

Set up Vivado Project

https://source.mnt.re/amiga/zz9000-firmware/-/merge_requests/18

When building with Vivado 2023.2 I would get the following error:

ERROR: [Common 17-142] Invalid property name 'dsa.emu_dir'. Please use the 'list_property' command to find properties supported by the target 'project' object.

Commenting out the line allowed the build to proceed, as I couldn't find this property documented anywhere.

The other change is adding "-force" flag to create_project directive, which allows the script to proceed when there are some remnants in place of a failed tcl script run.

Allows building/rebuilding with Vivado 2023.2

 Open Adam Polkosnik requested to merge [apolkosnik/zz9000-firmwa...](#) into master 55 minutes ago

Overview 0 Commits 1 Pipelines 0 **Changes 1**

 Compare master and latest version

zz9000_project.tcl

		@@ -103,7 +103,7 @@ if { \$::argc > 0 } {
103	103	set orig_proj_dir "[file normalize "\$origin_dir/"]"
104	104	
105	105	# Create project
106		- create_project \${_xil_proj_name_} ./\${_xil_proj_name_} -part xc7z020clg400-1
	106	+ create_project -force \${_xil_proj_name_} ./\${_xil_proj_name_} -part xc7z020clg400-1
107	107	
108	108	# Set the directory path for the new project
109	109	set proj_dir [get_property directory [current_project]]
		@@ -115,7 +115,7 @@ set_property -name "dsa.accelerator_binary_content" -value "bitstream" -objects
115	115	set_property -name "dsa.accelerator_binary_format" -value "xclbin2" -objects \$obj
116	116	set_property -name "dsa.description" -value "Vivado generated DSA" -objects \$obj
117	117	set_property -name "dsa.dr_bd_base_address" -value "0" -objects \$obj
118		- set_property -name "dsa.emu_dir" -value "emu" -objects \$obj
	118	+ #set_property -name "dsa.emu_dir" -value "emu" -objects \$obj
119	119	set_property -name "dsa.flash_interface_type" -value "bpix16" -objects \$obj
120	120	set_property -name "dsa.flash_offset_address" -value "0" -objects \$obj
121	121	set_property -name "dsa.flash_size" -value "1024" -objects \$obj

As Vivado projects are not suitable for version control, the Vivado project / block design is exported as a TCL script `zz9000_project.tcl`.

Start Vivado from your terminal in tcl mode:

```
source /path/to/Xilinx/Vivado/2018.3/settings64.sh
```

```
cd /place/where/you/checked/out/zz9000-firmware
vivado -mode tcl
```

Then, execute in the Vivado TCL shell:

