How to Prepare and Run NPU System Demo (CPU + FPGA)

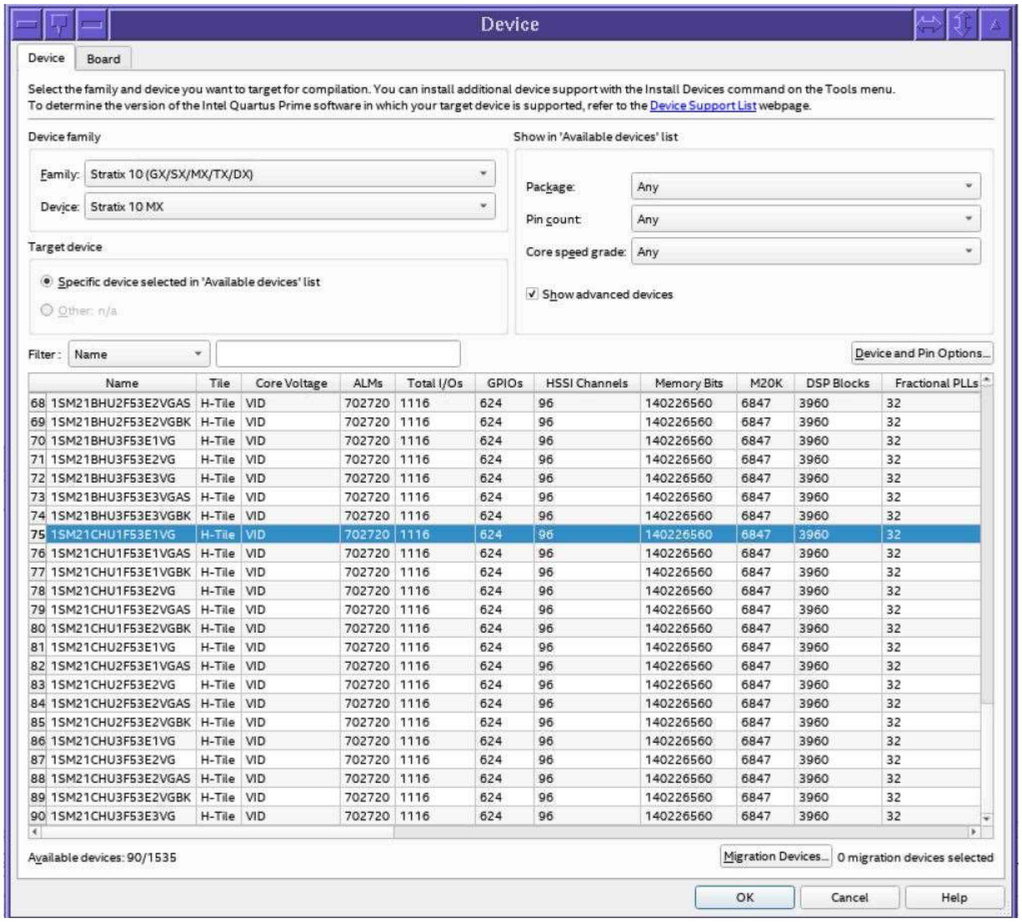
1. Start in the directory where the zip file has been unpacked

1) We call this directory the <project home> directory

2. PCIe EP example setup & Integration with NPU

2.1 PCIe EP example project setup

1) Go to <project home> directory
2) Create one new project with new project wizard & choose empty project
3) Choose S10 MX device with below item at first



Select the family and device you want to target for compilation. You can install additional device support with the Install Devices command on the Tools menu.
To determine the version of the Intel Quartus Prime software in which your target device is supported, refer to the Device Support List webpage.

Device family

Family: Stratix 10 (GX/SX/MX/TX/DX)

Device: Stratix 10 MX

Target device

● Specific device selected in 'Available devices' list

○ Other: n/a

Show in 'Available devices' list

Package: Any

Pin count: Any

Core speed grade: Any

☑ Show advanced devices

Filter: Name

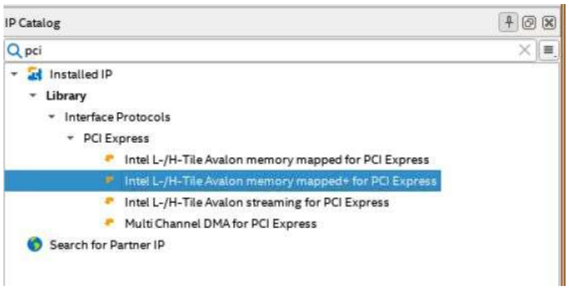
	Name	Tile	Core Voltage	ALMs	Total I/Os	GPIOs	HSSI Channels	Memory Bits	M20K	DSP Blocks	Fractional PLLs
68	1SM21BHU2F53E2VGAS	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
69	1SM21BHU2F53E2VGBK	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
70	1SM21BHU3F53E1VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
71	1SM21BHU3F53E2VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
72	1SM21BHU3F53E3VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
73	1SM21BHU3F53E3VGAS	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
74	1SM21BHU3F53E3VGBK	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
75	1SM21CHU1F53E1VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
76	1SM21CHU1F53E1VGAS	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
77	1SM21CHU1F53E1VGBK	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
78	1SM21CHU1F53E2VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
79	1SM21CHU1F53E2VGAS	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
80	1SM21CHU1F53E2VGBK	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
81	1SM21CHU2F53E1VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
82	1SM21CHU2F53E1VGAS	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
83	1SM21CHU2F53E2VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
84	1SM21CHU2F53E2VGAS	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
85	1SM21CHU2F53E2VGBK	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
86	1SM21CHU3F53E1VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
87	1SM21CHU3F53E2VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
88	1SM21CHU3F53E2VGAS	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
89	1SM21CHU3F53E2VGBK	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32
90	1SM21CHU3F53E3VG	H-Tile	VID	702720	1116	624	96	140226560	6847	3960	32

Available devices: 90/1535

0 migration devices selected

4) Directly click “finish”

5) Search “pci” in IP Catalog & choose that map + as below, then click “add..” button.

