May 1 2020

Enable 8 Port Ethernet with Linux in J7 EVM

Overview:

This is a document to introduce based on SDK6.02 software environment, how to enable 8 port(4xSGMIIi+4xRGMII) Eth interface with Linux in J7 EVM kit.

Software Requirement:	Hardware Requirement:
psdk_rtos_auto_j7_06_02_00_21	J7 Beta board
ti-processor-sdk-linux-automotive-j7-evm-06_02_00	GESI Expansion Board
	QP Ethernet Expansion Board

You need to make a sd card based on PSDKLA6.02 first, and there is one thing need to highlight is GESI board is supported by default 6.02 SDK, so the mainly work in this document is try to enable QSGMII in current SDK version, and the mainly step can be divided into three part:

1. Modify DTB file in Linux to disable PCIE0 serdes0.
2. Enable QSGMII in Ethfw.
3. Modify bootcmd in Uboot.

The detail shown as below:

Disable PCIE0 serdes0:

Apply patch in below link for dtb file to disable PCIE0 serdes0, or you can just replace the dtb file in sd_card/rootfs/boot with provided dtb files. After this, you can use lspci CMD to confirm the PCIE0 is disable in console1. And at current system, the 4 port Ethernet interface in GESI board is normally working, you can check this in console3, as below picture shown.

Patch, DTB file pls find in attachment

Console 3:

```
Enabling clocks for CPSW_9G!
=====================================================
        CPSW Ethernet Firmware Demo
=====================================================
ETHFW Version: 0. 1. 1
ETHFW Build Date (YYYY/MMM/DD):2020/Apr/27
ETHFW Commit SHA:ETHFW PermissionFlag:0x1ffffff, UART Connected:true,UART Id:2IPC_echo_test (core : mcu2_0) .....
CPSW_9G Test on MAIN NAVSS
Remote demo device (core : mcu2_0) .....
CpswPhy_bindDriver: PHY 12: OUI:080028 Model:23 Ver:01 <-> 'dp83867' : OK
CpswPhy_bindDriver: PHY 0: OUI:080028 Model:23 Ver:01 <-> 'dp83867' : OK
CpswPhy_bindDriver: PHY 3: OUI:080028 Model:23 Ver:01 <-> 'dp83867' : OK
CpswPhy_bindDriver: PHY 15: OUI:080028 Model:23 Ver:01 <-> 'dp83867' : OK
CpswPhy_bindDriver: PHY 16: OUI:0001c1 Model:27 Ver:00 <-> 'vsc8514' : OK
CpswPhy_bindDriver: PHY 17: OUI:0001c1 Model:27 Ver:00 <-> 'vsc8514' : OK
CpswPhy_bindDriver: PHY 18: OUI:0001c1 Model:27 Ver:00 <-> 'vsc8514' : OK
CpswPhy_bindDriver: PHY 19: OUI:0001c1 Model:27 Ver:00 <-> 'vsc8514' : OK
PHY 0 is alive
PHY 3 is alive
PHY 12 is alive
PHY 15 is alive
PHY 16 is alive
PHY 17 is alive
PHY 18 is alive
PHY 19 is alive
PHY 23 is alive
Host MAC address: 70:ff:76:1d:94:e3
[NIMU_NDK] CPSW has been started successfully
Cpsw_handleLinkUp: port 0: Link up: 1-Gbps Full-Duplex
Cpsw_handleLinkUp: port 2: Link up: 1-Gbps Full-Duplex
```

Console 1:

```
root@j7-evm:~# lspci
0000:00:00.0 PCI bridge: Texas Instruments Device b00d
0001:00:00.0 PCI bridge: Texas Instruments Device b00d
```

Enable QSGMII in Ethfw:

You need to make sure the below two action has been completed.

a. comment below in pdk\packages\ti\drv\cpsw\cpsw_component.mk
   #CPSW_CFLAGS += -DSDK_6_2_CORE_SDK_IMAGE
b. Add below in ethfw\apps\app_remoteswitchcfg_server\mcu_2_0\main_tirtos.c
   #define ENABLE_QSGMII_PORTS

After complete above change, you need to make a new app_remoteswitchcfg_server.xer5f and replace the default firmware in sd_card/rootfs/lib/firmware. Or you can download the firmware after changed in below link:

Ethfw pls find in attachment.

Modify Uboot:

You need to apply the below patch for SBL and generate a new tiboot3.bin file, and replace it in sd_card/boot partition, then set below CMD in uboot while SD boot Linux system. Or you can download the update tiboot3.bin directly.

