

Introduction

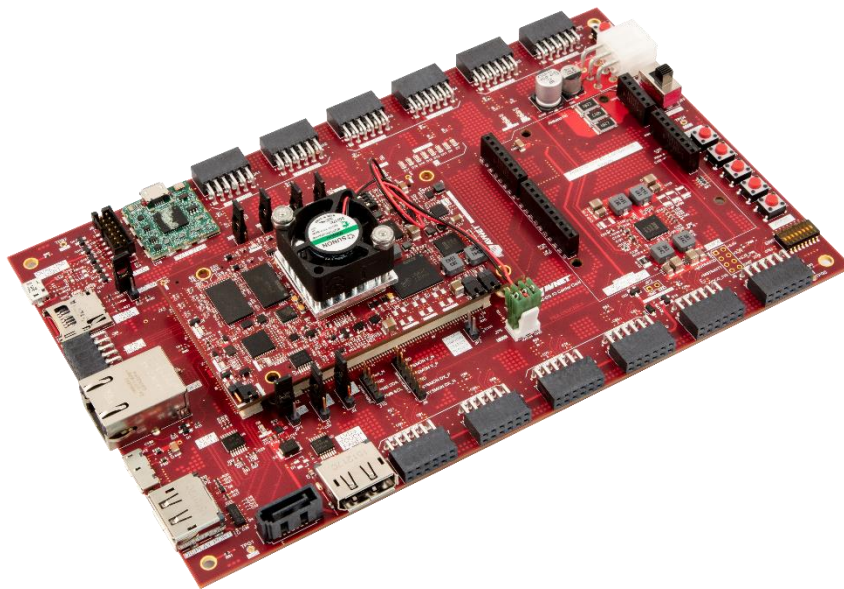
This document describes a Zynq UltraScale+ processor system design implemented and tested on the Avnet UltraZed-EG Starter Kit and running a simple standalone software application.

The Zynq UltraScale+ hardware development for this example design was performed with Xilinx Vivado v2016.2 tools and the Avnet HDL git repository.

UltraZed-EG Starter Kit Overview

The UltraZed-EG™ Starter Kit from Avnet Electronics Marketing provides engineers with everything needed to develop edge-to-cloud Internet-connected solutions which need to connect to cloud computing services such as IBM Watson. The kit is based on Avnet's UltraZed-EG System-on-Module (SOM) with a Xilinx Zynq® UltraScale+™ MPSoC. Several Pmod™ compatible expansion ports and an Arduino shield interface are also provided allowing for further connection of several low-cost sensor modules.

The versatility of this platform offers an excellent prototyping or proof-of-concept vehicle for your new product. Once you are finished prototyping your new product design and are ready to go into production, most of the components found on this platform can be purchased directly from Avnet. Indeed, the UltraZed-EG SOM is a great way to integrate a complete Zynq UltraScale+ solution into your product without worrying about the design complexities of designing your own chip-down system. Please contact your local Avnet FAE for further details.

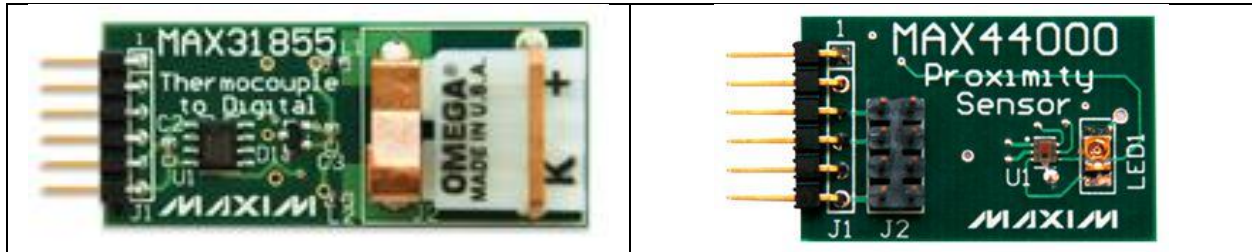


© 2016 Avnet. All rights reserved. All trademarks and registered trademarks are the property of their respective owners. All specifications are subject to change without notice.

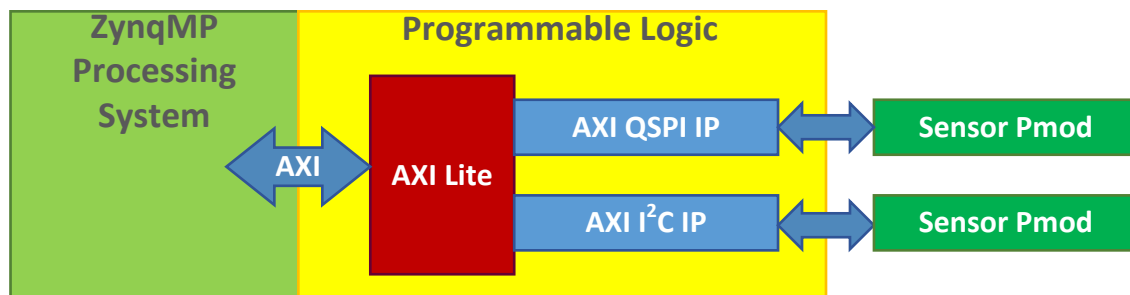
NOTICE OF DISCLAIMER: Avnet is providing this design, code, or information "as is." By providing the design, code, or information as one possible implementation of this feature, application, or standard, Avnet makes no representation that this implementation is free from any claims of infringement. You are responsible for obtaining any rights you may require for your implementation. Avnet expressly disclaims any warranty whatsoever with respect to the adequacy of the implementation, including but not limited to any warranties or representations that this implementation is free from claims of infringement and any implied warranties of merchantability or fitness for a particular purpose.

Design Overview

The UltraZed-EG I/O Carrier Card allows the UltraZed-EG System-on-Module (SOM) to connect to a variety of Pmod compatible expansion modules and Arduino “shields”. This example design uses the MAX31855PMB1 thermocouple temperature sensing module and MAX44000PMB1 proximity sensing module from Maxim Integrated.



These Pmods can easily be connected to the Programmable Logic of the Xilinx Zynq UltraScale+ MPSoC found on the UltraZed-EG SOM. Since the Processing System and Programmable Logic (PL) are asynchronous to each other, an AXI interconnect is used to connect the I²C and SPI IP cores to the shield and Pmod. This IP allows the IP cores to capture sensor data from the sensors while the ARM cores are able to perform other important functions.



Objectives

This tutorial is a guide for how to:

- Configure and build the Zynq MPSoC PS and PL for the UltraZed-EG Starter Kit
- Execute the example design on hardware

Example Design Requirements

Software

The software required to build and execute the example design is:

- TeraTerm or another serial terminal emulator
- Cloned Avnet HDL git repository
- Xilinx Vivado Design Suite 2016.2

Hardware

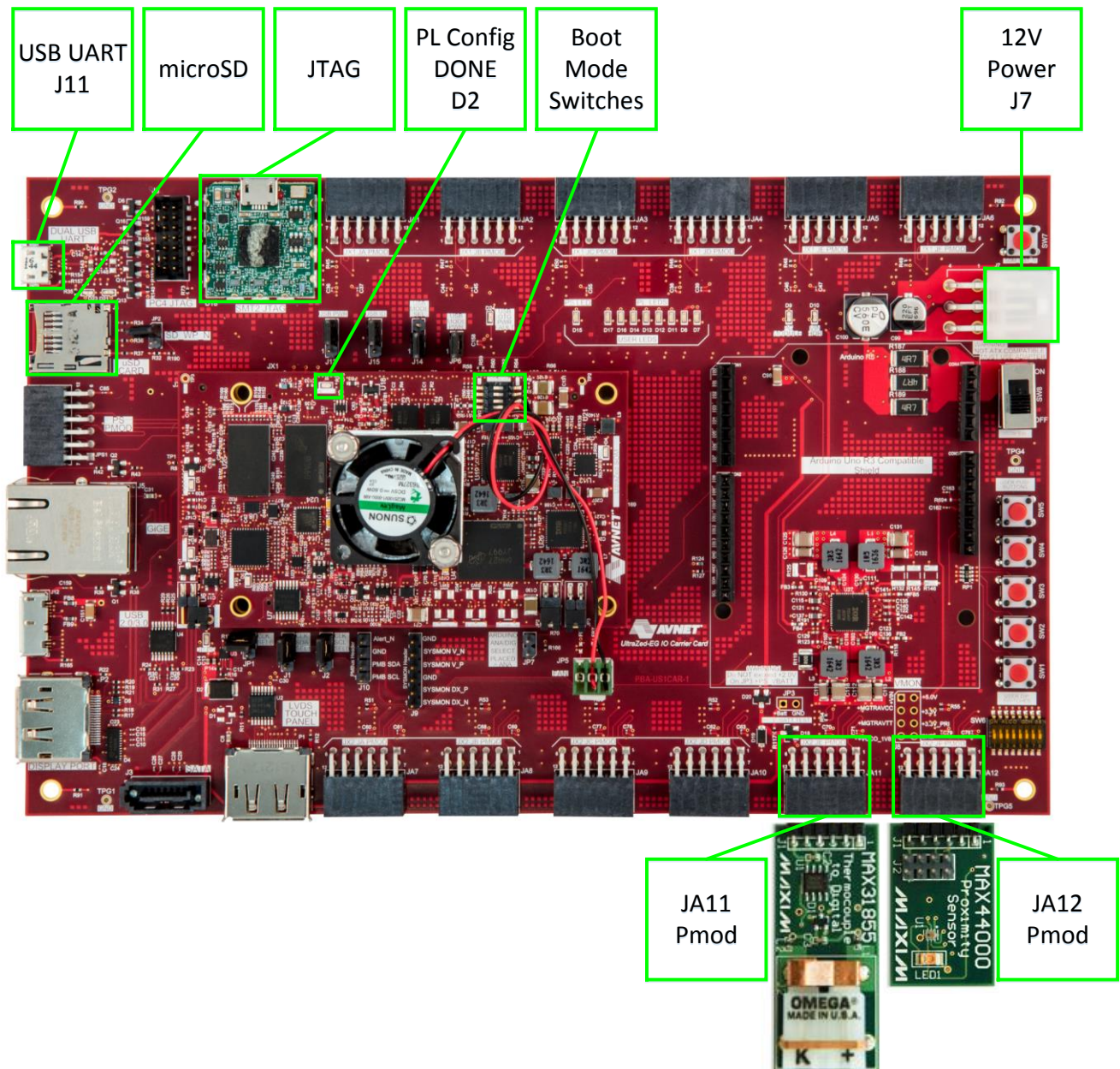
The hardware required to build and execute the reference design is:

- PC with at least 8GB RAM available for Xilinx tools
(www.xilinx.com/design-tools/vivado/memory.htm)
 - 4GB required, but 8GB recommended
- Avnet UltraZed-EG Starter Kit (AES-ZU3EGES-1-SK-G)
 - Avnet UltraZed-EG SOM (AES-ZU3EGES-1-SOM-G)
 - Avnet UltraZed-EG I/O Carrier Card (AES-ZU-IOCC-G)
 - 8GB microSD card
 - USB cables
 - 12V AC/DC supply
- Maxim Integrated Thermocouple Sensor Pmod (MAX31855PMB1)
- Maxim Integrated Proximity Sensor Pmod (MAX44000PMB1)

Experiment Setup

Setting Up the UltraZed-EG Starter Kit Hardware

Refer to the following figure and perform the following steps to set up the board.



