



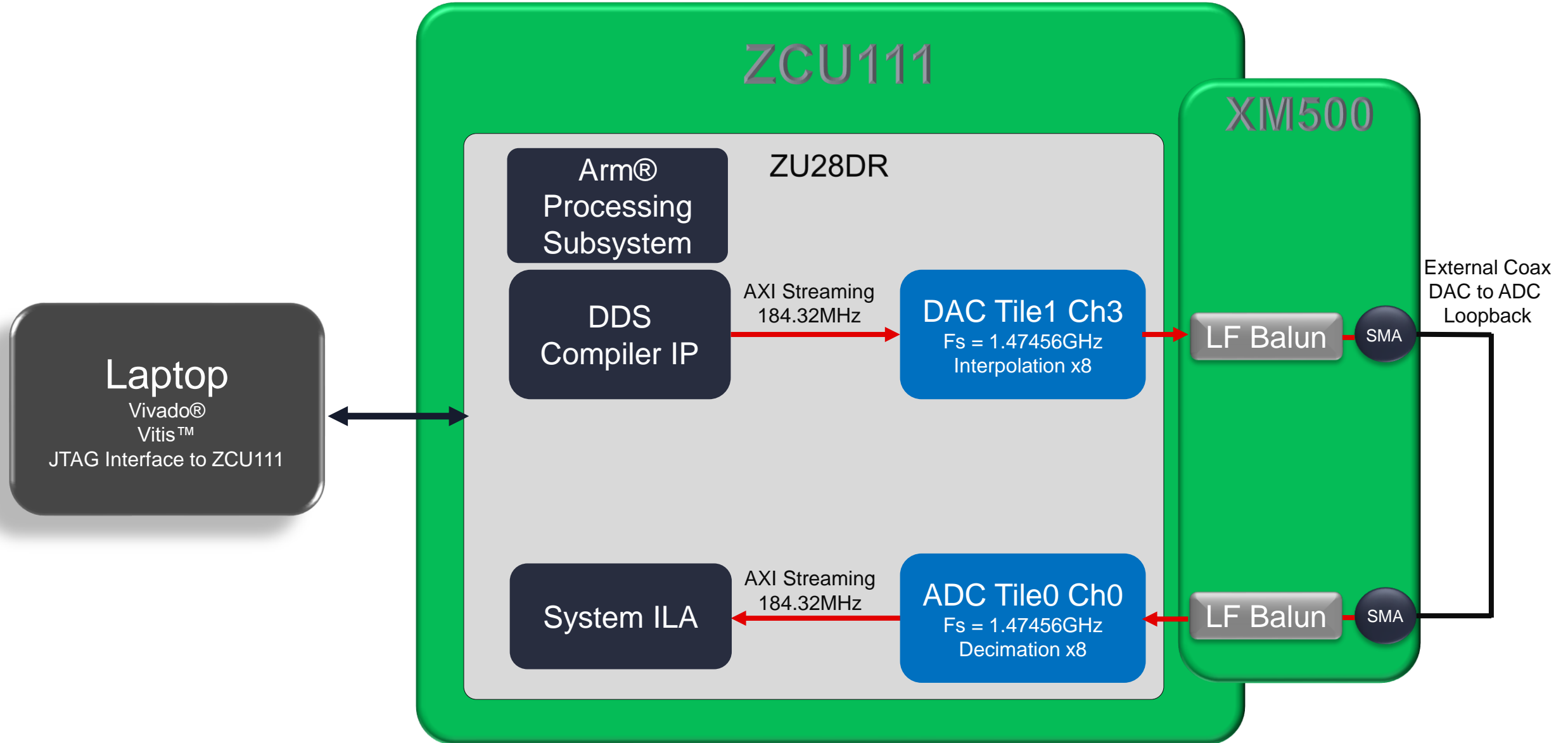
Zynq® UltraScale+™ RFSoc Example Design: ZCU111

DDS Compiler for DAC and System
ILA for ADC Capture – 2020.2

Introduction

- ▶ This is an example starter design for the RFSoc.
- ▶ It uses the ZCU111 board.
- ▶ It uses a DAC and ADC sample rate of 1.47456GHz.
- ▶ The DAC will continuously play 10MHz sine wave from the DDS Compiler IP.
- ▶ The ADC output will be sent to a System ILA to be displayed in the Hardware Manager.
- ▶ DAC Tile1 Ch3 will be used (LF balun).
- ▶ ADC Tile0 Ch0 will be used (LF balun).
- ▶ 2020.2 Xilinx tools (Vivado® Design Suite and Vitis™ unified software platform).
- ▶ Design tested in the directory `c:\rfsoc\ex_des\zcu111\v4\`
- ▶ This kit comes with the Vivado HW project and SW source files.

Demo Block Diagram



ZCU111 Clocks

- This design automatically programs the clocks to 1.47456GHz via the SW application.
- ZCU111 ADC/DAC clocks are generated from LMK04208 feeding 3 LMX2594 in parallel. Review 'RF Data Converter Clocking' in [UG1271](#) (ZCU111 board user guide).
- Clocking is configured via an I2C to SPI bridge.

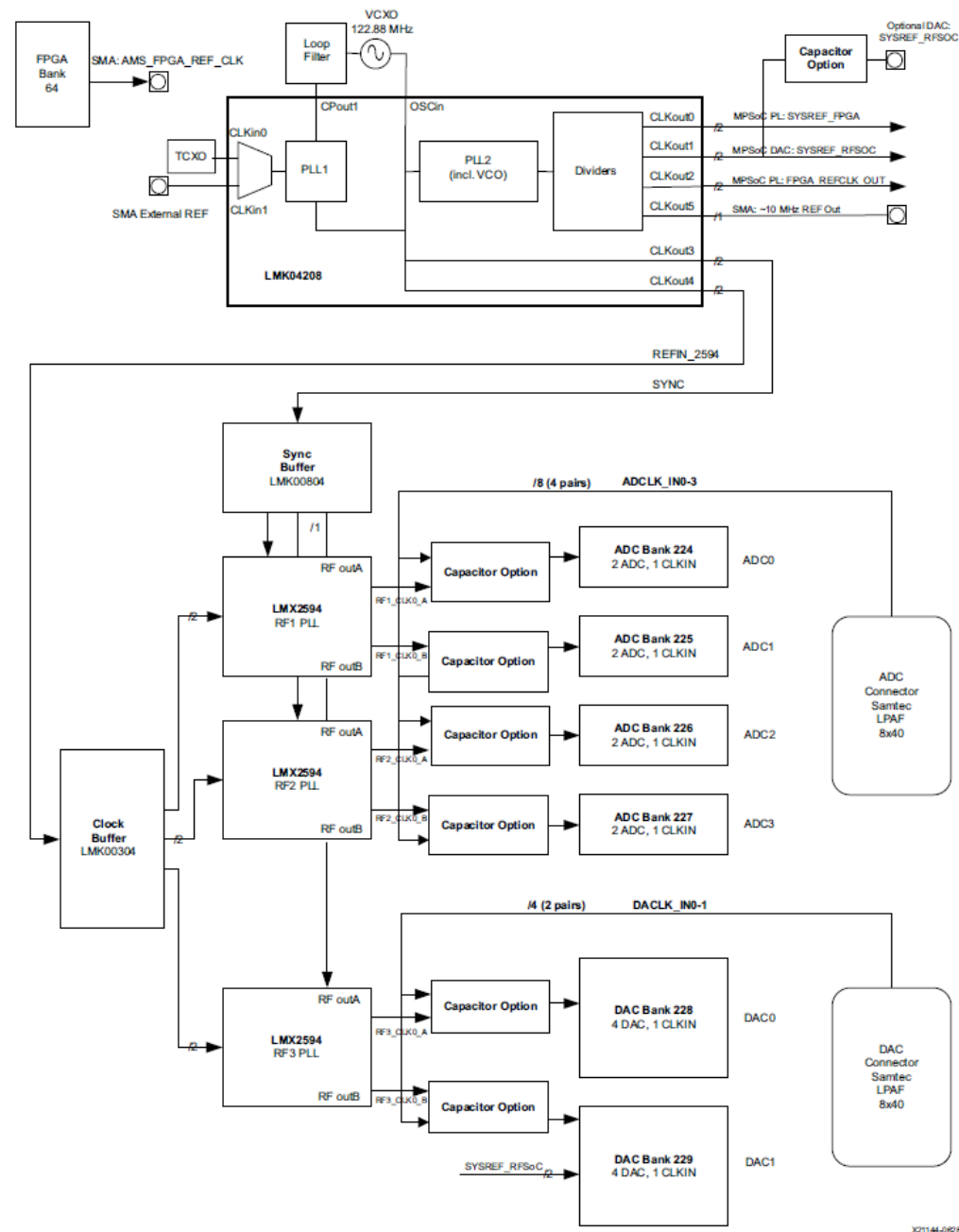


Figure 3-18: RF Clocking Structure for ADC and DAC Banks

DAC Setup

Basic

System Clocking

Advanced

Converter Setup

Converter Setup

Advanced

Changing Converter Setup to Simple will cause current Advanced IP configuration to be lost.

RF-ADC

RF-DAC

DAC Tile 228

DAC Tile 229

Multi Tile Sync

Converter Band Mode

☐ Enable Multi Tile Sync

Band

Single

Converter Configuration

DAC Pair 0,1

DAC Pair 2,3

DAC 2

DAC 3

☐ Enable DAC

☐ Invert Q Output

☐ Inverse Sinc Filter

Data Settings

Analog Output Data

Real

Interpolation Mode

Off

Samples per AXI4-Stream Cycle

16

Mixer Settings

Mixer Type

Off

Analog Settings

Nyquist Zone

Zone 1

Decoder Mode

SNR Optimized

☒ Enable DAC

☐ Invert Q Output

☐ Inverse Sinc Filter

Data Settings

Analog Output Data

Real

Interpolation Mode

8x

Samples per AXI4-Stream Cycle

1

Required AXI4-Stream clock: 184.320 MHz

Mixer Settings

Mixer Type

Bypassed

Mixer Mode

Real->Real

Analog Settings

Nyquist Zone

Zone 1

Decoder Mode

SNR Optimized

ADC Setup

Basic

System Clocking

Advanced

Converter Setup

Converter Setup

Advanced

Changing Converter Setup to Simple will cause current Advanced IP configuration to be lost.

RF-ADC

RF-DAC

ADC Tile 224

ADC Tile 225

ADC Tile 226

ADC Tile 227

Multi Tile Sync

Converter Band Mode

Link Coupling

☐ Enable Multi Tile Sync

Band

Single

Link Coupling

AC

Converter Configuration

ADC 0

ADC 1

☒ Enable ADC

☐ Invert Q Output

☐ Dither

☐ Bypass Background Calibration

Data Settings

Digital Output Data

Real

Decimation Mode

8x

Samples per AXI4-Stream Cycle

1

Required AXI4-Stream clock: 184.320 MHz

Mixer Settings

Mixer Type

Bypassed

Mixer Mode

Real->Real

Analog Settings

Nyquist Zone

Zone 1

Calibration Mode

Mode2

☐ Enable ADC

☐ Invert Q Output

☒ Dither

☐ Bypass Background Calibration

Data Settings

Digital Output Data

Real

Decimation Mode

Off

Samples per AXI4-Stream Cycle

8

Mixer Settings

Mixer Type

Off

Analog Settings

Nyquist Zone

Zone 1

Calibration Mode

Mode2

Data Converter Clocking

Basic

System Clocking

Advanced

AXI4-Lite Interface Configuration

AXI4-Lite Clock (MHz) 100.0

Tile Clocking Settings

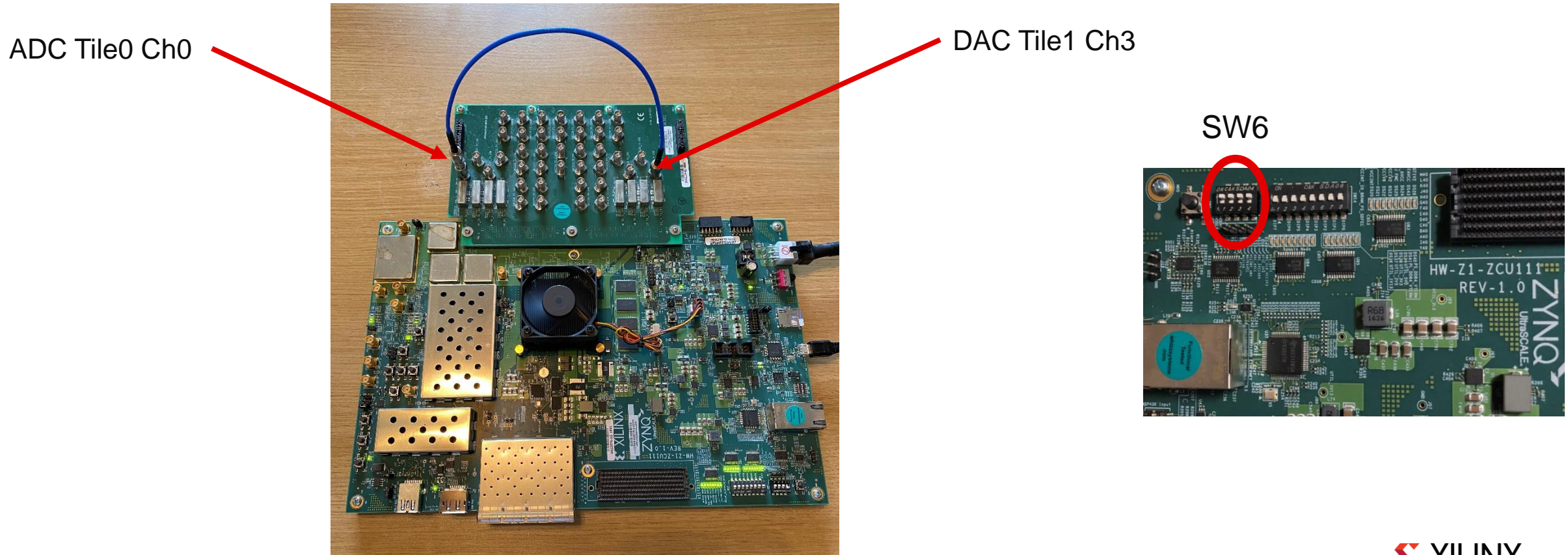
Tile	Sampling Rate (G SPS)	Max Fs (G SPS)	PLL	Reference Clock (MHz)	PLL Ref Clock (MHz)	Ref Clock Divider	Fabric Clock (MHz)	Clock Out (MHz)
ADC 224	1.47456	4.096	<input type="checkbox"/>	1474.560	-	1	184.320	92.160
ADC 225	2.0	4.096	<input type="checkbox"/>	2000.000	-	1	0.0	15.625
ADC 226	2.0	4.096	<input type="checkbox"/>	2000.000	-	1	0.0	15.625
ADC 227	2.0	4.096	<input type="checkbox"/>	2000.000	-	1	0.0	15.625
DAC 228	6.4	6.554	<input type="checkbox"/>	6400.000	-	1	0.0	50.000
DAC 229	1.47456	6.554	<input type="checkbox"/>	1474.560	-	1	184.320	184.320

PLL Summary Settings

Tile	Vco (MHz)	Fb Div	M	R
ADC 224	-	-	-	-
ADC 225	-	-	-	-
ADC 226	-	-	-	-
ADC 227	-	-	-	-
DAC 228	-	-	-	-
DAC 229	-	-	-	-

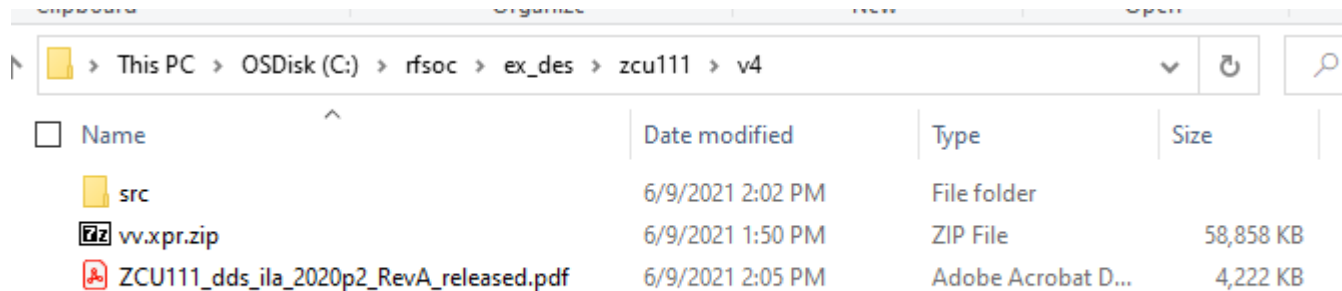
Board Setup for the Upcoming Designs

- ▶ Connect DAC Tile 229 Ch3 output to ADC Tile 224 Ch0 input on XM500 (low frequency balun connections).
- ▶ Set SW6 to on,on,on,on (JTAG boot mode).
- ▶ Connect USB to host for JTAG, PS UART, and System Controller UART access.






Design Kit Contents

1. Extract the design kit to an appropriate folder—be mindful of the Windows path length requirement.
2. Extract vv.xpr.zip, which is the Vivado® project.
3. Software source files in the “src” folder.
4. Design documentation in the .pdf file.



The screenshot shows a Windows File Explorer window with the address bar displaying the path: This PC > OSDisk (C:) > rfsoc > ex_des > zcu111 > v4. The main area displays a table of files and folders.

<input type="checkbox"/> Name	Date modified	Type	Size
 src	6/9/2021 2:02 PM	File folder	
 vv.xpr.zip	6/9/2021 1:50 PM	ZIP File	58,858 KB
 ZCU111_dds_ila_2020p2_RevA_released.pdf	6/9/2021 2:05 PM	Adobe Acrobat D...	4,222 KB

Extract vv.xpr.zip, open the design in Vivado®, and generate the bitstream.

