Gen 3 Space Grade Serial Dual QSPI P-SRAM™ Development Kit for Xilinx User Guide

AK3XXG208XILCC

Table 1: Revision History

Revision No.	Date	History
1.0	11/22/2022	Initial Release
1.1	12/15/2022 1/17/2023 3/28/2023 05/19/2023	Updated Development Kit Ordering Info Added information on Direct Load and Boot Using Avalanche ID Updated Avalanche ID Removed Eval Kit's Non-Socketed Ordering Option Added 96-Ball FBGA Socket Ordering Info
1.2	8/9/2023 9/12/2023	Updated instructions for PetaLinux from Installation Updated instructions for PetaLinux 2022.1 and 2022.2 releases

Contents

1. Overview

The Avalanche Serial Dual QSPI P-SRAM™ for Xilinx Versal ACAP VCK190 Development Kit enables users to configure an Avalanche Serial Dual QSPI P-SRAM™ to be a bootable device for a Xilinx Versal ACAP VCK190 FPGA platform. Intended to enable rapid evaluation and prototyping, the Development Kit connects easily to the VCK190 platform via Samtec connectors.

Xilinx's Versal ACAP/FPGA platform requires a complex high density code configuration and storage memory architecture including multiple memory types such as Boot ROM, SRAM and Flash. Each of these memory technologies faces technical challenges in achievable densities, scalability, endurance/reliability and radiation resilience. Avalanche's 3rd Generation (Gen 3) Space Grade Serial Dual Quad Serial P-SRAM™ offers scalable density migration from 1Gb to 8Gb in a common package available today and is designed specifically to address these challenges. As such, this single device replaces ROM, SRAM, Flash and associated support circuitry and is ideal for radiation environments. Leveraging our 22nm high reliability pMTJ STT-MRAM, data is always non-volatile with 10^{16} write cycles endurance and greater than 20-year retention @ 85°C.

The Versal ACAP platform provides six different boot modes options: JTAG, Octal SPI, Quad SPI, SD, eMMC1 and SelectMAP. The boot modes are categorized into master or slave boot modes. For the purpose of this user guide, it will focus on the Quad SPI (QSPI32 option, 4-Byte Address mode) master boot mode function. This Avalanche Development Kit is effectively a simple daughter card containing a custom socket for any of the Serial Dual Quad Serial P-SRAM densities (the MRAMs themselves are orderable separately), which leverage the 96-Ball FBGA device and connects to a Xilinx VCK190 evaluation board via Samtec high speed connector (240-pin, 8x20). Other boot options are outside of the scope of this user guide.

Writing to the Avalanche Serial Dual QSPI P-SRAM™ can be done through Linux U-Boot sf program or other means. This user guide will provide example and steps to follow on how to write to the device through Linux U-Boot.

2. Avalanche P-SRAM™ Product Support & Development Kit Ordering Options

The Avalanche P-SRAM™ daughterboard with a socket can be populated with either one of the following Serial Dual QSPI P-SRAM™ devices:

Table 2: Avalanche Serial Dual QSPI P-SRAM™ Product Support & Development Kit Socketed Ordering Options

Device Part #	Density	Voltage	Organization	Package	Development Kit Socketed Ordering Options
AS301G208-0108X0MCC	1Gb	2.7 V to 3.6V	Dual QSPI	96-Ball FBGA	AK301G208XILCC
AS302G208-0108X0MCC	2Gb	2.7 V to 3.6V	Dual QSPI	96-Ball FBGA	AK302G208XILCC
AS304G208-0108X0MCC	4Gb	2.7 V to 3.6V	Dual QSPI	96-Ball FBGA	AK304G208XILCC
AS308G208-0108X0MCC	8Gb	2.7 V to 3.6V	Dual QSPI	96-Ball FBGA	AK308G208XILCC