1. Plug the UltraZed-EG SOM onto the IO Carrier Card via JX1/JX2/JX3 connectors and connect the fan to the fan header (JP5) on the IO Carrier Card.
2. Set the UltraZed-EG SOM Boot Mode switch (SW2) (MODE[3:0] = SW2[4:1]) to ON, ON, ON, and ON positions (Boot Mode set to JTAG, MODE[3:0] = 0x0).

3. Install a jumper on the IO Carrier Card JP1.
4. Connect the USB-JTAG port (SMT2 JTAG module) on the I/O Carrier Card (U18) to a free USB port on your PC.
5. Connect the USB-UART port on the I/O Carrier Card (J11) to a free USB port on your PC.
6. Plug the Maxim MAX44000 Pmod into the top row of the JA12 Pmod connector.
7. Plug the Maxim MAX31855 Pmod into the top row of the JA11 Pmod connector. Plug the thermocouple probe wire into the Pmod.
8. Connect the 12V power cable, but do not turn on the board yet.

PC Setup

Install the Zynq UltraScale+ ES1 License

Use the software voucher shipped with the UltraZed-EG Starter Kit to obtain and install the Vivado license to unlock the Zynq UltraScale+ ES1 devices.

Install the Vivado Board Definition Files

A set of Vivado Board Definition Files are provided for the UltraZed-EG Starter Kit in order to automate the hardware platform generation. Please unzip the following file:

`<installation>\Vivado_files\AES-ZU3EGES-1-SOM-G-Board_Definition_Files_v2016_2_Release.zip`
to the following folder of the Vivado 2016.2 install directory:
`<Xilinx_install>\Vivado\2016.2\data\boards\board_files`

Enable the Zynq UltraScale+ ES1 Devices in Vivado

- Create a text file called `init.tcl` and enter the following line:
`enable_beta_device*`
- Save and close the `init.tcl` file.
- Place the `init.tcl` file in the `<Xilinx_install>\Vivado\2016.2\scripts` folder

Installing the UART Driver and Virtual COM Port

If the UltraZed-EG Starter Kit has not been connected to the host PC before, it may be necessary to install the software driver for the virtual COM port. The driver installation for the Silicon Labs CP210x USB-UART bridge is described in detail in [Appendix I: Installation of USB UART Driver](#).

Installing a Serial Console on a Windows 7 Host

Starting with Windows 7, Microsoft no longer includes the HyperTerminal terminal emulator software. However, this example design requires use of terminal emulation software for a serial console connection to the UltraZed-EG Starter Kit. A suitable free and open-source replacement for HyperTerminal is TeraTerm. Download and install instructions for TeraTerm can be found at <http://en.sourceforge.jp/projects/ttssh2>. As an alternative the Terminal applet in the Xilinx SDK may also be used.

Supplied Files

The following directory structure is included with this reference design:

boot: Contains the boot files for the hardware platform.

boot.bin: Golden boot image to configure the PL and run the sensors example software.

doc: Contains the documentation for this reference design.

UZ3EG_IOCC_Sensors_Example_VIV2016_2.pdf: This document.

vivado_files: Contains the files to add to the Vivado installation.

AES-ZU3EGES-1-SOM-G-Board_Definition_Files_v2016_2_Release.zip: Board definition files.

init.tcl: TCL script to enable Zynq UltraScale+ device targets with ES1 silicon.

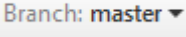
Reusable Components

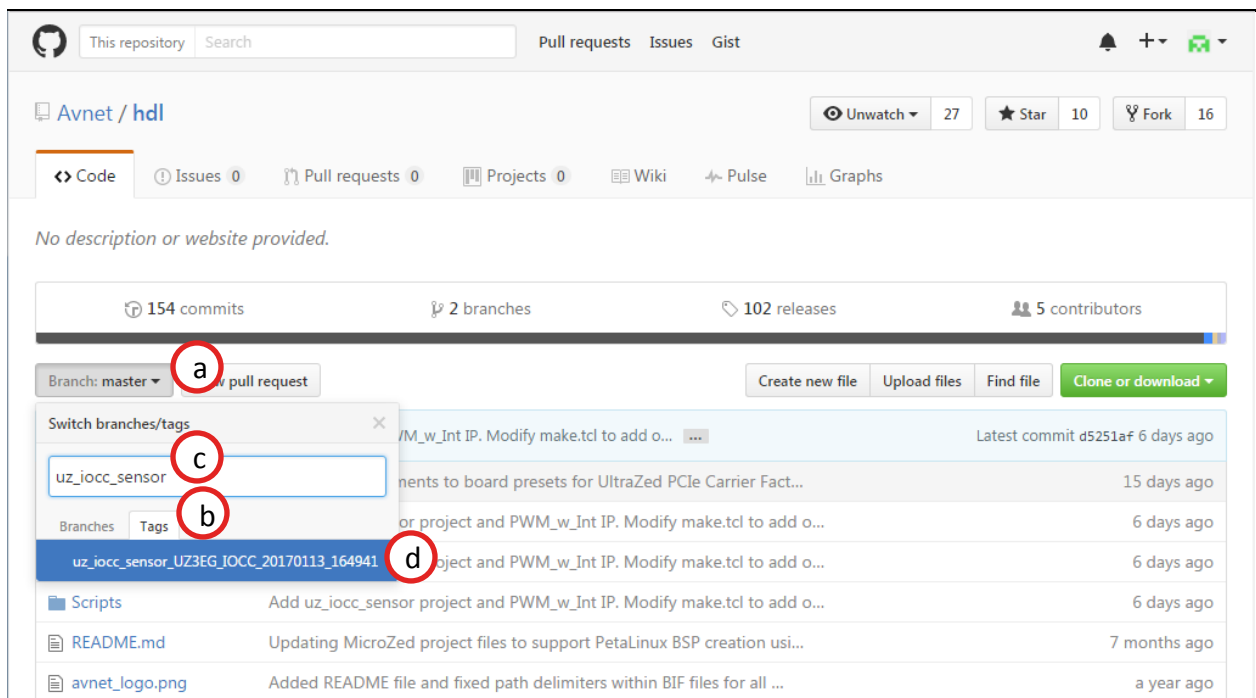
The tutorial will make use of components that can be reused in your own designs:


- TCL script: automatically build Vivado design from source code
 - **make_uz_iocc_sensor.tcl:** Builds entire project for UltraZed-EG.
- XDC constraints: Defines pinout and constraints for various carriers
 - **uz_iocc_sensor.xdc:** Constraints for UltraZed-EG Starter Kit platform

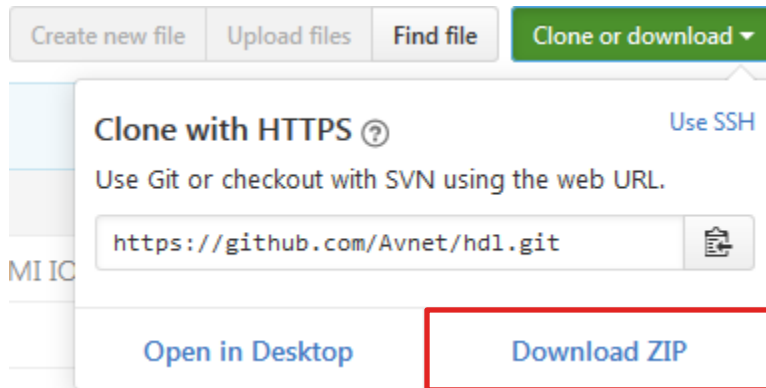
Experiment 1: Clone the Avnet HDL Repository

In this section, the design files for the reference design will be retrieved from the Avnet HDL git repository.

1. Use your favorite web browser and navigate to the following web site :
<https://github.com/Avnet/hdl>
2. The steps below describe how to fetch the specific project we want from the repository.
 - a. Click the **Branch:master**  button.
 - b. Click the **Tags** tab
 - c. Specify the following search criteria: **uz_iocc_sensor**
 - d. Select the **uz_iocc_sensor_UZ3EG_IOCC_20170113_164941** tag. This will retrieve a known working version of the design files for the Zynq hardware platform for the UltraZed-EG Starter kit.



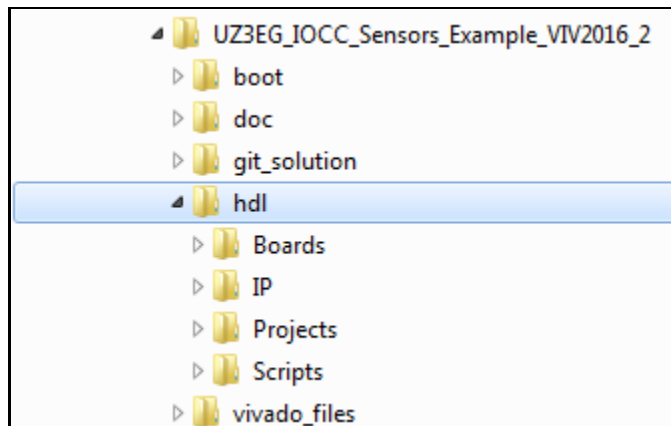
3. Click the Clone or download  button and select **Download zip**.



4. Save and extract the zip archive to a folder of your choice. Navigate to the cloned repository folder.

Note: To save space and make these tutorial instructions easier to read we will refer to this folder throughout this document as the <installation> folder.

5. Rename the **hdl-uz_iocc_sensor_UZ3EG_IOCC_20170113_164941** folder to **hdl**. You should see the following directory structure:



NOTE: The exact directory name is not critical, but it must remain short on Windows machines, due to the directory length limitation of Windows

The **<installation>\hdl** repository contains the following sub-directories:

Directory	Content Description
<installation>\hdl\Boards	contains board related files
<installation>\hdl\IP	contains the IP cores used by the ref designs
<installation>\hdl\Projects	contains project related files
<installation>\hdl\Scripts	contains scripts used to automatically build the designs

For the UltraZed-EG Sensors example design, the following content is of interest:

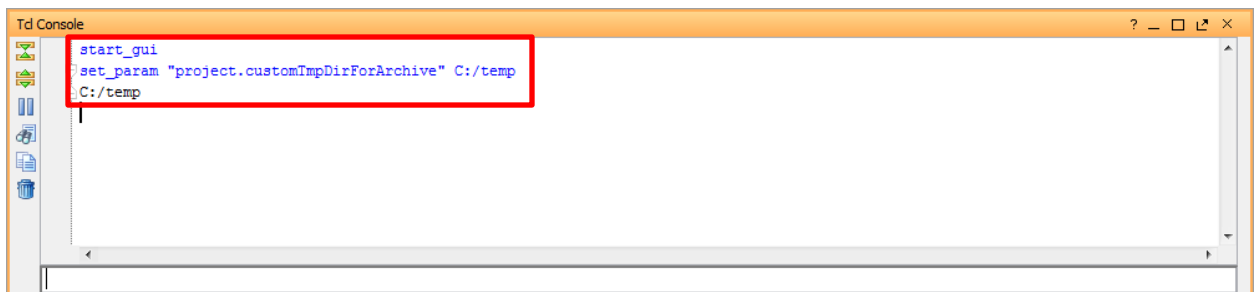
Directory	Content Description
< installation>\hdl\Projects\uz_iocc_sensor	Folder containing files for reference design on UltraZed-EG Starter Kit
< installation>\hdl\Scripts\make_uz_iocc_sensor.tcl	TCL script to launch the build of the reference design for a UltraZed-EG Starter target

Experiment 2: Build the UltraZed-EG Sensors Vivado Project

In this section, the Vivado project will be created and built with TCL scripts, implementing the Zynq UltraScale+ hardware platform reference design for UltraZed-EG Starter Kit.

