Learn The Steps Required to Configure NXP LayerScape SOCs to Boot From eSDHC

Jimmy Zhao PE, System & Application Engineering

6/2018

NXP

SECURE CONNECTIONS FOR A SMARTER WORLD

Agenda

- eSDHC boot basics for LayerScape (LS) SoCs
- Using LSDK pre-built images
- How to change RCW/PBI for SD boot
- How to change the DTS (Device Tree Structure) files
- How to build the u-boot image after code changes
- How to flash a SD card or an eMMC
- Common questions about eSDHC boot

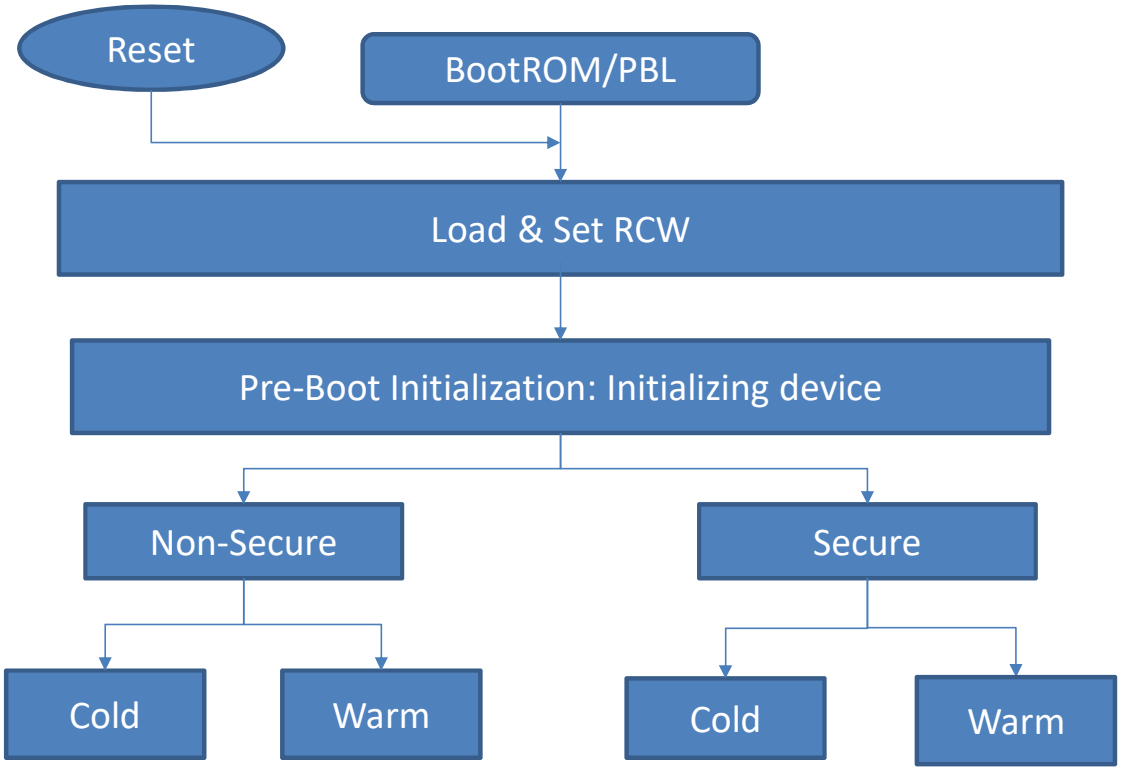
ESDHC BOOT BASICS FOR LAYERSCAPE PRODUCTS

Boot from eSDHC for LayerScape SoCs

- Pre-Boot Loader (PBL)
  Hardware state machine
  - LS1021A
  - LS1043A
  - LS1046A
- Service Processor (BootRom)
  ROM firmware
  - LS108xA
  - LS208xA
  - LX2160A
  - LS1028A
  - Future…??

SOCs	FlexSPI/ QSPI NAND	FlexSPI/ QSPI NOR	eSDHC (SD/eMMC)	I2C	IFC NAND	IFC NOR
LS1012A	√	√				
LS1021A/20A/22A	√	√	√		√	√
LS1043A/23A	√	√	√		√	√
LS1046A/26A	√	√	√		√	√
LS1088A/84A	√	√	√	√	√	√
LS2088A/48A	√	√			√	√
LS1028A/27A	√	√	√	√		
LX2160	√	√	√	√		

Initial Boot Flow



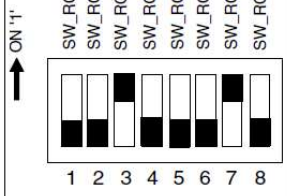
Initial SD Boot Flow

- RCW Phase
  - Load RCW to OCRAM (On-Chip RAM) **using 1-bit mode**
  - Offset for SD or eMMC is 0x1000 (FAT file system)
  - Check Valid Preamble/Load RCW command. Flags an error if not found
- Pre-Boot Initialization (PBI) (Optional)
  - Load PBI data to OCRAM **using 1-bit mode**
  - PBI data have to be both in the SD card or eMMC the same location as the RCW
- Load boot loader image (u-boot)
- Release Core
  - U-boot code configures the SDHC for 4-bit/8-bit mode and the interface speed

LayerScape Development Boards

- Support booting from a SD card
  - LS1021ATWR
  - LS1043ARDB
- Support booting from an eMMC or SD card
  - LS1046ARDB (see table below for SW setting)
  - LS1088ARDB Rev D (Common board design LS1043A is supported)
  - LX2160ARDB
  - LS1028ARDB
- Booting the Board
  - Set proper SW settings

Table 8. DIP switch settings (continued)

Switch figure	Switch	Name	Description
	SW5[1-8] SW4[1]	RCW_SRC[0-7] RCW_SRC8	RCW_SRC[0:8] select • 0010_0000_0: SDHC/eMMC • 0010_0010_0: QSPI (default value) • 0100_1XXX_X: Hard-coded RCW **NOTE:** The RCW_SRC field (9 bits) is spread over SW4 and SW5. **NOTE:** If you want to boot from eMMC, program a bootable image on the eMMC flash. When you boot from eMMC, you cannot insert an SD card. If you want to boot from an SD card, insert a bootable SD card. When an SD card is inserted, eMMC will be disabled.