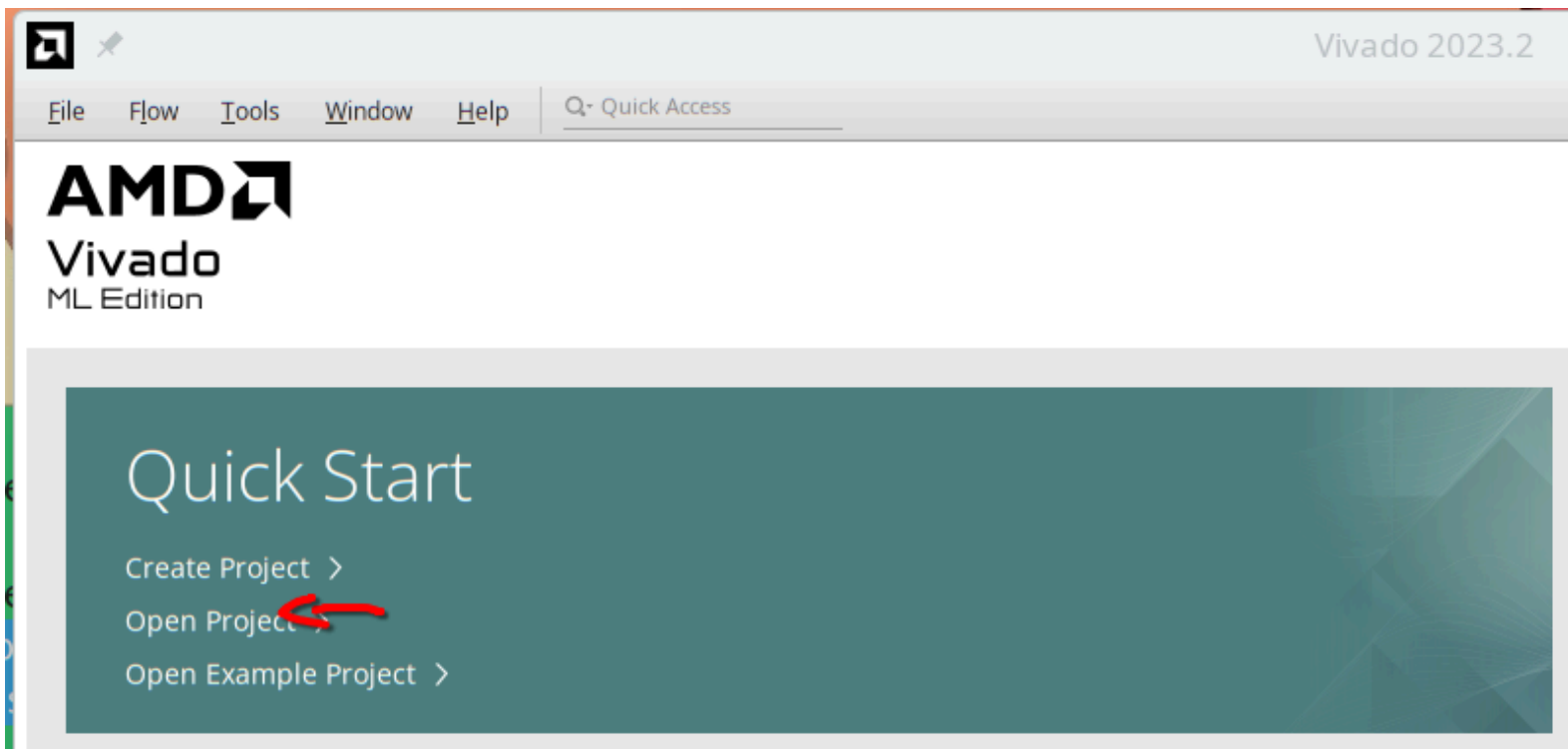
```
source zz9000_project.tcl
exit
```

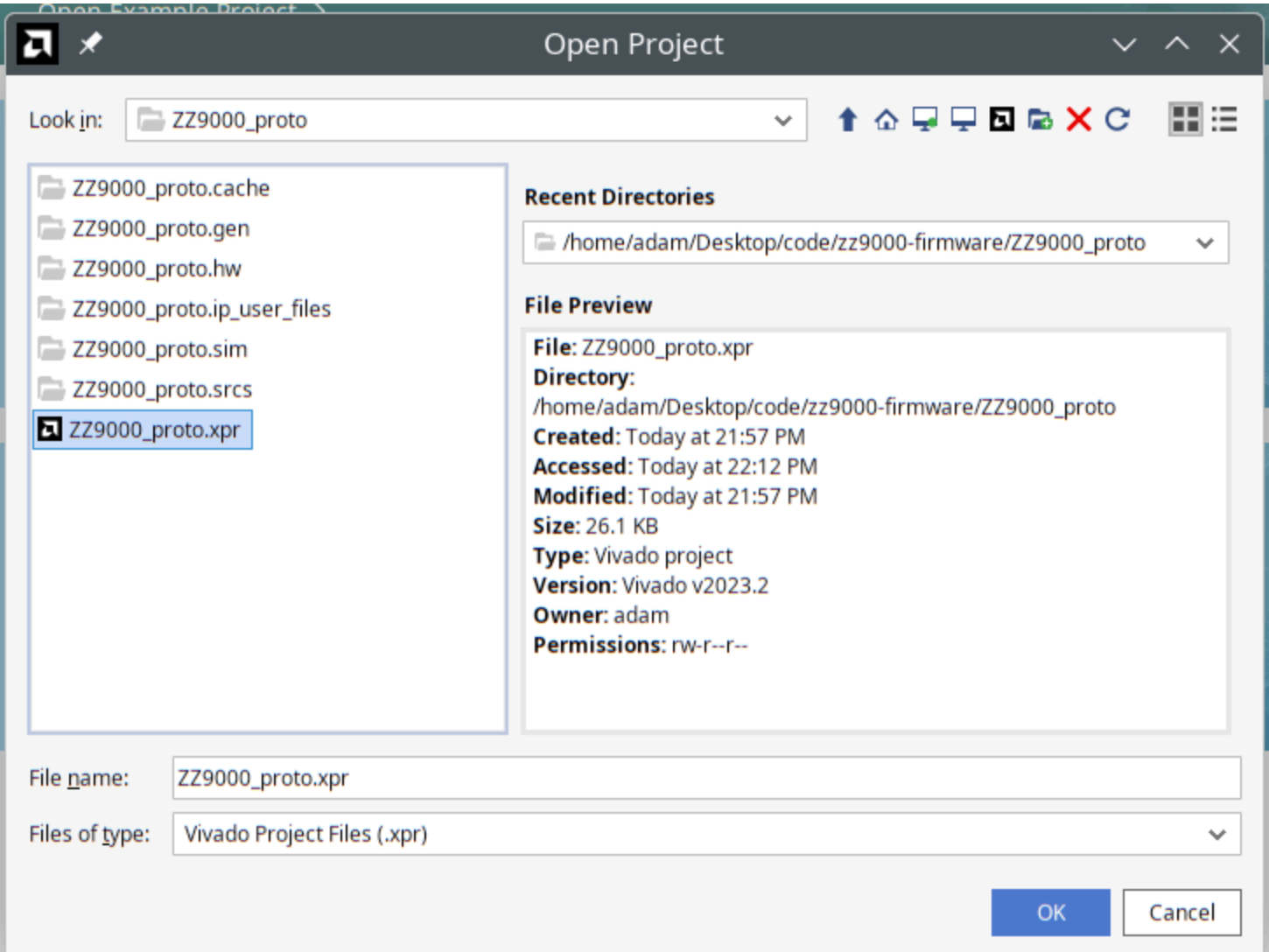
A folder `ZZ9000_proto` should have been created. Start Vivado normally (GUI) and navigate in the Open Project dialog to `zz9000-firmware/ZZ9000_proto` and open `ZZ9000_proto.xpr`.