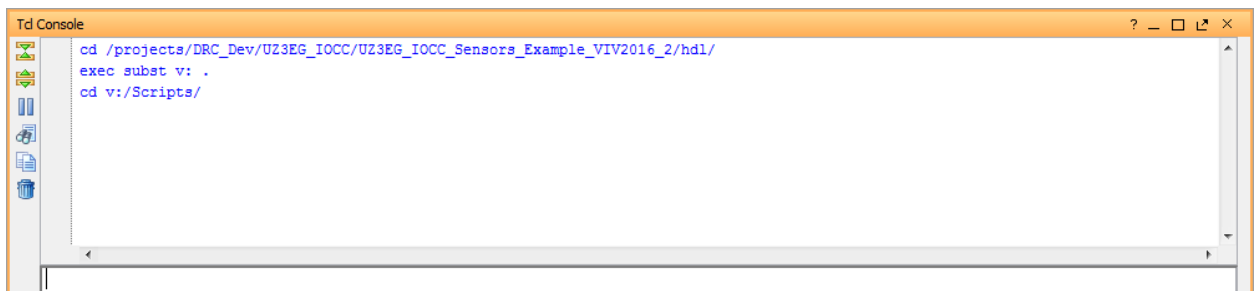
1. Launch Vivado 2016.2 from either the desktop icon or
Start → All Programs → Xilinx Design Tools → Vivado 2016.2 → Vivado 2016.2
2. If you are using the Vivado Design Suite on a Windows®-7 host, there is a known issue that Vivado file path names often exceed the maximum allowed by Windows. Please see [Appendix II](#) for more details. A workaround for this is to change the temp folder that Vivado uses. Create your own temporary directory named C:\temp, and force Vivado to use the new folder with the following TCL command:

```
set_param "project.customTmpDirForArchive" C:/temp
```



3. Another workaround for the 260 character path length limitation is to use the Windows **subst** command to substitute a drive letter for a long file path. Substitute the drive letter V: (or any other unused drive letter on your Windows PC) for the path where the zip archive of this example design was extracted. This can be done in the Vivado TCL Console:

```
cd <installation>/hdl  
exec subst v: .  
cd v:/scripts
```

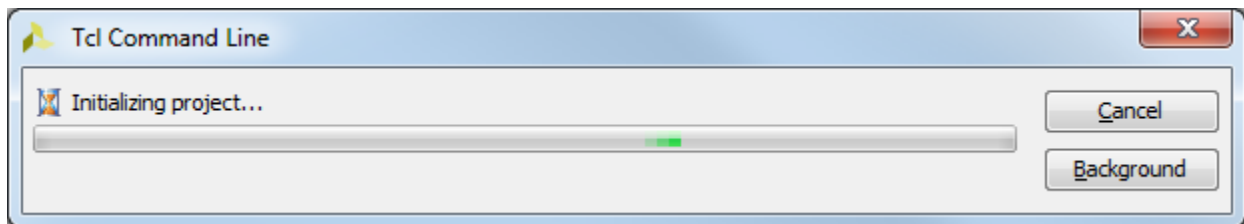


4. Launch the automated project build script with the following TCL command.

```
source ./make_uz_iocc_sensor.tcl
```



You will see the following window showing progress as Vivado works to build the design.



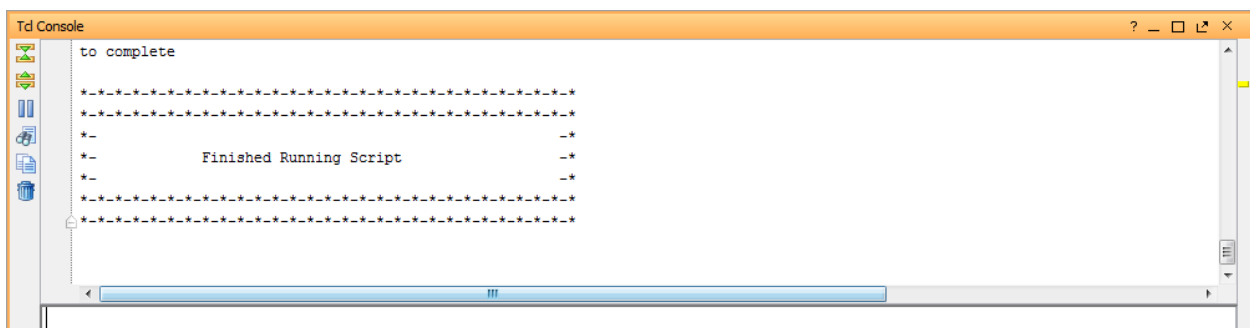
Wait about 5 to 10 minutes for the build to complete. The build will automatically perform the following steps for you.

- Create and build the hardware design with Vivado 2016.2, including the IP Integrator block design, and output a Programmable Logic bistream file. The resulting project resides at the following path:

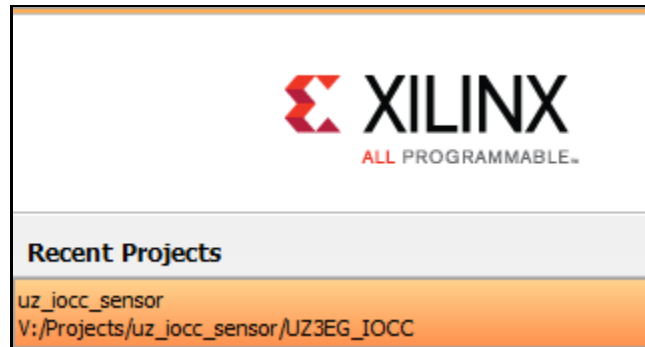
<installation>\hdl\Projects\uz_iocc_sensor\UZ3EG_IOCC\uz_iocc_sensor.xpr

- Create the FPGA bitstream file (uz_iocc_sensor_wrapper.bit)

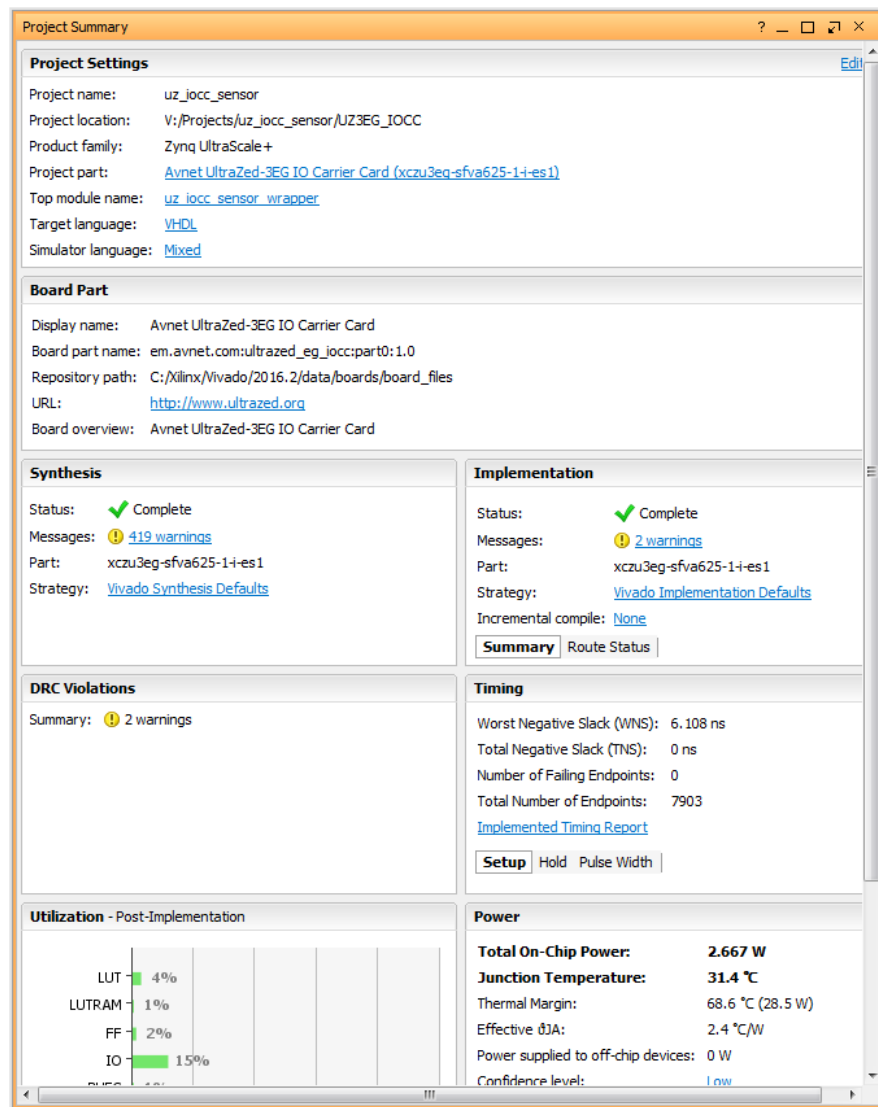
<installation>\hdl\Projects\uz_iocc_sensor\UZ3EG_IOCC\uz_iocc_sensor.runs\impl_1\uz_iocc_sensor_wrapper.bit



5. Open the completed design project by clicking on the **uz_iocc_sensor** project in the **Recent Projects** list.



6. In the **Project Summary** tab, you can find many high level details about the design such as an Implemented Timing Report, Device Utilization Report, and Power Report.



