download the images to p2041rdb board

Session 1 - Getting Started

Image Files

- The factory default images are located in :

C:/Program Files/tftpd32

```
rcw_14g_1500mhz.bin
u-boot-P2041RDB.bin
fsl_fman_ucode_P2041_106_1_4.bin
uImage-p2041rdb.bin
fsl-image-core-p2041rdb.rootfs.ext2.gz.u-boot
uImage-p2041rdb.dtb
uImage-p2041rdb-usdpaa.dtb
```

-

Session 1 - Getting Started

Notational Conventions : Command prompts

```
$ <Linux host command as user>
<output>

# <target Linux command as root>
<output>

=> <target u-boot command>
<output>
```

Session 1 - Getting Started

Documentation on Your Class PC

- *SDK 1.3 CD* (Yocto Source ISO): d:/yocto/documents

  `click START_HERE.html`

- *Board information*: Insert and browse USB key

Session 1 - Getting Started

Install Yocto

- Mount the ISO on a linux machine:

  $ sudo mount -o loop QorIQ-SDK-V1.3-<target>-<yyyymmdd>-yocto.iso /mnt/cdrom

- As a non-root user, install the Yocto:

  $ /mnt/cdrom/install

- When you are prompted to input the install path, ensure that the current user has the correct permission for the install path.

  The installation on host is:

  /home/fsldwf/QorIQ-SDK-V1.3-20121114-yocto/

Session 1 - Getting Started

Host Environment

- Yocto requires some packages to be installed on host.
- The following steps are used for preparing the environment for Yocto running.

  $ cd <yocto_install_path>

  $. /scripts/host-prepare.sh

- $ source ./fsl-setup-poky -m <machine> -j [# of bitbake jobs] -t [# of maketasks]

  For example:

  $ source ./fsl-setup-poky -m p2041rdb

  /* build_p2041rdb_release/ will be generated */

Session 1 - Getting Started

Builds

- $ cd QorIQ-SDK-V1.3-20121114-yocto/build_p2041rdb_release
- $ bitbake <image-target>
  - *fsl-image-minimal:* contains basic packages to boot up a board
  - *fsl-image-core:* contains common open source packages and Freescale-specific packages.
  - *fsl-image-full:* contains all packages in the full package list.
  - *fsl-image-flash:* contains all the user space apps needed to deploy the fsl-image-full image to a USB stick, hard drive, or other large physical media.
  - *fsl-image-kvm:* contains guest rootfs in qemu
  - *fsl-toolchain*: the cross compiler binary package
  - *package-name(usdpaa):* build a specific package

Session 1 - Getting Started
Builds --continued

- Example:

  $ bitbake fsl-image-core

```
<project>/tmp
├── buildstats
├── cache
├── ccache
├── deploy
│   ├── images
│   ├── licenses
│   ├── rpm
│   └── sdk
├── pkgdata
├── sstate-control
├── stamps
├── sysroots
│   ├── x86_64-linux
│   ├── p2041rdb
│   └── p2041rdb-tcbootstrap
└── work
    ├── all-fsl-linux
    ├── x86_64-linux
    ├── p2041rdb-fsl-linux
    └── ppce500mc-fsl-linux
```

Generated image files (kernel, u-boot, rcw, rfs, ...)

Generated toolchain

Session 2

Initial Board Bringup

Session 2 - Initial Board Bringup

Verify Physical Connections Between Board And PC

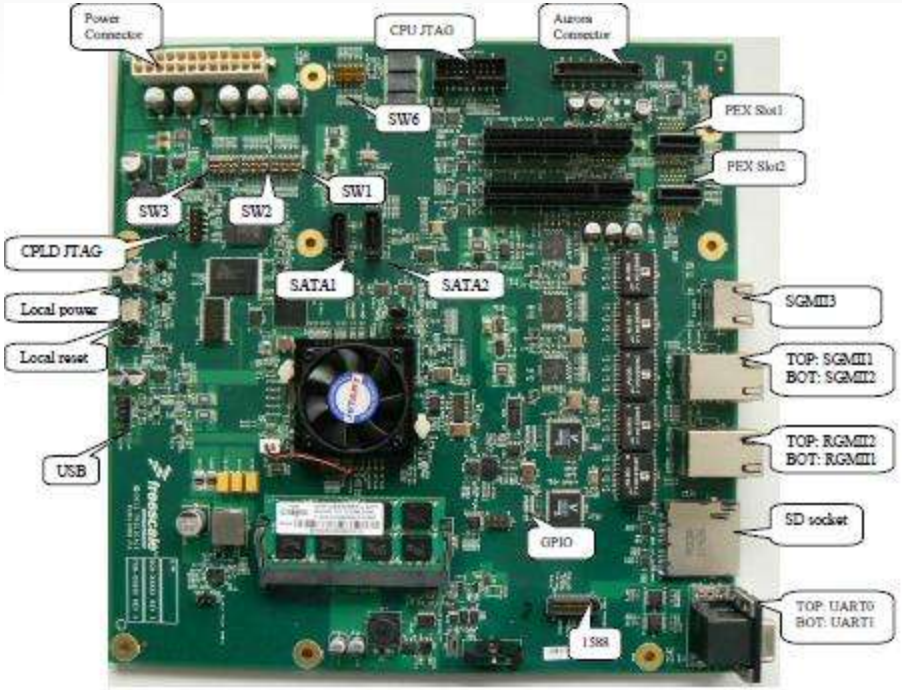
- ✓ Power cable to P2041RDB
- ✓ Ethernet from board SGMII port to PC
- ✓ Serial cable from board top RS-232 port to PC, using USB-serial adapter