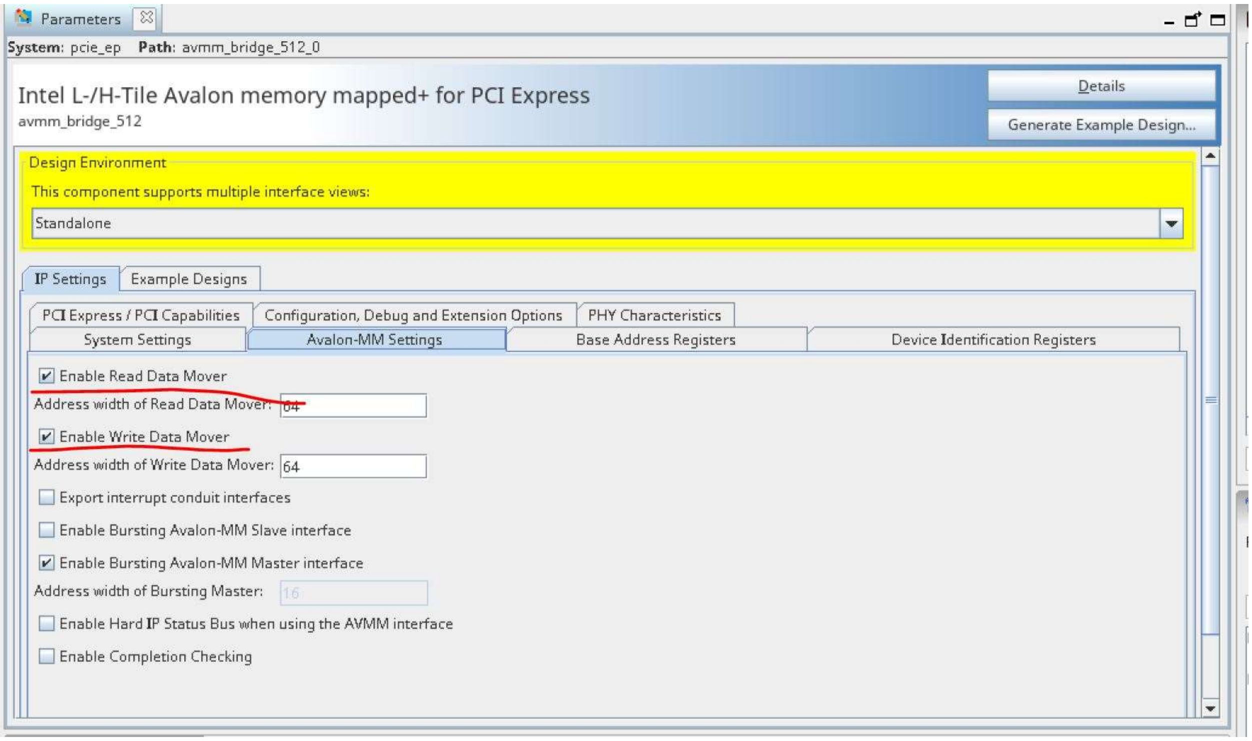


IP Catalog

pci

- Installed IP
  - Library
    - Interface Protocols
      - PCI Express
        - Intel L-/H-Tile Avalon memory mapped for PCI Express
        - Intel L-/H-Tile Avalon memory mapped+ for PCI Express
        - Intel L-/H-Tile Avalon streaming for PCI Express
        - Multi Channel DMA for PCI Express
  - Search for Partner IP

6) Create new PCIe endpoint IP
7) In PCIe EP config wizard, in “Avalon-MM settings” tab, make sure selection for “Enable Read Data Mover” & “Enable Write Data Mover”. (default selected)



Parameters

System: pcie_ep **Path:** avmm_bridge_512_0

Intel L-/H-Tile Avalon memory mapped+ for PCI Express

avmm_bridge_512

Details
Generate Example Design...

Design Environment

This component supports multiple interface views:

Standalone

IP Settings
Example Designs
PCI Express / PCI Capabilities
Configuration, Debug and Extension Options
PHY Characteristics
System Settings
Avalon-MM Settings
Base Address Registers
Device Identification Registers

☑ Enable Read Data Mover

Address width of Read Data Mover: 64

☑ Enable Write Data Mover

Address width of Write Data Mover: 64

☐ Export interrupt conduit interfaces

☐ Enable Bursting Avalon-MM Slave interface

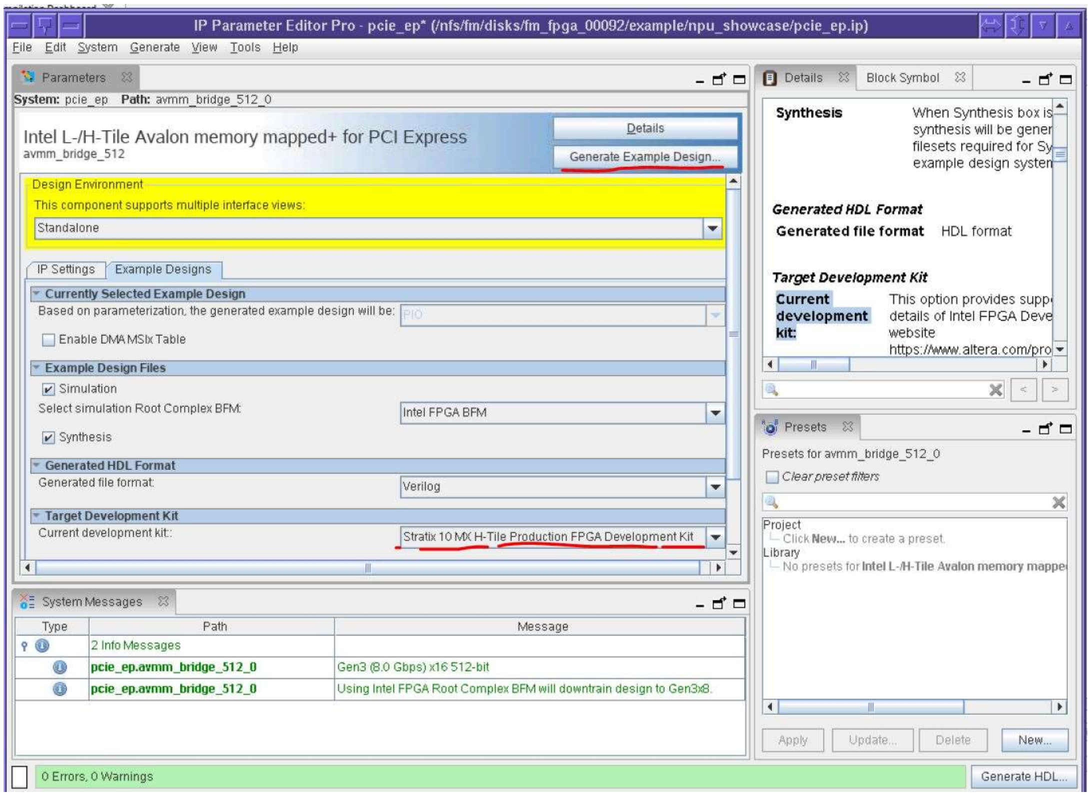
☑ Enable Bursting Avalon-MM Master interface