USING LSDK PRE-BUILT IMAGES

Using LSDK Pre-built Images

- Download pre-built image
  - wget http://www.nxp.com/lgfiles/sdk/lsdk1803/firmware_ls1046ardb_uboot_sdboot.img
- Flash the SD card
  - Option 1: flex-installer -f firmware_ls1046ardb_uboot_sdboot.img -s 8 -d /dev/sdx
  - Option 2*: dd if=* firmware_ls1046ardb_uboot_sdboot.img *of=/dev/sdx seek=8 bs=512*
- Booting the Board
  - Set proper Switch “SW” settings

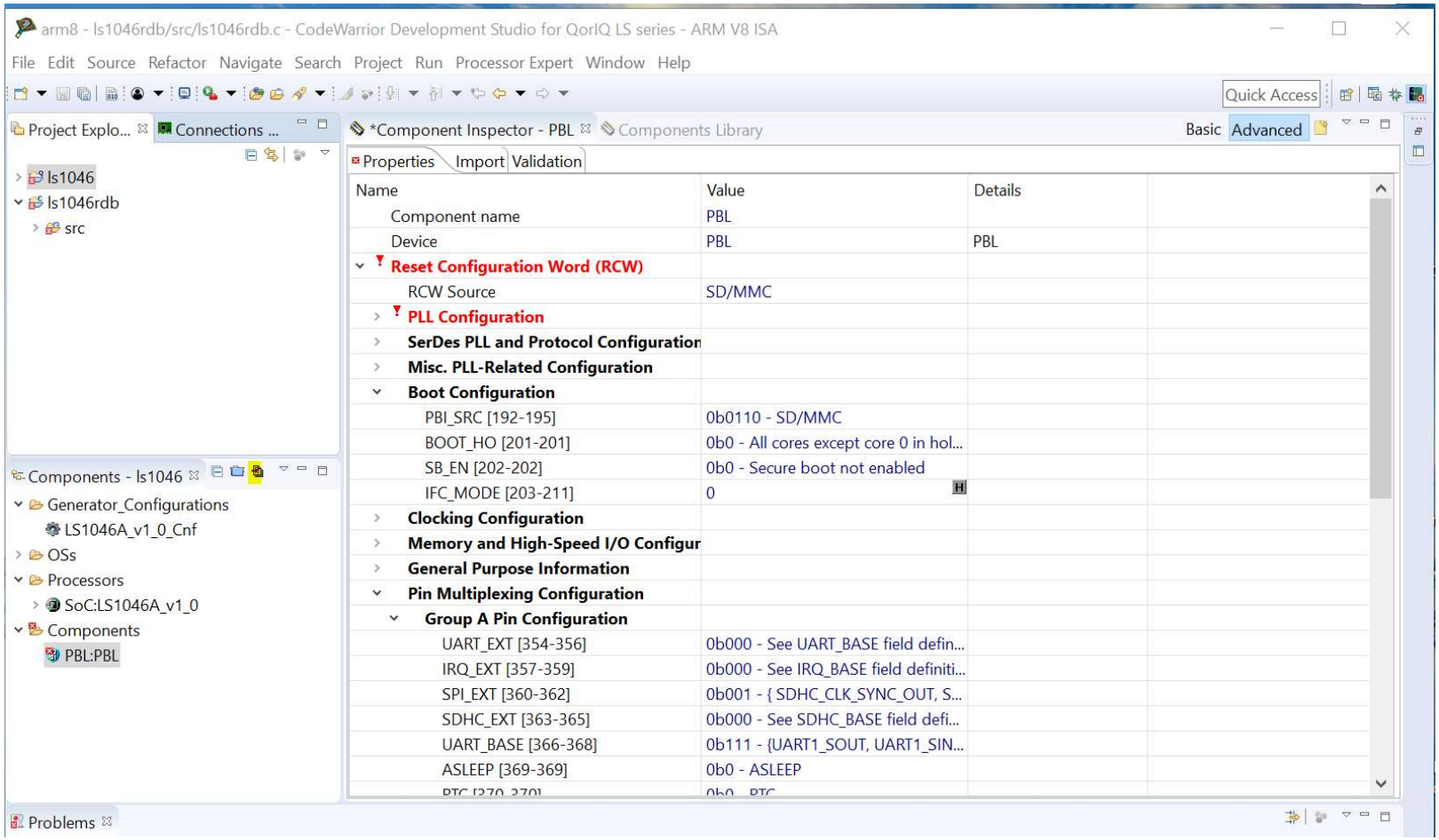
- Useful collaterals for reference
  - Layerscape Software Development(LSDK) Kit Document https://www.nxp.com/docs/en/supporting-information/LSDK-KC-REV18.03.pdf
  - For information on Flexbuilder: README.md under ~/flexbuild

HOW TO CHANGE RCW/PBI FOR SD BOOT

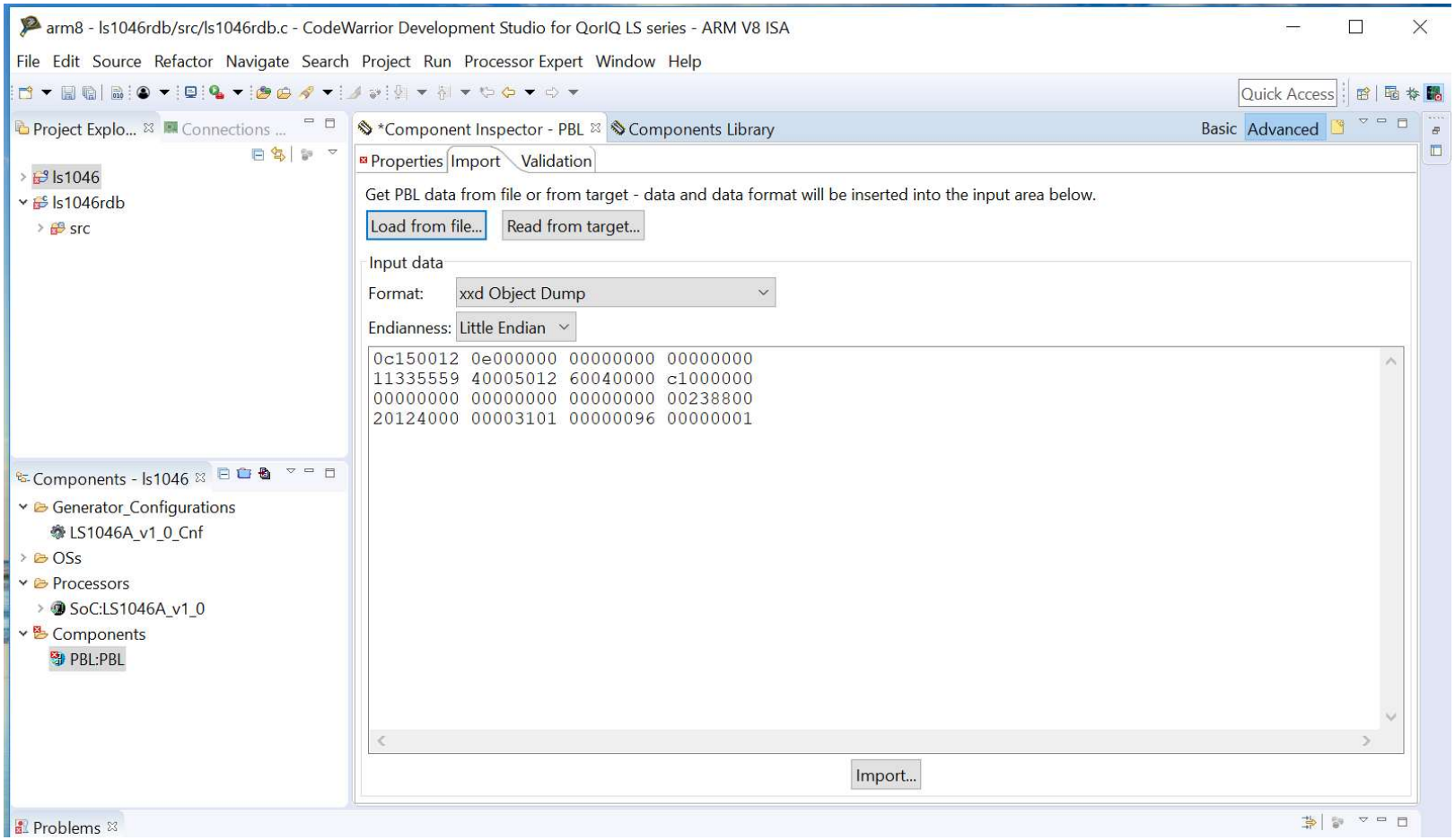
Change RCW/PBI for SD boot

- CodeWarrior
  - QorIQ Configuration and Validation Suite (QCVS)
- RCW/PBI is part of u-boot source code
  - PBL based only
- RCW is a separate source code
  - BootROM based