Session 2 - Initial Board Bringup

Verify board configuration

- ✓ Board configuration switches set to default

SW1[1:8] = 10110100
SW2[1:8] = 00100100
SW3[1:8] = 01010111
Processor: P2041
SysClk: 83.33
Core: 1500mhz
CCB: 750mhz
DDR: 1333mhz
Fman: 583
RCW: rcw_14g_1500mhz.bin



Session 2 - Initial Board Bringup

board memory map

Range Start	Range End	Definition	Size
0xeff80000	0xefffffff	u-boot (current bank)	512KB
0xeff60000	0xeff7ffff	u-boot env (current bank)	128KB
0xef000000	0xeff5ffff	FMAN Ucode (current bank)	16MB
0xed300000	0xeeffffff	Rootfs (alternate bank)	30MB
0xec020000	0xed2fffff	Linux.uImage (alternate bank)	19MB
0xec000000	0xec01ffff	Rcw (alternate bank)	128KB
0xebf80000	0xebffffff	u-boot (alternate bank)	512KB
0xebf60000	0xebf7ffff	u-boot env (alternate bank)	128KB
0xeb000000	0xebf5ffff	FMAN ucode (alternate bank)	16MB
0xe9300000	0xeaffffff	Rootfs (current bank)	30MB
0xe9000000	0xe92fffff	Guest image #3 (current bank)	3MB
0xe8d00000	0xe8ffffff	Guest image #2 (current bank)	2MB
0xe8a00000	0xe8cfffff	Guest image #1 (current bank)	3MB
0xe8900000	0xe89fffff	HV config device tree (current bank)	1MB
0xe8800000	0xe88fffff	Hardware device tree (current bank)	1MB
0xe8700000	0xe87fffff	HV.uImage (current bank)	1MB
0xe8020000	0xe86fffff	Linux.uImage (current bank)	7MB
0xe8000000	0xe801ffff	Rcw (current bank)	128K

•

Session 2 - Initial Board Bringup

Power On

- Log in on host.
- Start `teraterm` serial console
  - Select “Serial”, click “OK”
  - In “Setup”, click “Serial port”, change the “Baudrate” to 115200
- Start `Tftpd32`, make sure the “Server interface” is

  192.168.1.199
- Push front on/off switch on P2041RDB
  - what is the output from the console ?

  =>

Session 2 - Initial Board Bringup

Typical u-boot output

```
U-Boot 2011.12-00025-gc6d9d50 (Oct 24 2012 - 04:48:58)

   CPU0: P2041E, Version: 1.1, (0x82180111)
   Core: E500MC, Version: 2.2, (0x80230022)
   Clock Configuration:
   CPU0:1500 MHz, CPU1:1500 MHz, CPU2:1500 MHz, CPU3:1500 MHz,
   CCB:750 MHz,
   DDR:666.667 MHz (1333.333 MT/s data rate) (Asynchronous), LBC:93.750 MHz
   FMAN1: 583.333 MHz
   PME: 375 MHz
L1: D-cache 32 kB enabled
   I-cache 32 kB enabled
Board: P2041RDB, CPLD version: 4.1 vBank: 0
36-bit Addressing
Reset Configuration Word (RCW):
   00000000: 12600000 00000000 241c0000 00000000
   00000010: 648ea0c1 c3c02000 de800000 40000000
   00000020: 00000000 00000000 00000000 d0030f07
   00000030: 00000000 00000000 00000000 00000000
SERDES Reference Clocks: Bank1=100Mhz Bank2=125Mhz
[…]
```

Session 2 - Initial Board Bringup

Some u-boot Maintenance

- u-boot did not find a valid environment in flash, so used the compiled default

```
*** Warning - bad CRC, using default environment
```

- Check the network settings and update :

```
=> print
=> setenv ipaddr   192.168.1.100
=> setenv serverip 192.168.1.199
=> setenv bootargs root=/dev/ram rw
console=ttyS0,115200 usdpaa_mem=256M
```