Address width of Bursting Master: 16

☐ Enable Hard IP Status Bus when using the AVMM interface

☐ Enable Completion Checking

8) In “Example Design” tab, choose S10 MX development kit, then click “Generate Example Design” button.



IP Parameter Editor Pro - pcie_ep* (/nfs/fm/disks/fm_fpga_00092/example/npu_showcase/pcie_ep.ip)

File Edit System Generate View Tools Help
Parameters
Details
Block Symbol

System: pcie_ep **Path:** avmm_bridge_512_0

Intel L-/H-Tile Avalon memory mapped+ for PCI Express
avmm_bridge_512

Details
Generate Example Design...

Design Environment
This component supports multiple interface views:
Standalone

IP Settings
Example Designs

Currently Selected Example Design
Based on parameterization, the generated example design will be: PIO
□ Enable DMA MSIx Table

Example Design Files
☑ Simulation
Select simulation Root Complex BFM: Intel FPGA BFM
☑ Synthesis

Generated HDL Format
Generated file format: Verilog

Target Development Kit
Current development kit: Stratix 10 MX H-Tile Production FPGA Development Kit

Synthesis When Synthesis box is synthesis will be gener filesets required for Sy example design system

Generated HDL Format
Generated file format HDL format

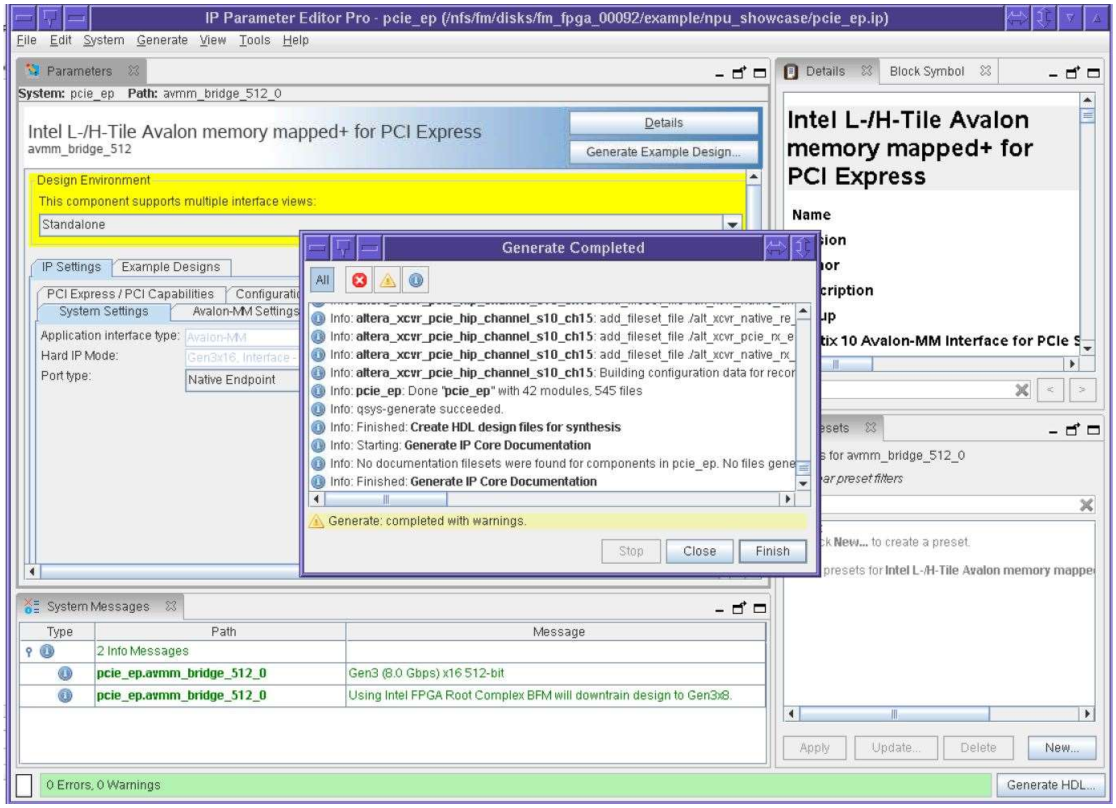
Target Development Kit
Current development kit: This option provides supp details of Intel FPGA Deve website https://www.altera.com/pro

Presets for avmm_bridge_512_0
□ *Clear preset filters*
Project
Click **New...** to create a preset.
Library
No presets for **Intel L-/H-Tile Avalon memory mappe**

Type	Path	Message
ⓘ	2 Info Messages	
ⓘ	**pcie_ep.avmm_bridge_512_0**	Gen3 (8.0 Gbps) x16 512-bit
ⓘ	**pcie_ep.avmm_bridge_512_0**	Using Intel FPGA Root Complex BFM will downtrain design to Gen3x8.