[illegible]

DAC Sine Wave Generator (DDS Compiler IP)

Re-customize IP

DDS Compiler (6.0)

Documentation IP Location

IP Symbol Information

☐ Show disabled ports

Component Name: dds_compiler_0

Configuration Implementation Detailed Implementation Output Frequencies

Configuration Options: Phase Generator and SIN COS LUT

System Requirements

System Clock (MHz): 184.32 [0.01 - 1000.0]

Number of Channels: 1

Mode Of Operation: Standard

Frequency per Channel (Fs): 184.31999999999999 MHz

Parameter Selection: SystemParameters

System Parameters

Spurious Free Dynamic Range (dB): 95 Range: 18...150

Frequency Resolution (Hz): 0.4 6.54836e-07...2.304e+07

Noise Shaping: Auto

clk M_AXIS_DATA

OK Cancel

Re-customize IP

DDS Compiler (6.0)

Documentation IP Location

IP Symbol Information

☐ Show disabled ports

Component Name: dds_compiler_0

Configuration Implementation Detailed Implementation Output Frequencies

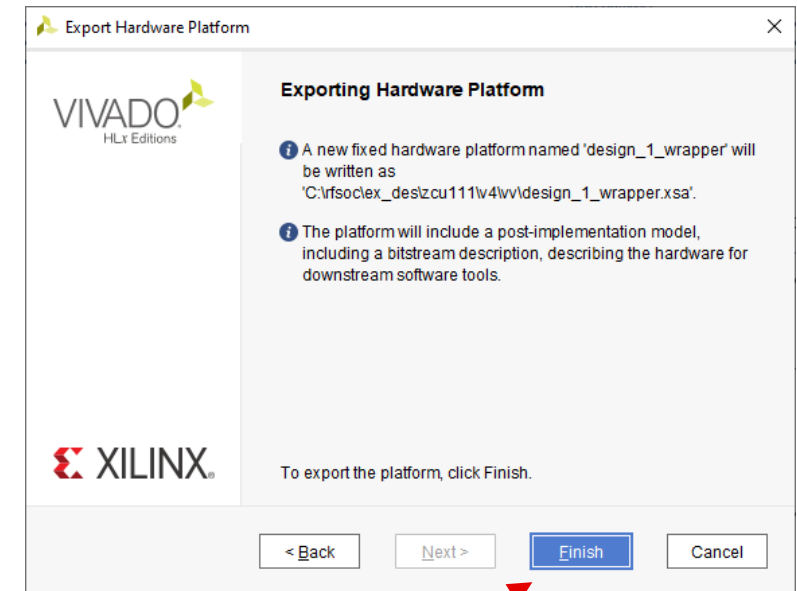
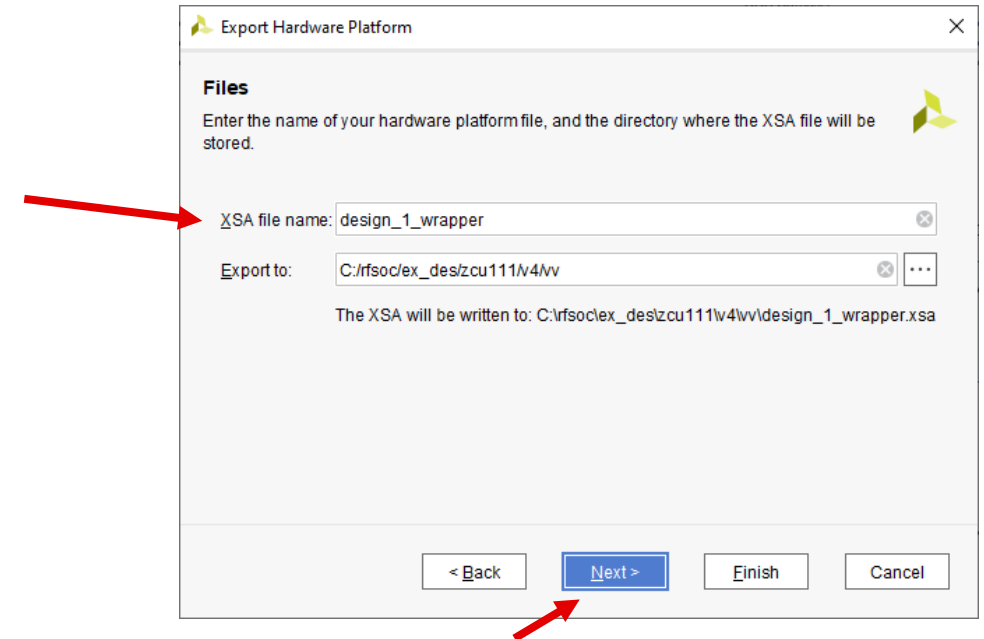
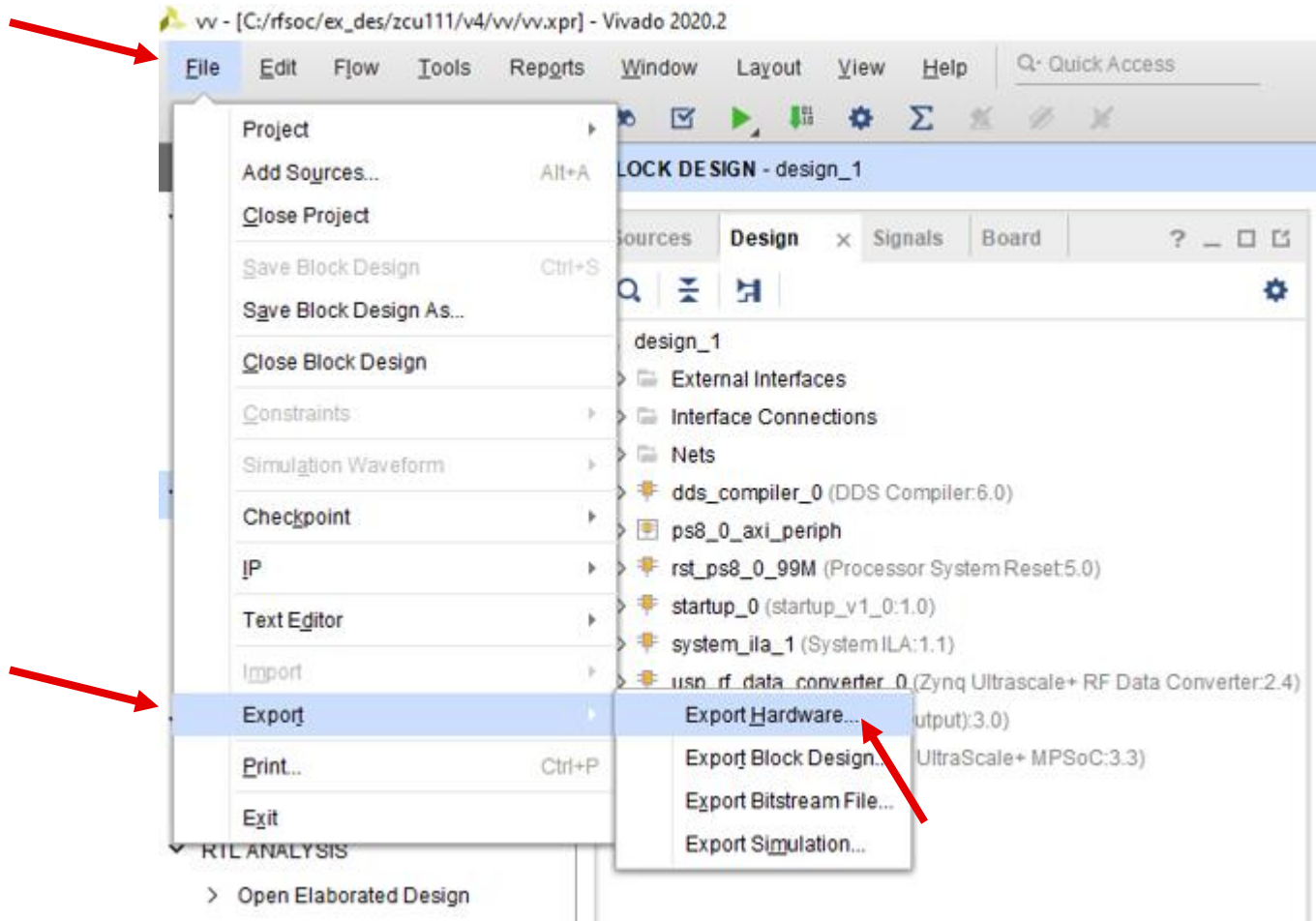
Channel	Output Frequency (MHz)
1	10
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0
16	0

Valid Range: 0.0...184.31999999999999 MHz

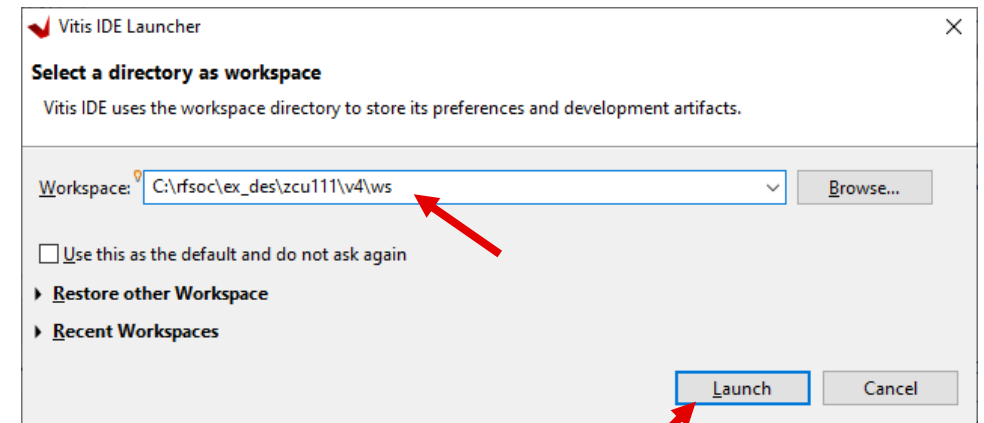
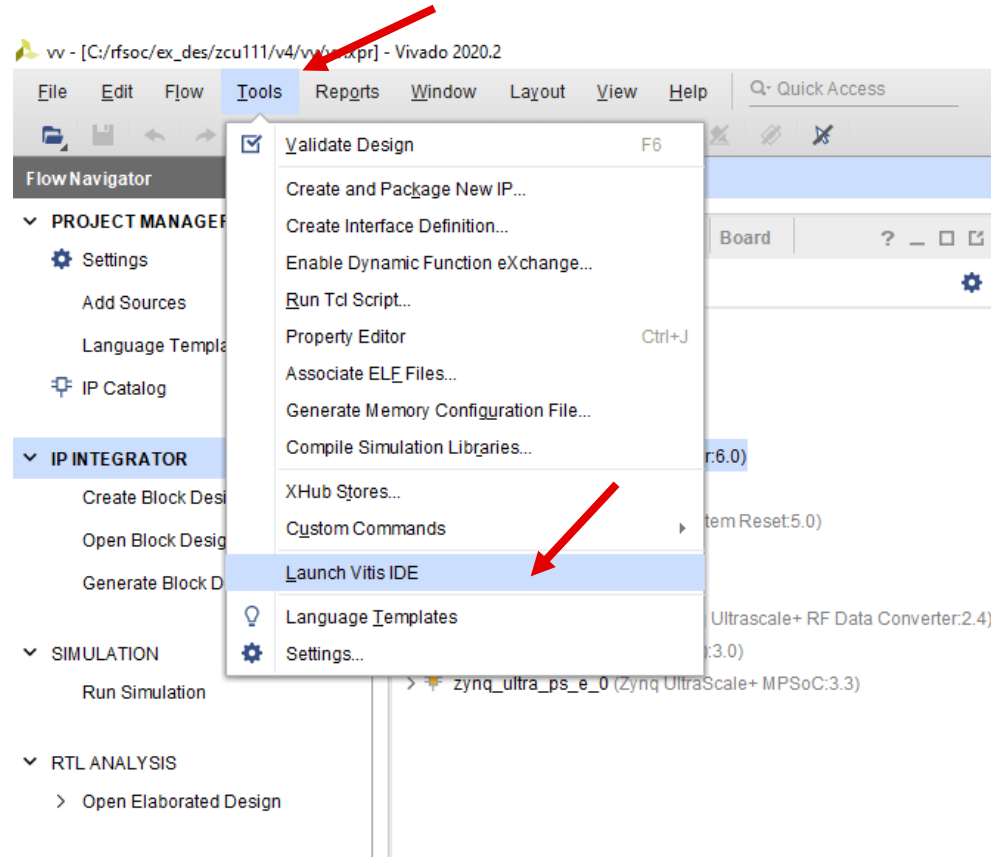
clk M_AXIS_DATA

OK Cancel

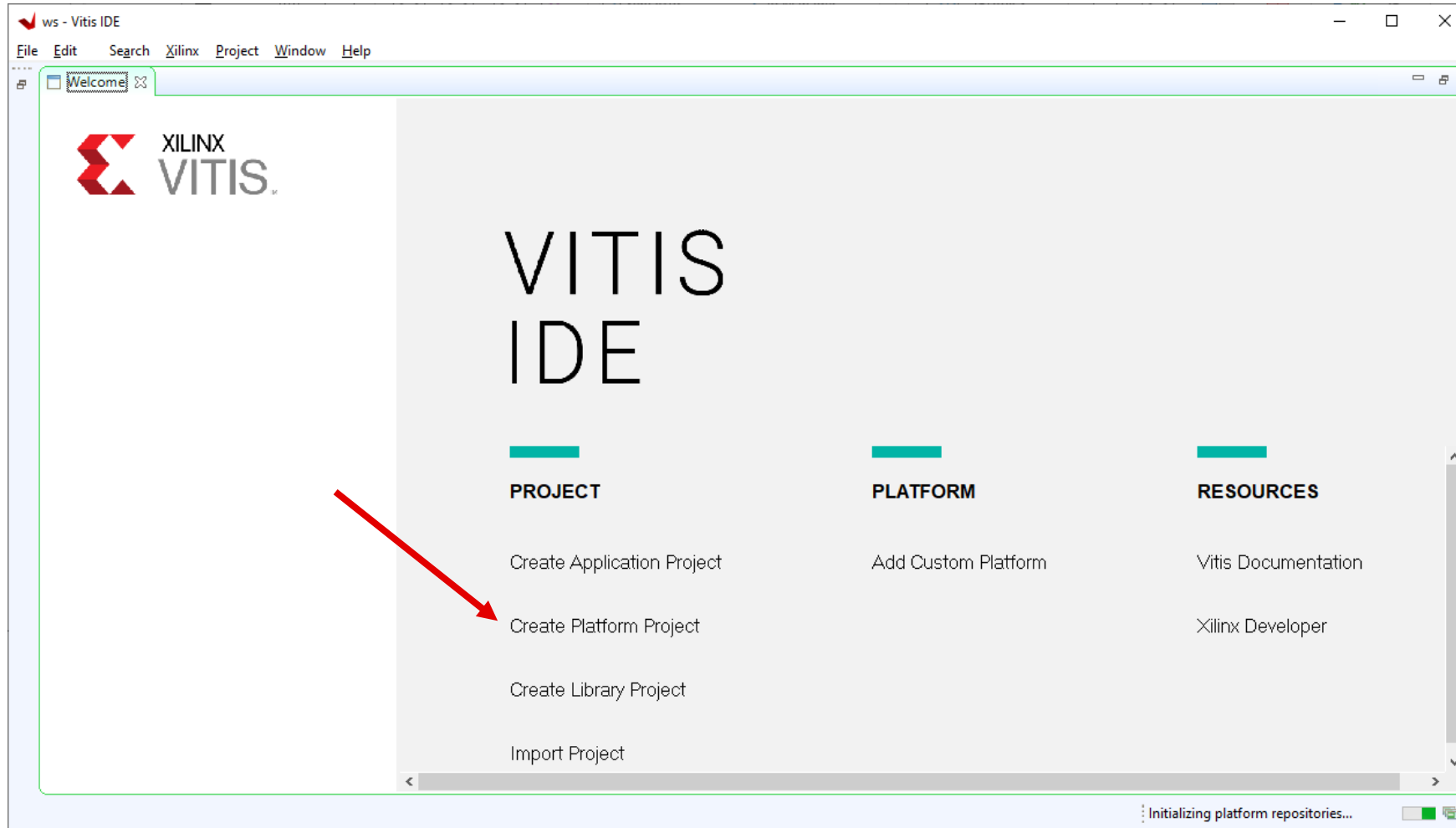
Export Hardware



Open Vitis™ Software Platform



Create Platform Project



Create Platform Project Cont'd

New Platform Project

Create new platform project

Enter a name for your platform project

This wizard will guide you through creation of a platform project from the output of Vivado [Xilinx Shell Archive (XSA)] or from an existing platform. A platform will enable you to specify options for the kernels, BSPs, as well as settings required for creating new applications. Platforms are currently supported for embedded software developers.

Platform project name: ZCU111

Platform Project

System Project

Processor

Domain

App

XSA

- A platform provides hardware information and software environment settings.
- A system project contains one or more applications that run at the same time.
- A domain provides runtime for applications, such as operating system or BSP.
- A workspace can contain unlimited platforms and unlimited system projects.

A new platform project can be created from one of the two inputs:

From hardware specification (XSA)

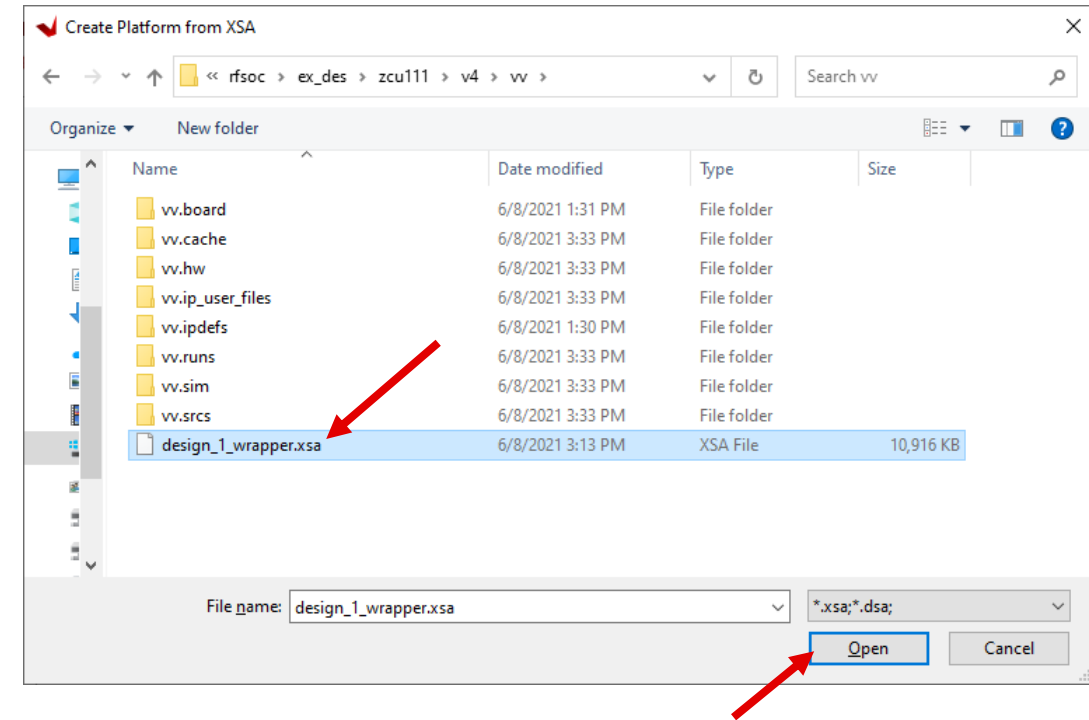
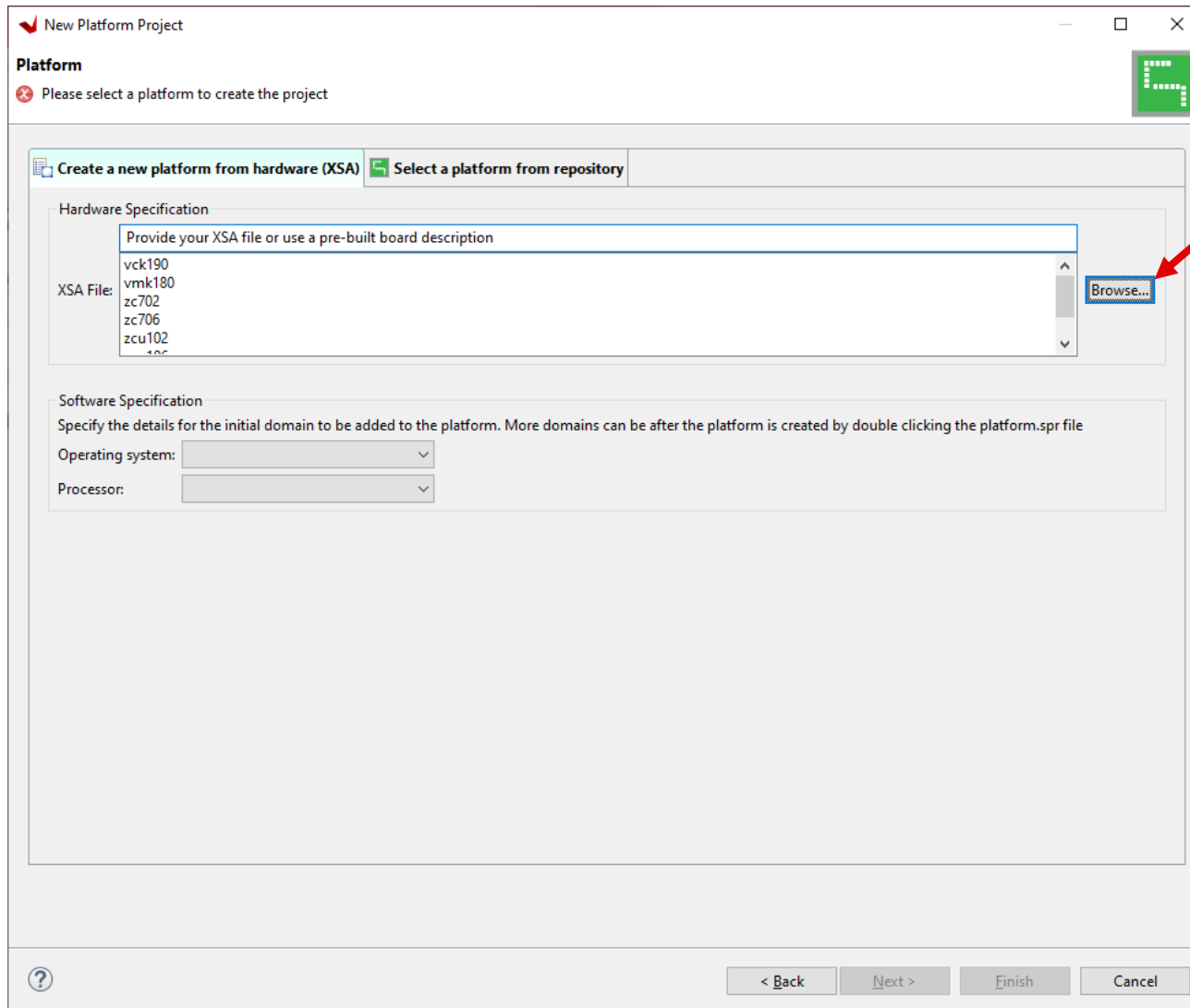
Create a new platform project from a hardware specification file. You can specify the OS and processor to start with. The platform can be customized later from the platform project editor.