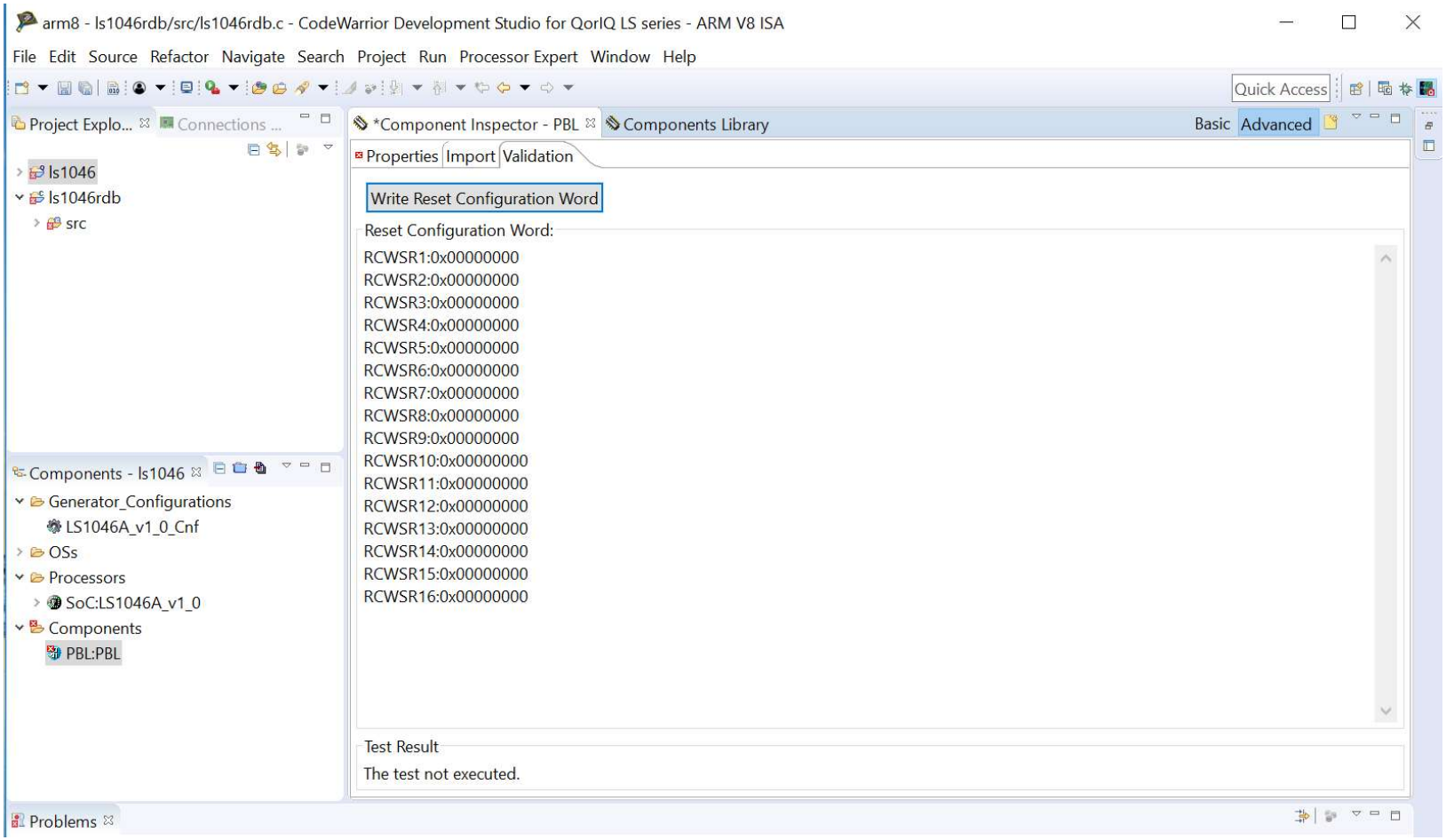
Change RCW/PBI for SD boot(QCVS)



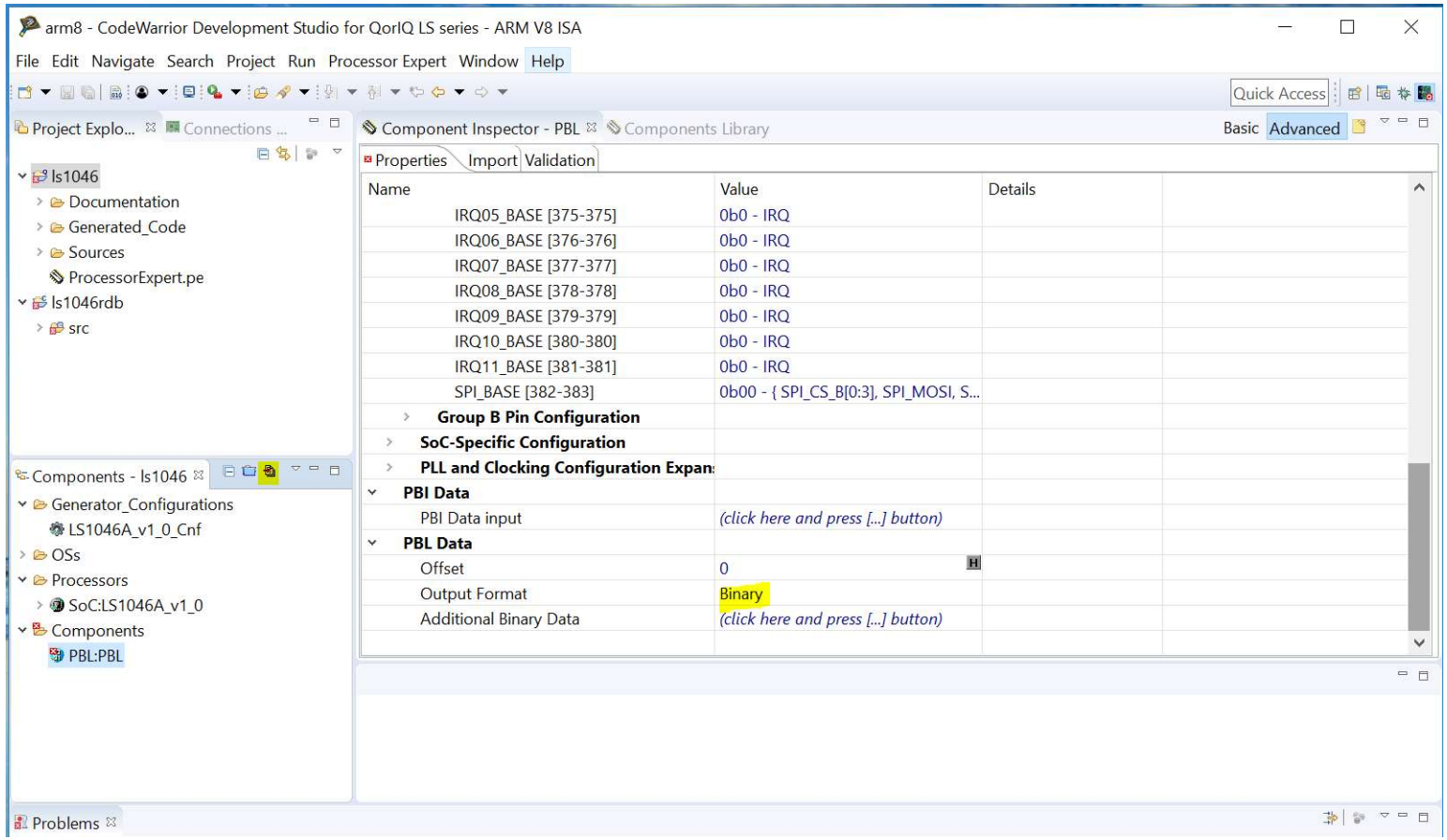
Change RCW/PBI for SD boot(QCVS)