0 Errors, 0 Warnings

9) Save & generate.



IP Parameter Editor Pro - pcie_ep (/nfs/fm/disks/fm_fpga_00092/example/npu_showcase/pcie_ep.ip)

System: pcie_ep **Path:** avmm_bridge_512_0

Intel L-/H-Tile Avalon memory mapped+ for PCI Express
avmm_bridge_512

Design Environment
This component supports multiple interface views:
Standalone

Application interface type: Avalon-MM
Hard IP Mode: Gen3x16, Interface -
Port type: Native Endpoint

Generate Completed

```
Info: altera_xcvr_pcie_hip_channel_s10_ch15: add_fileset_file ./alt_xcvr_native_re_
Info: altera_xcvr_pcie_hip_channel_s10_ch15: add_fileset_file ./alt_xcvr_pcie_rx_e
Info: altera_xcvr_pcie_hip_channel_s10_ch15: add_fileset_file ./alt_xcvr_native_rx_
Info: altera_xcvr_pcie_hip_channel_s10_ch15: Building configuration data for recor
Info: pcie_ep: Done "pcie_ep" with 42 modules, 545 files
Info: qsys-generate succeeded.
Info: Finished: Create HDL design files for synthesis
Info: Starting: Generate IP Core Documentation
Info: No documentation filesets were found for components in pcie_ep. No files gene
Info: Finished: Generate IP Core Documentation
```

⚠ Generate: completed with warnings.

Intel L-/H-Tile Avalon memory mapped+ for PCI Express

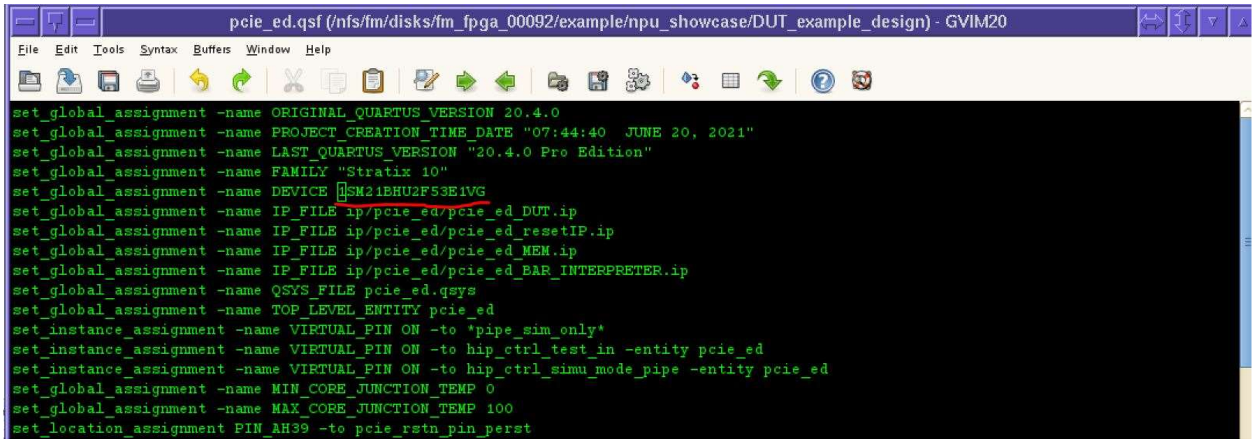
Name
ion
or
cription
up
tix 10 Avalon-MM Interface for PCIe S

s for avmm_bridge_512_0
ar preset filters
k **New...** to create a preset.
presets for **Intel L-/H-Tile Avalon memory mappe**

Type	Path	Message
ⓘ	2 Info Messages	
ⓘ	**pcie_ep.avmm_bridge_512_0**	Gen3 (8.0 Gbps) x16 512-bit
ⓘ	**pcie_ep.avmm_bridge_512_0**	Using Intel FPGA Root Complex BFM will downtrain design to Gen3x8.

