SDSoC 2016.2 Release Notes

For licensing and installation information, refer to the document An Introduction to the *SDSoC Environment* (UG1028) `<sdsoc_install_root>/docs/ug1028-intro-to-sdsoc.pdf`. The Tutorial Labs in UG1028 provide a hands-on introduction to the SDSoC Environment. Additional reference and tutorial information is contained in *SDSoC Environment User Guide* (UG1027).

Release Summary

2016.2 (July 13, 2016)

- Integrated installer including SDSoC 2016.2 and the Vivado Design Suite 2016.2 (System Edition with Vivado® HLS) for Zynq®-7000 and Zynq UltraScale+™ families
  - Complete installation environment containing tools, data files, and patches for supported targets
  - Web-based installer option
- Windows 64-bit support
  - Windows 7.1 and 10.0 Professional (64-bit), English
- Linux 64-bit host support
  - Red Hat Enterprise Workstation 6.6-6.7 and 7.0-7.1 (64-bit)
  - Ubuntu Linux 14.04.3 LTS (64-bit)
- ARM compiler toolchain support
  - Linaro-based GCC 4.9.2 32-bit and 64-bit toolchains
- Target OS support
  - Linux (kernel 4.4, Xilinx branch Xilinx-v2016.2_sdsoc), bare-metal, and FreeRTOS 8.2.3
  - Example PetaLinux BSP for ZC702 platform with documentation in *SDSoC Environment User Guide: Platforms and Libraries* (UG1146)
- Device Support
  - Zynq-7000 AP SoC support
  - Enhanced Zynq UltraScale+ MPSoC support, including AXI DMA scatter-gather and extended addressing for the ZCU102 platform (included with the release)
    - Supports Zynq UltraScale+ MPSoC Base Targeted Reference Design 2016.2 (embedded video processing platform)

- Supports Zynq UltraScale+ MPSoC Software Acceleration TRD 2016.2 (embedded signal processing platform)

- Documentation updates, including tutorial labs and a platform and libraries methodology guide

  o *SDSoC Environment User Guide* (UG1027): HTML version in Eclipse and `<sds_install_root>/docs/ug1027-user-guide.pdf`. Includes new content on system optimization and pragmas, updated chapter on performance measurement using the AXI performance monitor, and a new chapter on using external I/O.

  o *SDSoC Environment User Guide: An Introduction to the SDSoC Environment* (UG1028): Tutorial Labs 1-7, HTML version in Eclipse and `<sds_install_root>/docs/ug1028-intro-to-sdsoc.pdf`. Includes new tutorial on communication to platform I/O streams and tips for using non-ZC702 boards to complete tutorials.

  o *SDSoC Environment User Guide: Platforms and Libraries* (UG1146): HTML version in Eclipse and `<sds_install_root>/docs/ug1146-sdsoc-platforms-and-libraries.pdf`. Updated tutorials for direct I/O with lossless data capture example.

- SDSoC Eclipse UI with project creation, implementation, and debug

  o SDDebug, SDRelease, and user-defined build configurations with independent project settings

  o Hardware/software event trace build, data collection, and visualization

    - Updated trace support with reduced data collection times

  o Usability updates

    - Support to run performance estimation with a persistent configuration, allowing command line arguments and other settings to be specified

  o Online SDSoC help

- Compiler enhancements

  o Enhanced scheduling and resource allocation for multiple AXIMM ports

  o Enhanced data size analysis and checking

- Flow enhancements

  o FreeRTOS support enabled for MicroZed, Zed, and Zybo platforms

- Platform updates

  o Updated zc702_axis_io and zybo_axis_io platforms with direct I/O and lossless data capture

  o Renamed directories to reflect tool chain triplets

- Enhanced user-defined platform support

  o Improved error checking for hardware platform and software platform XML files

  o Updated documentation and tutorial examples for supporting and accessing platform streaming I/O.