Change RCW/PBI for SD boot(QCVS)



Change RCW/PBI for SD boot(QCVS)



Change RCW/PBI for SD boot (PBL)

- RCW source file
  - Link: https://lsdk.github.io/components.html
  - ~/flexbuild/packages/firmware/u-boot/board/freescale/ls**** directory
    - ls1046ardb_rcw_sd.cfg
    - ls1046ardb_rcw_emmc.cfg
  - Differences between these two files

b08938@apps-r620-2:~/qoriq/lsdk1803/u-boot

```
#PBL preamble and RCW header
aa55aa55 01ee0100
# RCW
0c150012 0e000000 00000000 00000000
11335559 40005012 60040000 c1000000
00000000 00000000 00000000 00238800
20124000 00003101 00000096 00000001
~
```

Differences	RCW	SD		eMMC	
EVDD_VSEL	439-440	3.3 V	0b10	1.8 V	0b00
IIC2_EXT	445-447	SDHC_CD/SDHC_WP	0b001	IIC2_SCL/IIC2_SDA	0b000

Change RCW/PBI for SD boot (PBL)

- PBI source file
  - ~/flexbuild/packages/firmware/u-boot/board/freescale/ls**** directory
  - ls1046ardb_pbi.cfg

```
#Configure Scratch register
09570600 00000000
09570604 10000000
#Disable CCI barrier tranaction
09570178 0000e010
09180000 00000008
#USB PHY frequency sel
09570418 0000009e
0957041c 0000009e
09570420 0000009e
#Serdes SATA
09eb1300 80104e20
09eb08dc 00502880
#PEX gen3 link
09570158 00000300
89400890 01048000
89500890 01048000
89600890 01048000
#Alt base register
09570158 00001000
#flush PBI data
096100c0 000fffff
```

<= A-010554 workaround For SATA
<= 6 Gbaud configuration

Steps to Build u-boot Image for SD boot (PBL)

- Use flexbuild
  - For SD: *flex-builder -c uboot -m ls1046ardb -s sd*
  - For eMMC: *flex-builder -c uboot -m ls1046ardb -s emmc*
  - Image is ./build/firmware/u-boot/ls1046ardb/uboot_ls1046ardb_sdcard.bin
- Use normal source code
  - Clone the u-boot source code
  - *export ARCH=arm64*
  - *export CROSS_COMPILE=/home/share/gcc-linaro-5.3.1-2016.05-x86_64_aarch64-linux-gnu/bin/aarch64-linux-gnu-*
  - *make distclean*
  - *make ls1046ardb_sdcard_defconfig* or
  - *make ls1046ardb_emmc_defconfig*
  - *make*
  - *Image is* ***u-boot-with-spl-pbl.bin***

Change RCW/PBI for SD boot (BootRom)

- Get RCW source code
  - *git clone https://source.codeaurora.org/external/qoriq/qoriq-components/rcw*
  - *cd rcw*
  - *git checkout LSDK-18.03*
- The some files under ls1088ardb directory
  - *bootlocptr_sdhc.rcw*
  - *bootlocptr_nor.rcw*
  - *bootlocptr_qspi.rcw*
  - *a008822.rcw*
  - *a008851.rcw*
  - *a009102_single.rcw*
  - *a010554_single.rcw*
  - *ls1088rdb.rcwi*
  - *README*
  - *tcpz_nosecure_region.rcw* *(should be tzpc…?)*

Change RCW/PBI for SD boot (BootRom)

- The source code:
  - ./Lsxxxxardb/FCQQQQQQQQ_PPP_H_0x1d_0x0d/rcw_1600_sd.rcw

```
SYS_PLL_RAT=7
MEM_PLL_RAT=21
CGA_PLL1_RAT=16
CGA_PLL2_RAT=16
HWA_CGA_M1_CLK_SEL=2
HWA_CGA_M2_CLK_SEL=1
DDR_REFCLK_SEL=2
DRAM_LAT=1
BOOT_LOC=21
FLASH_MODE=0x2
PBI_LENGTH=0x10
SYSCLK_FREQ=0x258
IIC3_EXT=1
UART_BASE=3
IIC2_BASE=2
IIC3_BASE=1
IIC4_BASE=1
SPI_PCS_BASE=3
```

Change RCW/PBI for SD boot (BootRom)

- The source code (continue):
  - ./ls1088ardb/FCQQQQQQQQ_PPP_H_0x1d_0x0d/rcw_1600_**sd**.rcw

```
IFC_GRP_A_BASE=3
IFC_GRP_FGHI_BASE=1
QSPI_OCT_EN=1
EC1=1
EC2=2
USB1_CLK_FSEL=39
USB2_CLK_FSEL=39
SRDS_PRTCL_S1_LN0=1
SRDS_PRTCL_S1_LN1=1
SRDS_PRTCL_S1_LN2=4
SRDS_PRTCL_S1_LN3=4
SRDS_PRTCL_S2_LN0=5
SRDS_PRTCL_S2_LN1=5
SRDS_PRTCL_S2_LN2=5
SRDS_PRTCL_S2_LN3=9
.pbi
blockcopy 0x40,0x00100000,0x1800a000,0x00015000
.end
```

Build RCW/PBI image for SD boot (BootRom)

- Make RCW binary image
  - *cd ls1088ardb*
  - Type *make* to create the binary image to flash
  - The RCW binary image would be in *FCQQQQQQQQ_PPP_H_0x1d_0x0d/*
- Python file: rcw.py
- README file

Build u-boot Image for SD boot (BootROM)

- Use flexbuild
  - For SD: *flex-builder -c uboot -m ls1088ardb -s sd*
  - For eMMC: *flex-builder -c uboot -m ls1088ardb -s emmc*
  - Image is build/firmware/u-boot/ls1088ardb/uboot_ls1088ardb_sdcard_qspi.bin
- Use normal source code
  - Clone the u-boot source code
  - *export ARCH=arm64*
  - *export CROSS_COMPILE=/home/share/gcc-linaro-5.3.1-2016.05-x86_64_aarch64-linux-gnu/bin/aarch64-linux-gnu-*
  - *make distclean*
  - *make ls1088ardb_sdcard_defconfig*
  - *make*
  - Image is ***u-boot-with-spl.bin***