After a while you should be able to select "Run Synthesis" in Flow Navigator, and after that completes, "Run Implementation" and finally "Generate Bitstream". Finally, select "Export Hardware" from the "File" menu and check "Include Bitstream". Export to "Local to Project". Then, open the SDK by selecting "Launch SDK" from the "File" menu. Leave everything at "Local to Project" and click OK.

Your SDK workspace will start with the `zz9000_ps_wrapper_hw_platform_0` project.

Step by step in screenshots





ZZ9000_proto - [/home/adam/Desktop/code/zz9000-firmware/ZZ9000_proto/ZZ9000_proto.xpr] - Vivado 2023.2

FileEditFlowToolsReportsWindowLayoutViewHelpQ- Quick AccessReady

Flow Navigator

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

Run Linter

Open Elaborated Design

SYNTHESIS

Run Synthesis

Open Synthesized Design

IMPLEMENTATION

Run Implementation

Open Implemented Design

PROGRAM AND DEBUG

Generate Bitstream

Open Hardware Manager

PROJECT MANAGER - ZZ9000_proto

Sources

Design Sources (4)

zz9000_ps_wrapper (zz9000_ps_wrapper.v) (1)

MNTZorro_v0_1_S00_AXI (mntzorro.v)

audio_clock (audio_clock.v)

video_formatter (video_formatter.v)

Constraints (1)

Simulation Sources (4)