- Save the updated current environment in flash :

```
=> saveenv
```

Session 2 - Initial Board Bringup

Some u-boot Maintenance

- Test network connectivity

```
=> ping $serverip
```

Session 2 - Initial Board Bringup

Some u-boot Maintenance

- Flash rcw, u-boot, ucode to the alternative bank

```
=> tftp 1000000 rcw_14g_1500mhz.bin
Using FM1@DTSEC2 device
TFTP from server 192.168.1.199; our IP address is 192.168.1.100
Filename 'rcw_14g_1500mhz.bin'.
Load address: 0x1000000
Loading: #
done
Bytes transferred = 248 (f8 hex)
=> erase ec000000 +100
.done
=> cp.b 1000000 ec000000 100
Copy to Flash... 9....done
=> tftp 1000000 u-boot-P2041RDB.bin;erase ebf80000 +80000; cp.b
  1000000 ebf80000 80000;
=> tftp 1000000 fsl_fman_ucode_P2041_106_1_4.bin; erase eb000000
  +7000; cp.b 1000000 eb000000 7000
=> cpld reset altbank  (vBank1)
```

Session 2 - Initial Board Bringup

Some u-boot maintenance

- Check the images in flash

```
=> imls
Legacy Image at E8020000:
 Image Name:   Linux-3.0.48-rt70
 Image Type:   PowerPC Linux Kernel Image (gzip compressed)
 Data Size:    3769254 Bytes = 3.6 MiB
 Load Address: 00000000
 Entry Point:  00000000
 Verifying Checksum ... OK
Legacy Image at E9300000:
 Image Name:   fsl-image-core-p2041rdb-20121025
 Image Type:   PowerPC Linux RAMDisk Image (gzip compressed)
 Data Size:    28838515 Bytes = 27.5 MiB
 Load Address: 00000000
 Entry Point:  00000000
 Verifying Checksum ... OK
```

Session 3

SMP Linux Scenario

Session 3 - SMP Linux Scenario

Deploy Linux Images

- Follow the steps again to set u-boot environment and verify the network connection

```
=> print
=> setenv ipaddr   192.168.1.100
=> setenv serverip 192.168.1.199
=> setenv bootargs root=/dev/ram rw
console=ttyS0,115200
```

- Save the updated current environment in flash :

```
=> saveenv
```

- Test network connectivity

```
=> ping $serverip
```

Session 3 - SMP Linux Scenario

Deploy Linux Images

- Download Linux kernel, RFS and device tree blob images

```
=> tftp 40000000 uImage-p2041rdb.bin
Using FM1@DTSEC2 device
TFTP from server 192.168.1.199; our IP address is 192.168.1.100
Filename 'Image-p2041rdb.bin'.
Load address: 0x1000000
- Loading: #####.............................######
- done
- Bytes transferred = 3769318 (3983e6 hex)
- => tftp 41000000 fsl-image-core-p2041rdb.rootfs.ext2.gz.u-boot
=> tftp c00000 uImage-p2041rdb.dtb
- => bootm 40000000 41000000 c00000
```

Session 3 - SMP Linux Scenario

Deploy Linux Images

- Boot SMP Linux using one of the added envvar commands :

```
=> setenv bootargs root=dev/ram rw console=ttyS0,115200
=> bootm 40000000 41000000 c00000
WARNING: adjusting available memory to 30000000
## Booting kernel from Legacy Image at 40000000 ...
   Image Name:   Linux-3.0.48-rt70
   Image Type:   PowerPC Linux Kernel Image (gzip compressed)
   Data Size:    3769254 Bytes = 3.6 MiB
   Load Address: 00000000
   Entry Point:  00000000
   Verifying Checksum ... OK
## Loading init Ramdisk from Legacy Image at 41000000 ...
    Image Name:   fsl-image-core-p2041rdb-20121025
    Image Type:   PowerPC Linux RAMDisk Image (gzip compressed)
    Data Size:    28838515 Bytes = 27.5 MiB
    Load Address: 00000000
    Entry Point:  00000000
    Verifying Checksum ... OK
## Flattened Device Tree blob at 00c00000
    Booting using the fdt blob at 0x00c00000
    Uncompressing Kernel Image ... OK
    Loading Ramdisk to 2e47f000, end 2ffffa73 ... OK
    Loading Device Tree to 03fe2000, end 03fff227 ... OK
```

Session 3 - SMP Linux Scenario

SMP Linux Boot

- Review the output for relevant and interesting sections
- Log into the target system :

```
Yocto (Built by Poky 7.0) 1.2 p2041rdb ttyS0

p2041rdb login: root
root@p2041rdb:~#
```

Session 3 - SMP Linux Scenario

A Few Quick Checks

```
# more /proc/cpuinfo
processor       : 0
cpu             : e500mc
clock           : 1499.985000MHz
revision        : 2.2 (pvr 8023 0022)
bogomips        : 46.87
[...]
processor       : 3
cpu             : e500mc
clock           : 1499.985000MHz
revision        : 2.2 (pvr 8023 0022)
bogomips        : 46.87

total bogomips  : 187.50
timebase        : 23437500
platform        : P2041 DS
model           : fsl,P2041RDB
Memory          : 4096 MB
```