From existing platform

Load the platform definition from an existing platform. You can choose any platform from the platform repository as a base for your platform project.

< Back Next > Finish Cancel

Create Platform Project Cont'd



Create Platform Project Cont'd

New Platform Project

Platform

Choose a platform for your project. You can also create an application from XSA through the 'Create a new platform from hardware (XSA)' tab.

Create a new platform from hardware (XSA) | Select a platform from repository

Hardware Specification

XSA File:

- C:\rfsoc\ex_des\zcu111\v4\vv\design_1_wrapper.xsa
- vck190
- vmk180
- zc702
- zc706
- zcu102
- zcu106
- zed
- C:\rfsoc\ex_des\zcu111\v4\vv\design_1_wrapper.xsa

Browse...

Software Specification

Specify the details for the initial domain to be added to the platform. More domains can be after the platform is created by double clicking the platform.spr file

Operating system: standalone

Processor: psu_cortexa53_0

Architecture: 64-bit

Boot components

☒ Generate boot components

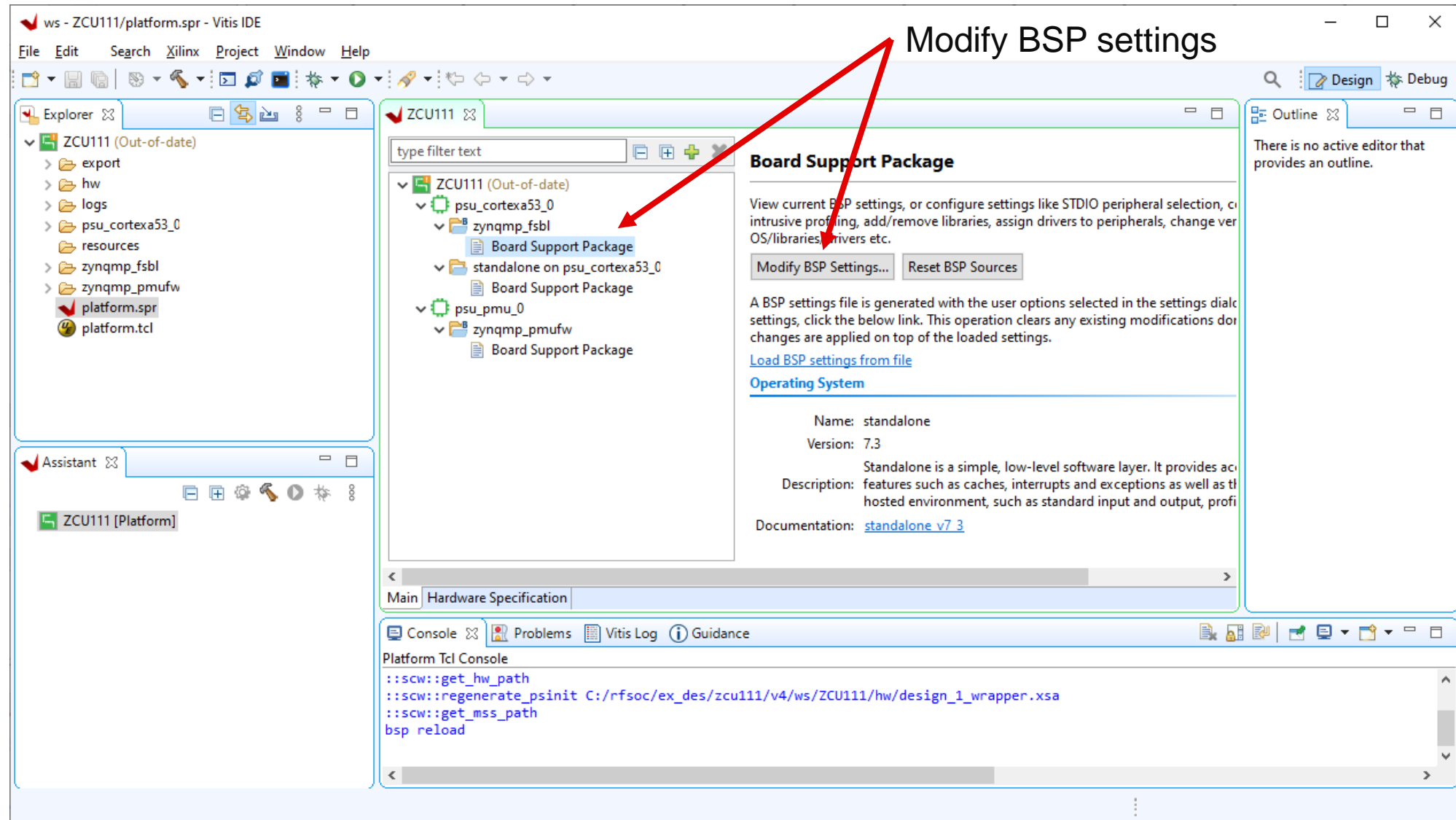
Target processor to create FSBL: ☒ psu_cortexa53_0 ☐ psu_cortexr5_0

Note: A domain with selected operating system and processor will be added to the platform. The platform project can be modified later to add new domains or change settings.

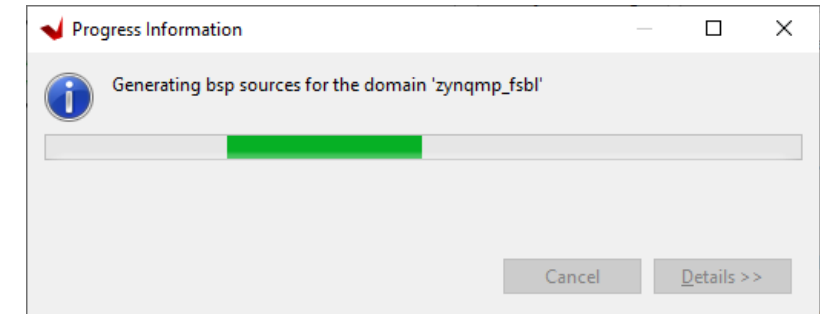
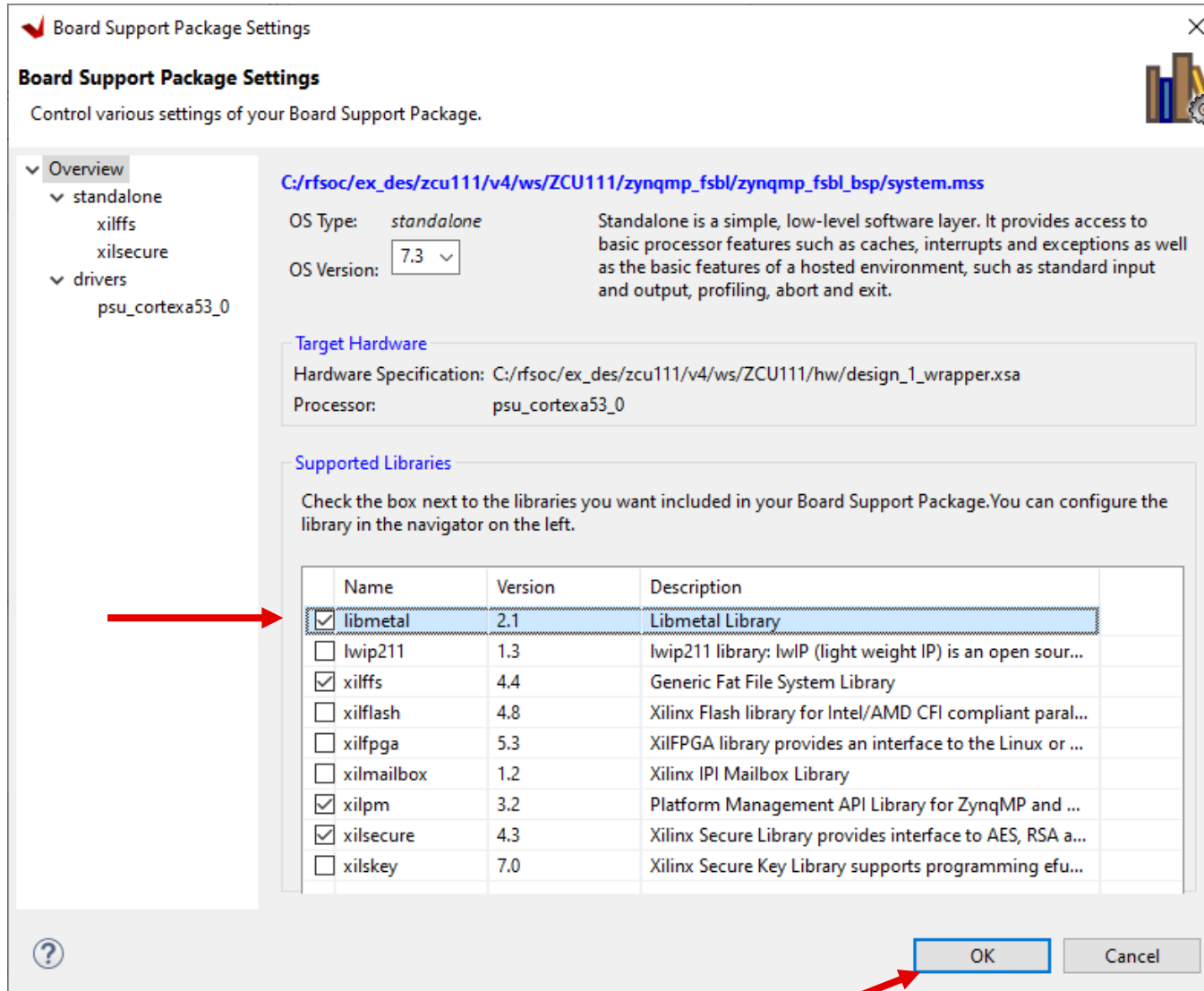
< Back Next > Finish Cancel

This may take a few minutes.