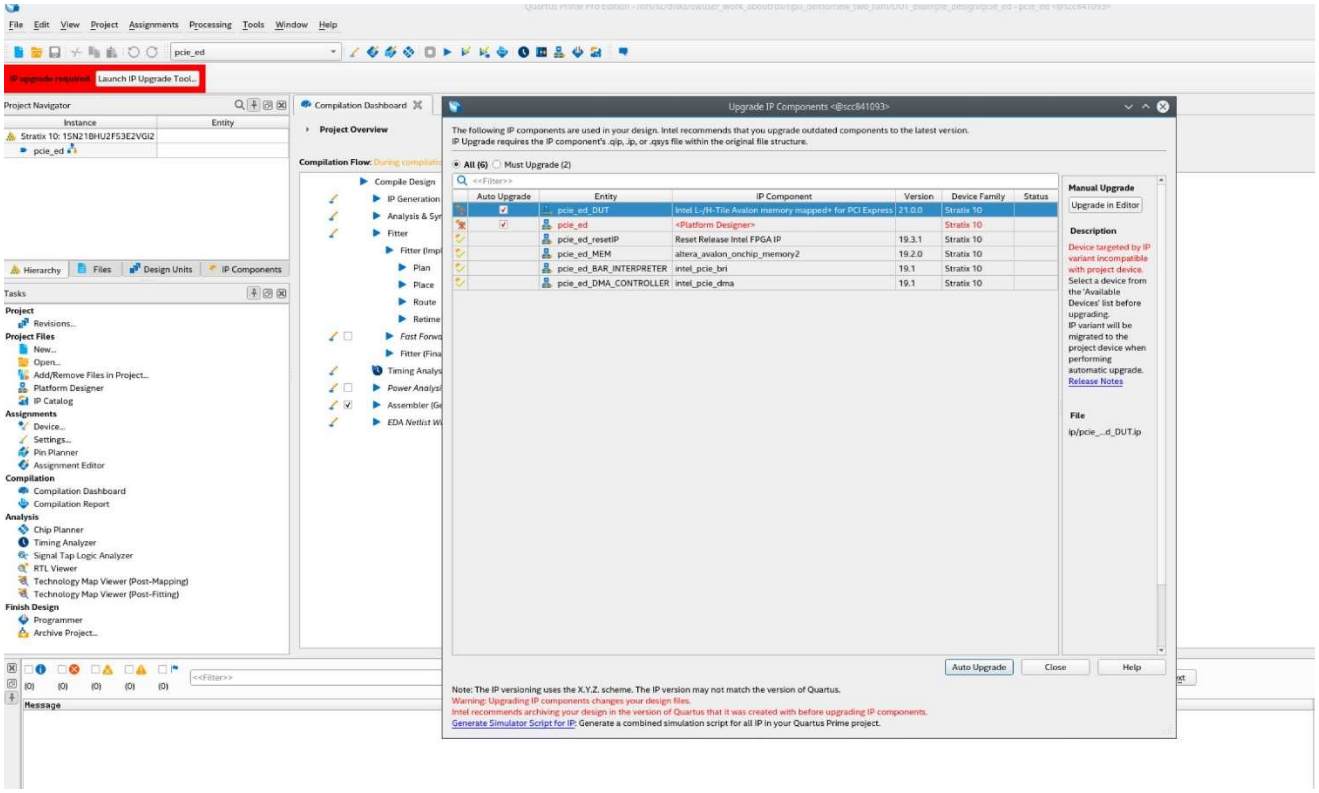
0 Errors, 0 Warnings

10) Since Stratix 10 NX can't be selected from Quartus GUI, you have to manually edit *.qsf under your new PCIe example generate folder to update device name to "1SN21BHU2F53E2VGI2"


pcie_ed.qsf (/nfs/fm/disks/fm_fpga_00092/example/npu_showcase/DUT_example_design) - GVIM20
File Edit Tools Syntax Buffers Window Help

```
set_global_assignment -name ORIGINAL_QUARTUS_VERSION 20.4.0
set_global_assignment -name PROJECT_CREATION_TIME_DATE "07:44:40  JUNE 20, 2021"
set_global_assignment -name LAST_QUARTUS_VERSION "20.4.0 Pro Edition"
set_global_assignment -name FAMILY "Stratix 10"
set_global_assignment -name DEVICE 1SM21BHU2F53E1VG
set_global_assignment -name IP_FILE ip/pcie_ed/pcie_ed_DUT.ip
set_global_assignment -name IP_FILE ip/pcie_ed/pcie_ed_resetIP.ip
set_global_assignment -name IP_FILE ip/pcie_ed/pcie_ed_MEM.ip
set_global_assignment -name IP_FILE ip/pcie_ed/pcie_ed_BAR_INTERPRETER.ip
set_global_assignment -name QSYS_FILE pcie_ed.qsys
set_global_assignment -name TOP_LEVEL_ENTITY pcie_ed
set_instance_assignment -name VIRTUAL_PIN ON -to *pipe_sim_only*
set_instance_assignment -name VIRTUAL_PIN ON -to hip_ctrl_test_in -entity pcie_ed
set_instance_assignment -name VIRTUAL_PIN ON -to hip_ctrl_simu_mode_pipe -entity pcie_ed
set_global_assignment -name MIN_CORE_JUNCTION_TEMP 0
set_global_assignment -name MAX_CORE_JUNCTION_TEMP 100
set_location_assignment PIN_AH39 -to pcie_rstn_pin_perst
```

11) Re-load project (*.qpf) under your new PCIe example generate folder, then you will see below notice with RED (you have to update IP for you manually change device name)



The following IP components are used in your design. Intel recommends that you upgrade outdated components to the latest version.
IP Upgrade requires the IP component's .qip, .ip, or .qsys file within the original file structure.

● All (6) ○ Must Upgrade (2)

Auto Upgrade	Entity	IP Component	Version	Device Family	Status
☑	pcie_ed_DUT	Intel L-/H-Tile Avalon memory mapped+ for PCI Express	21.0.0	Stratix 10	
☑	pcie_ed	<Platform Designer>		Stratix 10	
	pcie_ed_resetIP	Reset Release Intel FPGA IP	19.3.1	Stratix 10	
	pcie_ed_MEM	altera_avalon_onchip_memory2	19.2.0	Stratix 10	
	pcie_ed_BAR_INTERPRETER	intel_pcie_bri	19.1	Stratix 10	
	pcie_ed_DMA_CONTROLLER	intel_pcie_dma	19.1	Stratix 10	

Description

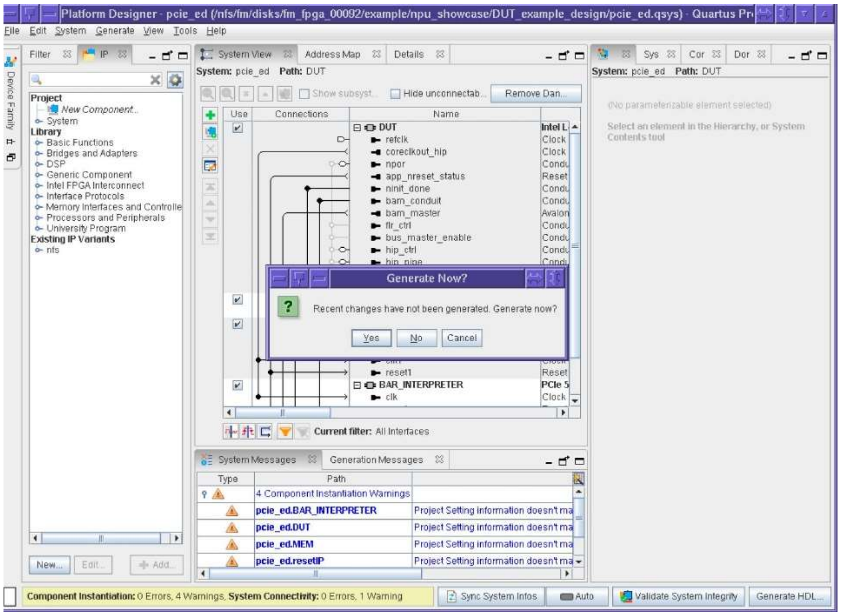
Device targeted by IP variant incompatible with project device. Select a device from the 'Available Devices' list before upgrading. IP variant will be migrated to the project device when performing automatic upgrade. Release Notes

File

ip/pcie_...d_DUT.ip

Note: The IP versioning uses the X.Y.Z. scheme. The IP version may not match the version of Quartus.
Warning: Upgrading IP components changes your design files.
Intel recommends archiving your design in the version of Quartus that it was created with before upgrading IP components.
Generate Simulator Script for IP: Generate a combined simulation script for all IP in your Quartus Prime project.