HOW TO CHANGE THE DTS FILES

DTS files for SD boot

- eSDHC driver (uboot) does not use Driver Model (DM) yet
- DTS file
  - Directory: arch/arm/dts
  - File:
    - fsl-ls1046a.dtsi
    - fsl-ls1046a-rdb.dts

```
/dts-v1/;
/include/ "fsl-ls1046a.dtsi"
/ {
	model = "LS1046A RDB Board";
	aliases {
		spi0 = &qspi;
	};
};

&qspi {
	bus-num = <0>;
	status = "okay";
	qflash0: s25fs512s@0 {
		#address-cells = <1>;
		#size-cells = <1>;
		compatible = "spi-flash";
		spi-max-frequency = <50000000>;
		reg = <0>;
	};

	qflash1: s25fs512s@1 {
		#address-cells = <1>;
		#size-cells = <1>;
		compatible = "spi-flash";
		spi-max-frequency = <50000000>;
		reg = <1>;
	};
};
```

Build u-boot Image Considering DTS

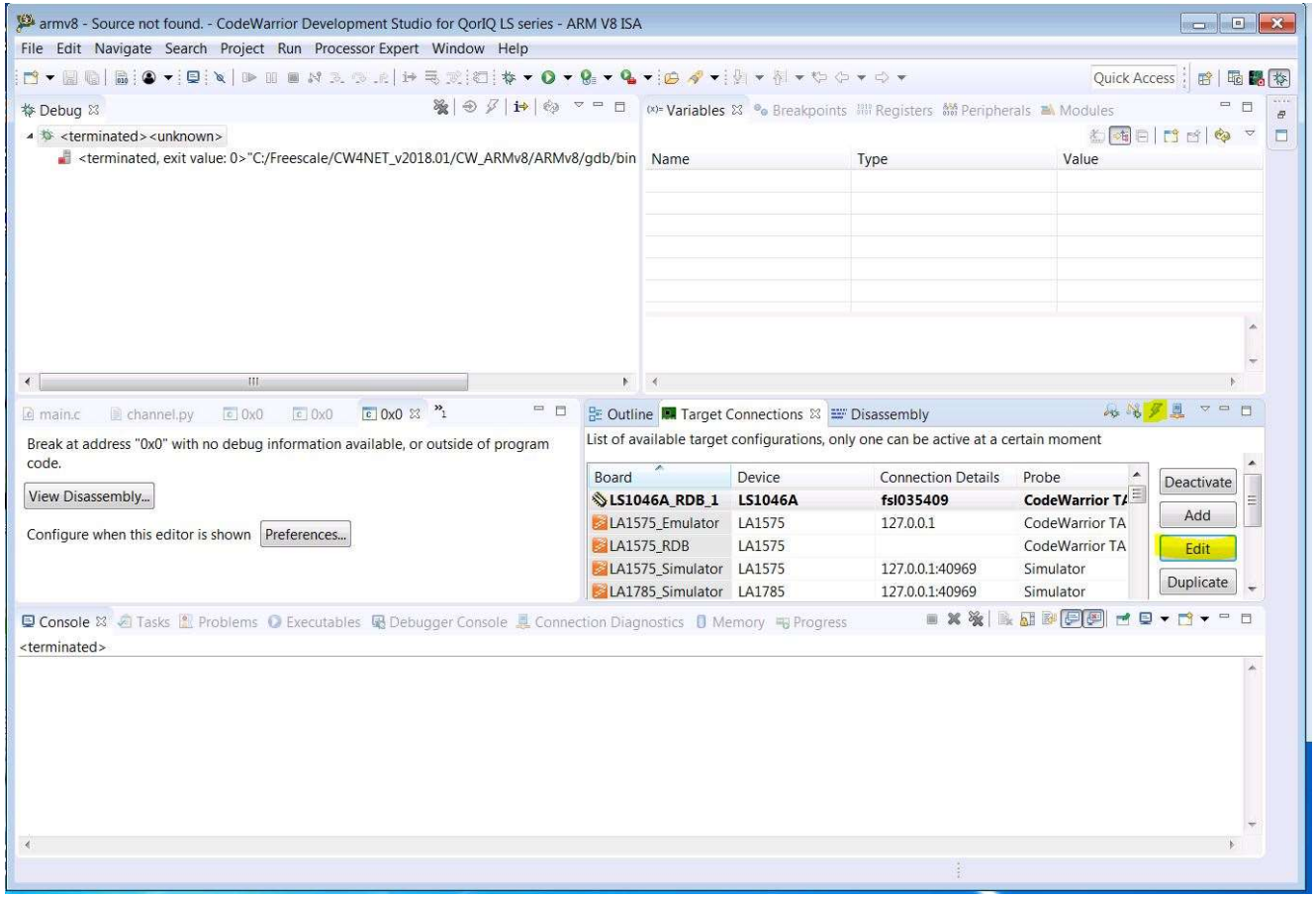
- Use normal source code
  - *export ARCH=arm64*
  - *export CROSS_COMPILE=/home/share/gcc-linaro-5.3.1-2016.05-x86_64_aarch64-linux-gnu/bin/aarch64-linux-gnu-*
  - *make distclean*
  - *make ls1046ardb_sdcard_defconfig* or
  - *make ls1046ardb_emmc_defconfig*
  - ***make menuconfig***
    - Enable Device Tree Support
    - Enable Drivers
  - *make*
  - Image is u-boot-with-spl-pbl.bin