Utility Sources

HierarchyIP SourcesLibrariesCompile Order

Properties

Select an object to see properties

Project Summary

OverviewDashboard

SettingsEdit

Project name: ZZ9000_proto

Project location: /home/adam/Desktop/code/zz9000-firmware/ZZ9000_proto

Product family: Zynq-7000

Project part: xc7z020clg400-1

Top module name: zz9000_ps_wrapper

Target language: Verilog

Simulator language: Mixed

SynthesisImplementation

Status: Not started

Messages: No errors or warnings

Part: xc7z020clg400-1

Strategy: Flow_PerfOptimized_high

Report Strategy: Vivado Synthesis Default Reports

Incremental synthesis: Automatically selected checkpoint

Status: Not started

Messages: No errors or warnings

Part: xc7z020clg400-1

Strategy: Flow_RunF

Report Strategy: Timing Cl

Incremental implementation: None

DRC ViolationsTiming

Run Implementation to see DRC results

Run Implementation to see timing

UtilizationPower

Run Synthesis to see utilization results

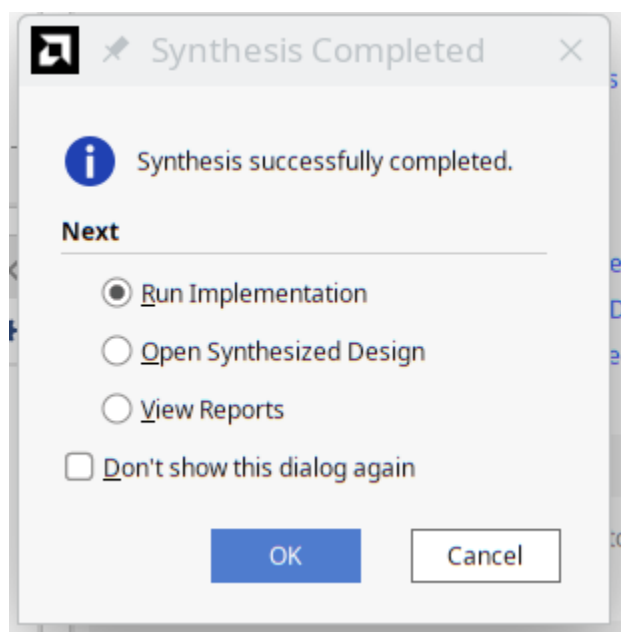
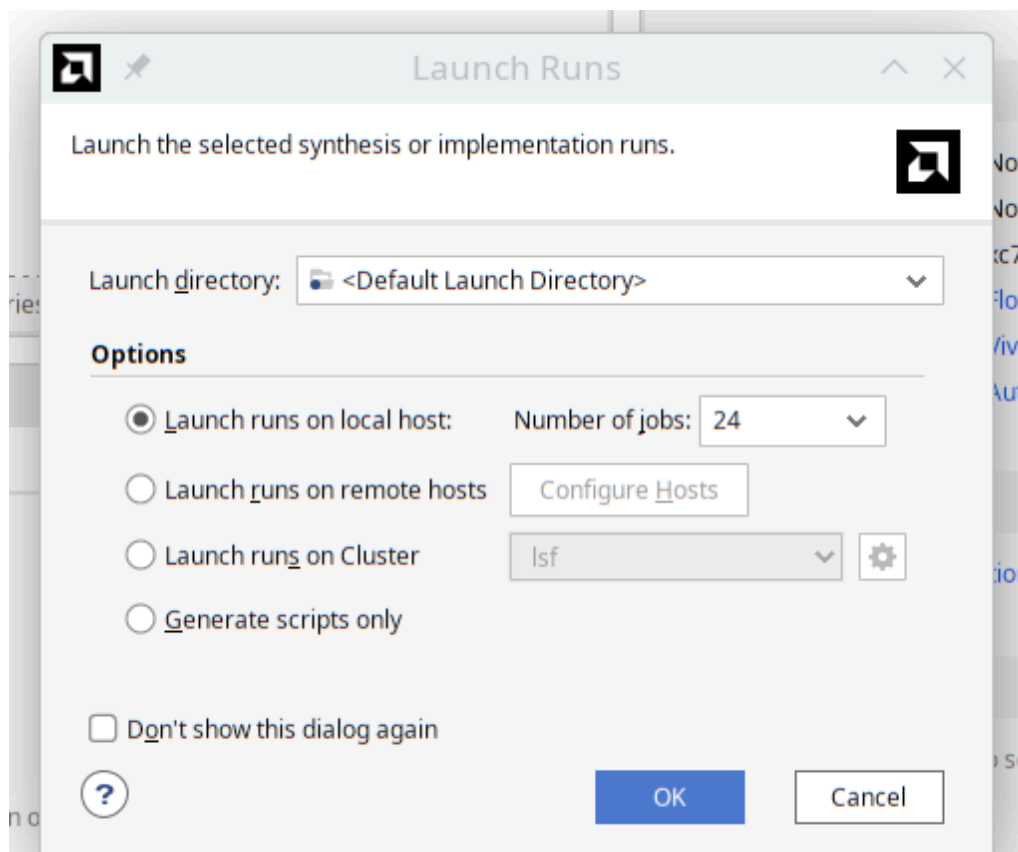
Run Implementation to see power

Tcl ConsoleMessagesLogReportsDesign Runs

Q- %

Name	Constraints	Status	WNS	TNS	WHS	THS	WBSS	TPWS	Total Power	Failed Routes	Methodology	RQA Score	QoR Suggestions	LUT	FF	BRAM	URA
synth_1	constrs_1	Not started															
impl_1	constrs_1	Not started															

ReportStrategy>



Launch Runs

Launch the selected synthesis or implementation runs.