3. Ordering Options

3.1 Development Kit

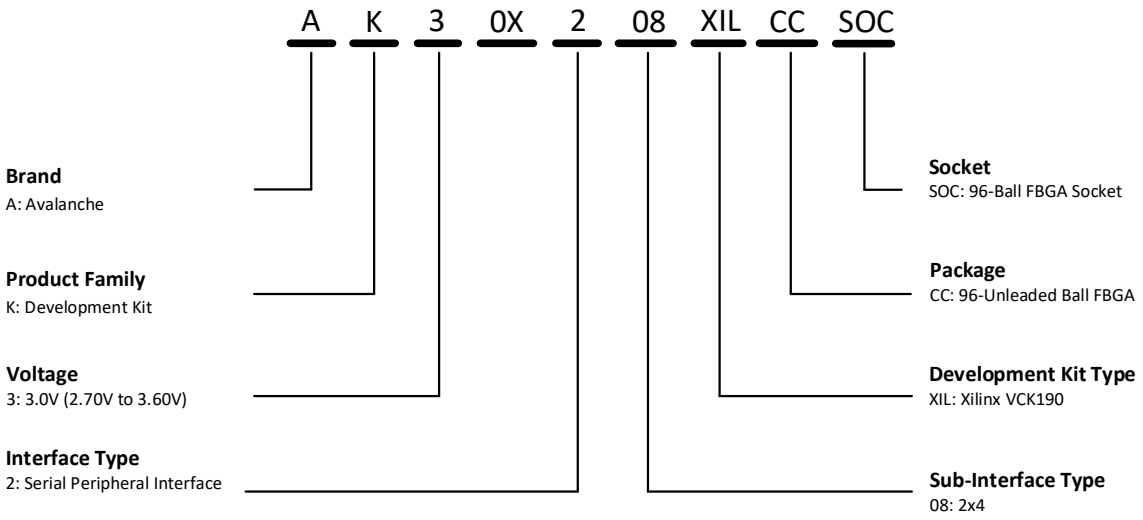


Figure 1: Development Kit Ordering Information

3.2 96-Ball FBGA Socket

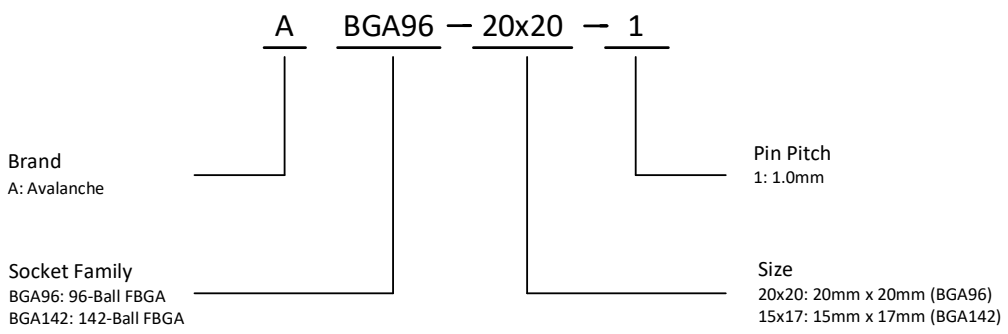


Figure 2: 96-Ball FBGA Socket Ordering Information

4. Development Kit Package Contents

An Avalanche daughterboard (3 x 2.5 inches) with a 96-FBGA socket

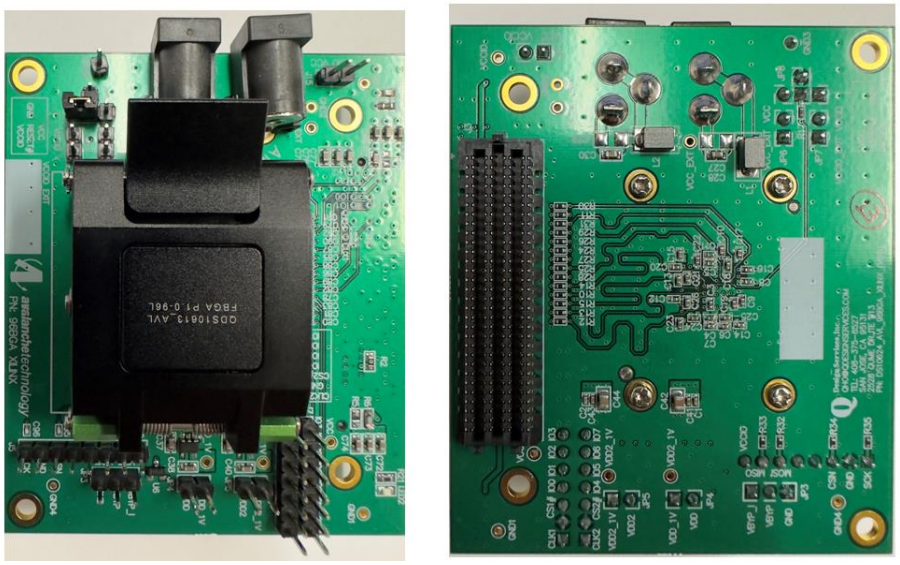


Figure 3: Serial Dual QSPI P-SRAM™ 96-Ball FPGA Socket Daughter Card (Front & Back)

Prerequisites

This user guide assumes the following prerequisite hardware, software and understanding:

- User is familiar with the Xilinx Vivado Design Suite & the Vitis software development platform
- Xilinx Vivado Design Suite has been installed
- Xilinx Vitis Software Platform has been installed
- Xilinx Vivado tools working environment is fully setup

- User has exported the hardware platform and .xsa image file is generated
- PetaLinux software tool has been installed
- User has created a new PetaLinux project or has an existing PetaLinux project

Requirements

- A PC system with one available USB 2.0/3.0 port
- Windows 10 or higher with 32/64-bit Operation System
- Linux OS to install PetaLinux tools
- Prerequires specified above
- Serial Dual QSPI P-SRAM™ for Xilinx Versal ACAP VCK 190 Development Kit:
  - Avalanche daughterboard with a 96-FBGA socket
- Xilinx Versal ACAP VCK190 Development Kit

5. QSPI32 Boot Commands Supported by Avalanche Serial Dual QSPI P-SRAM™ AS3XXG208 and Xilinx Versal ACAP VCK190

Table 3 below shows read commands in both 4-Byte Address (QSPI32) and 3-Byte Address (QSPI24) supported by Xilinx RCU. Avalanche AS3XXG208 Dual Quad SPI P-SRAM device fully supports these commands. Refer to Avalanche 1Gb-8Gb Dual Quad SPI P-SRAM™ datasheet for more information. Click here to download datasheet.

Table 3: Quad SPI Read/Write Commands Supported by Xilinx RCU (ROM Code Unit)

Boot Mode	Data Bus Width	SPI Command	Command Op-Code	Address Byte	Latency/Dummy Cycles Required	Avalanche AS3XXG208 Command Support
QSPI32	1	Normal Read	13h	4	-	✓
QSPI32	1	Fast Read	0Ch	4	8	✓
QSPI32	8	Quad Output Fast Read (Dual Parallel Quad SPI)	6Ch	4	8	✓
QSPI32	1	Normal Write	02h	4	-	✓
QSPI24	1	Fast Read	0Bh	3	-	✓
QSPI24	8	Quad Output Fast Read (Dual Parallel Quad SPI)	6Bh	3	8	✓

Source: Versal ACAP Technical Reference Manual (AM011).

Table 4: Xilinx Versal ACAP VCK190 Supported Quad SPI Boot Mode

Configuration	MRAM Device Count	Chip Select (CS#) Count	Data Bus Width	Avalanche AS3XXG208 Configuration Support
Single (1-bit, 4-bit)	1	1	4	✓
Dual-Parallel (8-bit)	2	2	8	✓

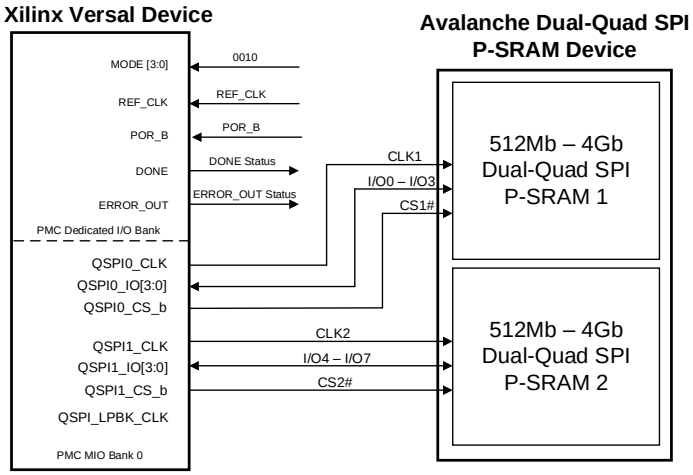


Figure 4: Dual-Parallel Quad SPI Interface Example

Note: For QSPI0_CLK > 37.5 MHz, QSPI_LPBK_CLK must be enabled in the design and unconnected on the board. Source: Versal ACAP Technical Reference Manual (AM011).