12) Click “Update in Editor” button, load platform designer, save → exit → generate.
(Note, please remember to choose Verilog for simulation)


Platform Designer - pcie_ed (/nfs/fm/disks/fm_fpga_00092/example/npu_showcase/DUT_example_design/pcie_ed.qsys) - Quartus Pr
File Edit System Generate View Tools Help
Filter IP
Project
New Component...
System
Library
Basic Functions
Bridges and Adapters
DSP
Generic Component
Intel FPGA Interconnect
Interface Protocols
Memory Interfaces and Controlle
Processors and Peripherals
University Program
Existing IP Variants
nfs
New... Edit... Add...
System View Address Map Details
System: pcie_ed Path: DUT
Show subsyst... Hide unconnectab... Remove Dan...
Use Connections Name
DUT Intel L
refclk Clock
coreclkout_hip Clock
npor Condu
app_nreset_status Reset
ninit_done Condu
bam_conduit Condu
bam_master Avalon
flr_ctrl Condu
bus_master_enable Condu
hip_ctrl Condu
hip_pipe Condu
reset1 Reset
BAR_INTERPRETER PCIe 5
clk Clock
Current filter: All Interfaces
Sys Cor Dor
System: pcie_ed Path: DUT
(No parameterizable element selected)
Select an element in the Hierarchy, or System Contents tool
System Messages Generation Messages
Sync System Infos Auto Validate System Integrity Generate HDL...

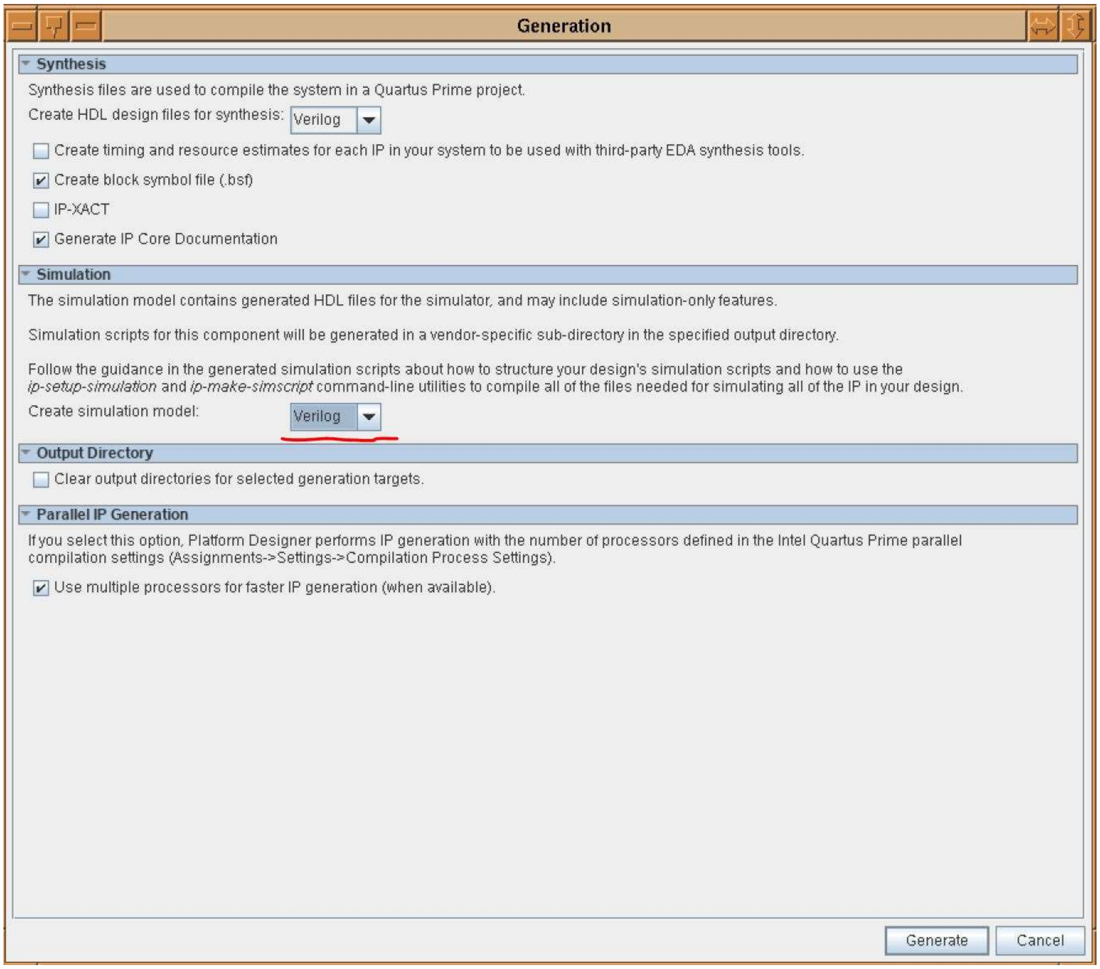
Generate Now?

Recent changes have not been generated. Generate now?