Create Platform Project Cont'd

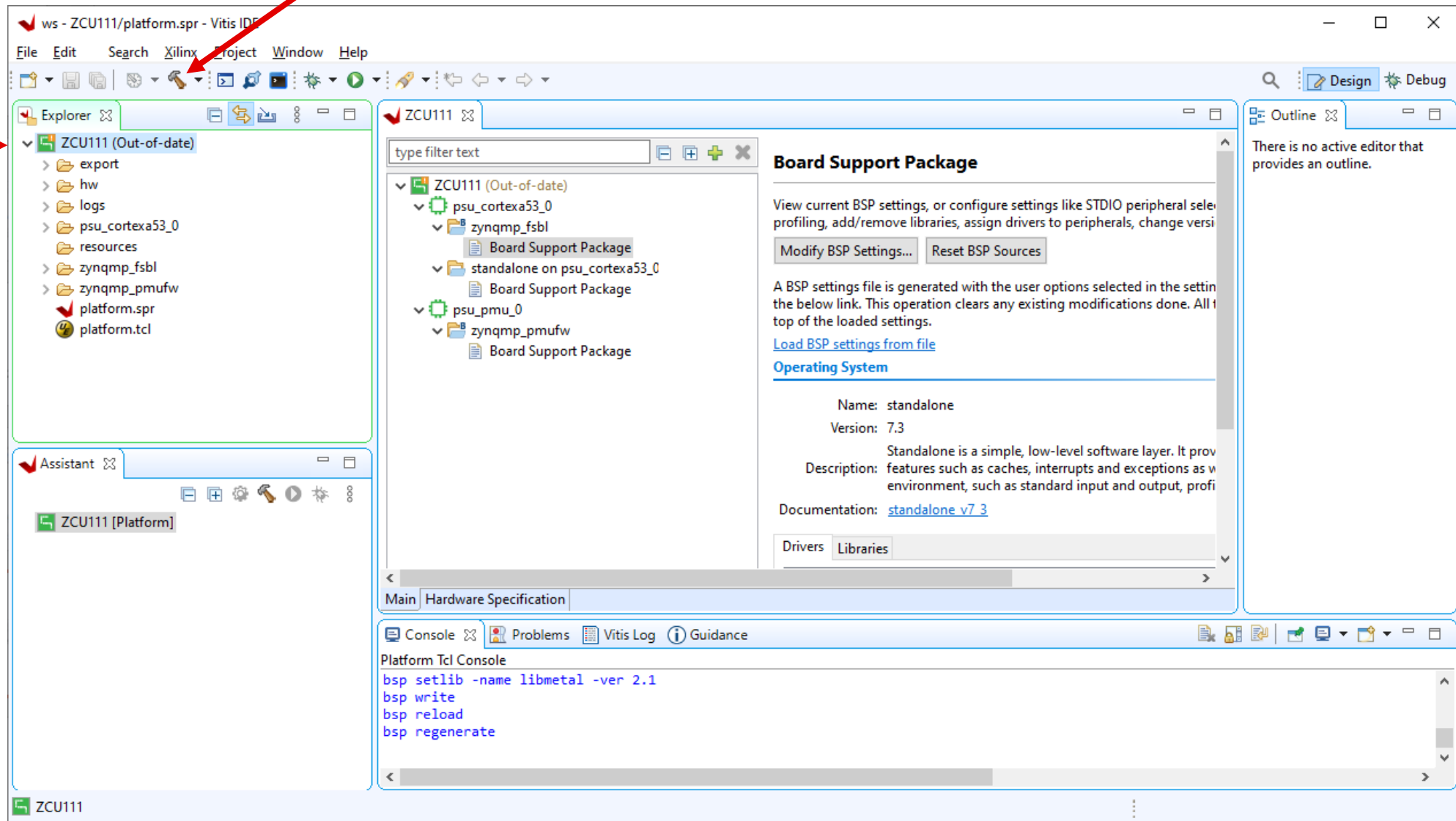


Enable libmetal

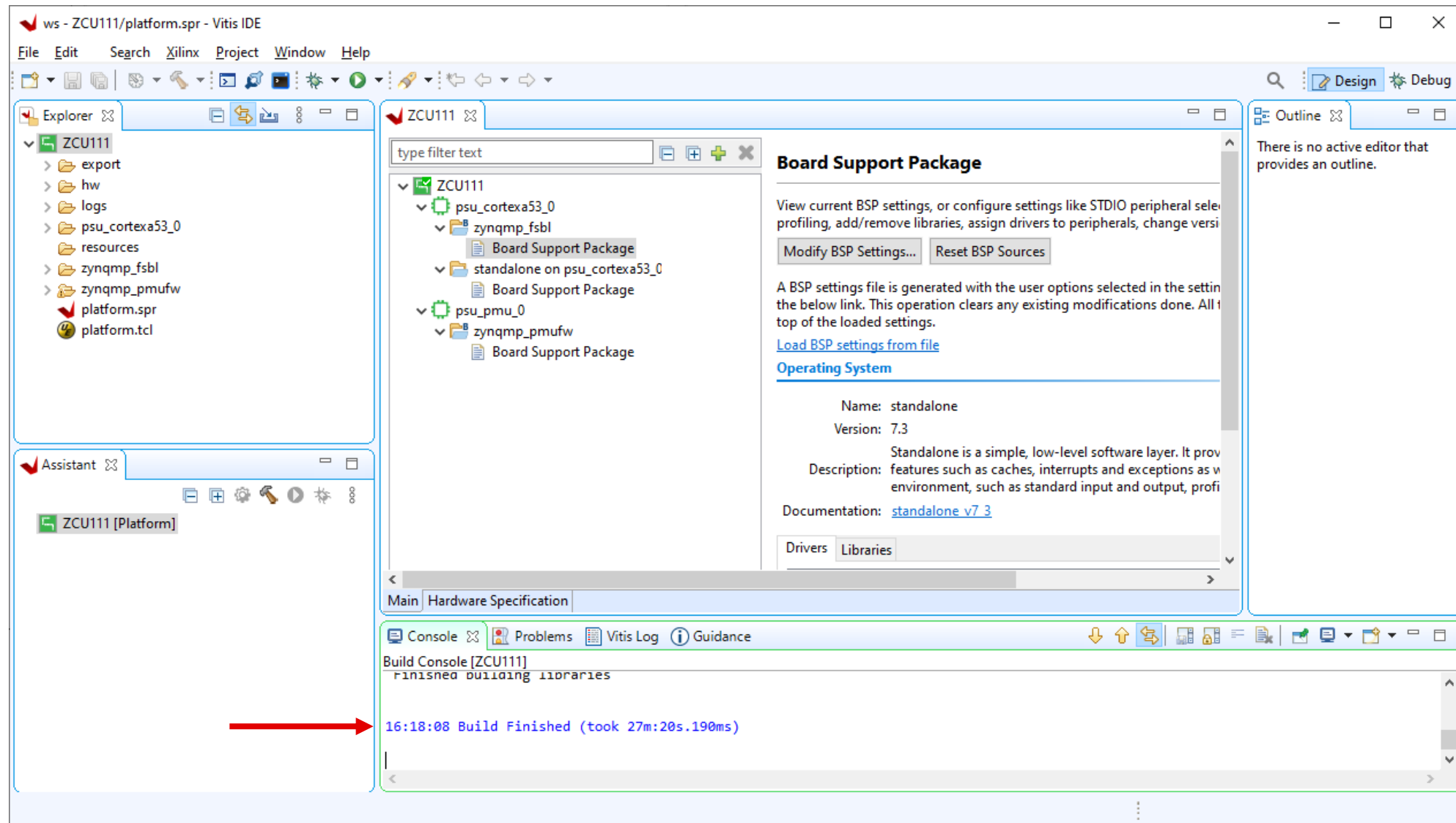


Build Project

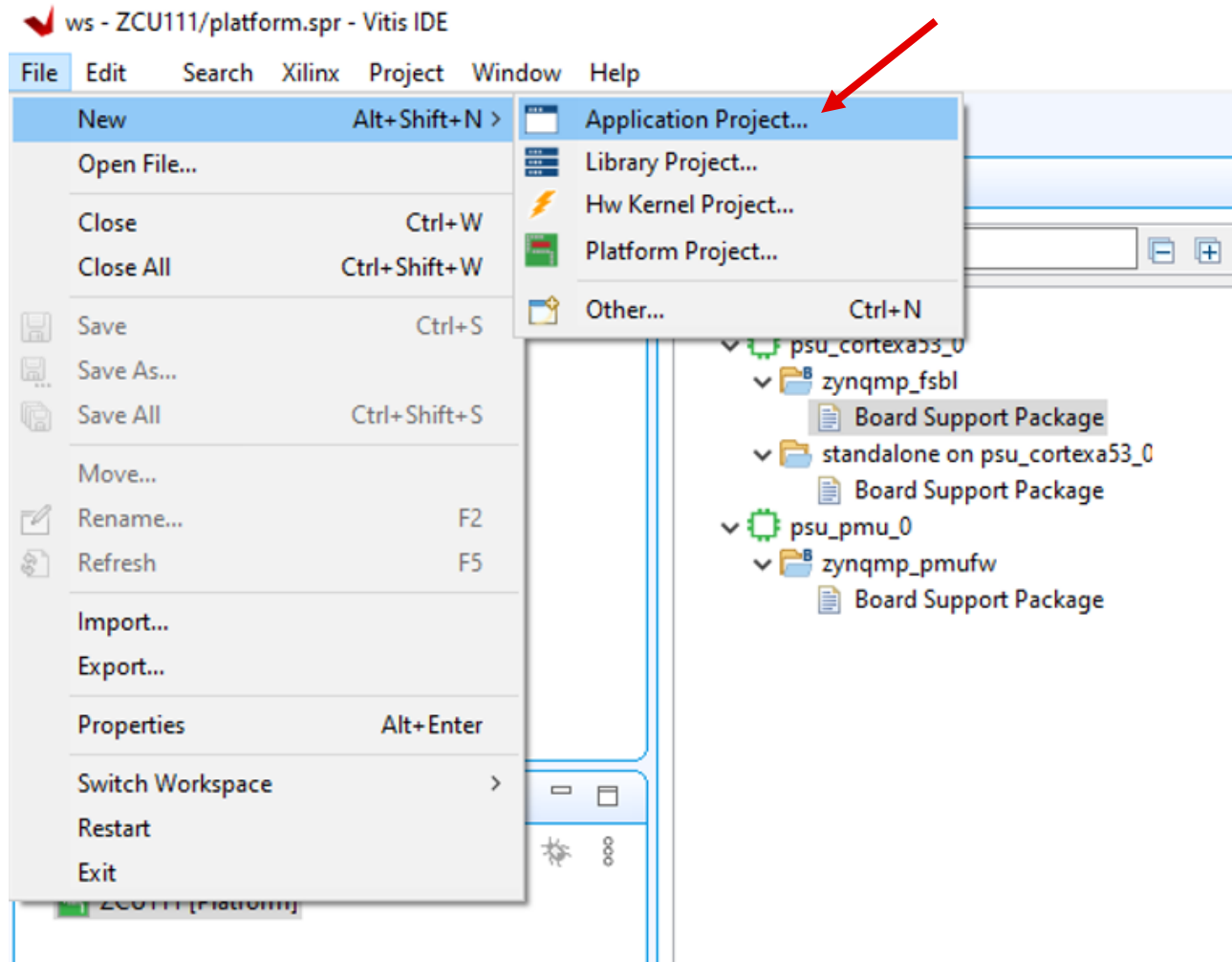
This may take
a few minutes.



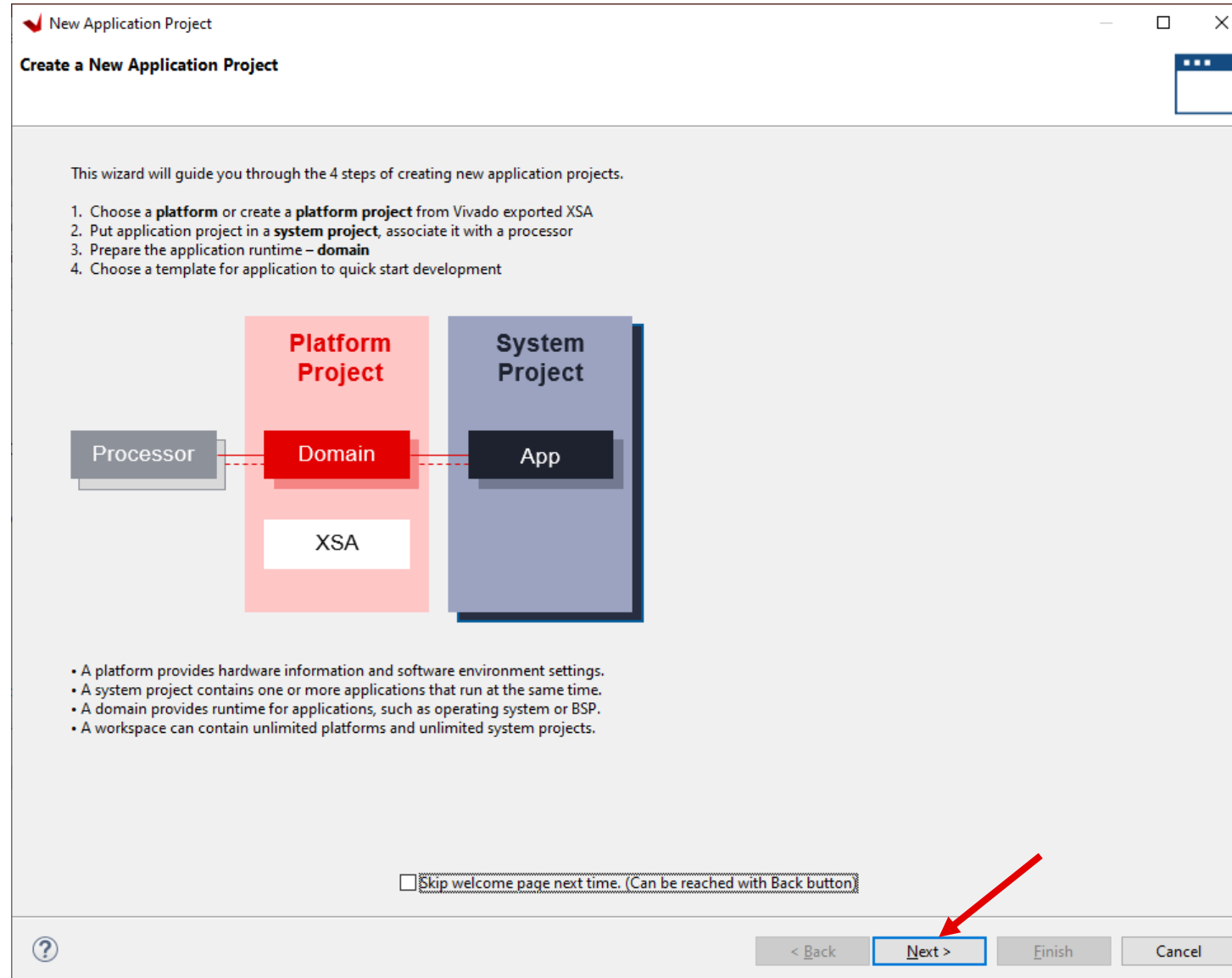
Build Complete



Create Application



Create Application Cont'd



Create Application Cont'd

New Application Project

Platform

Choose a platform for your project. You can also create an application from XSA through the 'Create a new platform from hardware (XSA)' tab.

Select a platform from repository | Create a new platform from hardware (XSA)

Find:

+ Add ⚙ Manage

Name	Board	Flow	Vendor	Path
ZCU111 [custom]	zcu111	Embedded SW Dev	xilinx	C:\rfsoc\ex_des\zcu111\v4\ws\ZCU111\export\ZCU111\ZCU111...

< >

Platform Info

General Info

Name:

Part:

Family:

Description:

Acceleration Resources

The selected platform does not have application acceleration capabilities

Domain Details

Domains

Domain name	Details
standalone on psu_corte...	CPU: psu_cortexa53...

< >

? < Back **Next >** Finish Cancel

Create Application Cont'd

New Application Project

Application Project Details

Specify the application project name and its system project properties

Application project name:

System Project

Create a new system project for the application or select an existing one from the workspace

Select a system project

+ Create new...

System project details

System project name:

Target processor

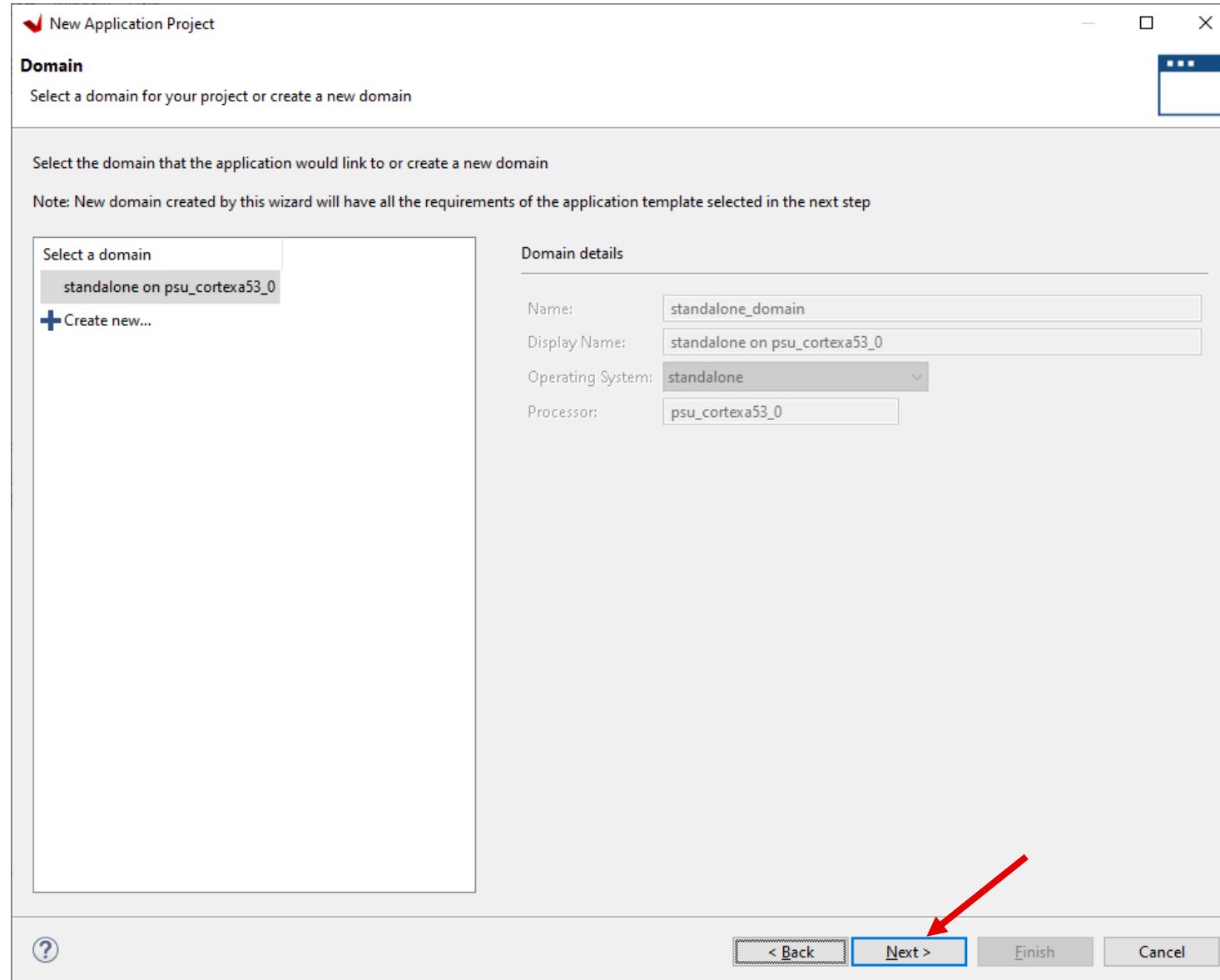
Select target processor for the Application project.

Processor	Associated applications
psu_cortexa53_0	RFSOC

Show all processors in the hardware specification ☐

< Back **Next >** Finish Cancel

Create Application Cont'd



The screenshot shows the 'New Application Project' wizard window. The title bar says 'New Application Project'. The main heading is 'Domain'. Below it, the instruction is 'Select a domain for your project or create a new domain'. A sub-instruction says 'Select the domain that the application would link to or create a new domain'. A note states: 'Note: New domain created by this wizard will have all the requirements of the application template selected in the next step'.

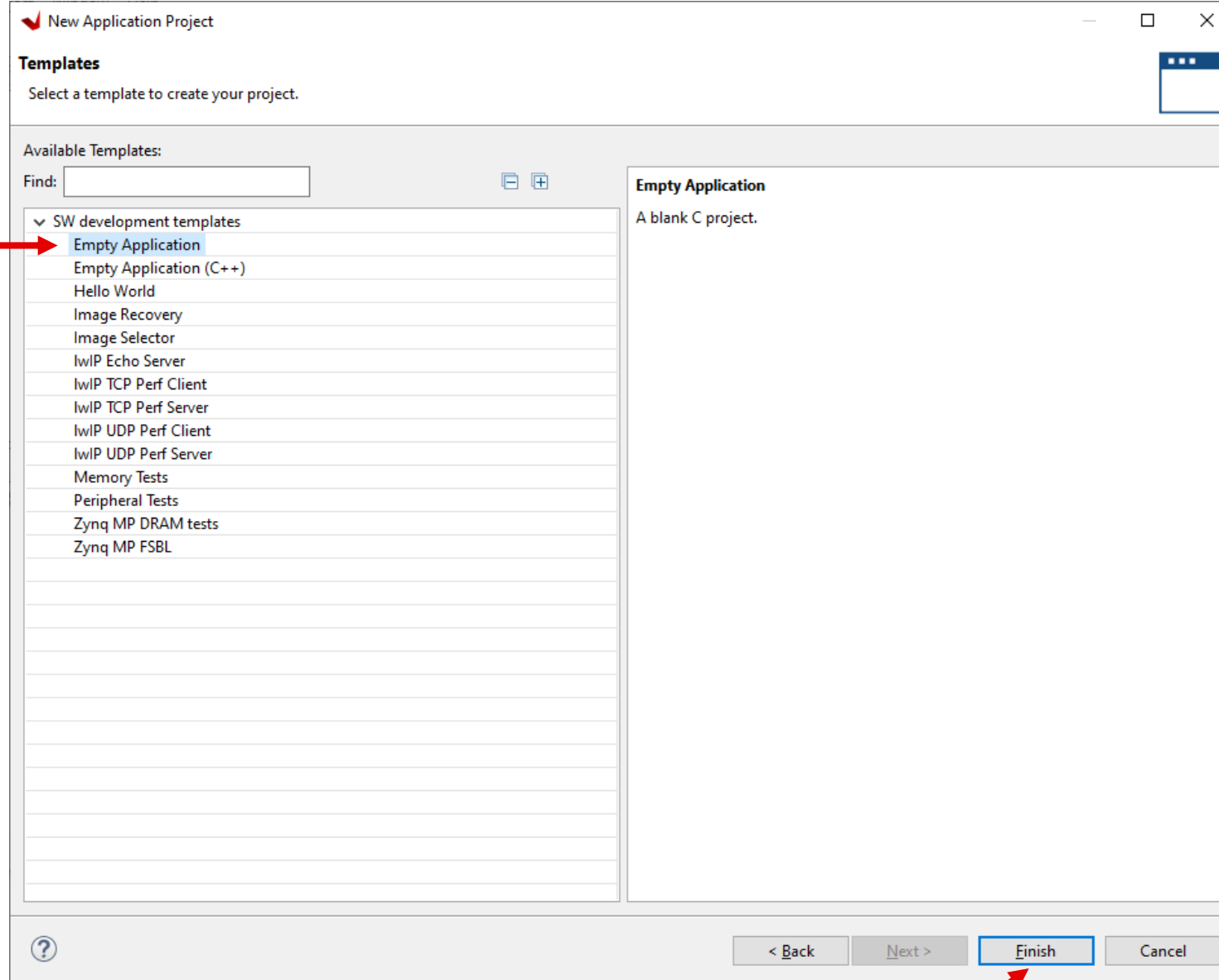
On the left, under 'Select a domain', there is a list with one item: 'standalone on psu_cortexa53_0'. Below the list is a '+ Create new...' button.