Project Summary

Project Settings

Project name: uz_iocc_sensor
Project location: V:/Projects/uz_iocc_sensor/UZ3EG_IOCC
Product family: Zynq UltraScale+
Project part: [Avnet UltraZed-3EG IO Carrier Card \(xczu3eg-sfva625-1-i-es1\)](#)
Top module name: [uz_iocc_sensor_wrapper](#)
Target language: [VHDL](#)
Simulator language: [Mixed](#)

Board Part

Display name: Avnet UltraZed-3EG IO Carrier Card
Board part name: em.avnet.com:ultrazed_eg_iocc:part0:1.0
Repository path: C:/Xilinx/Vivado/2016.2/data/boards/board_files
URL: <http://www.ultrazed.org>
Board overview: Avnet UltraZed-3EG IO Carrier Card

Synthesis

Status: ✔ Complete
Messages: ! 419 warnings
Part: xczu3eg-sfva625-1-i-es1
Strategy: [Vivado Synthesis Defaults](#)

Implementation

Status: ✔ Complete
Messages: ! 2 warnings
Part: xczu3eg-sfva625-1-i-es1
Strategy: [Vivado Implementation Defaults](#)
Incremental compile: [None](#)
Summary | [Route Status](#)

DRC Violations

Summary: ! 2 warnings

Timing

Worst Negative Slack (WNS): 6.108 ns
Total Negative Slack (TNS): 0 ns
Number of Failing Endpoints: 0
Total Number of Endpoints: 7903
[Implemented Timing Report](#)
Setup | [Hold](#) | [Pulse Width](#)

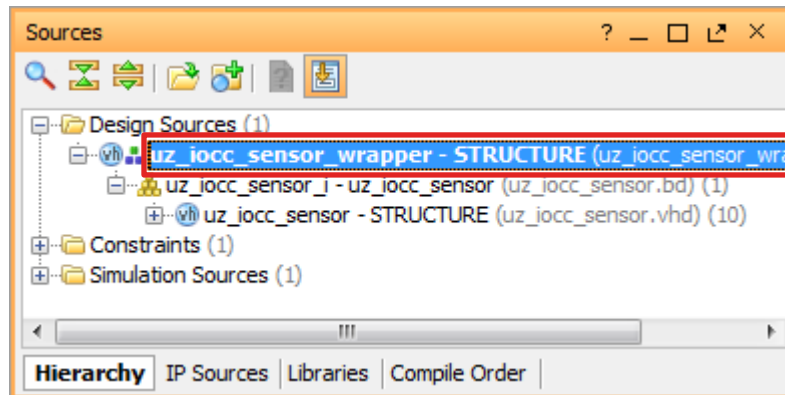
Utilization - Post-Implementation

Resource	Utilization
LUT	4%
LUTRAM	1%
FF	2%
IO	15%

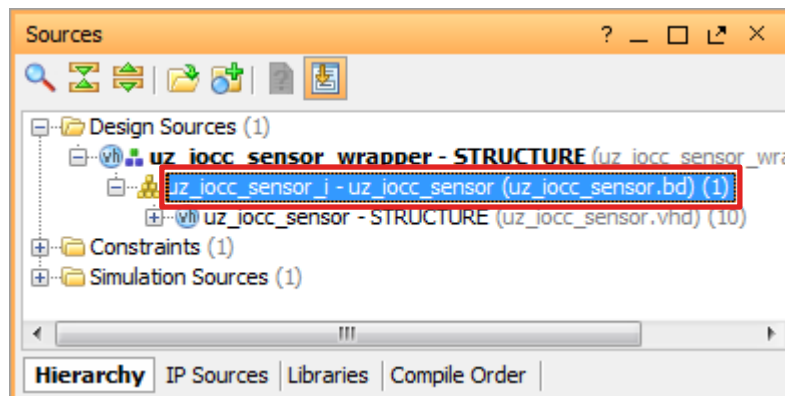
Power

Total On-Chip Power: 2.667 W
Junction Temperature: 31.4 °C
Thermal Margin: 68.6 °C (28.5 W)
Effective θ_{JA} : 2.4 °C/W
Power supplied to off-chip devices: 0 W
Confidence level: [Low](#)

7. In the Sources tab, expand the **uz_iocc_sensor_wrapper** branch of the **Design Sources** tree to see the items under the top level design wrapper.



8. Double-click the **uz_iocc_sensor.bd** entry to open the block design for the project

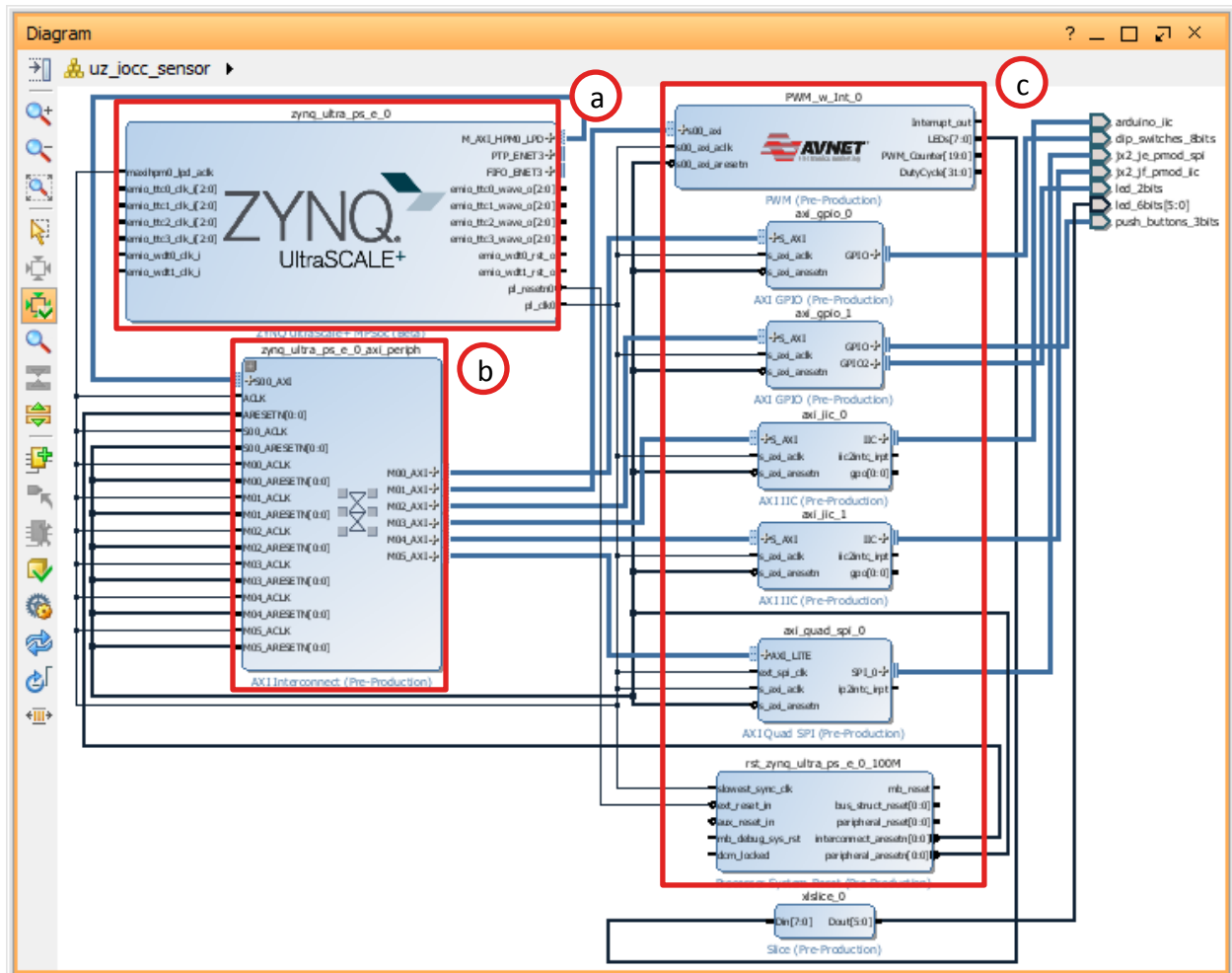


9. Explore the block design **Diagram** tab that is now opened. Note that there are three major components for this basic design:

- Zynq UltraScale+ Processing System (zynq_ultra_ps_e_0)
- AXI Interconnect which acts as means to move data between the processor and the peripherals in the PL (zynq_ultra_ps_e_0_axi_periph)
- AXI peripherals for GPIO, I2C, QSPI, and PWM


Vivado uses this block design to drive the synthesis and implementation capability of this tool to generate a Programmable Logic file which can be loaded into the FPGA portion of the Zynq device.

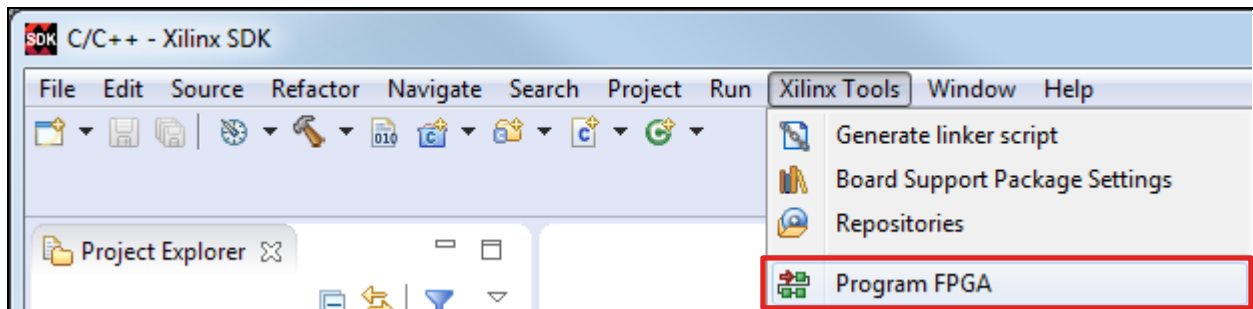
This block design is very customizable and could easily be used as a basis for expansion with other sensor modules to fit a particular application.



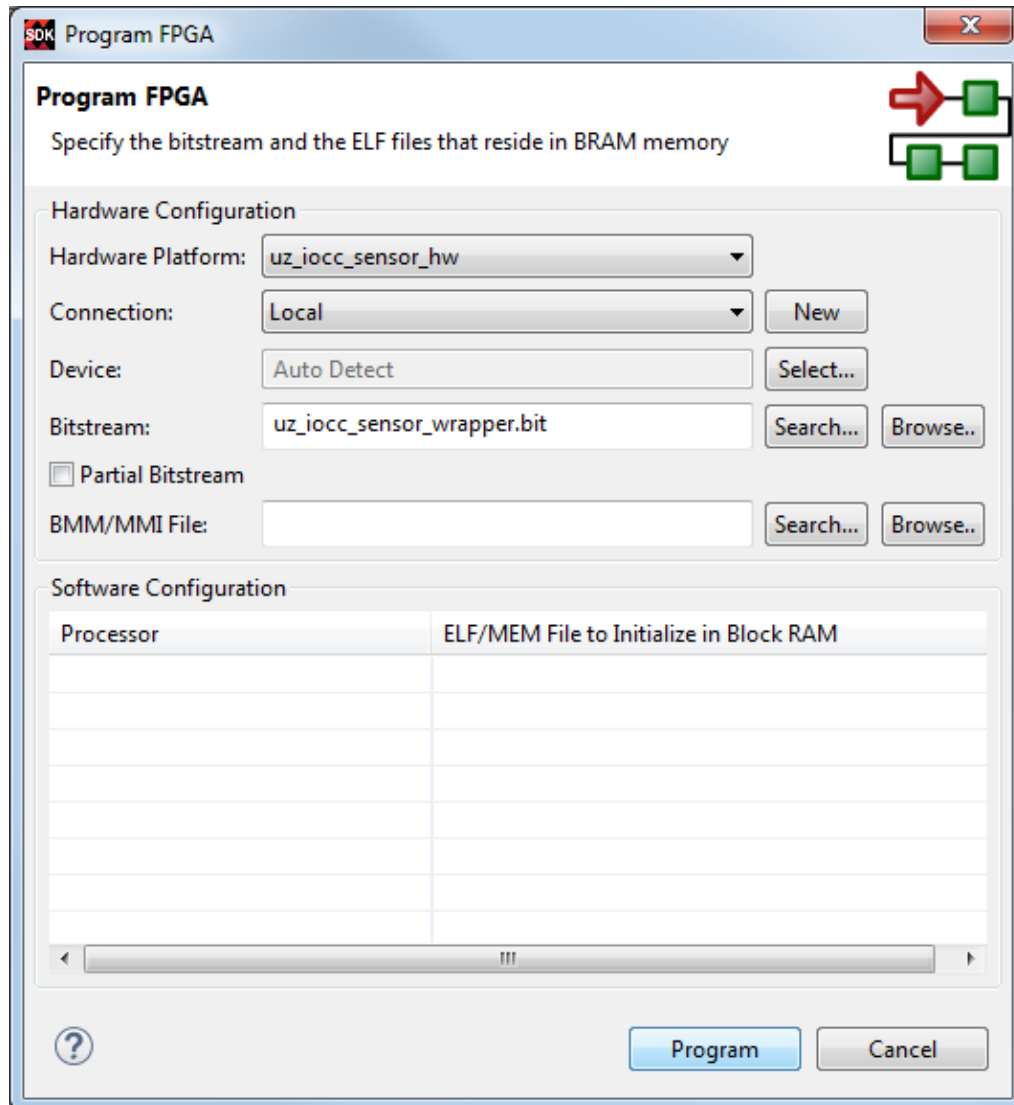
Experiment 3: Run the Supplied Software Application