Patch, Uboot3.bin pls find in attachment.

You need to set below command in uboot and make sure the load_ethfw and bootcmd are same as shown.

```
=> rproc init
=> load mmc 1:2 0x82000000 /lib/firmware/app_remoteswitchcfg_server.xer5f
=> rproc load 2 0x82000000 ${filesize}
=> rproc init 2
=> rproc start 2

=> setenv load_ethfw 'load mmc 1:2 0x82000000 /lib/firmware/app_remoteswitchcfg_server.xer5f;rproc init;rproc load 2 0x82000000 ${filesize};rproc start 2'
=> saveenv
=> printenv load_ethfw
load_ethfw=load mmc 1:2 0x82000000 /lib/firmware/app_remoteswitchcfg_server.xer5f;rproc init;rproc load 2 0x82000000 ${filesize};rproc start 2


=> setenv bootcmd 'run findfdt;setenv mmcdev 1;run init_${boot};run load_ethfw;run get_kern_${boot}; run get_fdt_${boot}; run get_overlay_${boot}; run run_kern'
=> saveenv
=> print bootcmd
bootcmd=run findfdt;setenv mmcdev 1;run init_${boot};run load_ethfw;run get_kern_${boot}; run get_fdt_${boot}; run get_overlay_${boot}; run run_kern
```

Test:

After all the above change, the QSGMII 4 port can worked with GESI 4 ethernet port whileLinux system running. Shown as below:

Console 3:

```
CPSW NIMU application, IP address I/F 1: 10.85.130.81

Rx Flow for Software Inter-VLAN Routing is up
Cpsw_handleLinkDown: port 0: Link down
Cpsw_handleLinkUp: port 2: Link up: 1-Gbps Full-Duplex
Cpsw_handleLinkDown: port 2: Link down
Cpsw_handleLinkUp: port 3: Link up: 1-Gbps Full-Duplex
Cpsw_handleLinkDown: port 3: Link down
CpswMacPort_enablePort: SGMII Link Parter Config port 6: Link Up: 1-Gbps Full-Duplex
Cpsw_handleLinkUp: port 6: Link up: 1-Gbps Full-Duplex
Cpsw_handleLinkDown: port 6: Link down
CpswMacPort_enablePort: SGMII Link Parter Config port 4: Link Up: 1-Gbps Full-Duplex
Cpsw_handleLinkUp: port 4: Link up: 1-Gbps Full-Duplex
Cpsw_handleLinkDown: port 4: Link down
CpswMacPort_enablePort: SGMII Link Parter Config port 5: Link Up: 1-Gbps Full-Duplex
Cpsw_handleLinkUp: port 5: Link up: 1-Gbps Full-Duplex
Cpsw_handleLinkDown: port 5: Link down
CpswMacPort_enablePort: SGMII Link Parter Config port 1: Link Up: 1-Gbps Full-Duplex
Cpsw_handleLinkUp: port 1: Link up: 1-Gbps Full-Duplex
Cpsw_handleLinkDown: port 1: Link down
Cpsw_handleLinkUp: port 7: Link up: 1-Gbps Full-Duplex
Cpsw_handleLinkDown: port 7: Link down
CpswMacPort_enablePort: SGMII Link Parter Config port 6: Link Up: 1-Gbps Full-Duplex
Cpsw_handleLinkUp: port 6: Link up: 1-Gbps Full-Duplex
```

Console 1:

```
j7-evm login: root
root@j7-evm:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 50:51:A9:71:DC:93
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

eth1      Link encap:Ethernet  HWaddr 70:FF:76:1D:94:E2
          inet addr:10.85.130.121  Bcast:10.85.130.255  Mask:255.255.255.0
          inet6 addr: fe80::72ff:76ff:fe1d:94e2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5 errors:0 dropped:0 overruns:0 frame:0
          TX packets:54 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1148 (1.1 KiB)  TX bytes:8712 (8.5 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:140 (140.0 B)  TX bytes:140 (140.0 B)

root@j7-evm:~# ping 10.85.130.[redacted]
PING 10.85.130.169 (10.85.130.[redacted]): 56 data bytes
64 bytes from 10.85.130.[redacted]: seq=0 ttl=64 time=0.497 ms
64 bytes from 10.85.130.[redacted]: seq=1 ttl=64 time=0.255 ms
64 bytes from 10.85.130.[redacted]: seq=2 ttl=64 time=0.424 ms
64 bytes from 10.85.130.[redacted]: seq=3 ttl=64 time=0.347 ms
```