HOW TO PROGRAM SD CARD OR EMMC

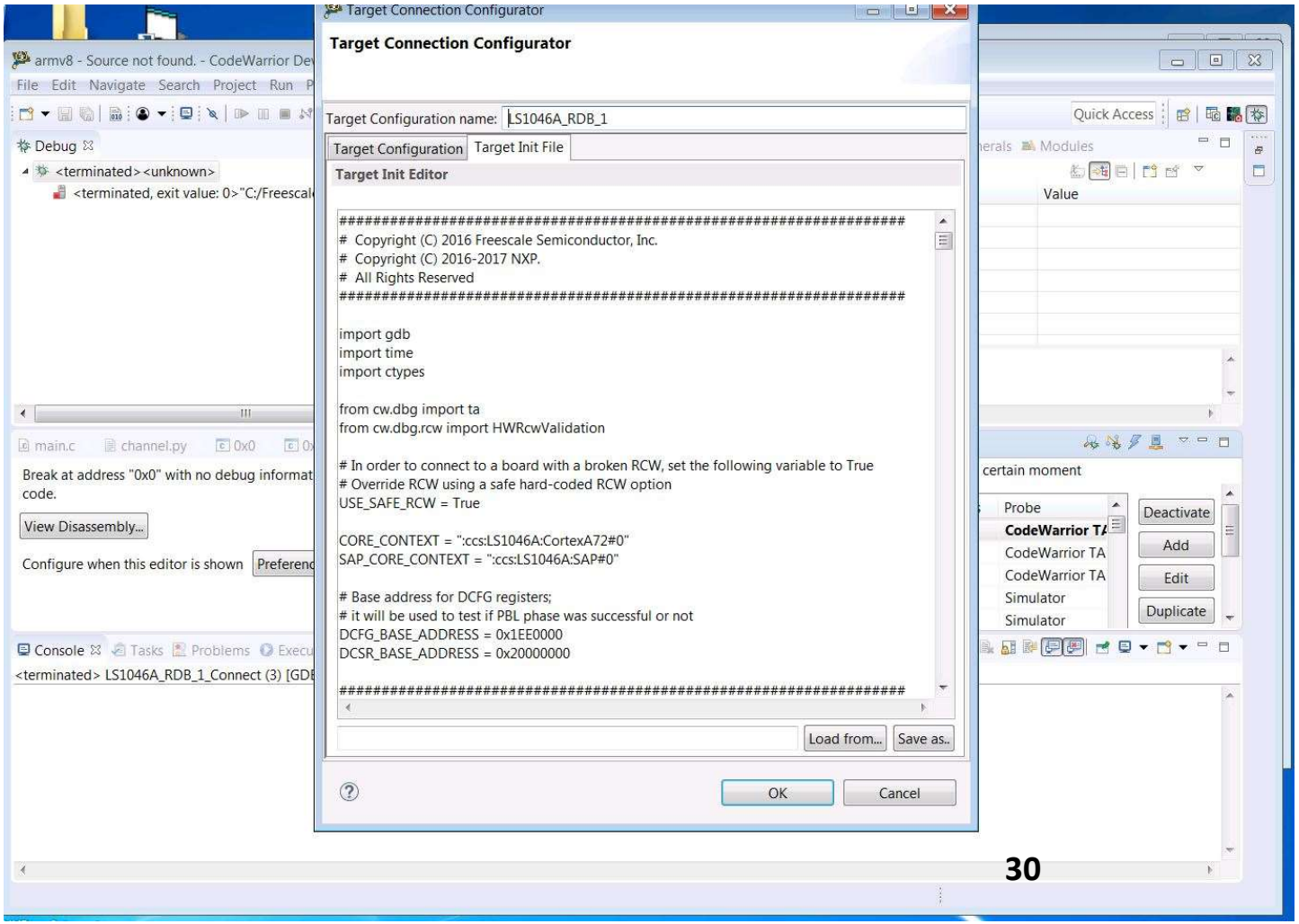
Program SD card or eMMC

- Linux Machine (SD only)
  - dd command
    - *dd if=u-boot-with-spl-pbl.bin of=/dev/sdx seek=8 bs=512*
  - Flexbuild
    - ***Full Image**: flex-installer -f firmware_ls1046ardb_uboot_sdboot.img -s 8 -d /dev/sdx*
    - ***U-boot Only:** flex-installer -f build/firmware/u-boot/ls1046ardb/**uboot_ls1046ardb_sdcard.bin** -s 8 -d /dev/sdx*
- U-boot commands
  - *tftp 0xa0000000 u-boot-with-spl-pbl.bin*
  - *mmcinfo*
  - *mmc write 0xa0000000 8 0x800*
- CodeWarrior
  - Demo