6. System Setup

Step 1: Set Configuration Switches on Xilinx Versal VCK190 for MicroSD Boot Mode

The mode DIP switch SW1 on Xilinx Versal VCK190 must be set at OFF, OFF, OFF, ON positions as indicated in table 6 below to enable MicroSD boot mode. Avalanche provides a MicroSD 8GB that contains a reference Linux image and other files for initial bootup process.

Table 5: Xilinx Versal VCK190 Eval Board Quad SPI Boot Mode Settings

Boot Mode	Mode Pins [3:0]	Mode Switch SW1 [4:1]	Data Bus Width	Address Byte
QSPI32	0010	ON, ON, OFF, ON	1-bit, 4-bit	4
MicroSD	1110	OFF,OFF,OFF,ON	N/A	N/A

Source: Versal ACAP Technical Reference Manual (AM011)

Step 2: Power Up the VCK190 to boot from the MicroSD card

1. Attach the Avalanche daughter card to the Versal VCK190 via Samtec connector
2. Connect USB cable between VCK190 and PC system
3. Turn on the VCK190 power using power switch SW13
   Note: to turn off the VCK190 power, use power switch SW13
4. System should boot as root user on PetaLinux as shown below

```
root@vcklinuxnew4:~# df
Filesystem           1K-blocks      Used Available Use% Mounted on
devtmpfs               3927176         4   3927172   0% /dev
tmpfs                  4071124         0   4071124   0% /dev/shm
tmpfs                  1628452      9780   1618672   1% /run
tmpfs                     4096         0      4096   0% /sys/fs/cgroup
tmpfs                  4071124         0   4071124   0% /tmp
tmpfs                  4071124        12   4071112   0% /var/volatile
/dev/mmcblk0p1         2088960     36000   2052960   2% /run/media/mmcblk0p1
/dev/mmcblk0p2         6028864   1208432   4489476  21% /run/media/mmcblk0p2
tmpfs                   814224         0    814224   0% /run/user/0
root@vcklinuxnew4:~# cd /run/media/mmcblk0p1
root@vcklinuxnew4:/run/media/mmcblk0p1# ls
BOOT.BIN                  System Volume Information  boot.scr                  image.ub
```

Figure 5: MicroSD bootup

Step 3: Copy PetaLinux Images from MicroSD to Avalanche AS3XXG208 P-SRAM Device

The MicroSD card contains the following folders, sub-folders and files:

- Folder: <VCK190>: VCK190 Linux image and boot script
- Sub-Folder: <2022.1>: PetaLinux Release 2022.1
- Sub-Folder: <2021.2>: PetaLinux Release 2022.2
- Sub-Folder: <Patch_files>: contains all the patch files to add add Avalanche AS3XXG208 P-SRAM device into existing Linux drivers:
  - Xloader_qspi.c/xloader_qspi.h: boot load for PLM update
  - Spi-nor.h/spi-nor-ids.c/spi-nor-core.c: U-boot updates
  - Cfi.h/core.c/micron-st.c: kernel updates
- BOOT.BIN: Linux reference boot image
- Image.ub: Linux fit image
- Boot.scr: Linux boot script
- vck190-Boot-from-SD-card.txt: instruction on how to transfer images from MicroSD card to AS3XXG208 P-SRAM Device
- vck190-Boot-from-QSPI.log: log file of booting from AS3XXG208 P-SRAM Device

Use the following command to copy PetaLinux images from MicroSD to Avalanche AS3XXG208 P-SRAM device:

1. root@vcklinux**:~# cd /run/media/mmcblk0p1**
2. root@vcklinux**:~ls /run/media/mmcblk0p1**
   a. You should see all three files **BOOT.BIN, boot.scr and image.ub**
3. **flashcp BOOT.BIN /dev/mtd0 (**copies boot sector to partition 0 of MRAM)
4. **flashcp image.ub /dev/mtd1 (**copies PetaLinux image to partition 1 of MRAM)

5. **flashcp boot.scr /dev/mtd2** (copies boot script to partition 2 of MRAM)

Step 4: Boot from Avalanche AS3XXG208 P-SRAM Device

1. Power down the VCK190
2. Set the boot mode switch SW1 to ON-ON-OFF-ON QSPI32 boot mode as indicated in table 6.
3. PetaLinux will now boot from the MRAM.

Note: it is configured by default to run from DRAM as that is the fastest mode.

To write files and have them persist across a reboot, you need to mount the MRAM filesystem and write to it as follows:

1. mount /dev/mtdblock4 /mnt
2. Write a file under the /mnt directory
3. Reboot
4. mount again: mount /dev/mtdblock4 /mnt
5. View the file.
6. In an actual system, automount will likely be used to create persistent partitions.

A successful boot will show screenshot below

```
[0.016]****************************************
[0.070]Xilinx Versal Platform Loader and Manager
[0.126]Release 2022.1   Mar  9 2023  -  18:37:18
[0.184]Platform Version: v2.0 PMC: v2.0, PS: v2.0
[0.248]BOOTMODE: 0x2, MULTIBOOT: 0x0
[0.299]****************************************
[0.518]Non Secure Boot
[3.425]PLM Initialization Time
[3.474]***********Boot PDI Load: Started***********
[3.534]Loading PDI from QSPI32
[3.582]Monolithic/Master Device
[3.653]***************************
[3.702]** AVALANCHE TECHNOLOGY ***
[3.752]**      BOOT MODULE     ***
[3.801]***************************
[3.851]FlashID=0xE6 0x21 0x2C 0x1
[3.905]AVALANCHE ID
 [3.947]HP Dual-Quad SPI - 3V
[3.994]40C~125C, 8Gbits
[4.282]0.724 ms: PDI initialization time
[4.338]+++Loading Image#: 0x1, Name: lpd, Id: 0x04210002
[4.404]---Loading Partition#: 0x1, Id: 0xC
[76.500] 72.012 ms for Partition#: 0x1, Size: 2896 Bytes
[81.349]---Loading Partition#: 0x2, Id: 0x0
[85.729] 0.527 ms for Partition#: 0x2, Size: 48 Bytes
```