- New and updated sample applications

  o Consuming an unpacketized input stream and writing to buffers in DDR memory

  o Converting an unpacketized input stream to a packetized stream

- o Using a software queue to mask data transfer setup and tear down overhead for lossless consumption of streaming I/O
- Bug fixes and infrastructure updates
  - o Improved stability and ease of use, including improved error checking
  - o Updated and enhanced accelerator driver API software

Zynq ARM Toolchain Changes

SDSoC 2016.1 and 2016.2 provide a Linaro-based 32-bit GCC compiler toolchain for the Zynq Cortex™-A9 processor. When compiling and linking applications, use only object files and libraries built using the same compiler toolchain and options used by sdscc, sds++, and SDSoC provided software libraries and software components (Linux kernel and root file system, BSP libraries, and other pre-built libraries). Object files and libraries created with SDSoC 2015.4 must be recompiled using SDSoC 2016.1 and 2016.2.

The table below summarizes changes related to the Zynq compiler toolchain.

Usage	SDSoC 2015.4	SDSoC 2016.1 and 2016.2
Zynq ARM bare-metal compiler and linker options	(built-in option: soft floating point ABI)	-mcpu=cortex-a9 -mfpu=vfpv3 -mfloat-abi=hard
Zynq ARM bare-metal linker options	(built-in option)	-Wl,--build-id=none -specs=<specfile> where the <specfile> contains *startfile: crti%O%s crtbegin%O%s
Zynq ARM bare-metal compiler	${SDSOC_install}/SDK/2015.4/gnu/arm/<host>/bin Toolchain prefix: arm-xilinx-eabi gcc executable: arm-xilinx-eabi-gcc g++ executable: arm-xilinx-eabi-g++	${SDSOC_install}/SDK/2016.1/gnu/aarch32/<host>/gcc-arm-none-eabi/bin Toolchain prefix: arm-none-eabi gcc executable: arm-none-eabi-gcc g++ executable: arm-none-eabi-g++
Zynq SDSoC bare-metal software (lib, include)	${SDSOC_install}/arm-xilinx-eabi	${SDSOC_install}/aarch32-none
Zynq ARM Linux compiler	${SDSOC_install}/SDK/2015.4/gnu/arm/<host>/bin Toolchain prefix: arm-xilinx-linux-gnueabi- gcc executable: arm-xilinx-linux-gnueabi-gcc g++ executable: arm-xilinx-linux-gnueabi-g++	${SDSOC_install}/SDK/2016.1/gnu/aarch32/<host>/gcc-arm-linux-gnueabi/bin Toolchain prefix: arm-linux-gnueabihf- gcc executable: arm-linux-gnueabihf-gcc g++ executable: arm-linux-gnueabihf-g++
Zynq SDSoC Linux software (lib, include)	${SDSOC_install}/arm-xilinx-linux-gnueabi	${SDSOC_install}/aarch32-linux

Known Issues (to be addressed in future releases)

1. On Linux hosts, the SDSoC IDE Welcome screen does not display all information (refer to Help Contents).

2. On RHEL Linux, connecting to a serial port from the Terminal View will crash SDSoC IDE. To work around the issue, use the SDK Terminal view or a separate terminal program to connect to your board.

3. If you cancel a build in the SDSoC IDE, you must delete the build target directory for the current build configuration (e.g., SDDebug, SDRelease) before rebuilding. Otherwise the behavior of the build is indeterminate.

4. If you encounter an "error copying" message on a Windows host, the SDSoC install directory or your workspace has a pathname that causes file pathnames generated by SDSoC to exceed the Windows limit of 260 characters. To mitigate this issue, you can choose short names for the install and workspace directories, e.g., `"C:\i"` or `"C:\w"`.

5. In performance estimation flows, do not use variables named clock_start and clock_end in a function marked for acceleration or the performance root function, otherwise a conflict occurs with variables created for the flow.

6. Standalone applications cannot write the file system on an SD card due to a bug in the `f_write()` function in the `xilffs` library.

7. The following critical warning can be ignored:

```
CRITICAL WARNING: [Pfi 67-13] Hardware Handoff file zc702_processing_system7_1_0.hwdef does not
exist for instance zc702_i/ps7/inst
```

Revision History

The following table shows the revision history for this document:

Date	Version	Revisions
07/13/2016	2016.2	Updated entire text for current release.

Send Feedback

Notice of Disclaimer

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx's limited warranty, please refer to Xilinx's Terms of Sale which can be viewed at www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at www.xilinx.com/legal.htm#tos.