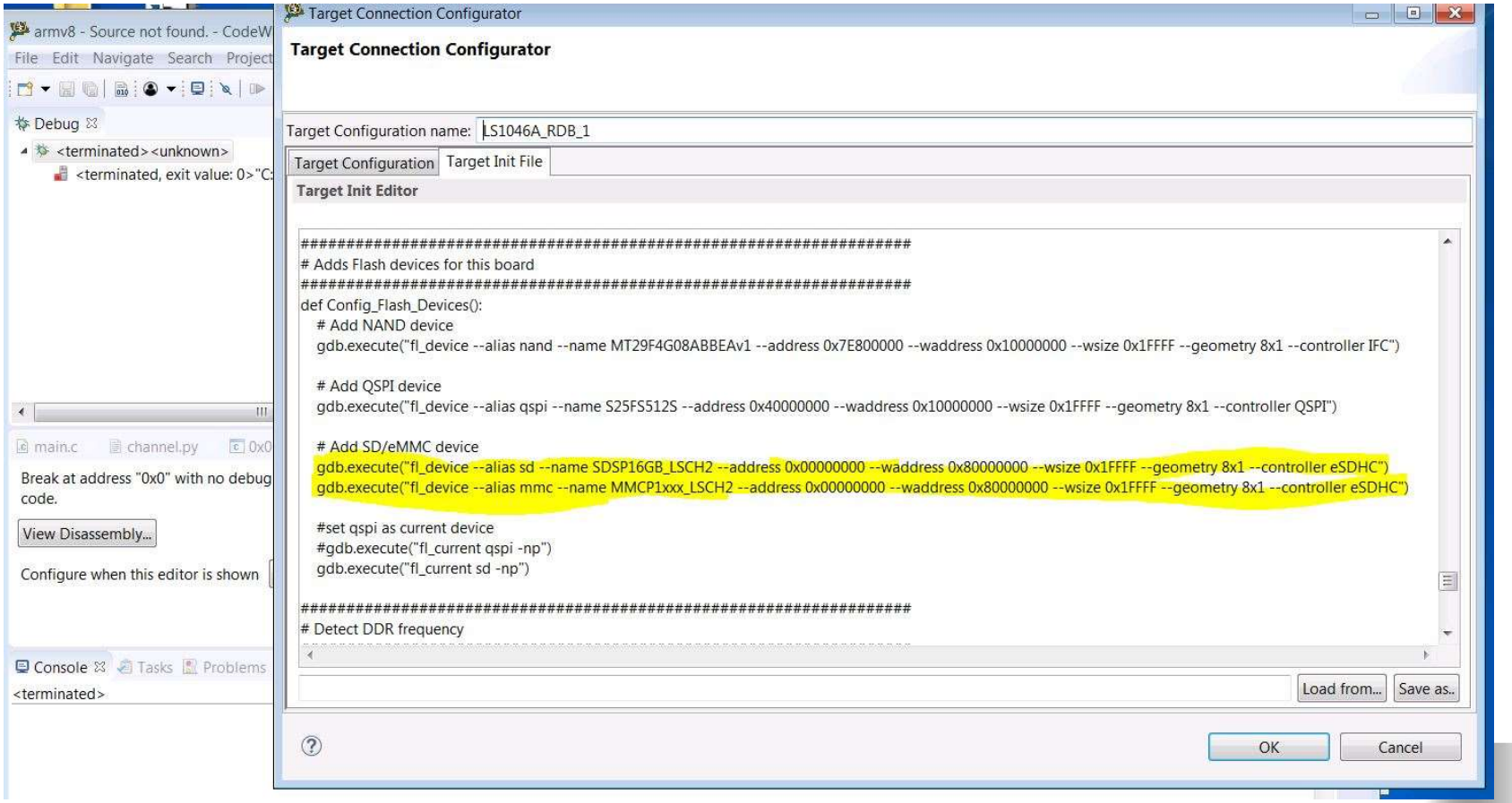
Program SD Card using CodeWarrior



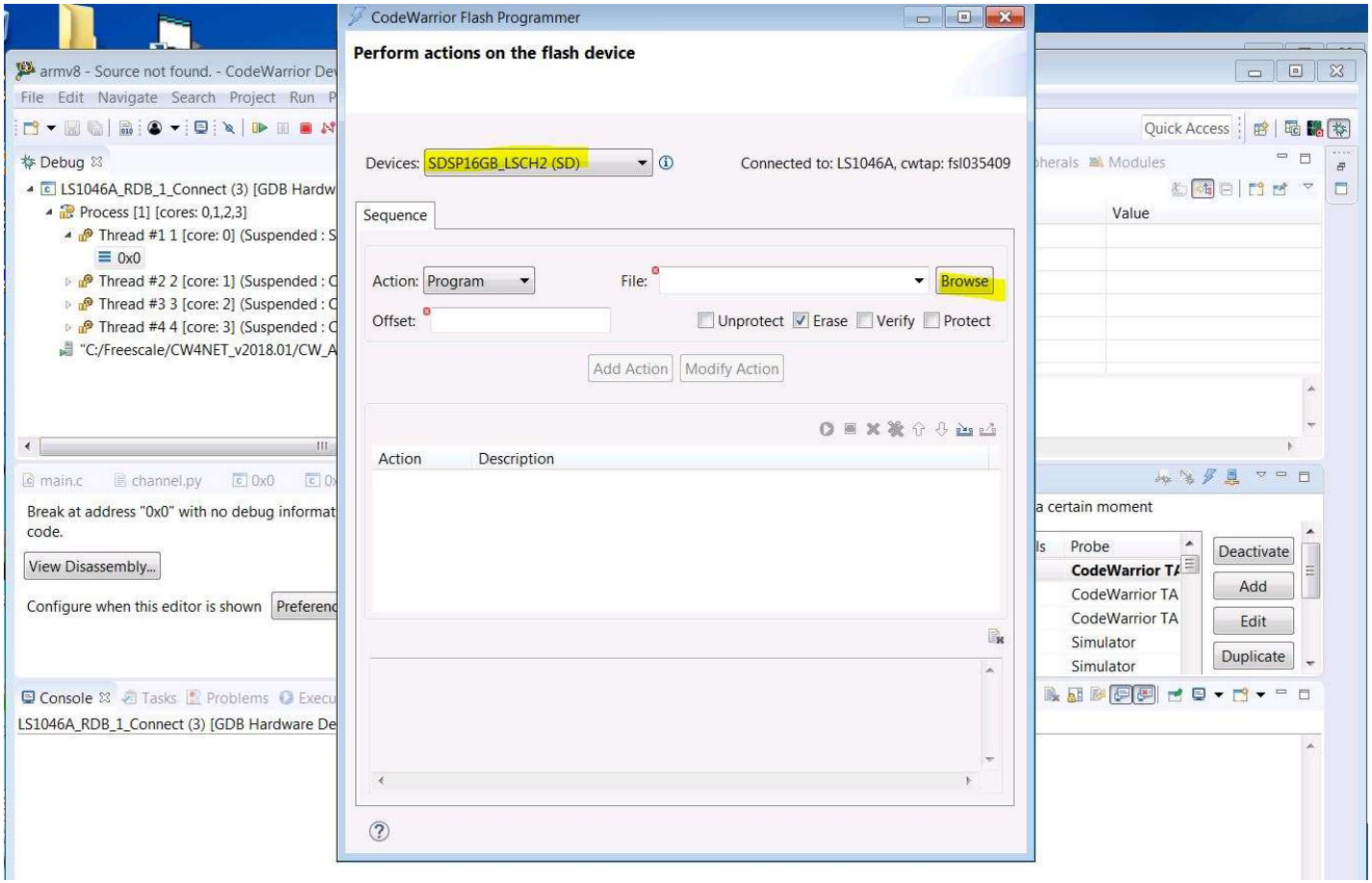
Program SD Card using CodeWarrior



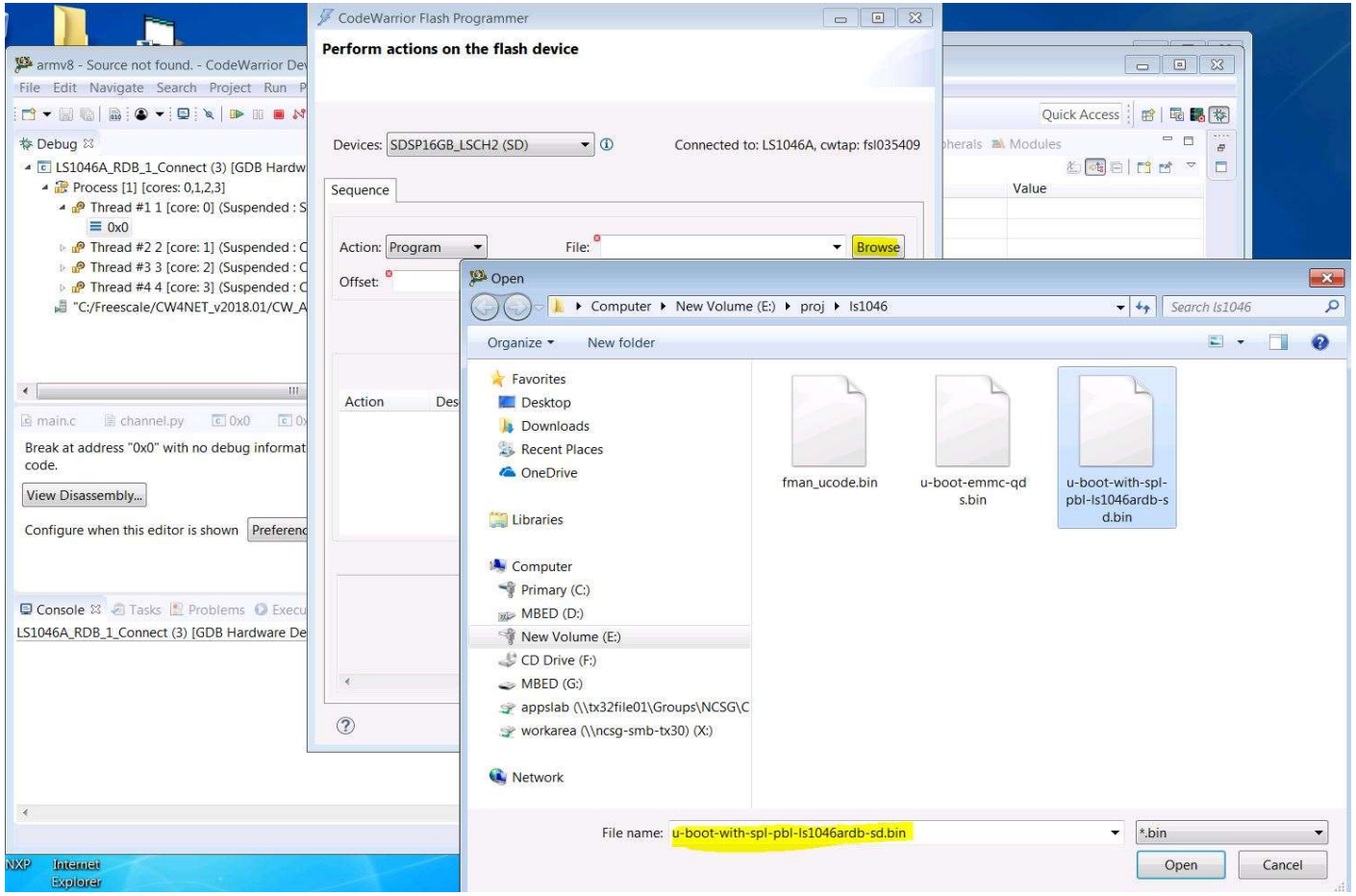
Program SD Card using CodeWarrior



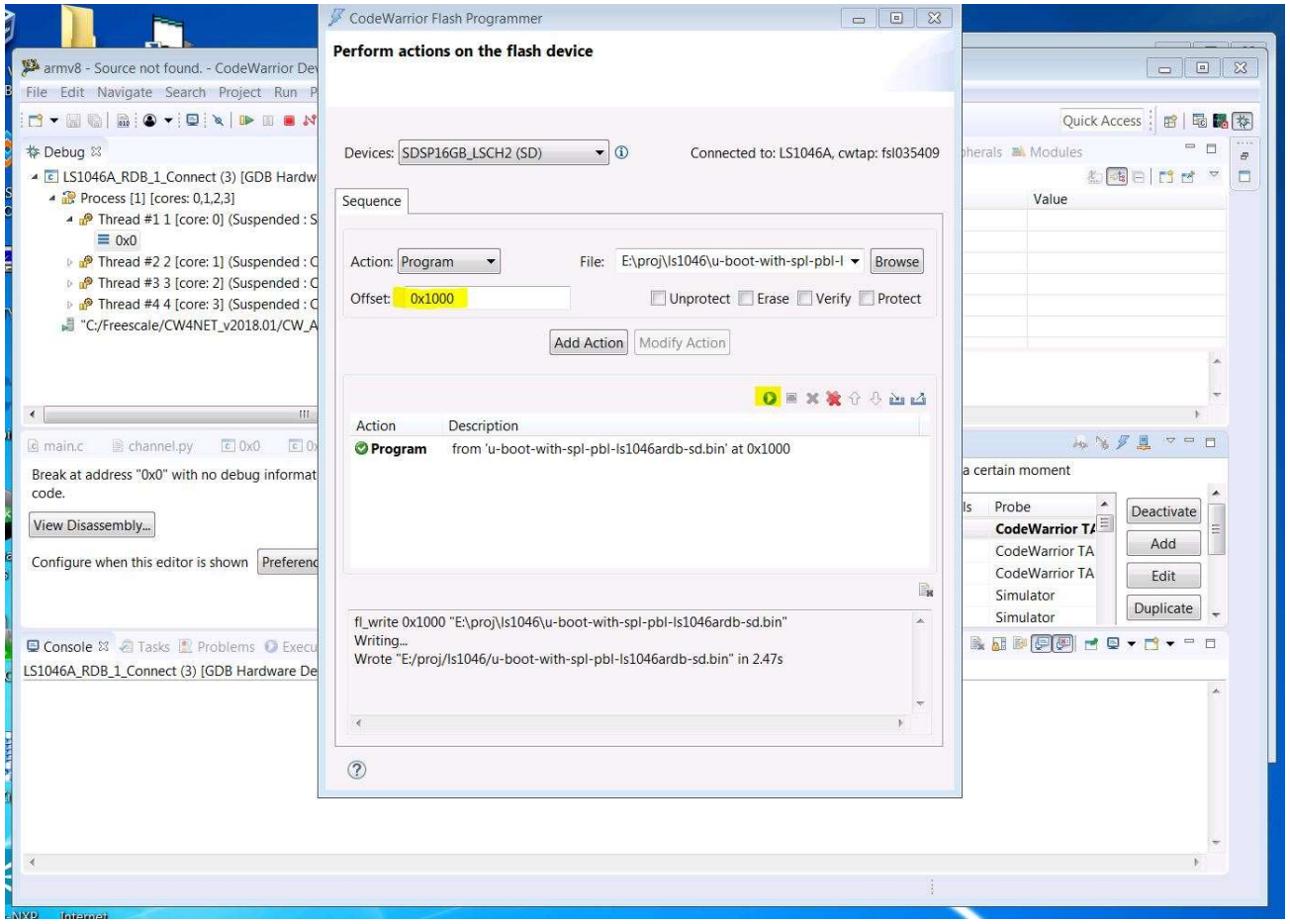
Program SD Card using CodeWarrior



Program SD Card using CodeWarrior



Program SD Card using CodeWarrior



COMMON QUESTIONS ABOUT ESDHC BOOT

Common questions about eSDHC boot

- Which version of SD or eMMC supported
  - SD card/SDIO: Up to SD specification 3.01 (SD, SDHC, SDXC, UHS-I)
  - eMMC: Up to eMMC specification 5.1
- What are the differences for using eMMC vs SD card
  - I/O voltage
    - SD card always starts at 3.3V and switch to 1.8V for SDR or DDR mode
    - eMMC should be 1.8V if HS200/HS400 is supported.
  - Bus Width
    - SD card: 1-bit, 4-bit, (uSD: 8-bit)
    - eMMC: 1-bit, 4-bit, 8-bit
  - Speed Modes
    - SD Card: Default, HS, SDR12, SDR25, SDR50, SDR104, DDR50
    - eMMC: Default, HS, DDR, HS200, HS400
  - Commands
    - SD Card: Table 4-21 to Table 4-30 of SD specification 3.01
    - eMMC: Table 49 - 59 of eMMC specification 5.1
    - SDIO: C.1 of SDIO specification 3.0

Common questions about eSDHC boot

- Does booting from an SD card use 4-bit mode
  - PBL/ROM boots eSDHC in 1-bit mode