Figure 6: Avalanche MRAM bootup

7. How to re-build the Linux Images to accommodate Avalanche AS3XXG208 P-SRAM Device

Step 1: Configuring Boot Image Using Vivado Hardware Platform

To configure a bootable image, user can use Vivado hardware design to configure the Avalanche Serial Dual QSPI P-SRAM™ daughter card. Use the following steps to generate the .xsa then use PetaLinux to create Linux boot images:

1. Select "Boot Mode"
2. Select "QSPI" as a Boot/Storage
3. Select "Dual Parallel" as QSPI Mode
4. Select "x4" as Data Mode
5. Enter "80" for Requested QSPI Reference Clock Frequency (MHz)

MRAM QSPI Interface Frequency (MHz) = QSPI Reference Clock Frequency (MHz)/2

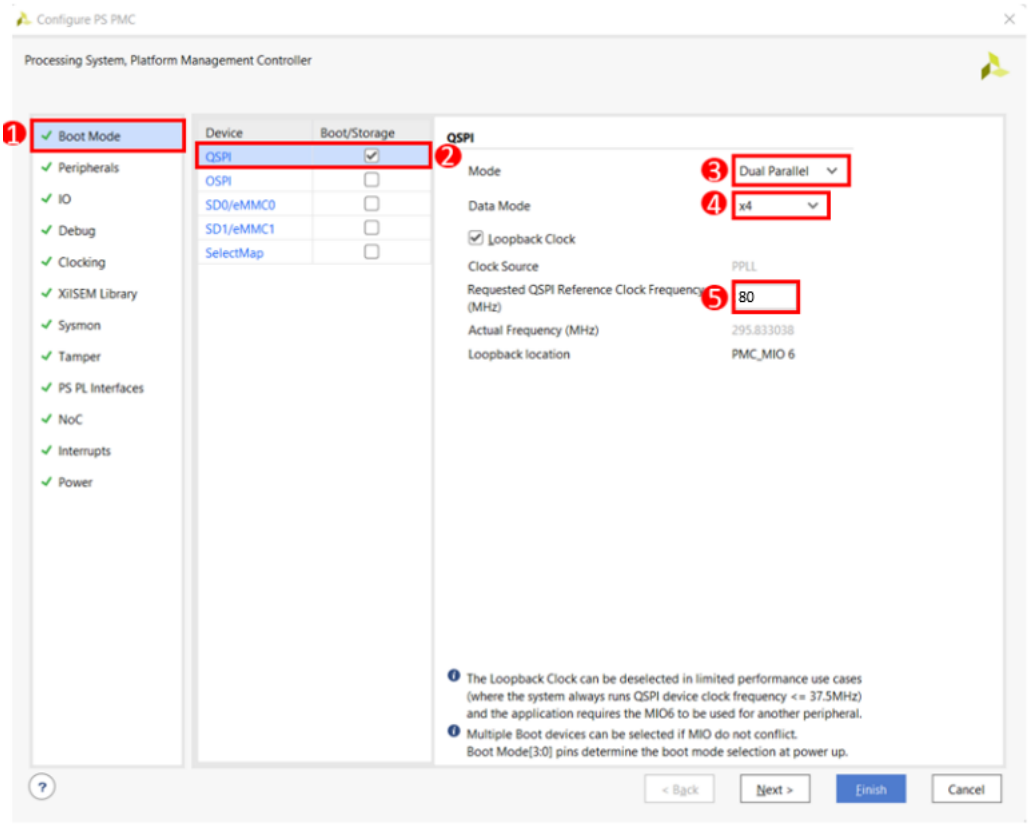


Figure 7: Steps to Configure Boot Images Using Vivado Hardware Platform

Step 2: Modifying Existing QSPI driver and PLM (Platform Loader and Manager) to add Avalanche ID to "Xloader_QSPI.C"

Assuming user has exported the hardware platform, the *.xsa image file has been generated, and the PLM has been created, the Vitis software platform will create the PLM application project and edit versal wrapper platform under the Explorer view as indicated in Figure 7 and Figure 8 below.

User is required to modify xloader_qspi.c to add Avalanche ID to it (refer to Figure 8 and Figure 9 below). The xloader_qspi.c is located in the sub-folder, Figure 7 below.