Session 3 - SMP Linux Scenario

A Few Quick Checks

```
# top ← hit '1' to show all cores
top - 03:54:48 up 7 min,  1 user,  load average: 0.14, 0.08, 0.05
Tasks:  55 total,   1 running,  54 sleeping,   0 stopped,   0 zombie
Cpu(s):  0.1%us,  0.4%sy,  0.0%ni, 99.5%id,  0.1%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu0  :  0.0%us,  0.2%sy,  0.0%ni, 99.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu1  :  0.0%us,  0.1%sy,  0.0%ni, 99.8%id,  0.1%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu2  :  1.8%us,  4.8%sy,  0.0%ni, 89.6%id,  3.7%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu3  :  0.1%us,  0.5%sy,  0.0%ni, 99.4%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   4024852k total,   124364k used,  3900488k free,      920k buffers
Swap:        0k total,        0k used,        0k free,     8776k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
    1 root      20   0  1968  476  424 S    0  0.0   0:02.71 init
    2 root      20   0     0    0    0 S    0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S    0  0.0   0:00.00 ksoftirqd/0
    4 root      20   0     0    0    0 S    0  0.0   0:00.00 kworker/0:0
    5 root      20   0     0    0    0 S    0  0.0   0:00.01 kworker/u:0
    6 root      RT   0     0    0    0 S    0  0.0   0:00.00 migration/0
    7 root      RT   0     0    0    0 S    0  0.0   0:00.00 migration/1
```

Session 3 - SMP Linux Scenario

Bring Up and Test The Network

- Available interfaces :

```
root@p2041rdb:~# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: fm1-gb0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
    link/ether 00:e0:0c:00:ea:00 brd ff:ff:ff:ff:ff:ff
3: fm1-gb1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
    link/ether 00:e0:0c:00:ea:01 brd ff:ff:ff:ff:ff:ff
4: fm1-gb3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
    link/ether 00:e0:0c:00:ea:02 brd ff:ff:ff:ff:ff:ff
5: fm1-gb3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
    link/ether 00:e0:0c:00:ea:03 brd ff:ff:ff:ff:ff:ff
6: fm1-10g: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
    link/ether 00:e0:0c:00:ea:05 brd ff:ff:ff:ff:ff:ff
7: tunl0: <NOARP> mtu 1480 qdisc noop state DOWN
    link/ipip 0.0.0.0 brd 0.0.0.0
8: sit0: <NOARP> mtu 1480 qdisc noop state DOWN
    link/sit 0.0.0.0 brd 0.0.0.0
```

  - `fm1-gb0/1` are the SGMII ports
  - `fm2-gb3/4` are the RGMII ports
  - `fm2-10g` is the 10G port, it cannot be used.

Session 3 - SMP Linux Scenario

Bring Up and Test The Network Connection

- Assign the same IP address as for `ipaddr` in `u-boot`

```
# ifconfig fm1-gb1 192.168.1.100

# ping 192.168.1.199
PING 192.168.1.199 (192.168.1.199): 56 data bytes
64 bytes from 192.168.1.199: icmp_seq=0 ttl=64 time=3.203 ms
64 bytes from 192.168.1.199: icmp_seq=1 ttl=64 time=0.277 ms
--- 192.168.1.199 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.277/1.740/3.203/1.463 ms

# ping -f 192.168.1.199
PING 192.168.1.199 (192.168.1.199): 56 data bytes
--- 192.168.1.199 ping statistics ---
296 packets transmitted, 296 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.192/0.261/0.551/0.036 ms
```

Session 3 - SMP Linux Scenario

Check Interrupts

```
# cat /proc/interrupts
           CPU0       CPU1       CPU2       CPU3
 16:          0          0          0          0   OpenPIC   Level     mpic-error-int
 24:          0          0          0          0   OpenPIC   Level     pamu
 25:          0          0          0          0   OpenPIC   Level     fsl-lbc
 28:          0          0          0          0   OpenPIC   Level     dma-uio0-0
 29:          0          0          0          0   OpenPIC   Level     dma-uio0-1
[...]
36:       8617          5        141          5   OpenPIC   Level     serial
 38:          0          0         41          0   OpenPIC   Level     i2c-mpc,i2c-mpc
 39:          0          0          0          0   OpenPIC   Level     i2c-mpc,i2c-mpc
 44:          0          1          0          0   OpenPIC   Level     ehci_hcd:usb2
 45:          1          0          0          0   OpenPIC   Level     ehci_hcd:usb1
 48:          0          0          0          0   OpenPIC   Level     mmc0
 53:          0          0          2          0   OpenPIC   Level     fsl_espi
 68:        664          0          0          0   OpenPIC   Level     fsl-sata
 [...]
 96:          0          0          0          0   OpenPIC   Level     fman
104:          0          0          0          0   OpenPIC   Level     qman-uio-0
105:          0          0          0          0   OpenPIC   Level     bman-uio-0
[...]
```