On the right, under 'Domain details', there are four fields:

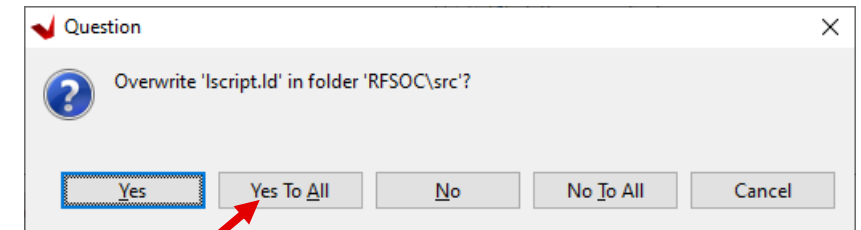
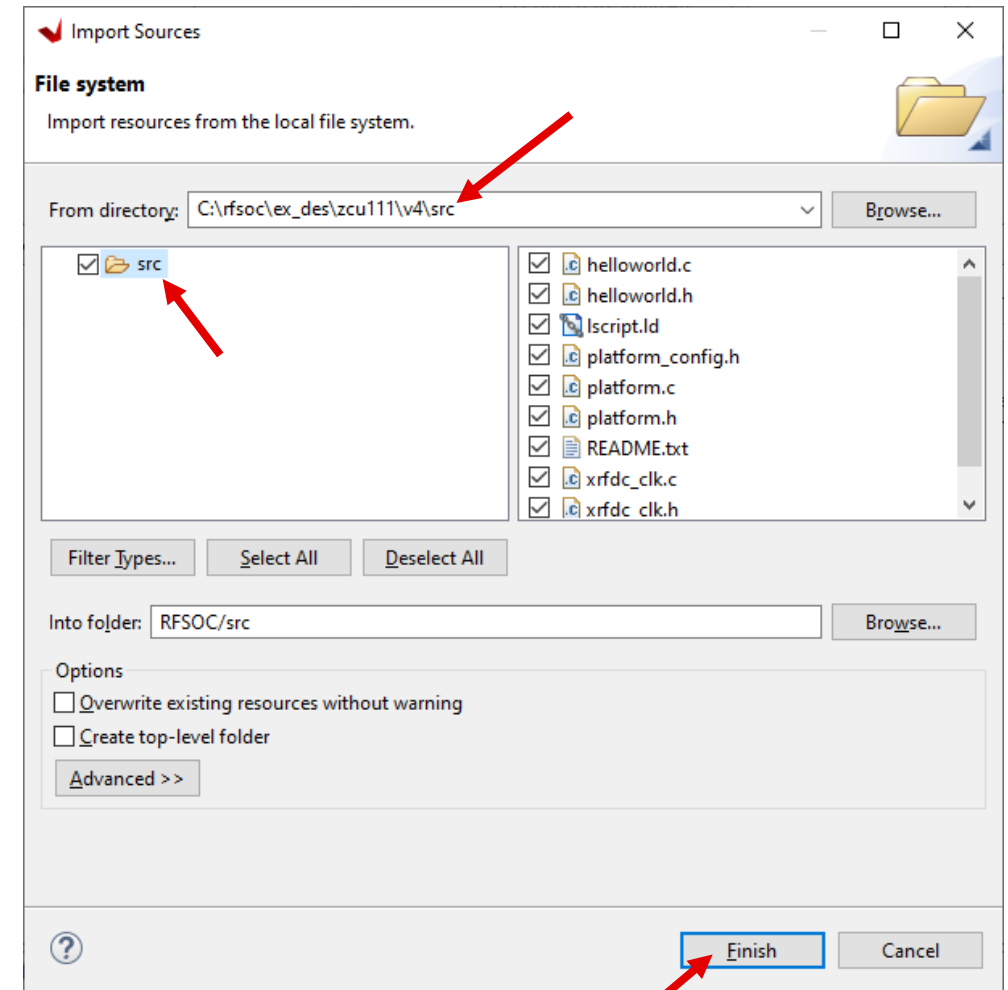
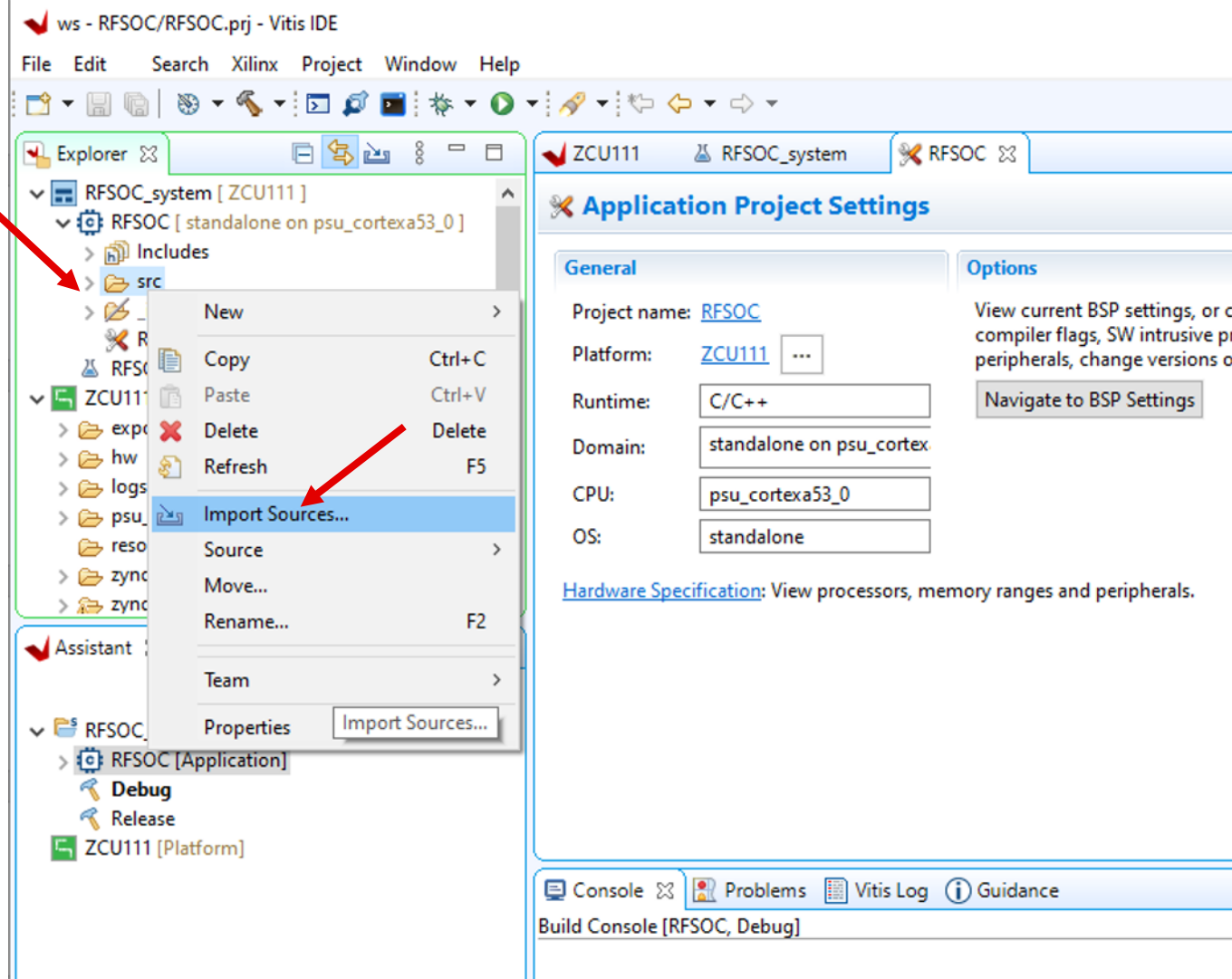
- Name: standalone_domain
- Display Name: standalone on psu_cortexa53_0
- Operating System: standalone (dropdown menu)
- Processor: psu_cortexa53_0

At the bottom right, there are four buttons: '< Back', 'Next >', 'Finish', and 'Cancel'. A red arrow points to the 'Next >' button.

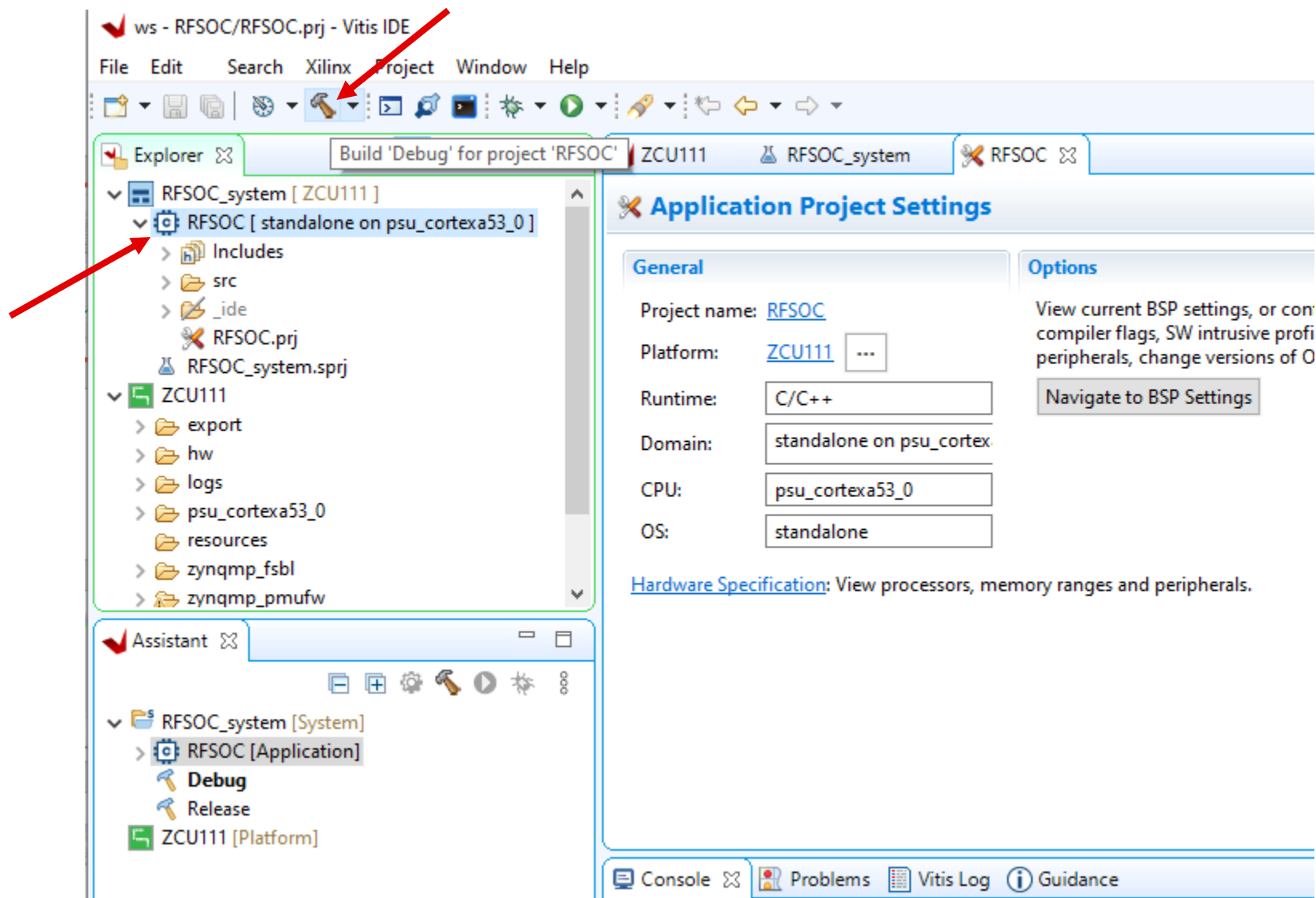
Create Application Cont'd



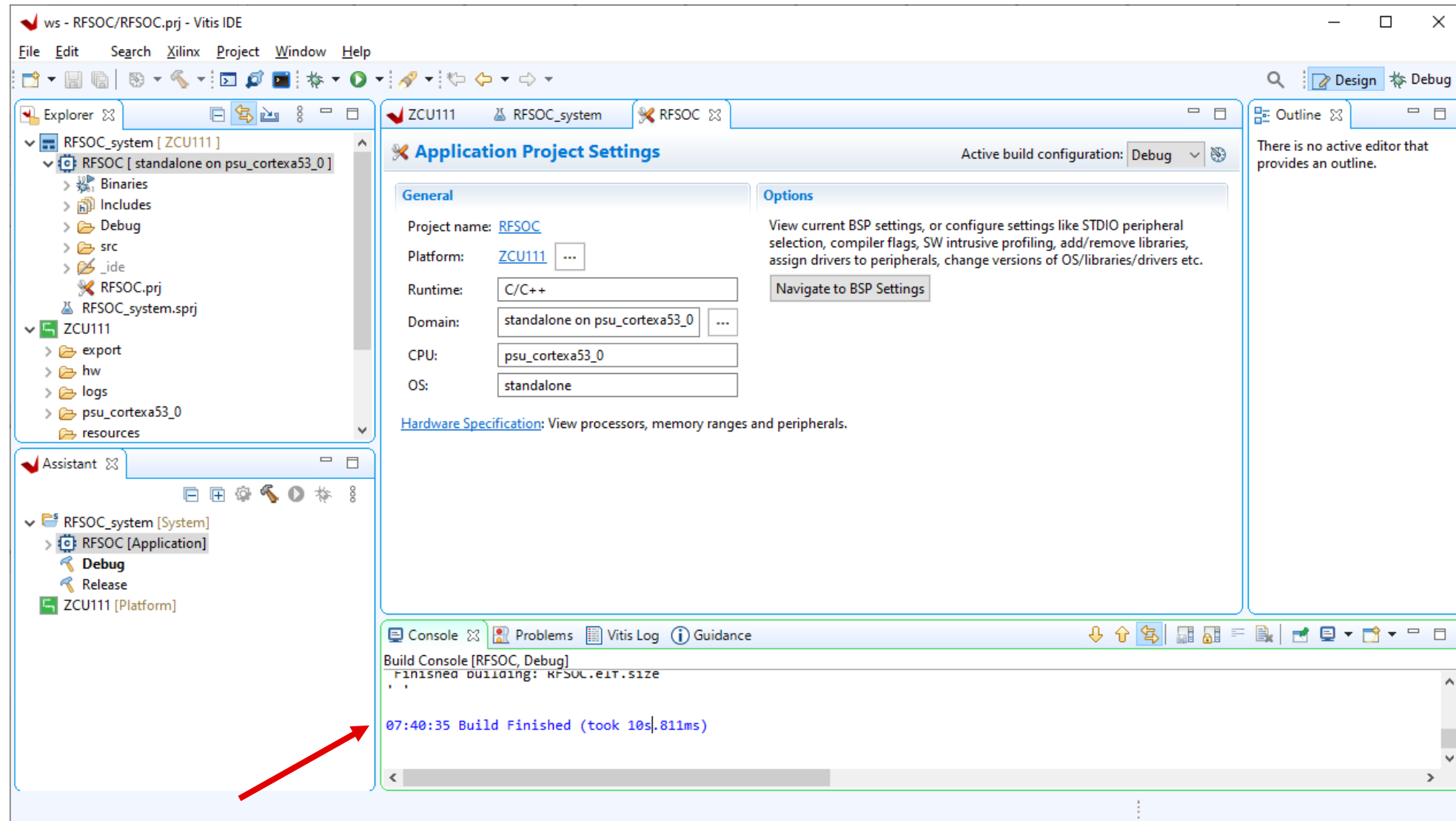
Import Sources



Build Application



Build Complete

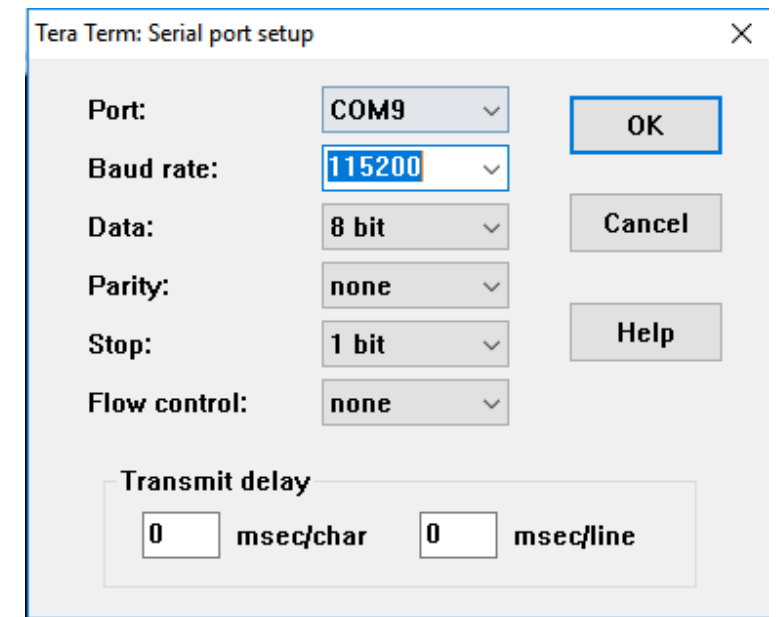
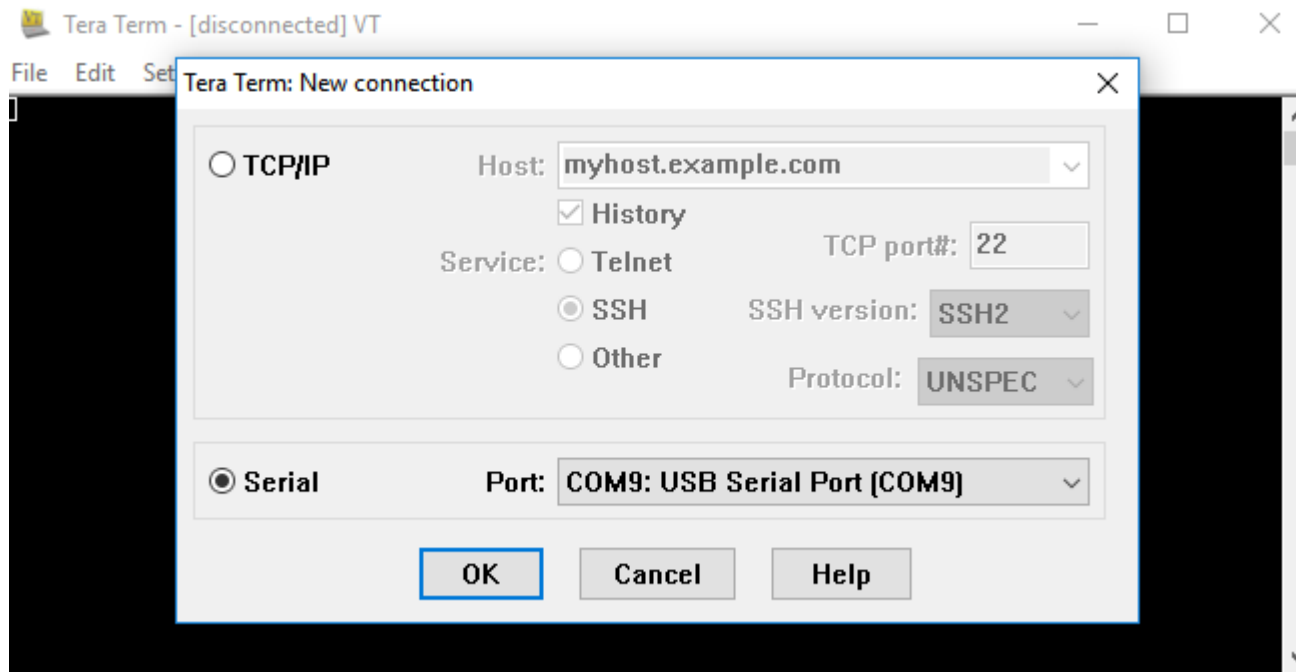


Run Design

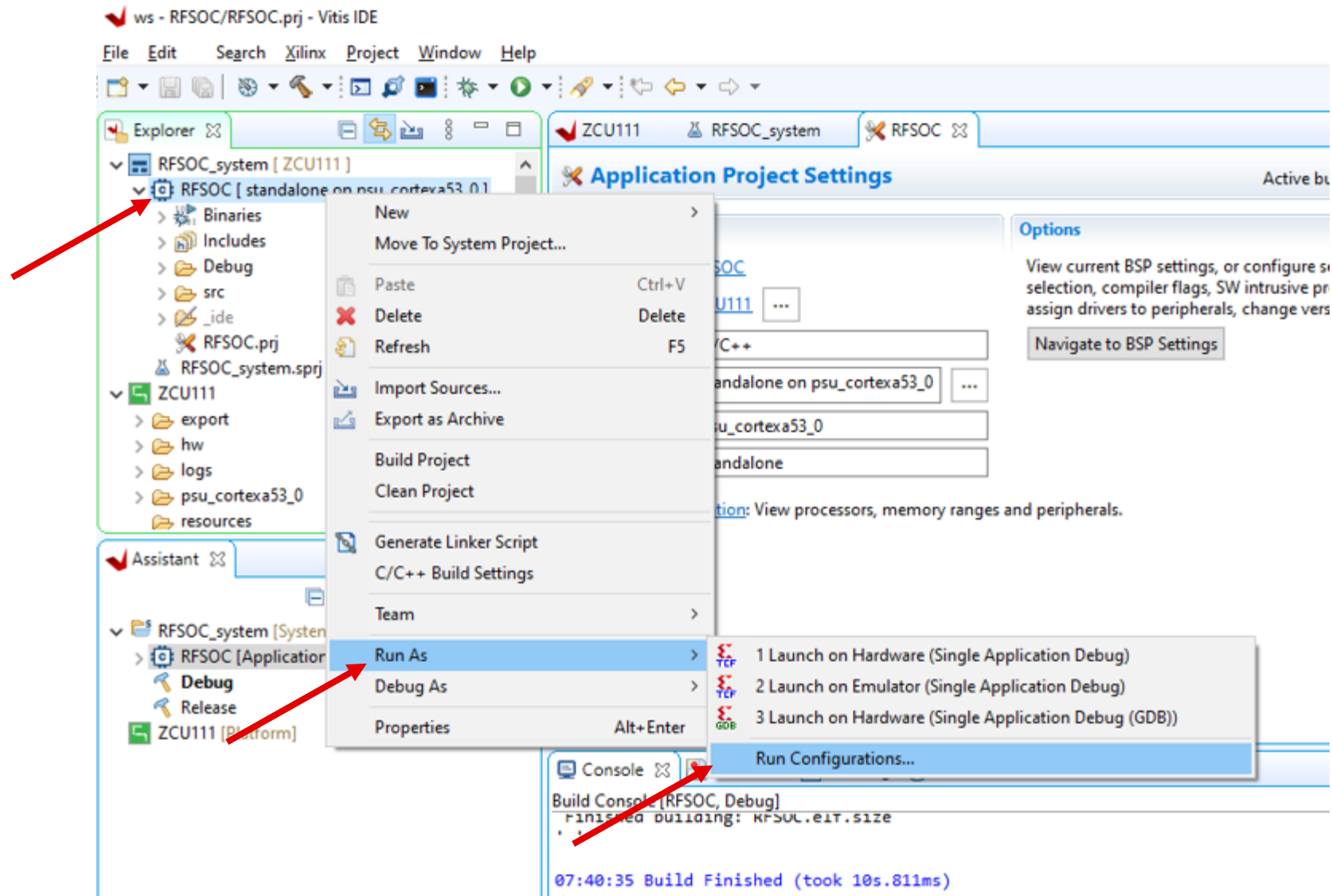
Open a Terminal Window

Open the COM port on the compute and set the rate to 115200.