The script we ran to build the Zynq UltraScale+ hardware platform in Vivado also created a new SDK software workspace with a new First Stage BootLoader (FSBL) and supplied software applications.

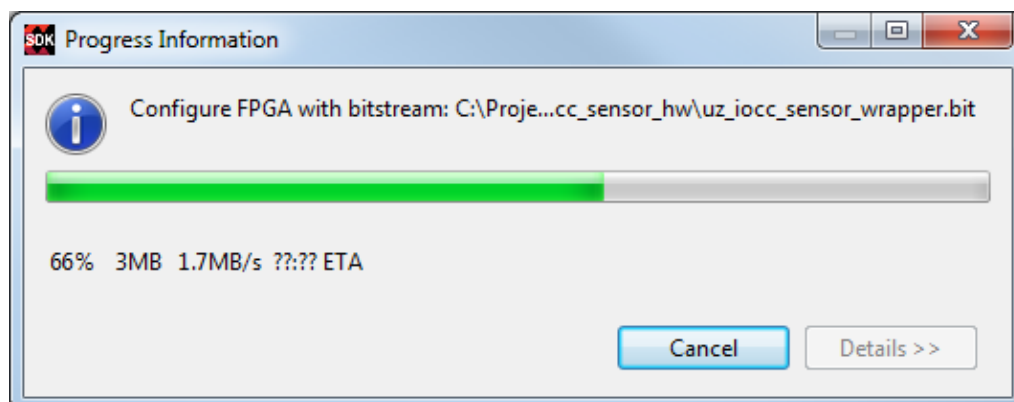
1. If not already done, connect the UltraZed-EG Starter Kit to the host PC as described earlier in [Setting Up the UltraZed-EG Starter Kit Hardware](#).
2. Turn on the UltraZed-EG Start Kit. The power switch (SW8) is near the 12V power connector on the I/O Carrier Card.
3. Start a serial terminal session using your terminal software of choice and set the serial port parameters to **115200** baud rate, **no** parity, **8** bits, **1** stop bit and no flow control. Determine the COM port used by the board as described in [Appendix I: Determining the Virtual COM Port](#).
4. In the SDK main menu, select **Xilinx Tools** → **Program FPGA** or click on the  on the SDK toolbar (hidden by the menu in the screenshot below):




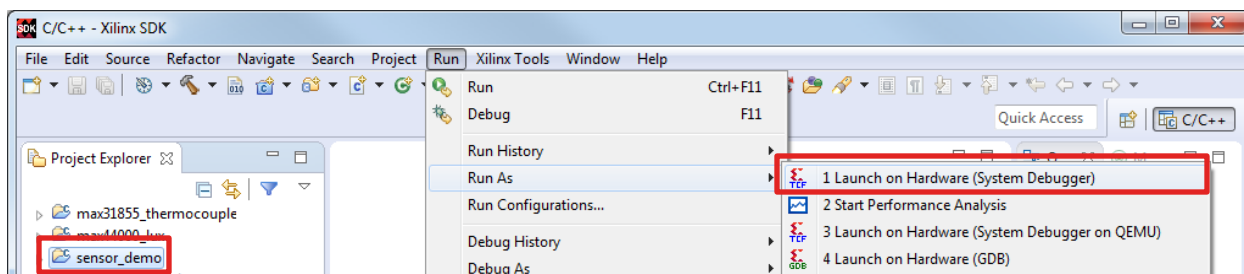
5. Accept the defaults for the **Hardware Platform**, **Connection**, and **Bitstream**. Click **Program** to configure the PL with the hardware bitstream.



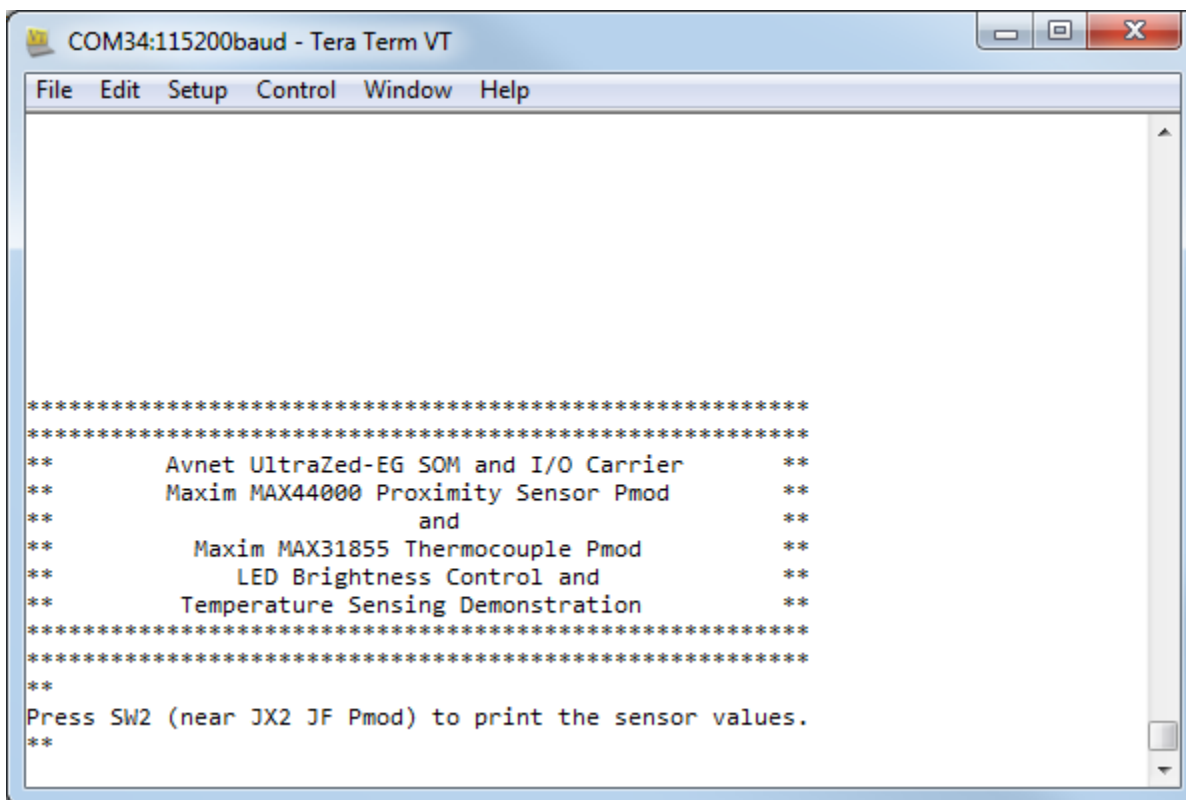
6. You will see a progress window as the PL is configured.



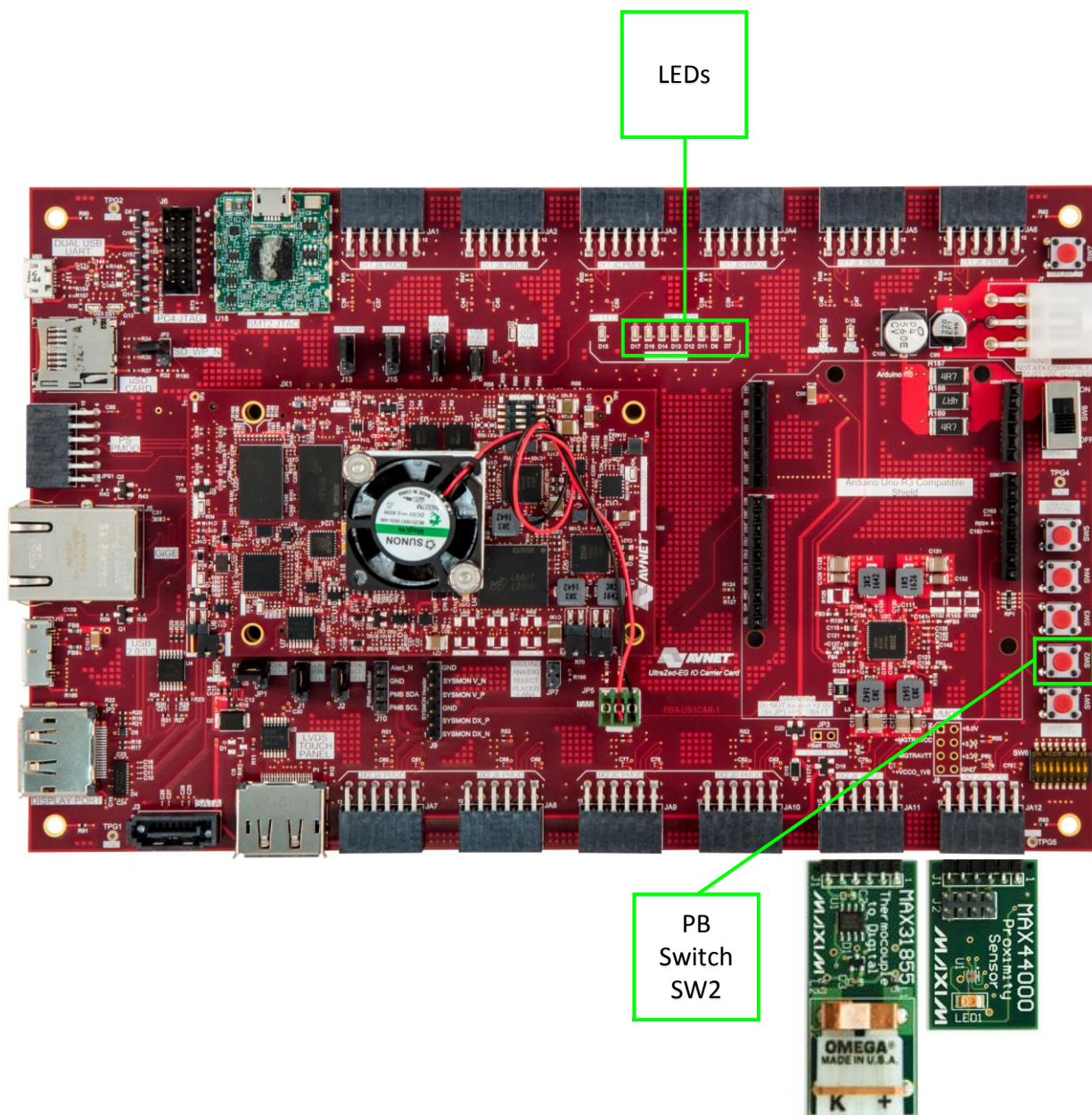
7. There are a few supplied software applications that are pre-built in the SDK. Two for individually demonstrating the Maxim thermocouple and proximity sensors, and another, **sensor_demo**, that displays the readings from each sensor. This is the application that we want to run now. The mechanism to download the sensor demo application and run it on the board is to create a run configuration. Click on the **sensor_demo** application in the **Project Explorer** pane and in the SDK main menu, select **Run → Run As → Launch on Hardware (System Debugger)** or click on the  on the SDK toolbar (hidden by the menu in the screenshot below).