Yes No Cancel

Type	Path	
⚠	4 Component Instantiation Warnings	
⚠	pcie_ed.BAR_INTERPRETER	Project Setting information doesn't ma
⚠	pcie_ed.DUT	Project Setting information doesn't ma
⚠	pcie_ed.MEM	Project Setting information doesn't ma
⚠	pcie_ed.resetIP	Project Setting information doesn't ma

Component Instantiation: 0 Errors, 4 Warnings, **System Connectivity:** 0 Errors, 1 Warning



Generation

Synthesis

Synthesis files are used to compile the system in a Quartus Prime project.

Create HDL design files for synthesis: Verilog

☐ Create timing and resource estimates for each IP in your system to be used with third-party EDA synthesis tools.

☑ Create block symbol file (.bsf)

☐ IP-XACT

☑ Generate IP Core Documentation

Simulation

The simulation model contains generated HDL files for the simulator, and may include simulation-only features.

Simulation scripts for this component will be generated in a vendor-specific sub-directory in the specified output directory.

Follow the guidance in the generated simulation scripts about how to structure your design's simulation scripts and how to use the *ip-setup-simulation* and *ip-make-simscript* command-line utilities to compile all of the files needed for simulating all of the IP in your design.

Create simulation model: Verilog

Output Directory

☐ Clear output directories for selected generation targets.

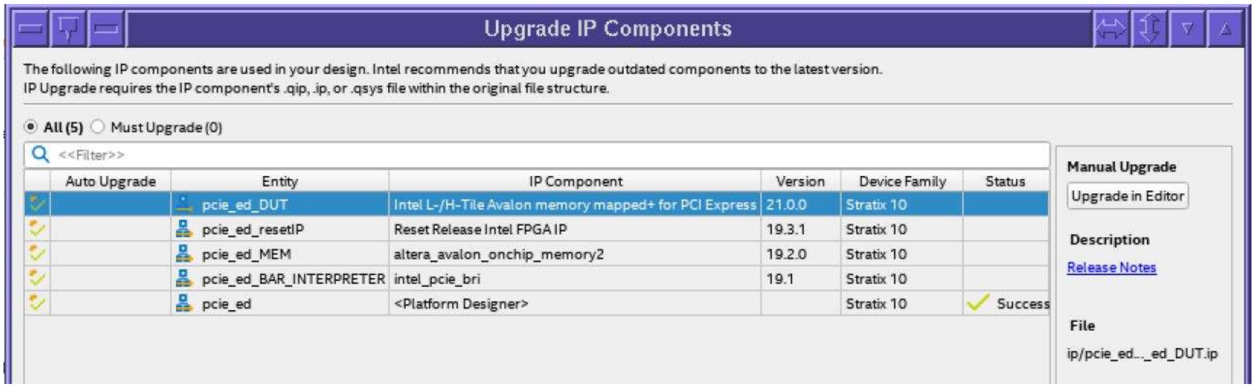
Parallel IP Generation

If you select this option, Platform Designer performs IP generation with the number of processors defined in the Intel Quartus Prime parallel compilation settings (Assignments->Settings->Compilation Process Settings).

☑ Use multiple processors for faster IP generation (when available).

Generate Cancel

13) Update IP in red one by one to finally all IP updated & without highlighted with red.



Upgrade IP Components

The following IP components are used in your design. Intel recommends that you upgrade outdated components to the latest version.
IP Upgrade requires the IP component's .qip, .ip, or .qsys file within the original file structure.

● All (5) ○ Must Upgrade (0)

<<Filter>>

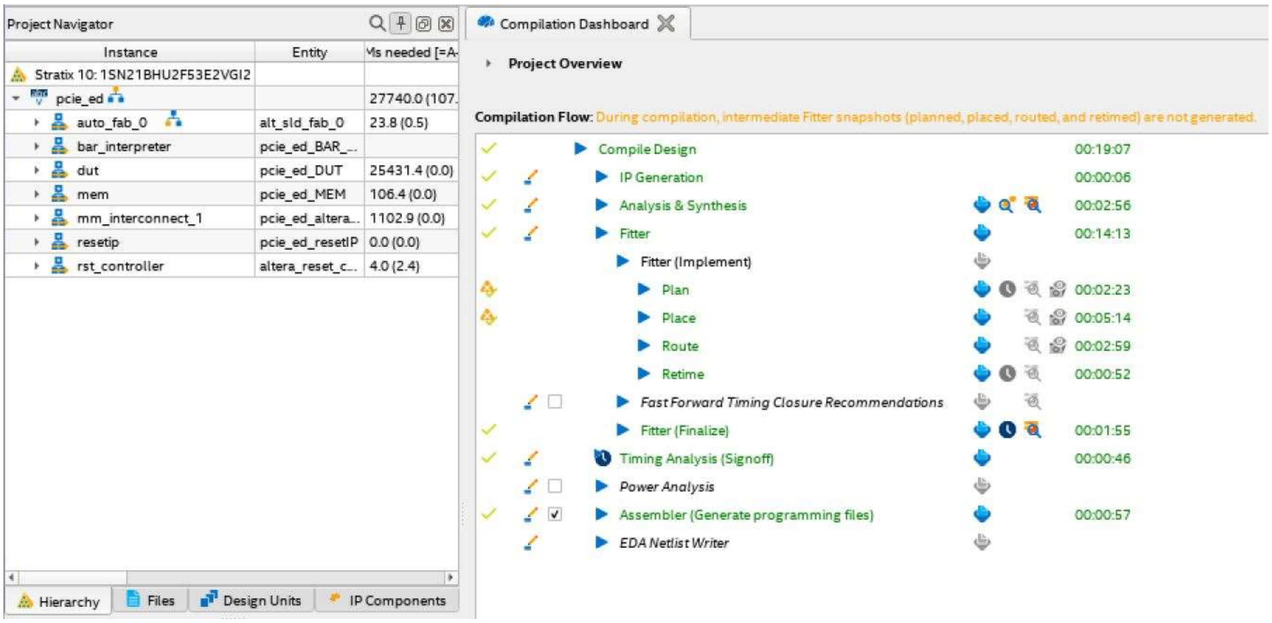
Auto Upgrade	Entity	IP Component	Version	Device Family	Status
✓	pcie_ed_DUT	Intel L-/H-Tile Avalon memory mapped+ for PCI Express	21.0.0	Stratix 10	
✓	pcie_ed_resetIP	Reset Release Intel FPGA IP	19.3.1	Stratix 10	
✓	pcie_ed_MEM	altera_avalon_onchip_memory2	19.2.0	Stratix 10	
✓	pcie_ed_BAR_INTERPRETER	intel_pcie_bri	19.1	Stratix 10	
✓	pcie_ed	<Platform Designer>		Stratix 10	✓ Success

Manual Upgrade
Upgrade in Editor

Description
Release Notes

File
ip/pcie_ed..._ed_DUT.ip

14) Click "run" button to generate one PCIe EP image at first before integrating with NPU IP.