Launch directory: <Default Launch Directory>

Options

☒ Launch runs on local host: Number of jobs: 24

☐ Launch runs on remote hosts: Configure Hosts

☐ Launch runs on Cluster: Isf

☐ Generate scripts only

☐ Don't show this dialog again

?

OK Cancel

Implementation Completed

i Implementation successfully completed.

Next

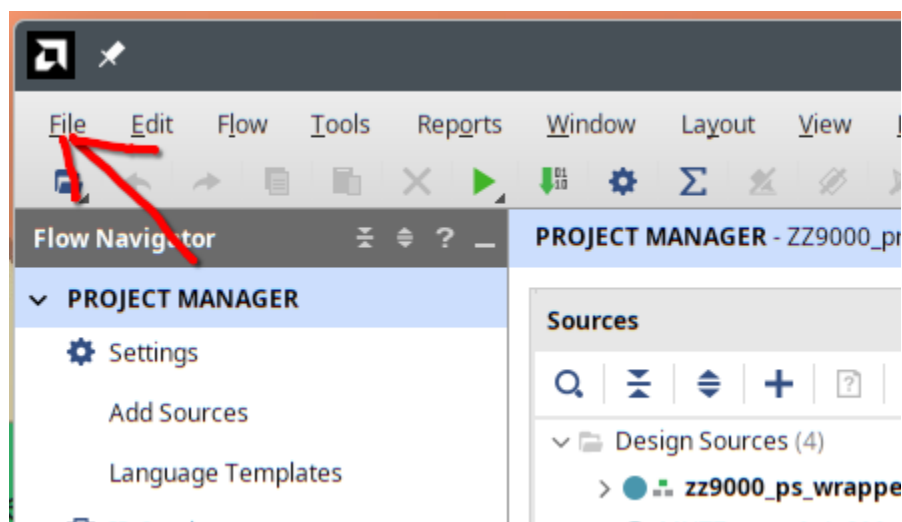
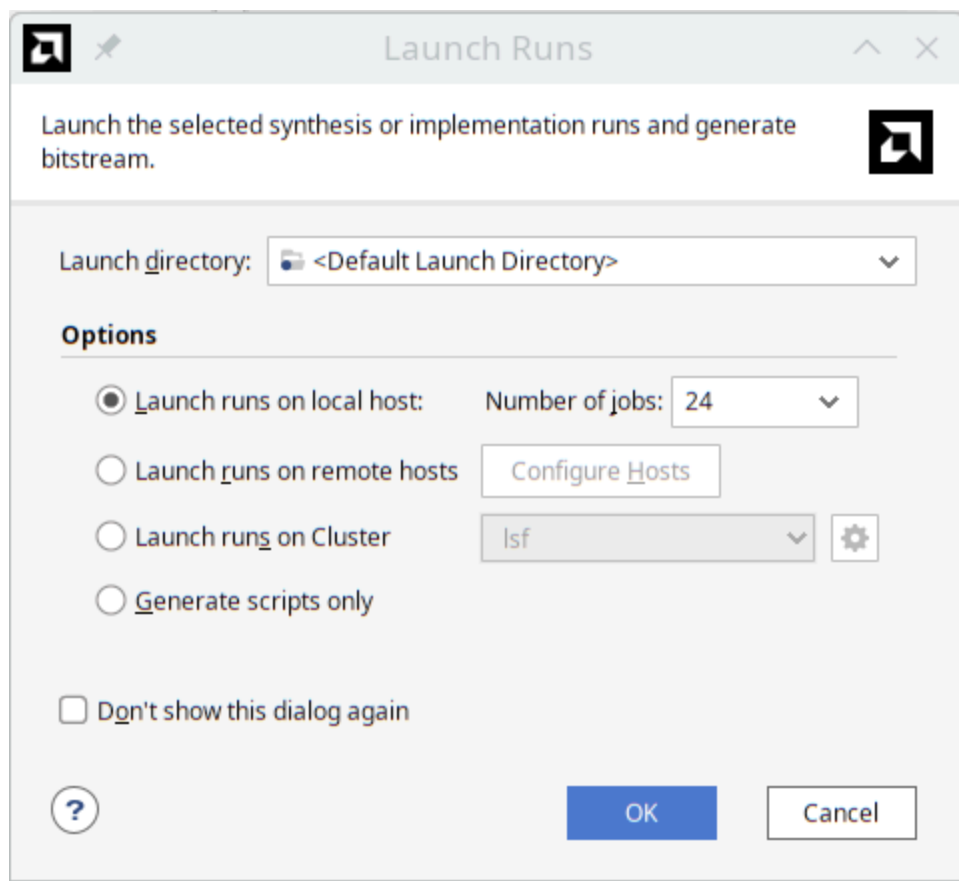
☒ Open Implemented Design