TeraTerm can be used. See [UG1036](#).

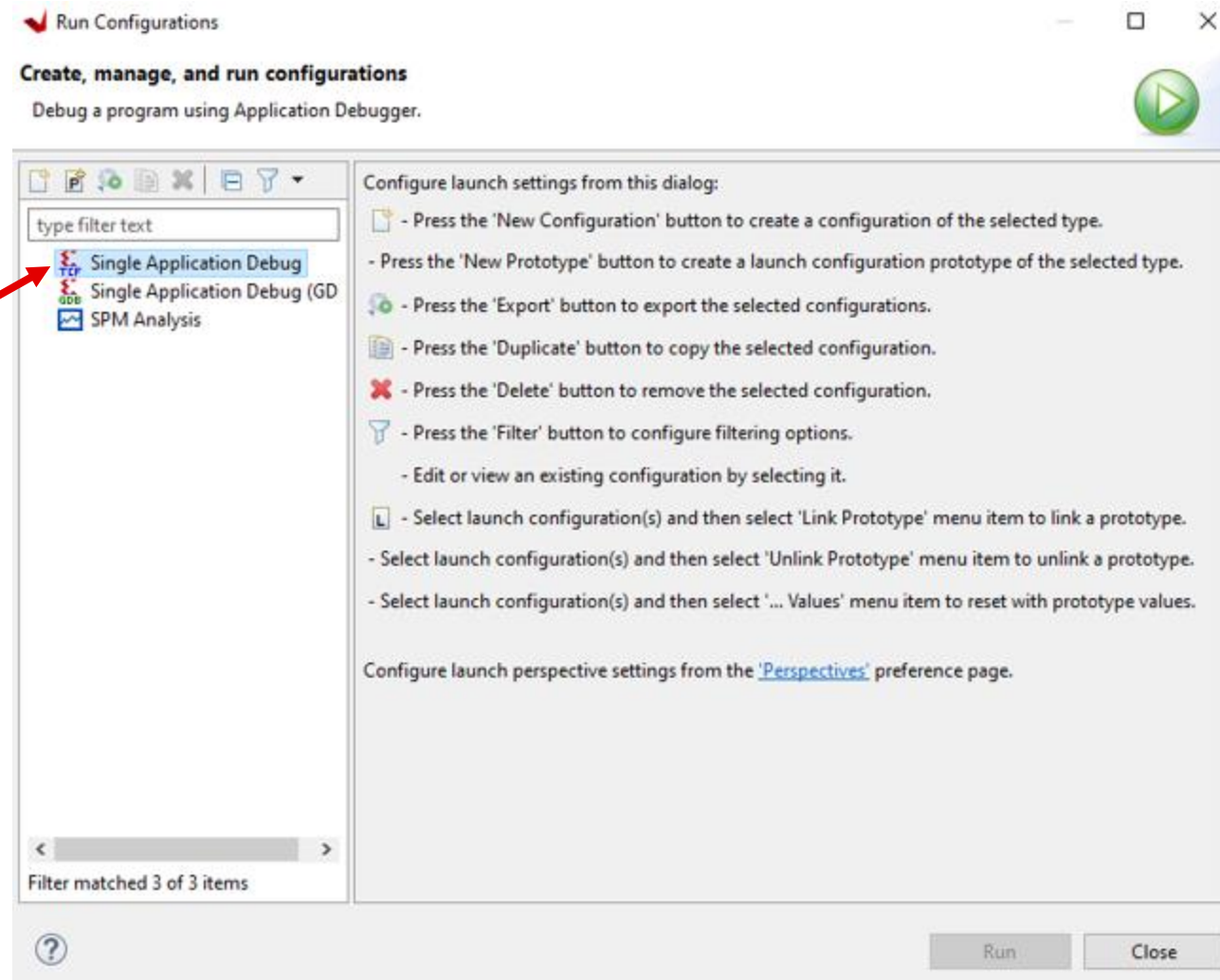


Setup Run Configuration

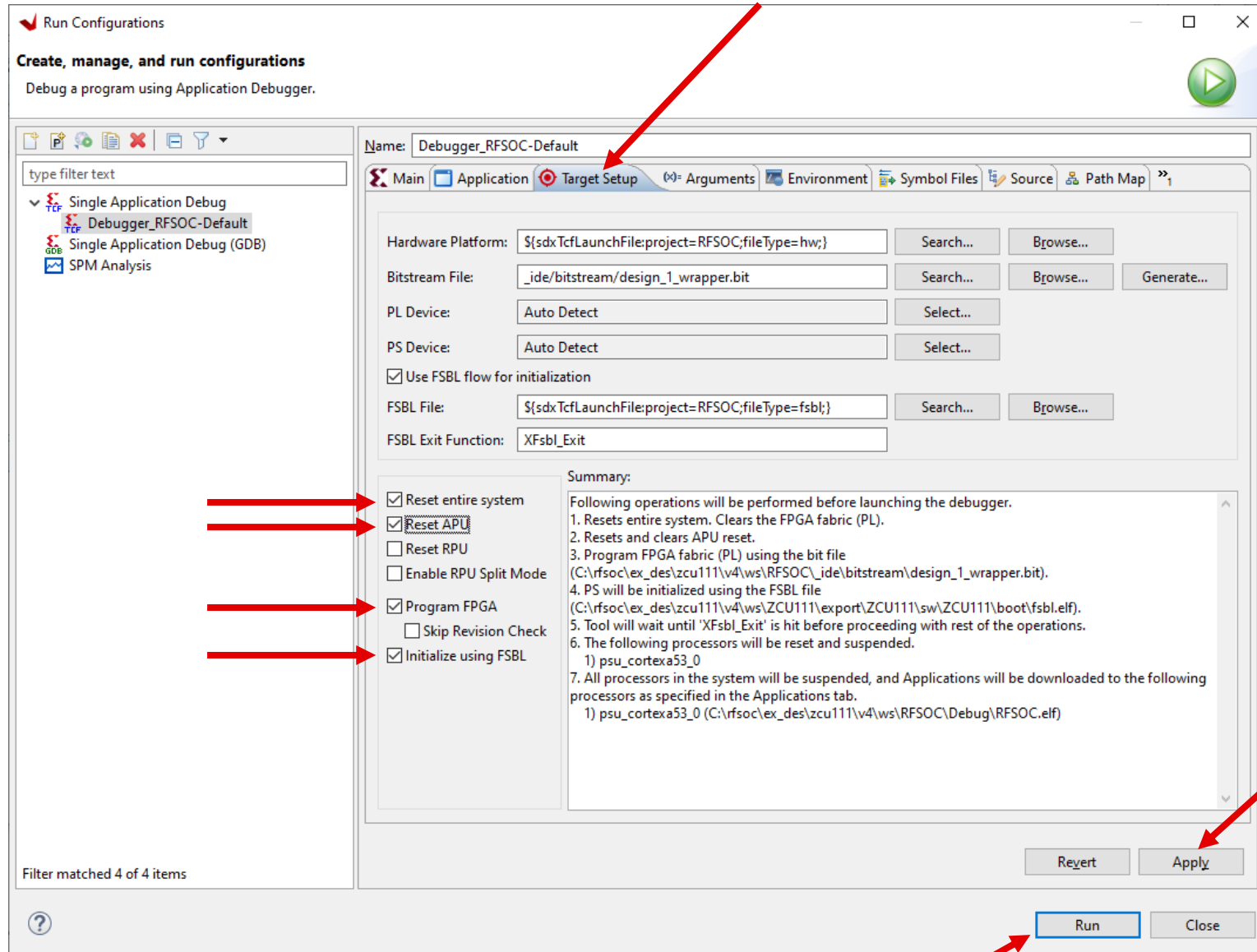


Run Configuration Cont'd

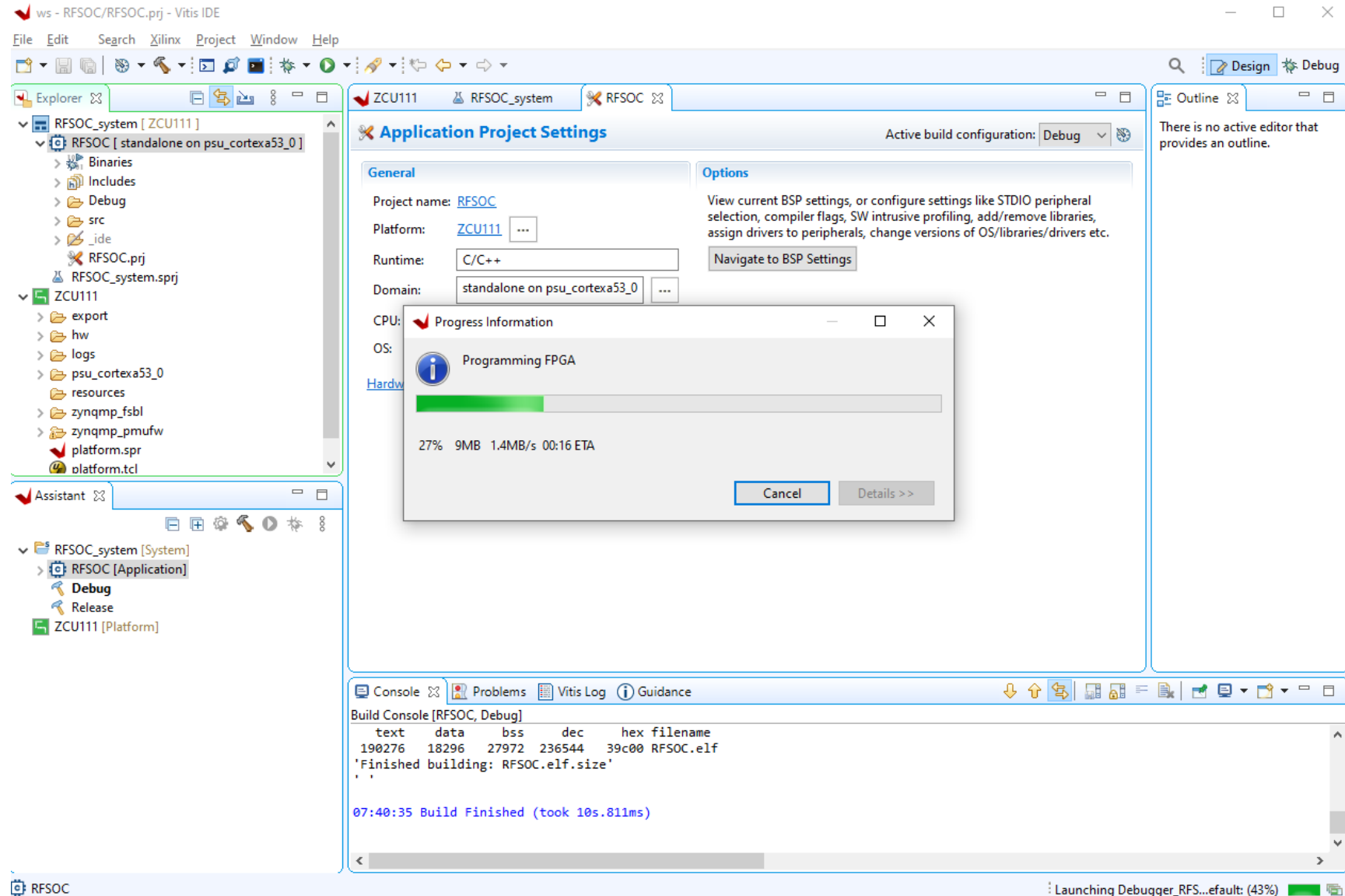
Double
Click



Run Configuration Cont'd



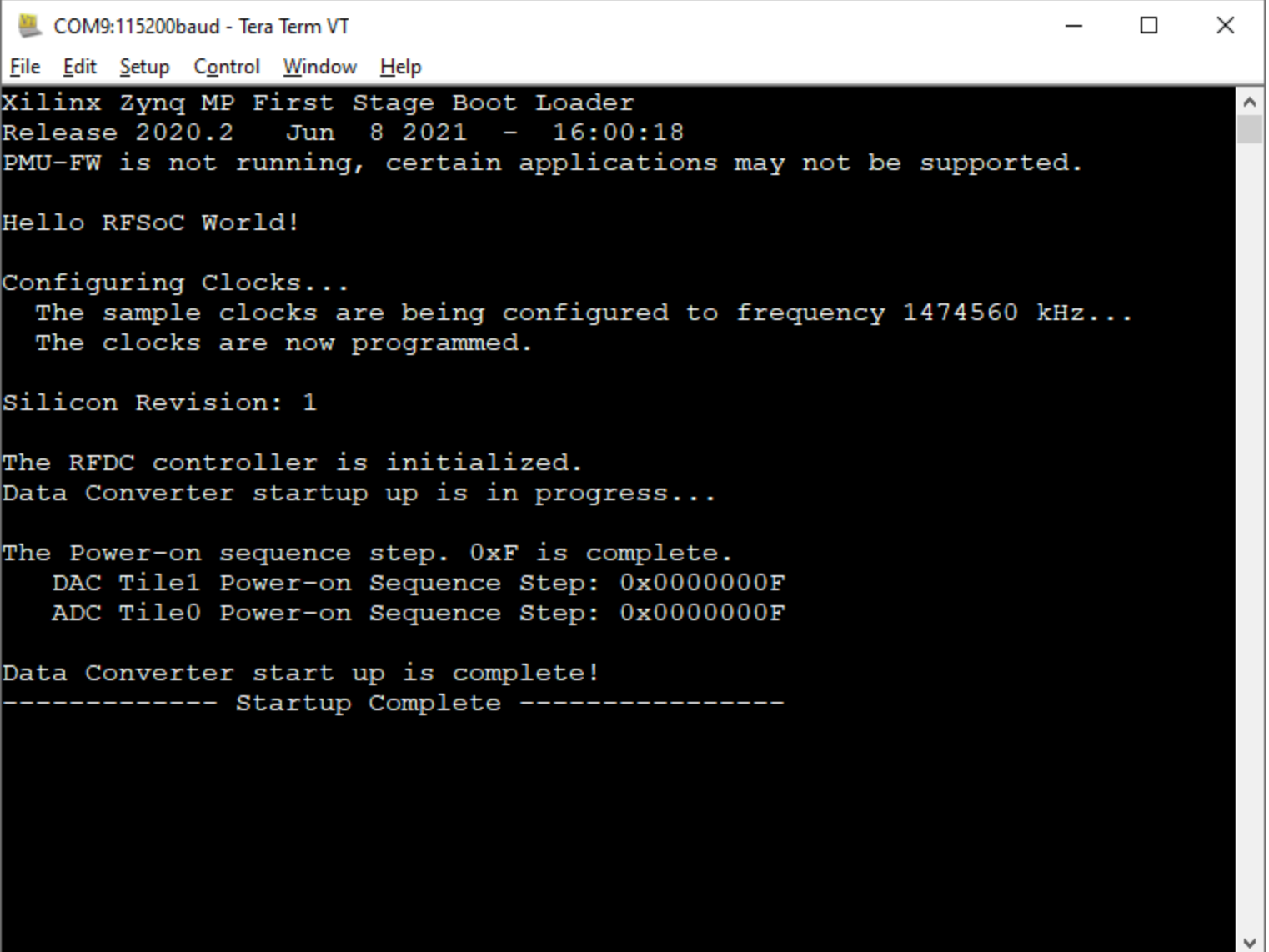
Run Design



Application Startup

The application...