- What speed mode is used during eSDHC booting
  - High Speed mode
- When do I need a voltage translator
  - EVDD is equal to 1.8V when SD card is used (3.3V is needed for SD card)
  - EVDD is equal to 1.8V when 3.3V eMMC is used
- When SDHC_CLK_SYNC_xx are needed
  - DDR mode
  - SDR50 Highly recommended (No needed if Fixed tuning is used)

Common questions about eSDHC boot

- Which pin require the pullups
  - SDHC_Datax
  - SDHC_CMD
  - Recommend 10kΩ to 50kΩ for eMMC and 10kΩ to 100kΩ pull-up value for SD
- What steps are needed when booting from eSDHC fails
  - Check cfg_rcw_src termination
  - RCW/boot loader image offset: 8 sections (0x1000 bytes)
  - Check RCW configuration values
  - Check whether "**Power-on reset sequencing**" is followed as outlined in the H/W spec
  - Check if EVDD I/O Voltage level is set properly
  - Check SDHC_CLK setting and clock is present
- Input clock for HS200/HS400/SDR104 mode
  - Peripheral clock (eSDHCCTL[PCS] = 1) must be used

Common questions about eSDHC boot

- How to select eMMC or SD card boot on LS1046ARDB
  - It is done automatically. If no SD card is inserted, LS1046ARDB board will try to boot from eMMC
- SDHC clock changes many times during booting
  - RCW+PBI phase: SYSCLK
  - Load boot loader: Platform Clock

SECURE CONNECTIONS
FOR A SMARTER WORLD