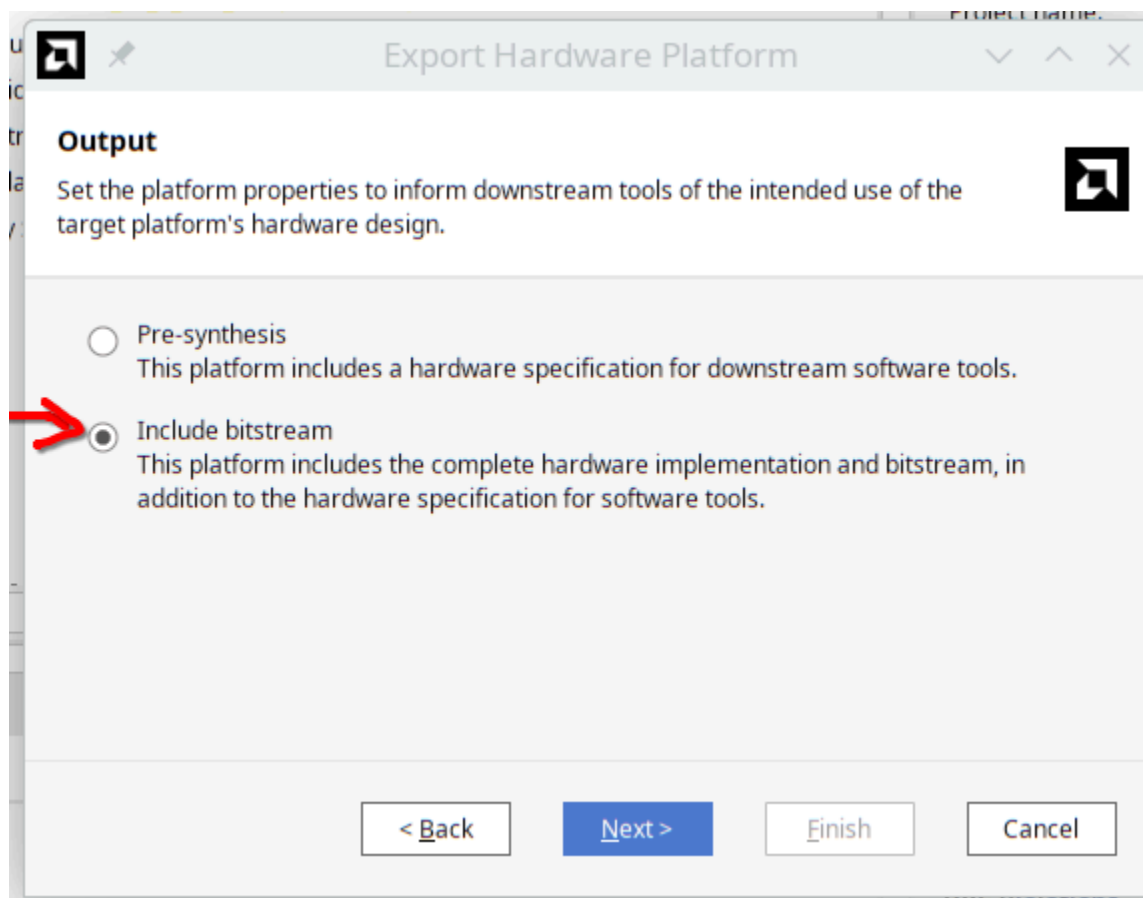
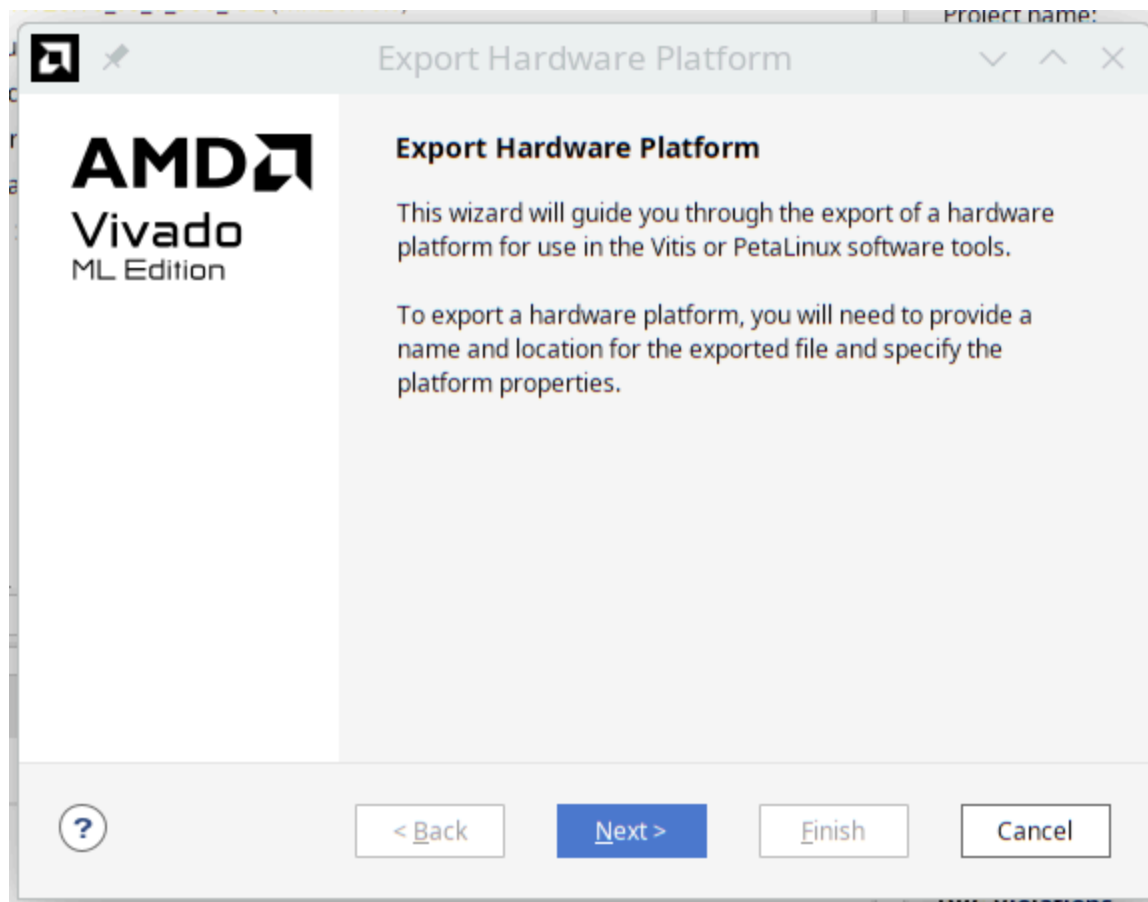
☐ Generate Bitstream

☐ View Reports

☐ Don't show this dialog again

OK Cancel





Export Hardware Platform

Files

Enter the name of your hardware platform file, and the directory where the XSA file will be stored.



XSA file name:

Export to:

The XSA will be written to: /home/adam/Desktop/code/zz9000-firmware/ZZ9000_proto/zz9000_ps_wrapper.xsa

< Back

Next >

Finish

Cancel

Export Hardware Platform



Exporting Hardware Platform

-  A new fixed hardware platform named 'zz9000_ps_wrapper' will be written as '/home/adam/Desktop/code/zz9000-firmware/ZZ9000_proto/zz9000_ps_wrapper.xsa'.
-  The platform will include a post-implementation model, including a bitstream description, describing the hardware for downstream software tools.

To export the platform, click Finish.



< Back

Next >

Finish

Cancel