1. Programs the clocks.
2. Issues the data converters master reset.
3. Displays the Power-on Sequence Step of the data converters.



```
COM9:115200baud - Tera Term VT
File Edit Setup Control Window Help
Xilinx Zynq MP First Stage Boot Loader
Release 2020.2 Jun 8 2021 - 16:00:18
PMU-FW is not running, certain applications may not be supported.

Hello RFSoc World!

Configuring Clocks...
  The sample clocks are being configured to frequency 1474560 kHz...
  The clocks are now programmed.

Silicon Revision: 1

The RFDC controller is initialized.
Data Converter startup up is in progress...

The Power-on sequence step. 0xF is complete.
  DAC Tile1 Power-on Sequence Step: 0x0000000F
  ADC Tile0 Power-on Sequence Step: 0x0000000F

Data Converter start up is complete!
----- Startup Complete -----
```

Open Hardware Manager

The screenshot shows the Vivado 2020.2 Project Manager window. The left sidebar contains the 'Flow Navigator' with the following sections:

- PROJECT MANAGER
 - Settings
 - Add Sources
 - Language Templates
 - IP Catalog
- IP INTEGRATOR
 - Create Block Design
 - Open Block Design
 - Generate Block Design
- SIMULATION
 - Run Simulation
- RTL ANALYSIS
 - Open Elaborated Design
- SYNTHESIS
 - Run Synthesis
 - Open Synthesized Design
- IMPLEMENTATION
 - Run Implementation
 - Open Implemented Design
- PROGRAM AND DEBUG
 - Generate Bitstream
 - Open Hardware Manager** (highlighted with a red arrow)

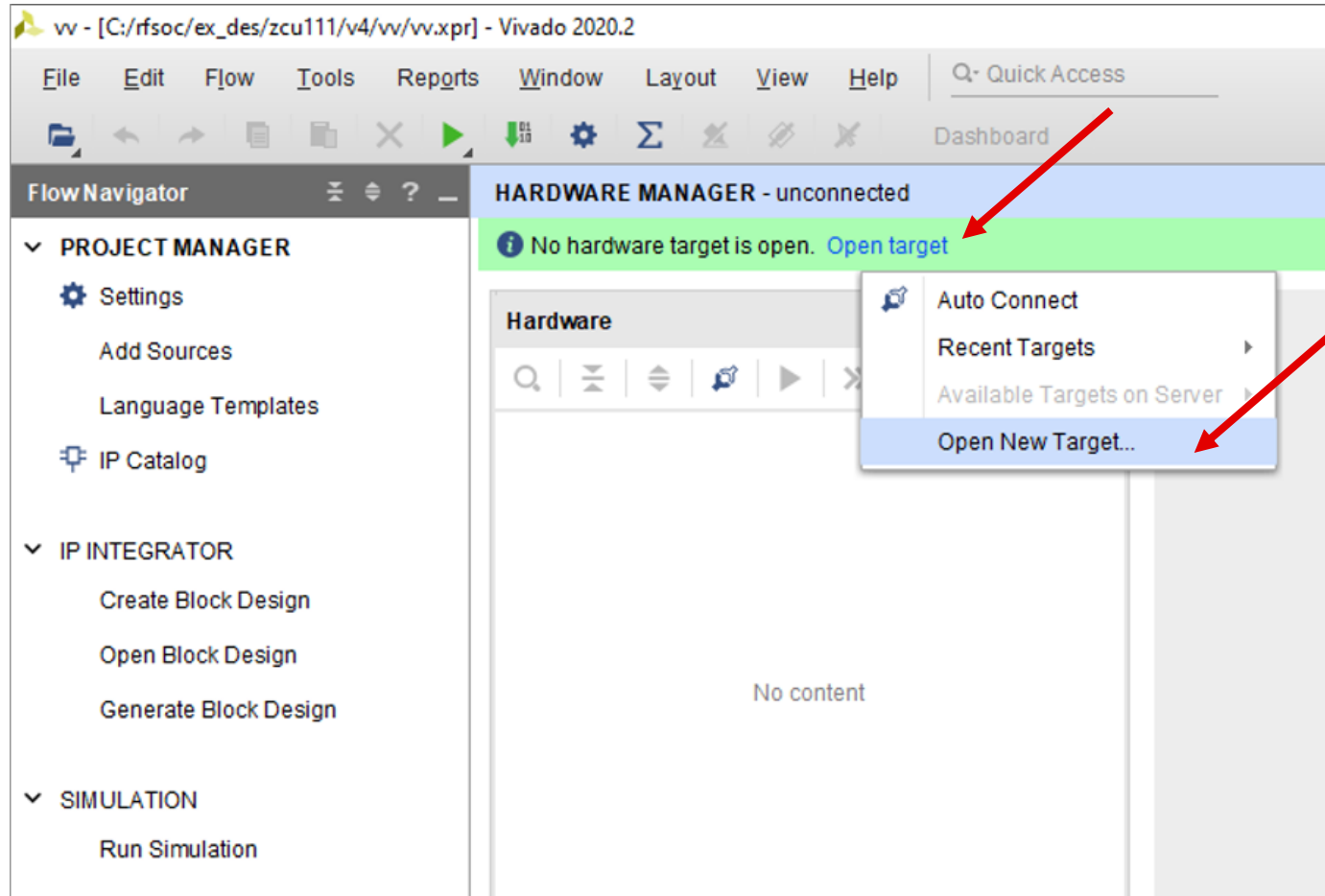
The main area is divided into three panes:

- Sources:** Shows design sources for 'design_1_wrapper', including 'startup' and 'Text'.
- Project Summary:** Displays project details such as name, location, family, and board part information. It includes a 'Board Part' section with a photo of the ZCU111 evaluation board.
- Properties:** A section for object properties, currently empty.

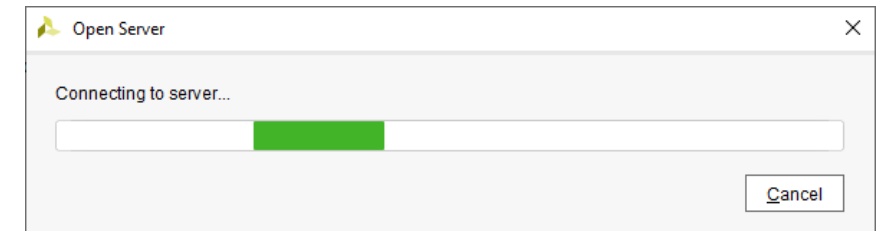
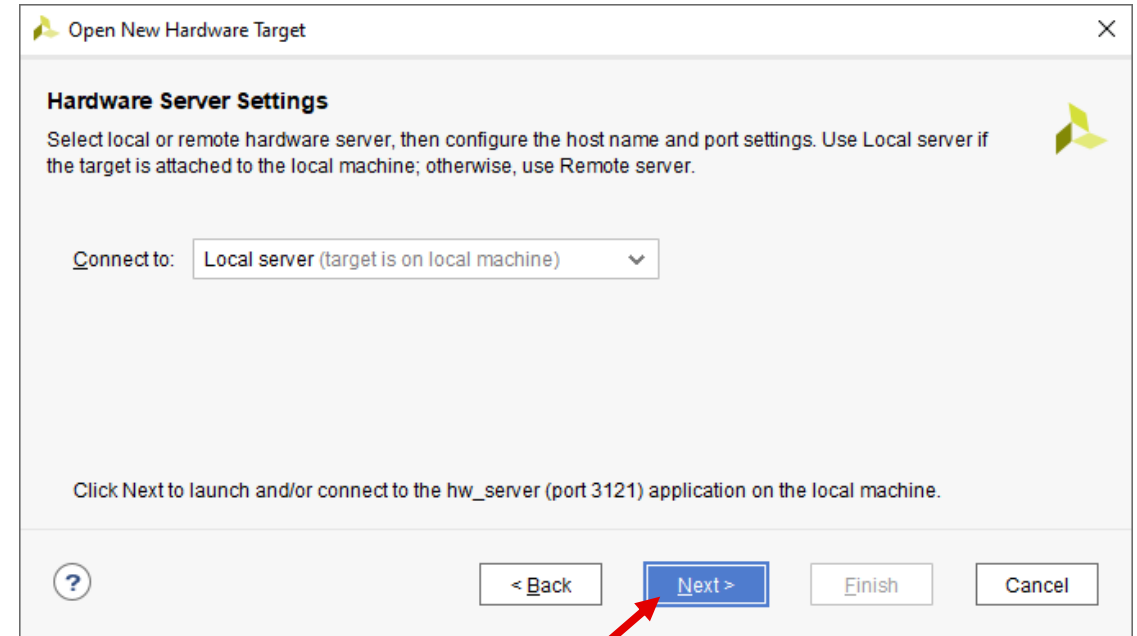
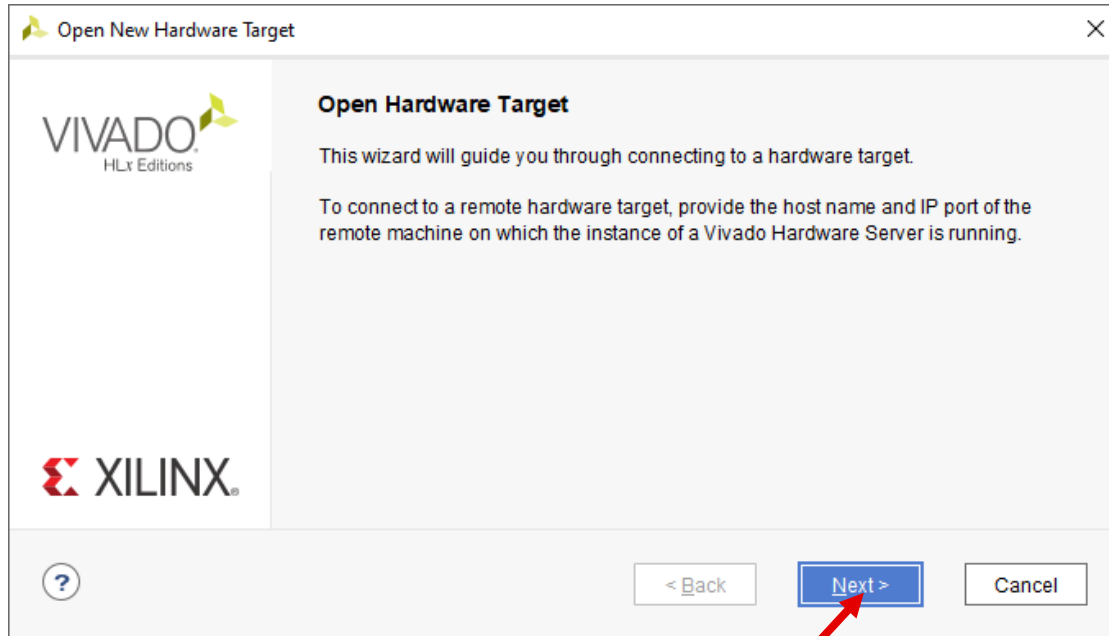
At the bottom, the 'Design Runs' table shows the status of various runs:

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report
synth_1 (active)	constrs_1	synth_design Complete!								0	0	0.0	0	0	6/8/21, 1:40 PM	00:01:51	Vivado Synthesis Defaults (Vivado Synthesis 2018)	Vivado
impl_1	constrs_1	write_bitstream Complete!	1.804	0.000	0.014	0.000	0.000	1.632	0	4917	5833	6.5	0	0	6/8/21, 1:42 PM	00:10:25	Vivado Implementation Defaults (Vivado Implementation 2018)	Vivado
Out-of-Context Module Runs																		
design_1		Submodule Runs Complete													6/8/21, 1:36 PM	00:03:46		

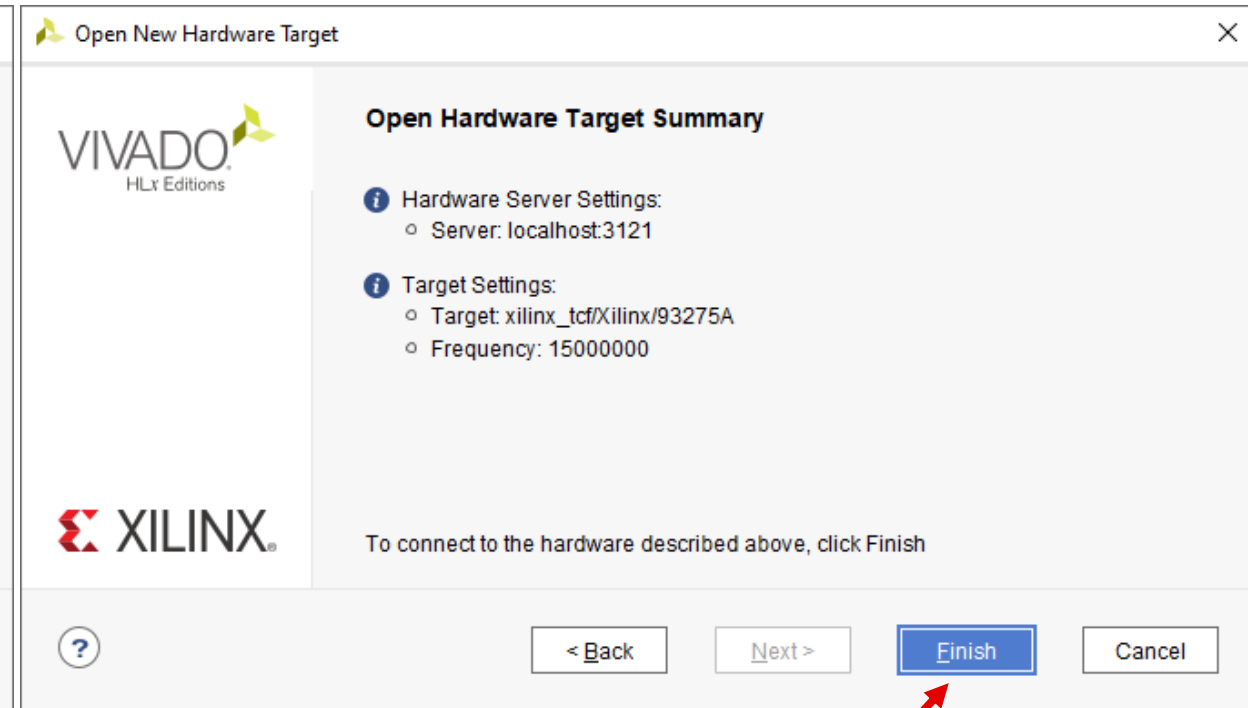
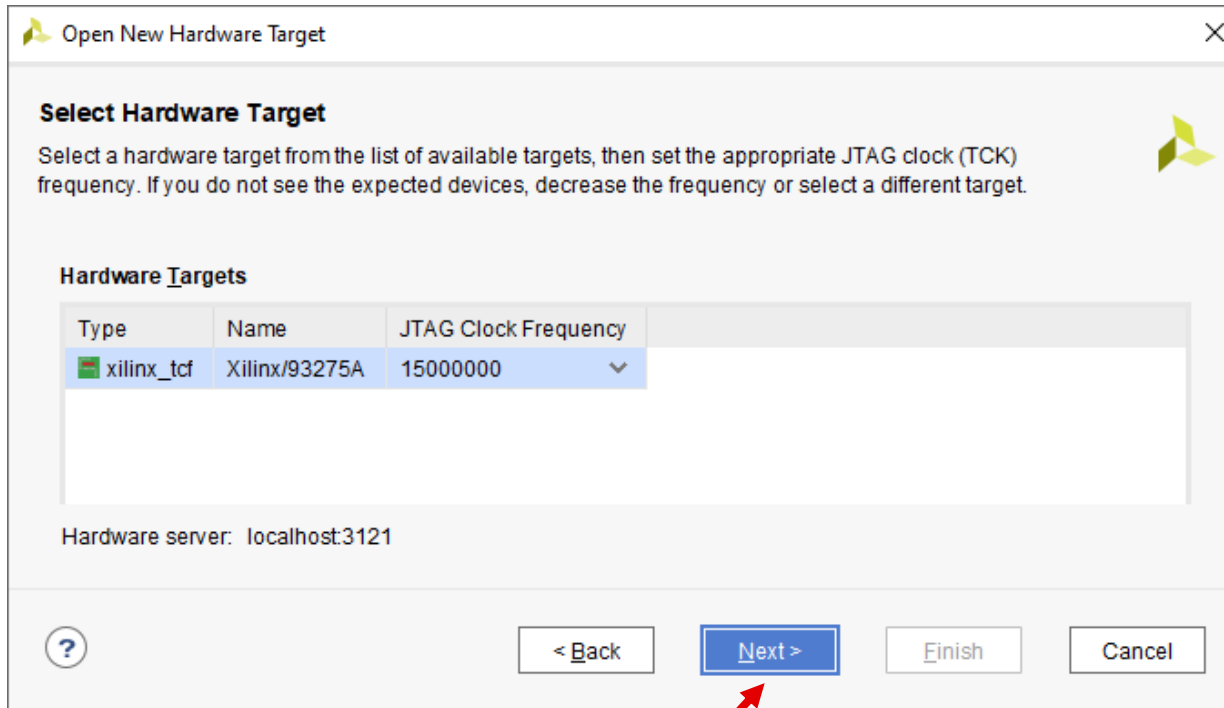
Open New Target



Open Hardware Target



Open Hardware Target Cont'd



Convert Data to the Analog Waveform Style

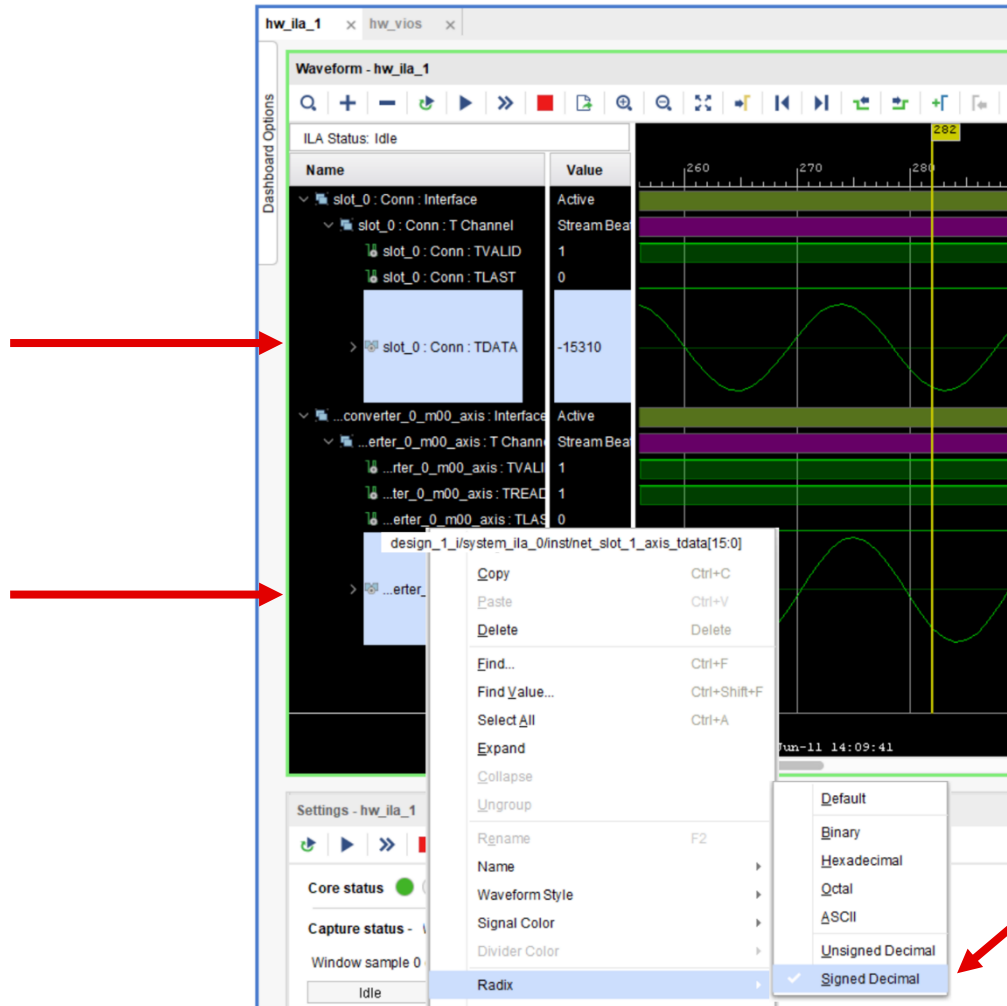
10MHz sine wave
going from the DDS
compiler to the DAC.

ADC capture to the
System ILA.

The screenshot displays the Xilinx ILS interface with the following components:

- Waveform - hw_ila_1:** The main window showing a list of signals and their values. A red arrow points from the text "10MHz sine wave going from the DDS compiler to the DAC." to the signal `slot_0 : dds_compiler_0_M_AXIS_DATA : TDATA`. Another red arrow points from the text "ADC capture to the System ILA." to the signal `slot_1 : usp_rf_data_converter_0_m00_axis : TDATA`.
- Settings - hw_ila_1:** A panel on the left showing the core status as "Idle" and the capture status as "Window 1 of 1".
- Waveform Style Menu:** A context menu is open, showing options like Cut, Copy, Paste, Delete, Find..., etc. The "Waveform Style" option is selected, and a sub-menu is displayed with "Digital", "Analog" (checked), and "Analog Settings..." options. Two red arrows point to the "Analog" option.