Session 3 - SMP Linux Scenario

Check Interrupts

```
           CPU0       CPU1       CPU2       CPU3
116:          0          0          0         33   OpenPIC   Level     QMan portal 3
117:          0          0          0          0   OpenPIC   Level     BMan portal 3
118:          0          0         33          0   OpenPIC   Level     QMan portal 2
119:          0          0          0          0   OpenPIC   Level     BMan portal 2
120:          0         32          0          0   OpenPIC   Level     QMan portal 1
121:          0          0          0          0   OpenPIC   Level     BMan portal 1
122:         89          0          0          0   OpenPIC   Level     QMan portal 0
123:          0          0          0          0   OpenPIC   Level     BMan portal 0
[...]
480:          0          0          0          0   OpenPIC   Edge      fman-err
481:          0          0          0          0   OpenPIC   Edge      bman-err
482:          0          0          0          0   OpenPIC   Edge      qman-err
484:          0          0          0          0   OpenPIC   Edge      pme-err
493:          0          0          0          0   OpenPIC   Edge      [EDAC] PC
I err
LOC:       5596       3593        900       1310   Local timer interrupts
SPU:          0          0          0          0   Spurious interrupts
CNT:          0          0          0          0   Performance monitoring interr
upts
MCE:          0          0          0          0   Machine check exceptions
```

Note all cores handling Rx IRQs servicing the same pool channel

Session 4

DPAA debug

Session 4 – DPAA debug

Do You Experience Packet Lost?

- What is FMan doing with the packets it receives?
  - Dropping the packets?
  - Enqueueing the packets to QMan?
- Starting out you want to determine whether or not the packets made it through the MAC and all the way to an FMAN port.

- Check the FMan RX port statistics.
  - There's an easy way to do this when running Linux.

Session 4 – DPAA debug

Check sysfs for Fman device statistics

- sysfs allows kernel to export statistic registers value to user process via an in-memory filesystem

```
root@p2041rdb:~# ls /sys/devices/ffe000000.soc/ffe400000.fman/statistics
bus_error                          pcd_prs_hard_prs_cycle_incl_stall_cycles
cmq_not_empty                      pcd_prs_l2_parse_result_returned
deq_0                              pcd_prs_l2_parse_result_returned_with_err
deq_1                              pcd_prs_l3_parse_result_returned
deq_2                              pcd_prs_l3_parse_result_returned_with_err
deq_confirm                        pcd_prs_l4_parse_result_returned
deq_from_context                   pcd_prs_l4_parse_result_returned_with_err
deq_from_default                   pcd_prs_muram_read_cycles
deq_from_fd                        pcd_prs_muram_read_stall_cycles
deq_total_frame                    pcd_prs_muram_write_cycles
enq_total_frame                    pcd_prs_muram_write_stall_cycles
pcd_enq_total_frame                pcd_prs_parse_dispatch
pcd_kg_total                       pcd_prs_shim_parse_result_returned
pcd_plcr_length_mismatch           pcd_prs_shim_parse_result_returned_with_err
pcd_plcr_recolored_to_red          pcd_prs_soft_prs_cycles
pcd_plcr_recolored_to_yellow       pcd_prs_soft_prs_stall_cycles
pcd_plcr_red                       read_buf_ecc_error
pcd_plcr_total                     write_buf_ecc_fm_error
pcd_plcr_yellow                    write_buf_ecc_sys_error
pcd_prs_fpm_command_stall_cycles
```

Session 4 – DPAA debug

Check sysfs for Fman device statistics

```
root@p2041rdb:~# cat /sys/devices/ffe000000.soc/ffe400000.fman/statistics/*
            FM 0 counter:  F
            FM 0 counter:  F
            FM 0 counter:  760
            FM 0 counter:  0
            FM 0 counter:  0
            FM 0 counter:  760
            FM 0 counter:  0
            FM 0 counter:  760
            FM 0 counter:  1996
            FM 0 counter:  0
            FM 0 counter:  0
            FM 0 counter:  0
            FM 0 counter:  0
            […]
```