8. You should see the following on the serial console:



9. Each time SW2 (near the JX2 JF Pmod) is pressed a new set of temperature and light sensor readings will be displayed. Refer to the figure below for the location of the Pushbutton switches and LEDs.

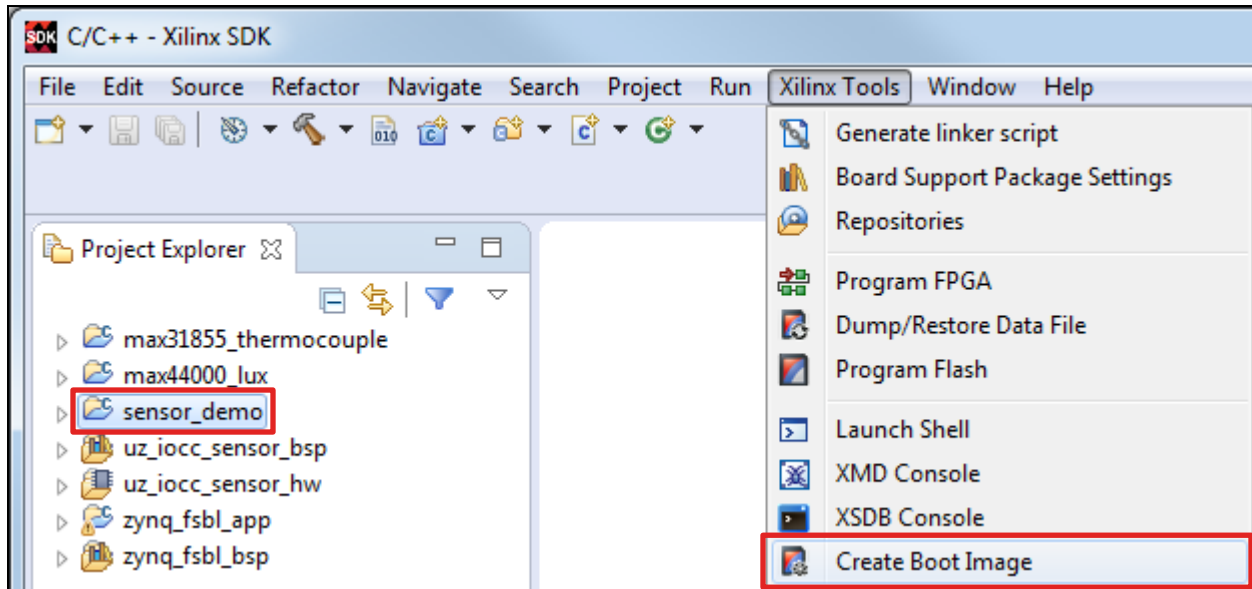


10. Notice that as the light reading from the MAX44000 sensor changes, the LEDs (D7-D14) get brighter or dimmer. If the light into the sensor is especially dim the D16 LED will turn on. Conversely, if the light reaching the sensor is extremely bright the D17 LED will turn on.
11. Also experiment with the temperature sensor. Hold the end of the thermocouple wire in your fingertips or place it near something that is hotter or colder than the ambient temperature. Notice the temperature reading change as you press SW2.

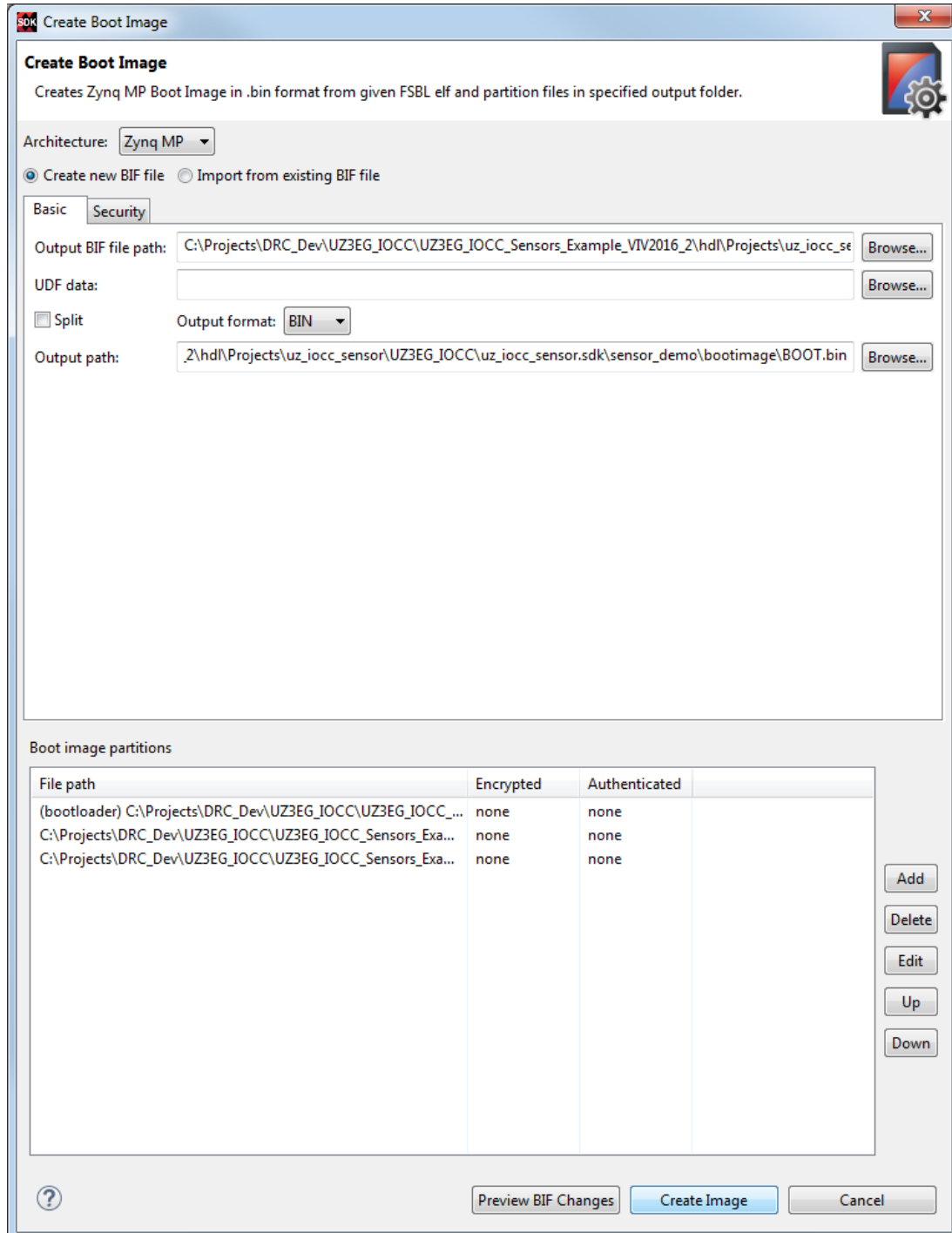
Experiment 4: Create a New Zynq Boot File

With the PL bitstream built and the FSBL and sensor demo software application created, we are ready to create the boot file to boot the UltraZed-EG from a microSD card.

1. Click on the sensor_demo software application in the Project Explorer pane and on the SDK main menu go to **Xilinx Tools** → **Create Boot Image**.



- The **Create Boot Image** window will pre-populate with default settings for the required components of the FSBL, PL bitstream, and sensor demo software application. Review the partitions and their order. The FSBL must come first, followed by the PL bitstream, and then the software application binary. Accept the defaults and click **Create Image** to create the new boot.bin boot image. Note the **Output Path** is set to `<installation>\hdl\Projects\uz_iocc_sensor\UZ3EG_IOCC\uz_iocc_sensor.sdk\sensor_demo\bootimage\BOOT.bin`. We will need this in the next steps.



Create Boot Image

Creates Zynq MP Boot Image in .bin format from given FSBL elf and partition files in specified output folder.

Architecture: Zynq MP

☒ Create new BIF file ☐ Import from existing BIF file

Basic Security

Output BIF file path: C:\Projects\DRC_Dev\UZ3EG_IOCC\UZ3EG_IOCC_Sensors_Example_VIV2016_2\hdl\Projects\uz_iocc_sensor\UZ3EG_IOCC\uz_iocc_sensor.sdk\sensor_demo\bootimage\BOOT.bin Browse...

UDF data: Browse...

☐ Split Output format: BIN

Output path: C:\Projects\DRC_Dev\UZ3EG_IOCC\UZ3EG_IOCC_Sensors_Example_VIV2016_2\hdl\Projects\uz_iocc_sensor\UZ3EG_IOCC\uz_iocc_sensor.sdk\sensor_demo\bootimage\BOOT.bin Browse...