For more information on how to create the PLM, click here

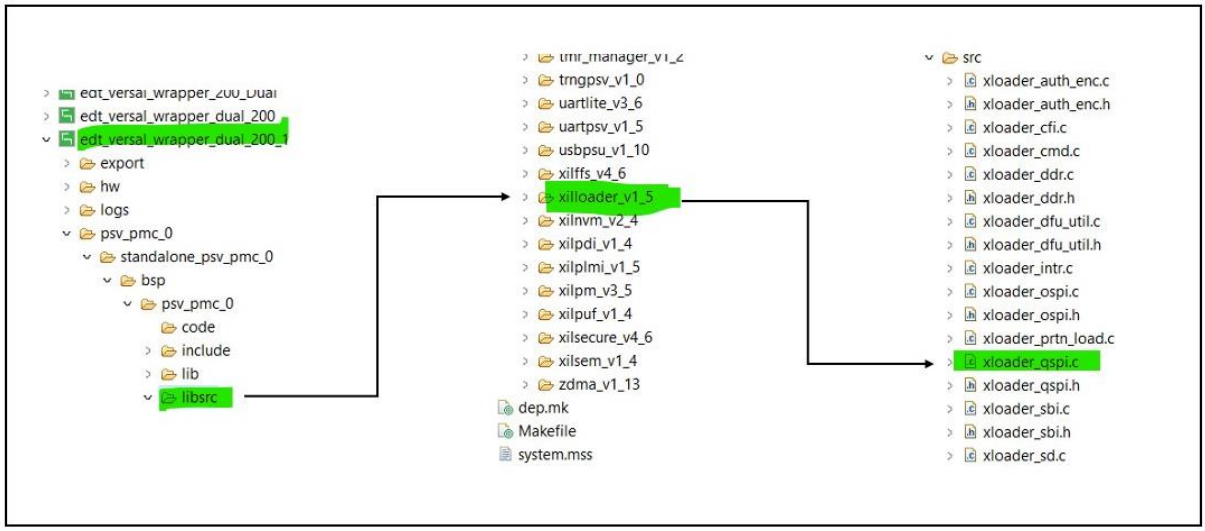


Figure 8: xloader_qspi.c file location

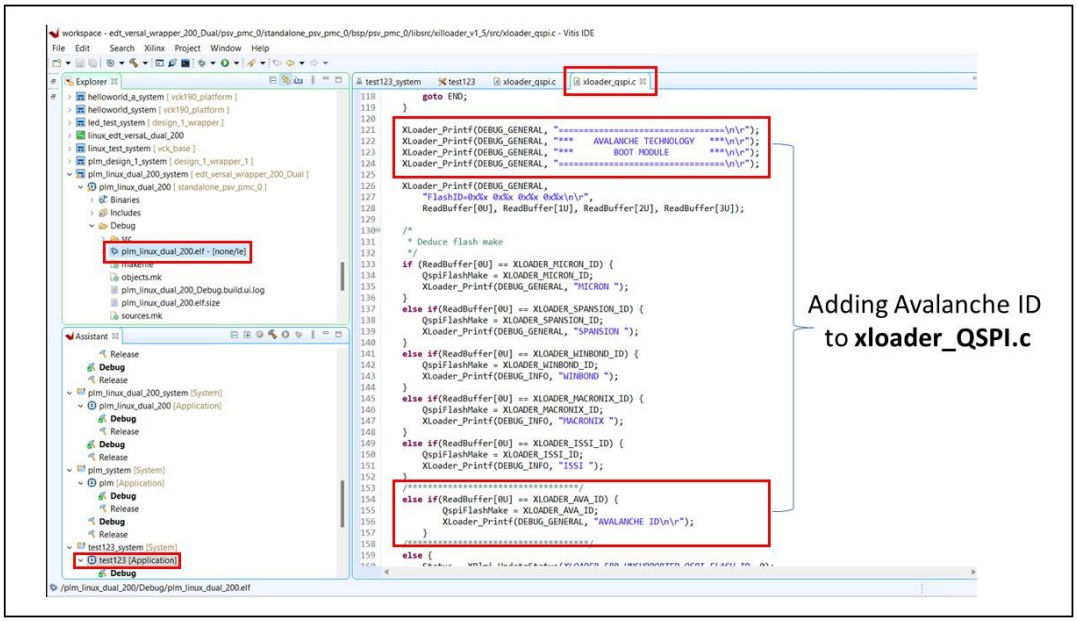


Figure 9: Adding Avalanche ID to Xloader_QSPI.c

```
171         * Deduce flash Size
172         */
173 /*****************************************************************/
174     if(ReadBuffer[0U] == XLOADER_AVA_ID) {
175         if (ReadBuffer[1U] == 0X21U) {
176             XLoader_Printf(DEBUG_GENERAL, "HP Dual-Quad SPI - 3V\r\n");
177         }
178         else {
179 //          Status = XPlmi_UpdateStatus(XLOADER_ERR_UNSUPPORTED_QSPI_FLASH_ID, 0);
180             XLoader_Printf(DEBUG_GENERAL, "Unsupported Interface or Voltage\r\n");
181 //          goto END;
182         }
183         if (ReadBuffer[2U] == 0X28U) {
184             QspiFlashSize = XLOADER_FLASH_SIZE_512Mb;
185             XLoader_Printf(DEBUG_GENERAL, "40C~125C, 512Mb\r\n");
186         }
187         else if (ReadBuffer[2U] == 0X29U) {
188                 QspiFlashSize = XLOADER_FLASH_SIZE_1G;
189                 XLoader_Printf(DEBUG_GENERAL, "40C~125C, 1Gb\r\n");
190             }
191         else if (ReadBuffer[2U] == 0X2AU) {
192                 QspiFlashSize = XLOADER_FLASH_SIZE_2G;
193                 XLoader_Printf(DEBUG_GENERAL, "40C~125C, 2Gb\r\n");
194             }
195         else if (ReadBuffer[2U] == 0X2CU) {
196                 QspiFlashSize = XLOADER_FLASH_SIZE_4G;
197                 XLoader_Printf(DEBUG_GENERAL, "40C~125C, 4Gb\r\n");
198             }
199         else {
200 //          Status = XPlmi_UpdateStatus(XLOADER_ERR_UNSUPPORTED_QSPI_FLASH_SIZE, 0);
201         QspiFlashSize = XLOADER_FLASH_SIZE_1G;
202         XLoader_Printf(DEBUG_GENERAL, "Unsupported Density\r\n");
203 //      goto END;
```

Figure 10: Adding Avalanche ID Information to xloader_QSPI.c

A fully modified xloader_QSPI.c sample code that includes Avalanche ID and device propriety can be downloaded from Avalanche website. Click here to download the sample code.

Step 3: Building booting Linux image with Modified PLM.elf

The PLM.elf file needs to be re-built and U-Boot source needs to be repackaged. The following commands are used:

- Petaliux-package --boot –plm plm_new.elf --u-boot –force

Step 4: Re-building U-Boot to include Avalanche ID

User should download source code using external Kernel and U-Boot with PetaLinux. Click here to download the source code. For more information on how to repackage the U-Boot source, click here

Below is an example of modifications made to the spi-nor-ids.c to include Avalanche device ID. Similar modifications can be made to other files indicated in the “Patch Folder”.

```
{ INFO("AVALANCHE_2x512Mb",   0xE62128, 0, 64 * 1024, 1024, SECT_4K | USE_FSR | SPI_NOR_QUAD_READ | SPI_NOR_4B_OPCODES | NO_CHIP_ERASE) },
{ INFO("AVALANCHE_2x1Gb",     0xE62129, 0, 64 * 1024, 2048, SECT_4K | USE_FSR | SPI_NOR_QUAD_READ | SPI_NOR_4B_OPCODES | NO_CHIP_ERASE) },
{ INFO("AVALANCHE_2x2Gb",     0xE6212a, 0, 64 * 1024, 4096, SECT_4K | USE_FSR | SPI_NOR_QUAD_READ | SPI_NOR_4B_OPCODES | NO_CHIP_ERASE) },
{ INFO("AVALANCHE_2x4Gb",     0xE6212c, 0, 64 * 1024, 8192, SECT_4K | USE_FSR | SPI_NOR_QUAD_READ | SPI_NOR_4B_OPCODES | NO_CHIP_ERASE) },
{ INFO6("mt25ql256a",    0x20ba19, 0x104400, 64 * 1024,  512,
        SECT_4K | SPI_NOR_DUAL_READ | SPI_NOR_QUAD_READ |
        SPI_NOR_4B_OPCODES | USE_FSR) },
```

Figure 11: Adding Avalanche Device ID info to Linux SPI-NOR-IDS.c

The "SPI-NOR-IDs.c" file is located in "u-boot-xlnx/drivers/mtd/spi/spi-nor-ids.c. User is required to add Avalanche ID information to the file as indicated in figure 11 above.