Session 4 – DPAA debug

Check sysfs for Fman Rx port statistics

```
root@p2041rdb:~# ls /sys/devices/ffe000000.soc/ffe400000.fman/fff488000.port/statistics/*
/* FM1@DTSEC1 */
/sys/devices/ffe000000.soc/ffe400000.fman/ffe488000.port/statistics/port_dealloc_buf
/sys/devices/ffe000000.soc/ffe400000.fman/ffe488000.port/statistics/port_discard_frame
/sys/devices/ffe000000.soc/ffe400000.fman/ffe488000.port/statistics/port_enq_total
/sys/devices/ffe000000.soc/ffe400000.fman/ffe488000.port/statistics/port_frame
/sys/devices/ffe000000.soc/ffe400000.fman/ffe488000.port/statistics/port_rx_bad_frame
/sys/devices/ffe000000.soc/ffe400000.fman/ffe488000.port/statistics/port_rx_filter_frame
/sys/devices/ffe000000.soc/ffe400000.fman/ffe488000.port/statistics/port_rx_large_frame
/sys/devices/ffe000000.soc/ffe400000.fman/ffe488000.port/statistics/port_rx_out_of_buffers_di
  scard
```

Session 4 – DPAA debug

Check sysfs for Fman Rx port statistics

```
root@p2041rdb:~# cat /sys/devices/ffe000000.soc/ffe400000.fman/ffe488000.port/statistics/*
            FM 0 Port 0 counter:  0
            FM 0 Port 0 counter:  0
            FM 0 Port 0 counter:  1389
            FM 0 Port 0 counter:  23638
            FM 0 Port 0 counter:  0
            FM 0 Port 0 counter:  0
            FM 0 Port 0 counter:  0
            FM 0 Port 0 counter:  0
```

Session 4 – DPAA debug

Check sysfs for Fman Tx port statistics

```
root@p2041rdb:~# ls /sys/devices/ffe000000.soc/ffe400000.fman/fff4a8000.port/statistics/*
/* FM1@DTSEC1 */
/sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/port_dealloc_buf
/sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/port_deq_confirm
/sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/port_deq_from_default
/sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/port_deq_total
/sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/port_discard_frame
/sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/port_enq_total
/sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/port_frame
/sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/port_length_error
/sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/port_unsupported_format
```

Session 4 – DPAA debug

Check sysfs for Fman Tx port statistics

```
root@p2041rdb:~# cat /sys/devices/ffe000000.soc/ffe400000.fman/ffe4a8000.port/statistics/*
            FM 0 Port 0 counter:  0
            FM 0 Port 0 counter:  703
            FM 0 Port 0 counter:  703
            FM 0 Port 0 counter:  703
            FM 0 Port 0 counter:  0
            FM 0 Port 0 counter:  703
            FM 0 Port 0 counter:  22955
            FM 0 Port 0 counter:  0
            FM 0 Port 0 counter:  0
```

Session 4 – DPAA debug

Check debugfs for dpaa_eth port

```
root@p2041rdb:~# mount –t debugfs none /sys/kernel/debug
root@p2041rdb:~# ls /sys/kernel/debug
bdi  caam hid      mmc0    qman      rcu           usb
Bman gpio memblock powerpc qoriq-dgb sched_feature wakeup_sources
root@p2041rdb:~# cat /sys/kernel/debug/powerpc/fsl_dpa/eth0
DPA counters for fm1-gb0:
[…]
Device congestion stats:
[…]
DPA RX Errors:
[…]
DPA ERN counters:
```

Session 4 – DPAA debug

Check debugfs for Qman

```
root@p2041rdb:~# cd /sys/kernel/debugfs/qman

root@p2041rdb: /sys/kernel/debugfs/qman/fqd #  echo 1 > query_fq_fields

root@p2041rdb: /sys/kernel/debugfs/qman/fqd #  cat query_fq_fields
```

Query FQ Programmable Fields Result fqid 0x1
- orprws: 0
- oa: 0
- olws: 0
- cgid: 0
- fq_ctrl: None
- dest_channel: 0
- dest_wq: 0
- ics_cred: 0
- td_mant: 0
- td_exp: 0
- ctx_b: 0
- ctx_a: 0x0
- ctx_a_stash_exclusive: None
- ctx_a_stash_annotation_cl: 0
- ctx_a_stash_data_cl: 0
- ctx_a_stash_context_cl: 0

Session 4 – DPAA debug

Check debugfs for Qman

```
root@p2041rdb:~# cd /sys/kernel/debugfs/qman/fqd

root@p2041rdb: /sys/kernel/debugfs/qman/fqd #  cat summary
```

/ Provides a summary of all the fields in all frame queue descriptors. This is a read only file.*/*

Out of Service count = 32201
Retired count = 0
Tentatively Scheduled count = 566
Truly Scheduled count = 0
Parked count = 0
Active, Active Held or Held Suspended count = 0

Prefer in cache count = 560
Hold active in portal count = 0
Avoid Blocking count = 528
High-priority SFDRs count = 0
CPC Stash Enable count = 0
Context-A stashing count = 528
ORP Enable count = 0
Tail-Drop Enable count = 560