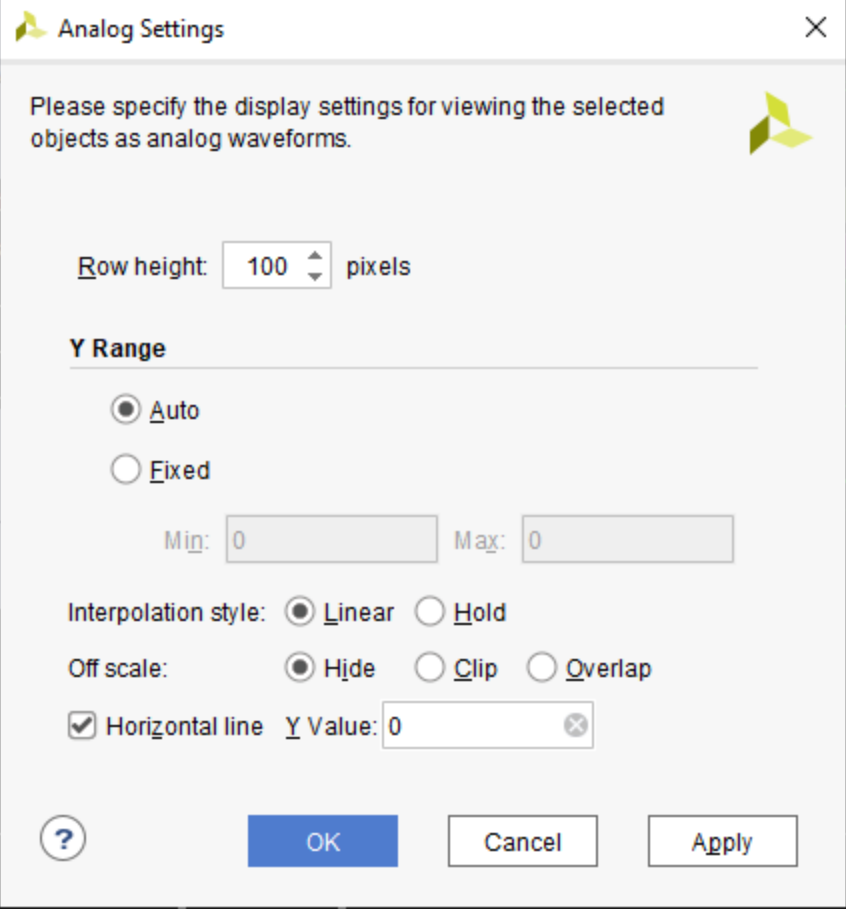
Radix



Use Radix of Signed Decimal.

Analog Settings

Set Row height to 100.



Analog Settings

Please specify the display settings for viewing the selected objects as analog waveforms.

Row height: 100 pixels

Y Range

☒ Auto

☐ Fixed

Min: 0 Max: 0

Interpolation style: ☒ Linear ☐ Hold

Off scale: ☒ Hide ☐ Clip ☐ Overlap

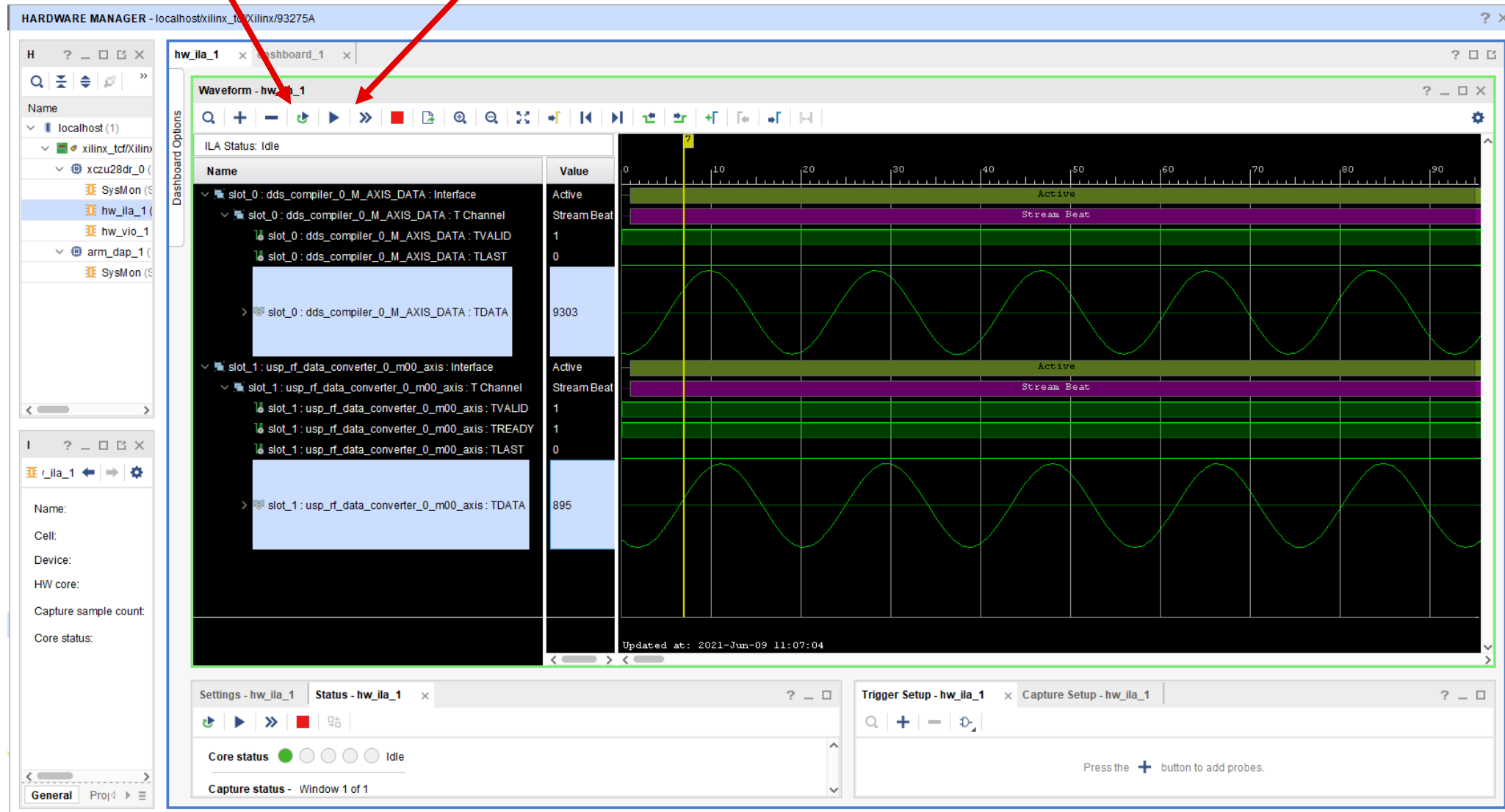
☒ Horizontal line Y Value: 0

? OK Cancel Apply

System ILA Capture

Automatically retrigger

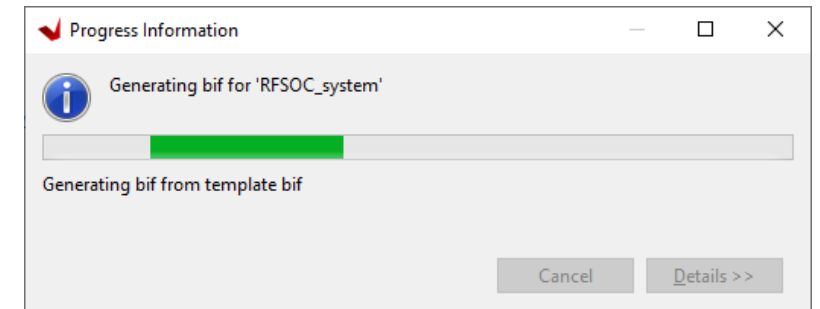
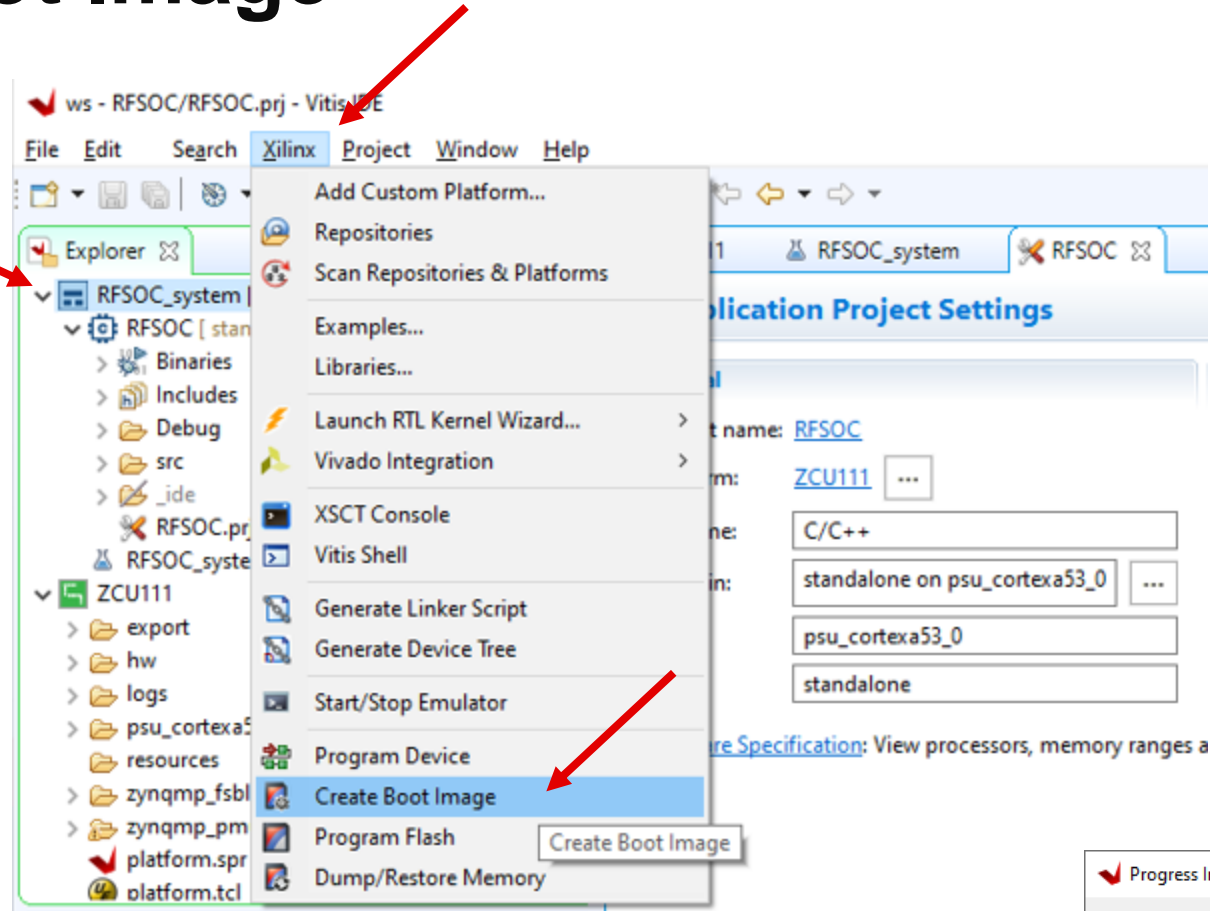
Trigger ILA capture



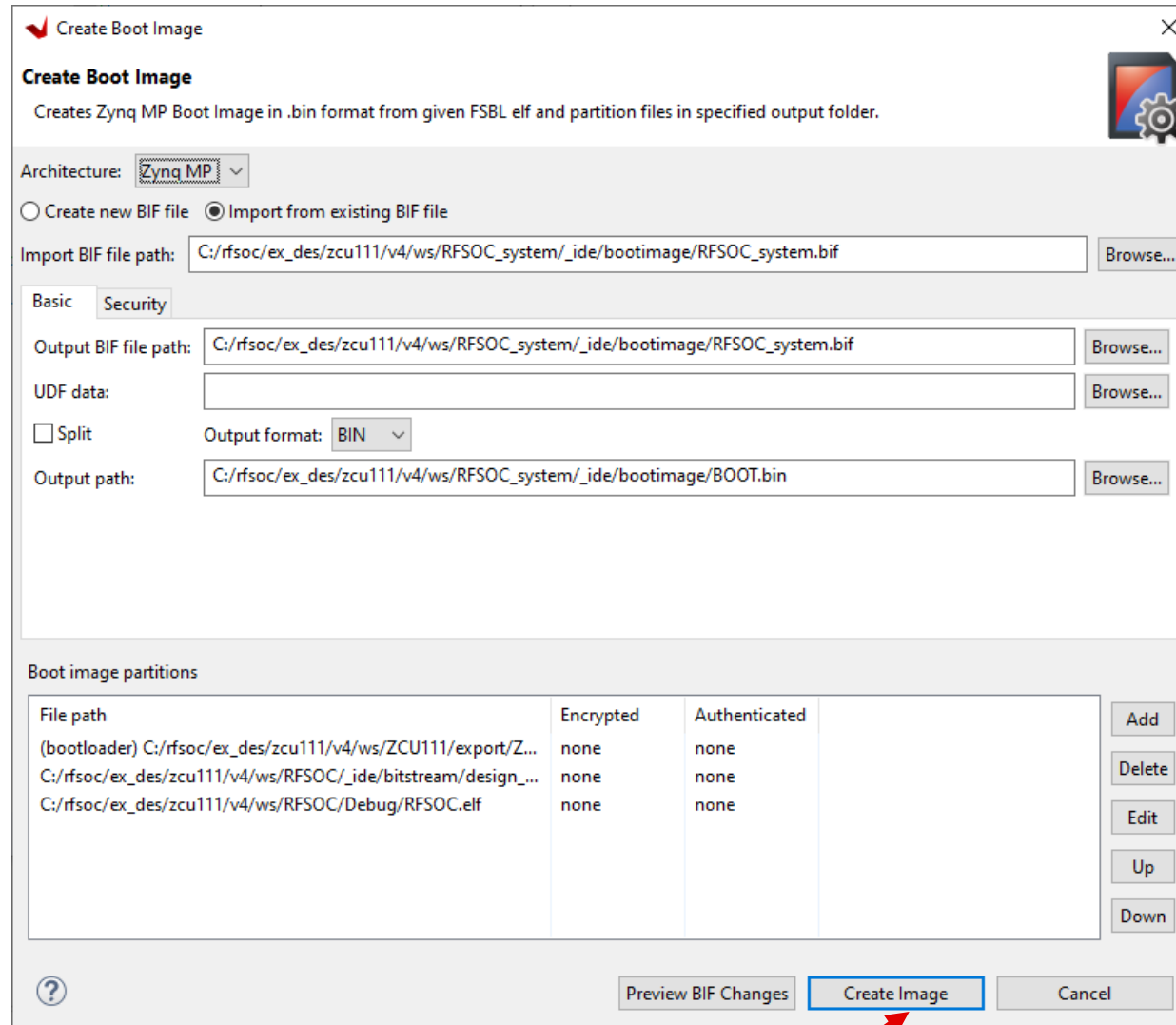
Boot Image

Create Boot Image

To run the application from the SD card rather than directly from Vitis™, create the boot.bin file



Create boot.bin file



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

Import BIF file path:

Basic **Security**

Output BIF file path:

UDF data:

☐ Split

Output path:

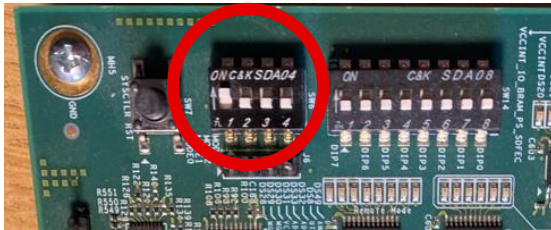
Boot image partitions

File path	Encrypted	Authenticated	
(bootloader) C:/rfsoc/ex_des/zcu111/v4/ws/ZCU111/export/Z...	none	none	
C:/rfsoc/ex_des/zcu111/v4/ws/RFSOC/_ide/bitstream/design_...	none	none	
C:/rfsoc/ex_des/zcu111/v4/ws/RFSOC/Debug/RFSOC.elf	none	none	

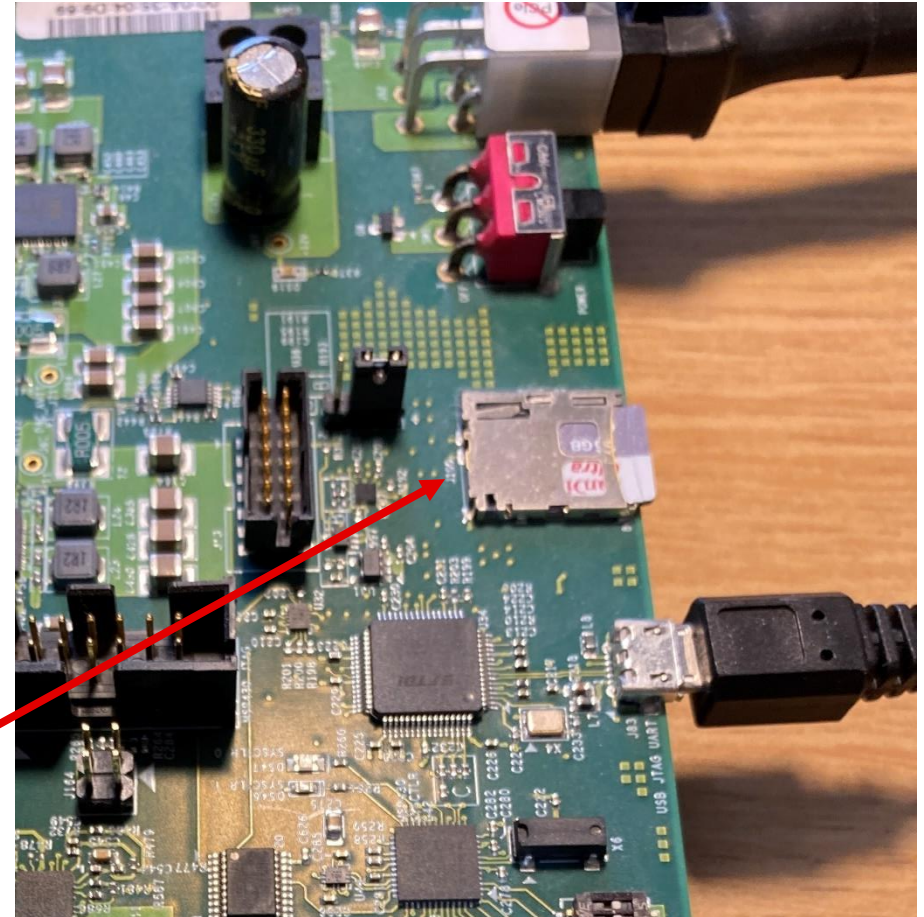
Note the location of the boot.bin file to put at the root level of the SD card.

Boot from SD Card

Set SW6 to on,off,off,off (SD Card boot mode).



Load boot.bin on the SD card, insert it into the board, and turn on the power.





Thank You