Step 5: Rebuilding Kernel that includes Avalanche ID

Below is an example of modification made to the micron-st.c to include Avalanche device ID. Similar modifications can be made to other files indicated in the "Patch Folder"

```
                    SPI_NOR_HAS_LOCK | SPI_NOR_HAS_LOCK) },
{ "n25q00a",      INFO(0x20bb21, 0, 64 * 1024, 2048, SECT_4K |
                  USE_FSR | SPI_NOR_QUAD_READ | NO_CHIP_ERASE ||
                  SPI_NOR_HAS_LOCK) },
{ "AVALANCHE_2x512Mb",     INFO(0xE62128, 0, 64 * 1024, 1024, SECT_4K |
                  USE_FSR | SPI_NOR_QUAD_READ | SPI_NOR_4B_OPCODES | NO_CHIP_ERASE |
                  SPI_NOR_HAS_LOCK) },
{ "AVALANCHE_2x1Gb",     INFO(0xE62129, 0, 64 * 1024, 2048, SECT_4K |
                  USE_FSR | SPI_NOR_QUAD_READ | SPI_NOR_4B_OPCODES | NO_CHIP_ERASE |
                  SPI_NOR_HAS_LOCK) },
{ "AVALANCHE_2x2Gb",     INFO(0xE6212a, 0, 64 * 1024, 4096, SECT_4K |
                  USE_FSR | SPI_NOR_QUAD_READ | SPI_NOR_4B_OPCODES | NO_CHIP_ERASE |
                  SPI_NOR_HAS_LOCK) },
{ "AVALANCHE_2x4Gb",     INFO(0xE6212c, 0, 64 * 1024, 8192, SECT_4K |
                  USE_FSR | SPI_NOR_QUAD_READ | SPI_NOR_4B_OPCODES | NO_CHIP_ERASE |
                  SPI_NOR_HAS_LOCK) },
{ "mt25ql02g",   INFO(0x20ba22, 0, 64 * 1024, 4096,
                       SECT_4K | USE_FSR | SPI_NOR_QUAD_READ |
                       NO_CHIP_ERASE | SPI_NOR_HAS_LOCK) },
```

Figure 12: Adding Avalanche info to Linux micron-st.c

Step 6: Rebuilding PetaLinux with updated U-Boot and Kernels that includes Avalanche ID

The following Linux commands are used:

- Petaliux-build
- Petaliux-package --boot –plm plm_new.elf --u-boot –force

In the buit folder, it should now contain modified Linux images:

- BOOT.BIN
- boot.src
- Image.ub

The last step is to copy these modified images to a MicroSD card and follow the procedures defined in section 6.

8. Avalanche Serial Dual QSPI P-SRAM™ Daughter Card

The Avalanche Serial Dual QSPI P-SRAM™ daughter card (part # AK3XXG208XILCCSOC) is an 8-layer board.

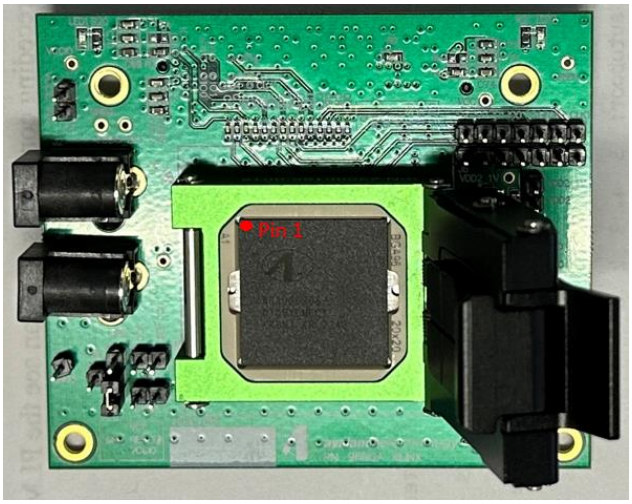
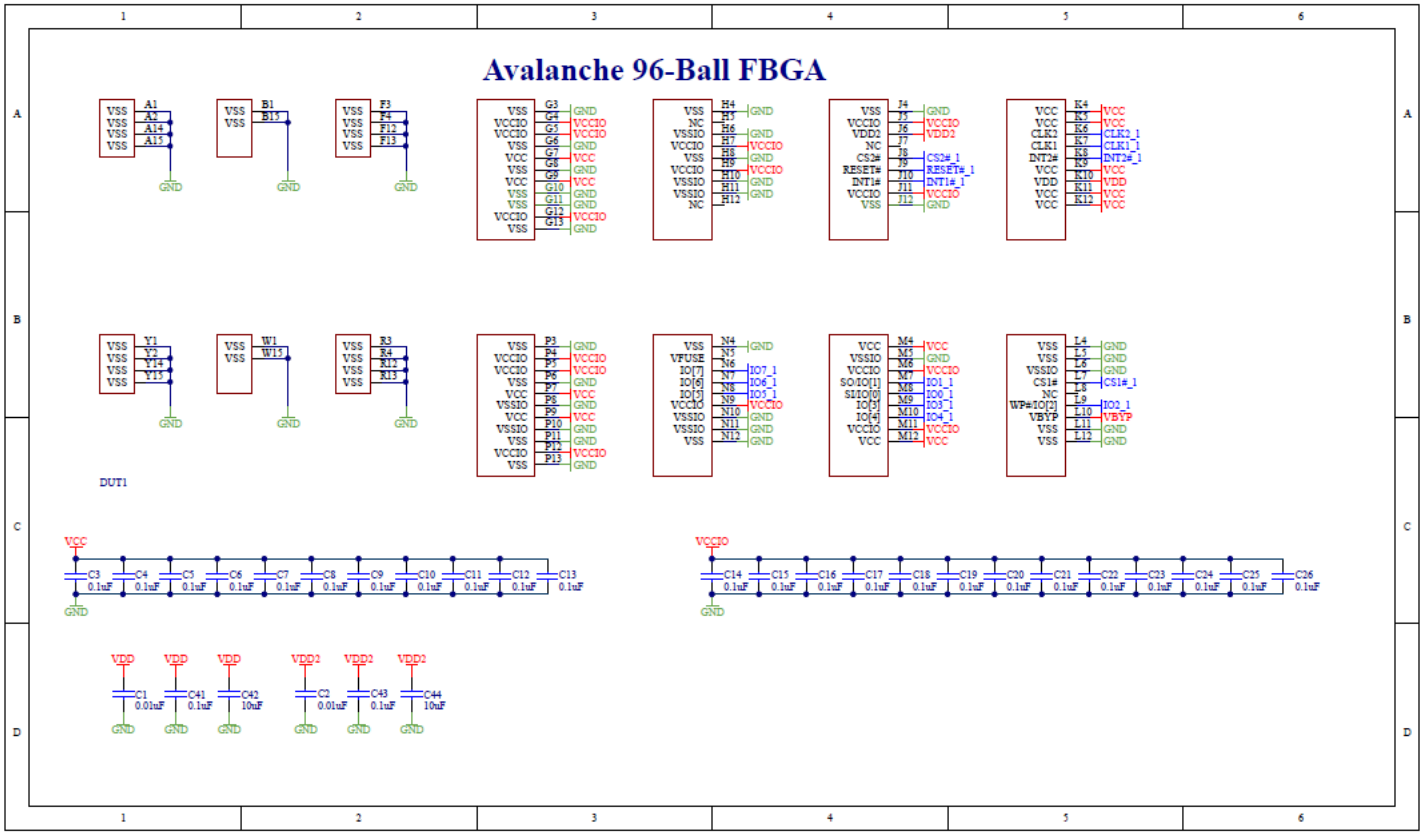