Session 5

USDPAA Scenario

Session 5 - USDPAA Scenario

`reflector` : Packet Flow

1. Host pings a P2041 IP address
   → sends out an `ICMP_Echo` request
2. Switch floods the frame to all P2041 ports
   → frame will be dropped by all but the correct one
3. P2041 receives the frame, swaps the IP and MAC src and dest addresses, and sends it back
   → swap turns frame into `ICMP_Echo` to host
4. Host receives this frame and does `ICMP_Echo_Reply`
5. P2041 receives, swaps and sends this frame back too.
   → swap turns frame into `ICMP_Echo_Reply` to host
6. Host receives this frame and the original ping is now complete
   → the host just "pinged itself" via the P2041

Session 5 - USDPAA Scenario

Setting Up `reflector`

- Software images (besides FMan ucode and u-boot):
  - Device tree : uImage-p2041rdb-usdpaa.dtb
  - Kernel and rootfs built with USDPAA support
- Uses the RGMII and SGMII ports
- Note MAC addresses of P2041 SGMII ports
  - cfr. u-boot envvars **`eth[x]addr`**

Session 5 - USDPAA Scenario

Deploying Device Tree Image

- A different device tree blob image must be used, because some ports must be reserved for USDPAA use, and not configured as Linux network devices

- `=> setenv bootargs root=/dev/ram rw console=ttyS0,115200 usdpaa_mem=256M`
- `=> tftpboot 40000000 uImage-p2041rdb.bin`
- `=> tftpboot 41000000 fsl-image-core-p2041rdb.rootfs.ext2.gz.u-boot`

  `=> tftpboot c00000 uImage-p2041rdb-usdpaa.dtb`

  `=> bootm 40000000 41000000 c00000`

Session 5 - USDPAA Scenario

Reconfigure and Boot

- The `usdpaa_mem=<memory size>` boot argument must be passed to the kernel, so that memory is reserved by the `fsl_usdpaa` driver.

- Boot using a modified command and check if the configure got applied correctly

```
[Linux boots...]
# more /proc/cmdline
root=/dev/ram rw console=ttyS0,115200 usdpaa_mem=256M
```

→ bootargs = OK

```
# ip link | grep fm
```

→ none of ports is assigned for use by Linux, all are reserved for usdpaa

Session 5 - USDPAA Scenario

Testing `reflector` : Network Setup

- Traffic will be injected in FM1-DTSEC1.
  Look up its MAC address :

```
# cd /proc/device-tree; find -name local-mac-address | sort
./soc@ffe000000/fman@400000/ethernet@e0000/local-mac-address // FM1-DTSEC1
./soc@ffe000000/fman@400000/ethernet@e2000/local-mac-address // FM1-DTSEC2
./soc@ffe000000/fman@400000/ethernet@e4000/local-mac-address // FM1-DTSEC3
./soc@ffe000000/fman@400000/ethernet@e6000/local-mac-address // FM1-DTSEC4
./soc@ffe000000/fman@400000/ethernet@e8000/local-mac-address // FM1-DTSEC5
./soc@ffe000000/fman@400000/ethernet@f0000/local-mac-address // FM1-TGEC1

# od -t x1 ./soc\@ffe000000/fman\@400000/ethernet\@e0000/local-mac-address
0000000 00 04 9f 01 70 f7  // MAC address
0000006
```

Session 5 - USDPAA Scenario

Testing `reflector` : Network Setup

- On the Linux host, create alias interface `eth0:1`

```
$ sudo ifconfig eth0:1 192.168.10.1 netmask 255.255.255.0
```

- Add an ARP entry for the P2041ds ingress port

```
$ sudo arp -s 192.168.10.100 00:04:9f:01:70:f7
```

Session 5 - USDPAA Scenario

Testing `reflector` : Usage

```
root@p2041rdb:~# reflector --help
Usage: reflector [OPTION...] [cpu-range]

USDPAA PPAC-based application

  -b, --buffers=x:y:z        Number of buffers to allocate
  -c, --fm-config=FILE       FMC configuration XML file
  -d, --dma-mem=SIZE         Size of DMA region to allocate
  -i, --fm-interfaces=FILE   FMAN interfaces
  -n, --non-interactive      Ignore stdin
  -p, --fm-pcd=FILE          FMC PCD XML file
  cpu-range                  'index' or 'first'..'last'
  -?, --help                 Give this help list
      --usage                Give a short usage message
  -V, --version              Print program version
```

Session 5 - USDPAA Scenario

Testing `reflector` : Startup

- Execute the `/root/SOURCE_THIS` script on the target
  - a PCD configuration and policy will be applied to the hardware by the `fmc` utility
  - the reflector application starts as a single thread on cpu 1

```
# reflector
Found /fsl,dpaa/dpa-fman0-oh@1, Tx Channel = 46, FMAN = 0, Port ID = 1
Found /fsl,dpaa/ethernet@4, Tx Channel = 40, FMAN = 0, Port ID = 0
Found /fsl,dpaa/ethernet@7, Tx Channel = 63, FMAN = 1, Port ID = 2
Found /fsl,dpaa/ethernet@8, Tx Channel = 64, FMAN = 1, Port ID = 3
Found /fsl,dpaa/ethernet@9, Tx Channel = 60, FMAN = 1, Port ID = 0
Configuring for 4 network interfaces
Allocated DMA region size 0x1000000
reflector starting
Released 0 bufs to BPID 7
Released 0 bufs to BPID 8
Released 8192 bufs to BPID 9
Thread uid:0 alive (on cpu 1)
reflector>
```