Boot image partitions

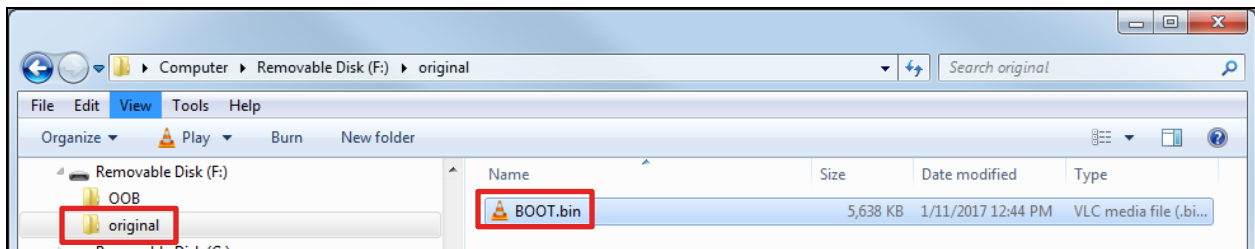
File path	Encrypted	Authenticated
(bootloader) C:\Projects\DRC_Dev\UZ3EG_IOCC\UZ3EG_IOCC_Sensors_Example_VIV2016_2\hdl\Projects\uz_iocc_sensor\UZ3EG_IOCC\uz_iocc_sensor.sdk\sensor_demo\bootimage\BOOT.bin	none	none
C:\Projects\DRC_Dev\UZ3EG_IOCC\UZ3EG_IOCC_Sensors_Example_VIV2016_2\hdl\Projects\uz_iocc_sensor\UZ3EG_IOCC\uz_iocc_sensor.sdk\sensor_demo\bootimage\BOOT.bin	none	none
C:\Projects\DRC_Dev\UZ3EG_IOCC\UZ3EG_IOCC_Sensors_Example_VIV2016_2\hdl\Projects\uz_iocc_sensor\UZ3EG_IOCC\uz_iocc_sensor.sdk\sensor_demo\bootimage\BOOT.bin	none	none

? Preview BIF Changes Create Image Cancel

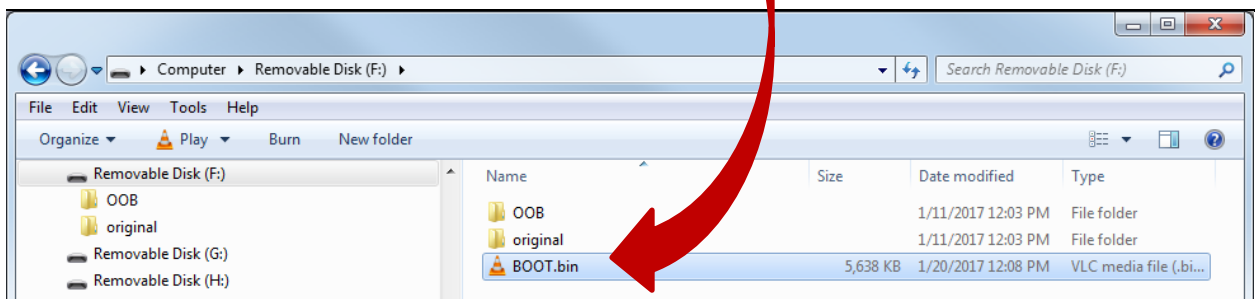
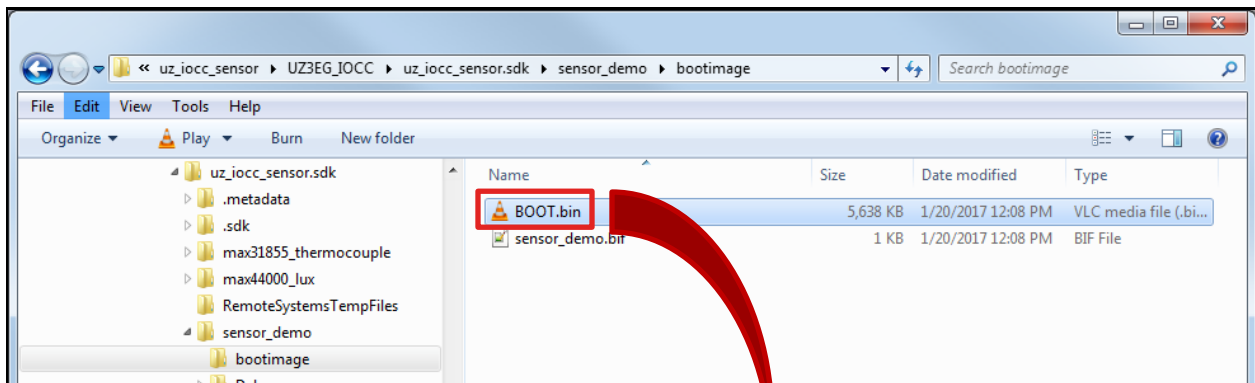
Add Delete Edit Up Down

Experiment 5: Boot the System From microSD Card

1. Retrieve the microSD card that came with the UltraZed-EG Starter Kit. If it is not available you can use a different microSD card. Insert the microSD card into an available port on your PC.
2. Using the file manager of your choice, navigate to the microSD card and backup the original **boot.bin** file if it exists. Create a backup folder named **original** to hold the original file.

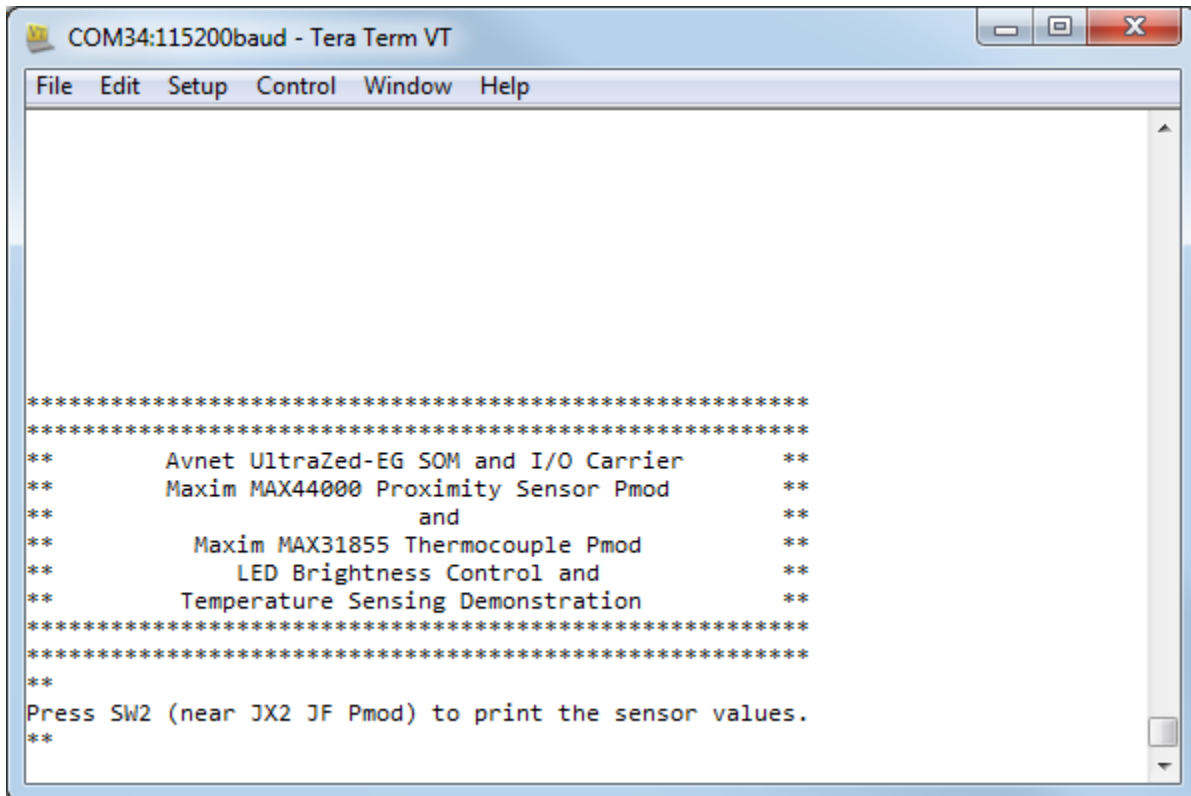


3. Navigate to the `<installation>\hdl\Projects\uz_iocc_sensor\UZ3EG_IOCC\uz_iocc_sensor.sdk\sensor_demo\bootimage` folder and copy the new **boot.bin** file to the root directory of the microSD card. Eject the microSD card from the PC when finished.



4. Turn off the UltraZed-EG Starter Kit if it is turned on.
5. Set the UltraZed-EG SOM Boot Mode switch (SW2) (MODE[3:0] = SW2[4:1]) to ON, OFF, ON, and OFF positions (Boot Mode set to JTAG, MODE[3:0] = 0xA).

6. Insert the microSD card in the cage (J4).
7. Turn the UltraZed-EG Starter Kit back on. You should see the FSBL run in the serial console, then the PL get configured (the DONE LED will turn on), and then the sensor demo software application will start.



```
*****
*****
** Avnet UltraZed-EG SOM and I/O Carrier **
** Maxim MAX44000 Proximity Sensor Pmod **
** and **
** Maxim MAX31855 Thermocouple Pmod **
** LED Brightness Control and **
** Temperature Sensing Demonstration **
*****
*****
**
Press SW2 (near JX2 JF Pmod) to print the sensor values.
**
```

8. Feel free to examine the Vivado hardware platform or SDK software applications for how you might adapt them to your own designs. What improvements or changes would you make?

This concludes this design tutorial.

Appendix I: Installation of USB UART Driver

Many of the Avnet evaluation boards are equipped with the Silicon Labs CP2102 USB-to-UART Bridge IC. This connects a PC's USB port to the evaluation board and looks like a UART to the PC. A virtual COM port will be created on the PC by means of a Silicon Labs CP2102 USB-to-UART bridge driver. Follow the instructions listed below to install the Silicon Labs drivers.

Download and Install the Required Software

1. Using your web browser, navigate to the Silicon Labs website:

<http://www.silabs.com/products/mcu/pages/usbtouartbridgevcpdrivers.aspx>

2. Download the **VCP Driver Kit** for your PC's operating system. Drivers for MacOS and Linux are also available.



SILICON LABS

Silicon Labs » Products » MCUs » USB to UART Bridge VCP Drivers

CP210x USB to UART Bridge VCP Drivers

The CP210x USB to UART Bridge Virtual COM Port (VCP) drivers are required for device operation as a Virtual COM Port to facilitate host communication with CP210x products. These devices can also interface to a host using the [USBXpress](#) direct access driver. These drivers are static examples detailed in application note 197: The Serial Communications Guide for the CP210x, download an example below:

 [AN197: The Serial Communications Guide for the CP210x](#)