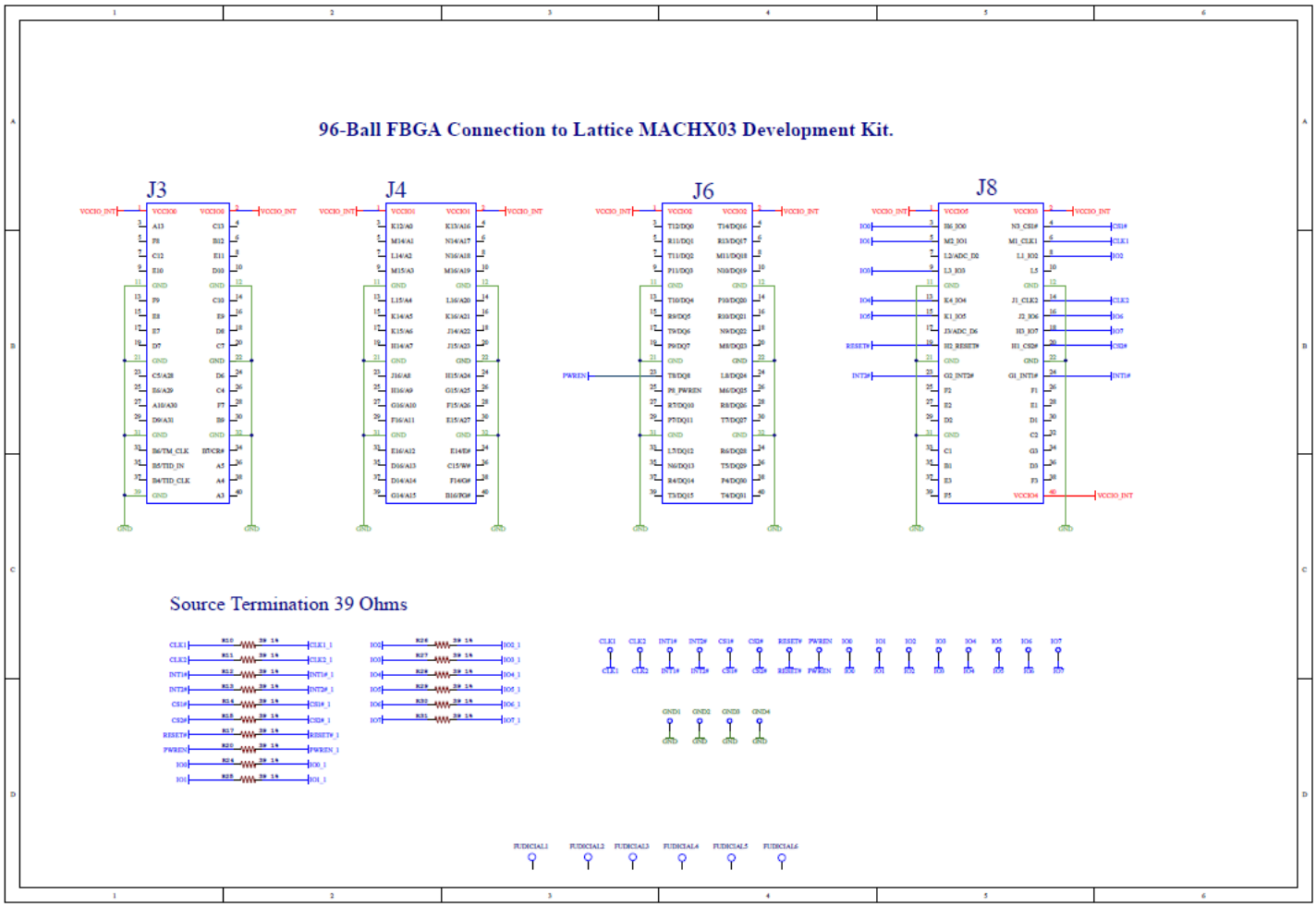
Figure 13: Avalanche Serial Dual QSPI P-SRAM™ Daughter Card with AS30XG208 Device