Session 5 - USDPAA Scenario

Testing `reflector`

- Reflector CLI commands

```
reflector> help
Available commands:
offline rm macs list add help
```

- Threads can be listed/added/removed using the CLI, with the exception of the primary thread on CPU 1

Session 5 - USDPAA Scenario

Testing `reflector`

- Start injecting ICMP traffic :

```
$ ping 192.168.10.100
PING 192.168.10.100 (192.168.10.100) 56(84) bytes of data.
64 bytes from 192.168.10.100: icmp_req=1 ttl=64 time=0.201 ms
[...]
^C
--- 192.168.10.100 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1999ms
rtt min/avg/max/mdev = 0.201/0.278/0.320/0.056 ms

$ sudo ping -f ping 192.168.10.100
PING 192.168.10.100 (192.168.10.100) 56(84) bytes of data.
.
```

Session 6

Initial Board Bringup with CW

Session 6 - Initial Board Bringup with CW

Using a CodeWarrior Bare Board Project

- Assume that the CW PA 10.3 is installed under a linux host, in path /opt/CodeWarrior_PA_10.3
- Start a host shell in Terminal or Terminator console
- Start the CodeWarrior:

```
$ cwide &
```

Session 6 - Initial Board Bringup with CW

Using a CodeWarrior Bare Board Project

- Create a new project :
  - File→CodeWarrior→Bareboard Project Wizard
    - Project name = Bareboard
    - Processor = P2041
    - Accept Debug Target Settings and Build Settings
    - Configurations :
      - AMP (One project per core)
      - Core 0
    - Uncheck all Trace Configuration options
    - Finish

Session 6 - Initial Board Bringup with CW

Using a CodeWarrior Bare Board Project

- Show the C/C++ perspective from the top right tab
- Right-click the Bareboard-core0 project to open its context menu [Bareboard-core0 context]:
  - Build Project

    ... when complete, an additional project branch is shown :

    Binaries → Bareboard-core0.elf
- Since the platform is not configured, a RCW must be provided by the debugger
- Window → Show View → Remote Systems
  - Open Properties of Bareboard-core0_RAM_Download Target

Session 6 - Initial Board Bringup with CW

Using a CodeWarrior Bare Board Project

- Edit the Target Type:
  - Import from File System :

    `/opt/CodeWarrior_PA_10.3/PA_Support/\`
    `Initialization_Files/jtag_chains/P2041RDB_RCW_1500-750-1333.txt`
  - Set Target Type = `P2041RDB_RCW_1500-750-1333.txt`
- In the Initialization Tab:
  - Check : Execute target reset (applies to...)
  - Uncheck for all cores: Core reset / Run out of reset
  - Check for `e500mc-0` only : Initialize target
    - Click on : Initialize target script field
    - Select : Workspace → Bareboard-core0 → CFG → `P2041RDB_init_core.tcl`

Session 6 - Initial Board Bringup with CW

Using a CodeWarrior Bare Board Project

- Close the window and return to the perspective
- Hit F11 to start the debug session
  - Follow startup in an Eclipse Console view
- After applying the RCW and running the target initialization file, the debugger will halt in the `main()` function
- You can exercise the debugger from here :
  - Resume, Suspend, show registers and variables, look at code disassembly, ...
- Now the board must be prepared for booting autonomously from NOR flash :
  - requires a good RCW, `u-boot` and FMan ucode

Session 6 - Initial Board Bringup with CW

Using a CodeWarrior Bare Board Project

- Suspend your active debug session
- Hit the Flash programmer button
  - Select Flash File To Target
    - Flash Configuration File : `P2041RDB_NOR_FLASH.xml`
    - File: `/tftpboot/p2041rdb/rcw/RR_PX_0x09/rcw_14g_1500mhz.bin`
    - Offset : `0xe8000000`
    - Hit Erase and Program
      - Save to framework only
  - Repeat for :
    - `/tftpboot/p2041rdb/fsl_fman_ucode_p2041_106_1_4.bin @ ef000000`
    - `/tftpboot/p2041rdb/u-boot-P2041RDB.bin @ eff80000`

Session 6 - Initial Board Bringup with CW

Using a CodeWarrior Bare Board Project

- Terminate the debug session
- Return to the `teraterm` window
- Push the reset button on the board
  - `u-boot` comes up


freescale™