Download Software

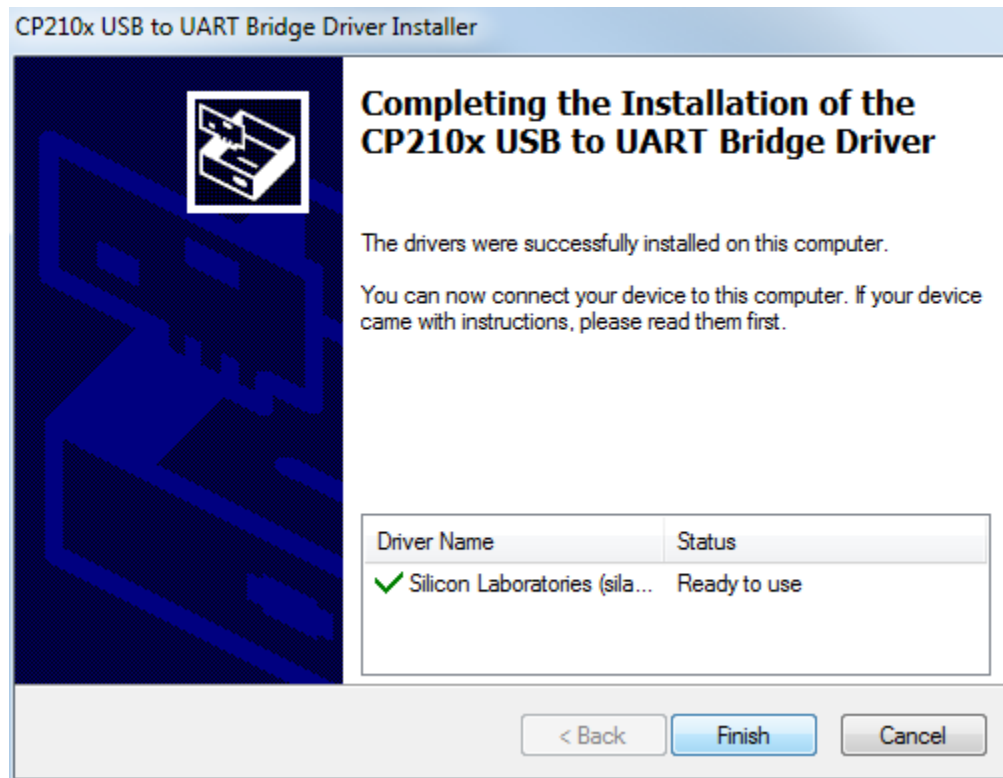
The CP210x Manufacturing DLL and Runtime DLL have been updated and must be used with v6.0 and later of the CP210x Windows VCP Driver. Application Note Software downloads affected are AN144SW.zip, AN205SW.zip and AN223SW.zip. If you are using a 5.x driver and need support you can download archived [Application Note Software](#).

Download for Windows XP/Server 2003/Vista/7/8/8.1 (v6.7)

Platform	Software	Release Notes
 Windows XP/Server 2003 Vista/7/8/8.1	Download VCP (3.66 MB)	Download VCP Revision History

GET THE LATEST DOCUMENTATION UPDATES.


3. Once the file is downloaded, extract the **CP210x VCP Driver Kit** archive. For example, for Windows XP/Vista/7 the file is CP210x_VCP_Windows.zip. Once the archive is extracted, open the folder where the archive was extracted and choose the correct installer for a 32-bit (CP210xVCPInstaller_x86.exe) or 64-bit (CP210xVCPInstaller_x64.exe) PC. The installer will guide you through the setup. Accept the license agreement and install the software on your PC. Click **FINISH** when completed.

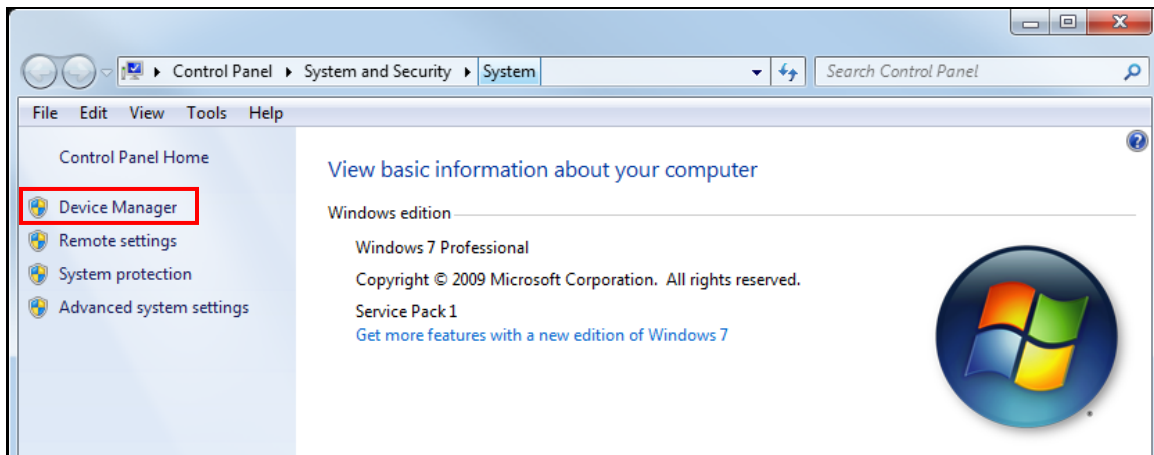


Determining the Virtual COM Port

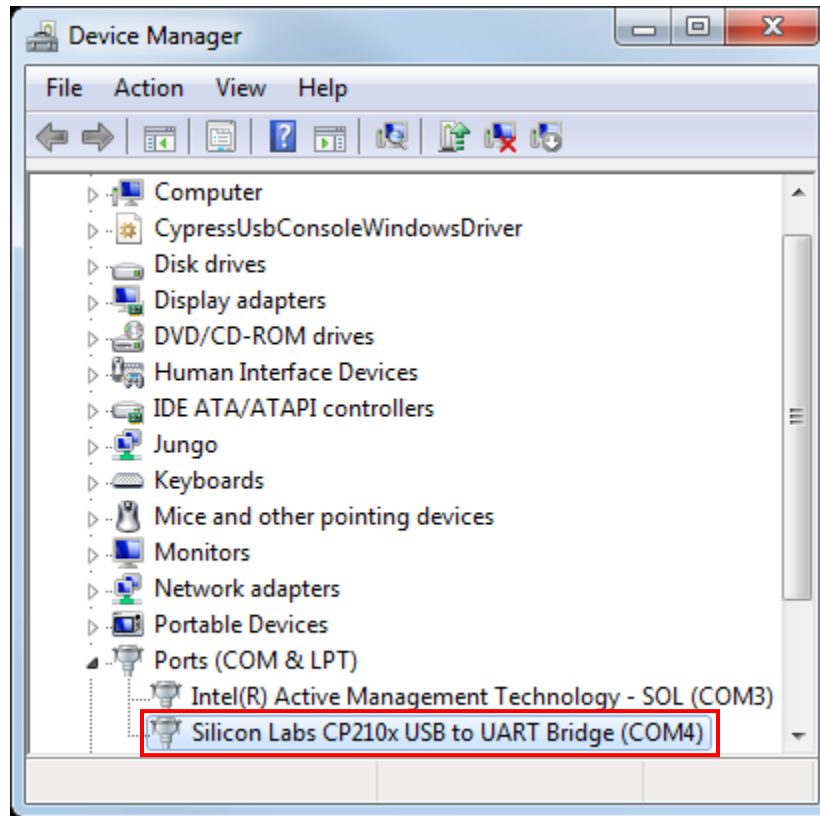
Now you can connect the evaluation board's USB-to-UART port to one of the USB ports on your PC. The new hardware detection will pop up and enumeration of the driver will be started. Once finished a virtual COMx port is created and you are ready to setup a connection using Windows HyperTerminal or comparable serial terminal emulation utility. Follow these instructions to determine the COMx port assigned to the USB-to-UART bridge:



1. Open the Device Manager by right-clicking on , select Properties, then click on the Device Manager.



2. In the Device Manager, scroll down to Ports and expand the list. You will see the Silicon Labs CP210x USB to UART Bridge and its assigned COM port. In the example below, it is COM4. Make note of this COM port number for use with the serial terminal you will use elsewhere in this design tutorial. This concludes these USB UART driver and virtual COM port installation instructions.



Appendix II: Windows 260 Character Path Limit

If you are using the Vivado Design Suite on a Windows-7 host, you may run into issues resulting from Vivado pathnames exceeding the maximum allowed. Vivado projects create a very deep file hierarchy, and it becomes very easy to violate the Windows limit if the project is not extracted near the root of the drive. This can even happen when the archive is decompressed, depending on where you choose to place the project in your existing file hierarchy.

It is not always convenient to place every Vivado project at the root of a drive. To work around this limitation, the recommended procedure is to place the Vivado archive in a shared folder on your host machine, then use Windows Explorer to map a network drive to the directory where the archive will be decompressed. This allows the Vivado project to be mapped to the root of the virtual (mapped) directory, eliminating any path issues.

Vivado also makes use of the Windows temp folder, which may be located several folders deep from the root drive, and this can also cause problems. You can create your own temporary directory in C:\temp, and force Vivado to use the new folder with the following TCL:

```
set_param "project.customTmpDirForArchive" C:/temp
```

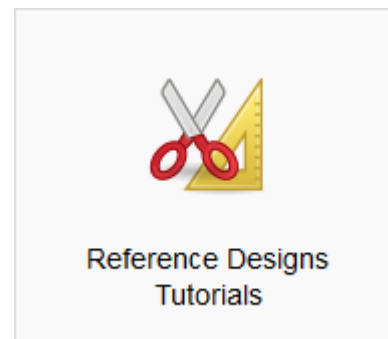
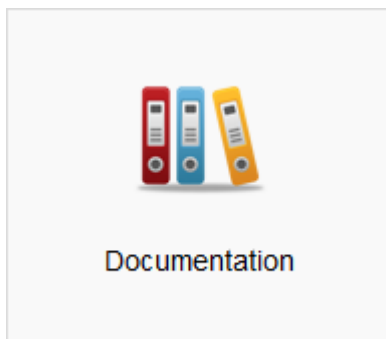
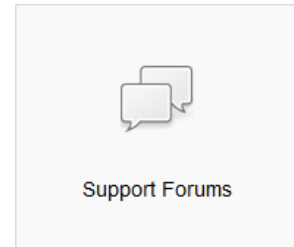
For further information, see the Xilinx answer record at:

<https://www.xilinx.com/support/answers/52787.html>

Appendix III: Getting Support

Avnet Support

- Technical support is offered online through the ultrazed.org website support forums. UltraZed-EG Starter Kit users are encouraged to participate in the forums and offer help to others when possible.
<http://ultrazed.org/forums/zed-english-forum>
<http://ultrazed.org/forums/software-application-development>
- For questions regarding the UltraZed-EG community website, please direct questions to the ultrazed.org Web Master (webmaster@ultrazed.org).
- To access the most current collateral for the UltraZed-EG Starter Kit, visit the community support page (www.ultrazed.org/content/support) and click one of the icons shown below:



- UltraZed-EG Starter Kit Documentation
<http://ultrazed.org/support/documentation/17596>
- UltraZed-EG Starter Kit Reference Designs
<http://ultrazed.org/support/design/17596/131>
- Instructions for how to setup the Ubuntu virtual machine if using a Linux host PC
http://ultrazed.org/sites/default/files/design/VirtualBox_Installation_Guide_2016_2.zip

Xilinx Support

For questions regarding products within the Product Entitlement Account, send an email message to the Customer Service Representative in your region:

- Canada, USA and South America - isscs_cases@xilinx.com
- Europe, Middle East, and Africa - eucases@xilinx.com
- Asia Pacific including Japan - apaccase@xilinx.com

For technical support, including the installation and use of the product license file, contact Xilinx Online Technical Support at www.xilinx.com/support. The following assistance resources are also available on the website:

- Software, IP and documentation updates
- Access to technical support Web tools
- Searchable answer database with over 4,000 solutions
- User forums

Revision History

Date	Version	Revision
20 Jan 2017	1.0	Initial Release