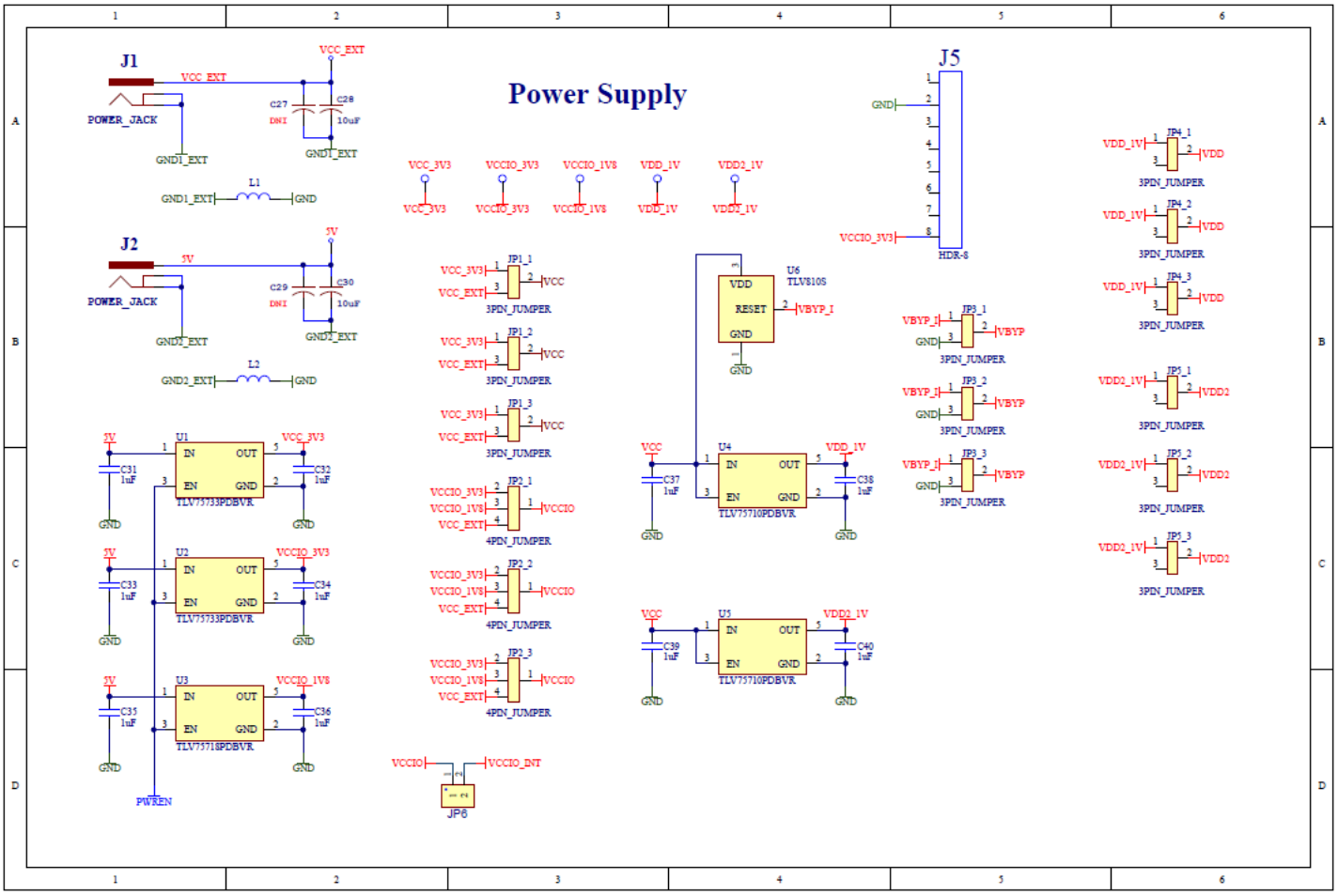
9. Schematic and BOMs

Click here to download gerber file, board layout and schematic in high resolution ALTIUM format.

Schematic




96-Ball FBGA Connection to Lattice MACHX03 Development Kit.
J3
J4
J6
J8
Source Termination 39 Ohms


Power Supply
J1
POWER_JACK
VCC_EXT
GND1_EXT
L1
J2
POWER_JACK
5V
GND2_EXT
L2
U1
TLV75733PDBVR
U2
TLV75733PDBVR
U3
TLV75718PDBVR
PWREN
VCC_3V3
VCCIO_3V3
VCCIO_1V8
VDD_1V
VDD2_1V
JP1_1
JP1_2
JP1_3
JP2_1
JP2_2
JP2_3
3PIN_JUMPER
4PIN_JUMPER
VCCIO_INT
JP6
U6
TLV810S
VDD
RESET
GND
VBYP_I
VBYP
U4
TLV75710PDBVR
U5
TLV75710PDBVR
J5
HDR-8
JP3_1
JP3_2
JP3_3
JP4_1
JP4_2
JP4_3
JP5_1
JP5_2
JP5_3
VDD
VDD2

BOMs

Item	Quantity	Description	Designator	Footprint
1	2	0.1µF	C1, C2	N_0402
2	26	0.1µF	C3, C4, C5, C6, C7, C8, C9, C10, C11, C12, C13, C14, C15, C16, C17, C18, C19, C20, C21, C22, C23, C24, C25, C26, C41, C43	N_0402
3	2	DNI	C27, C29	N_1206
4	5	10µF	C28, C30, C69, C72, C75	N_0603
5	10	1µF	C37, C38, C39, C40, C70, C71, C73, C74, C76, C77	N_0603
6	2	10µF	C42, C44	N_0805
7	18	5pF	C78, C79, C80, C81, C82, C83, C85, C86, C87, C88, C89, C90, C91, C92, C93, C94, C95, C96	N_0402
8	1	96-Ball FBGA footprint/socket	DUT1	Ava 96-Ball FBGA
9	8	TP	GND1, GND2, GND3, GND4, Vcc, Vccio, Vdd2_1V, Vdd_1V	Test Point
10	2	Power Jack	J1, J2	Power Jack
11	1	Samtec Connector (SEAM-30-02.0-L-08-1-A-K-TR)	J3	Samtec
12	1	HDR-8	J5	HDR 1x8
13	2	HDR-8	J6, J7	HDR 1x7
14	1	HDR-2-Pin	J15	HDR-2
15	1	3-Pin Jumper	JP3	3-Pin Jumper
16	4	HDR-TH 2P-P2.54	JP4, JP5, Jp6, JP7	1x2 Header with Shunt 051914
17	1	4-Pin Jumper	JP8	4-Pin Jumper
18	2	FB	L1, L2	FB
19	2	Red LED	LED1, LED2	N_0603
20	3	0R	R2, R3, R6	N_0603
21	2	DNI	R4	N_0603
22	19	39.1% Resistor	R10, R11, R12, R13, R14, R15, R17, R24, R25, R26, R27, R28, R29, R30, R31, R32, R33, R34, R35, R20	N_0402
23	1	666 Resistor	R20	N_0603
24	1	261 Resistor	R21	N_0603
25	2	TI TLV75710PDBVR	U4, U5	SOT23-5
26	1	TI TLV810S	U6	SOT23-3
27	2	TP	Vcc_ext, Vccio_ext	Test Point