Project Navigator

Instance	Entity	Ms needed [=A-
Stratix 10: 1SN21BHU2F53E2VGI2		
pcie_ed		27740.0 (107.
auto_fab_0	alt_sld_fab_0	23.8 (0.5)
bar_interpreter	pcie_ed_BAR_...	
dut	pcie_ed_DUT	25431.4 (0.0)
mem	pcie_ed_MEM	106.4 (0.0)
mm_interconnect_1	pcie_ed_altera...	1102.9 (0.0)
resetip	pcie_ed_resetIP	0.0 (0.0)
rst_controller	altera_reset_c...	4.0 (2.4)

Hierarchy | Files | Design Units | IP Components

Compilation Dashboard

Project Overview

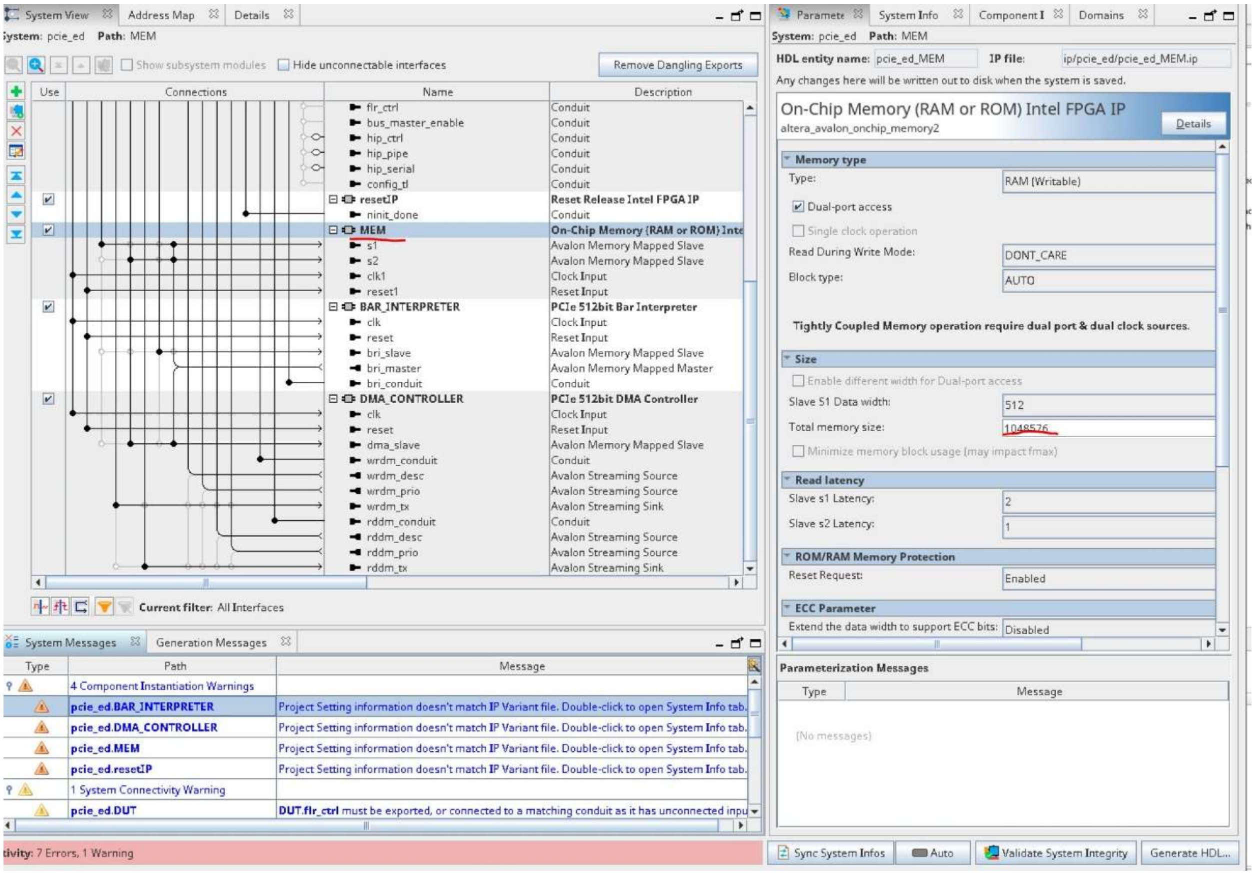
Compilation Flow: During compilation, intermediate Fitter snapshots (planned, placed, routed, and retimed) are not generated.

Task	Time
✓ Compile Design	00:19:07
✓ IP Generation	00:00:06
✓ Analysis & Synthesis	00:02:56
✓ Fitter	00:14:13
Fitter (Implement)	
Plan	00:02:23
Place	00:05:14
Route	00:02:59
Retime	00:00:52
□ *Fast Forward Timing Closure Recommendations*	
✓ Fitter (Finalize)	00:01:55
✓ Timing Analysis (Signoff)	00:00:46
□ *Power Analysis*	
✓ ☑ Assembler (Generate programming files)	00:00:57
EDA Netlist Writer	

2.2 How to get current dual-RAM FPGA module (1Mbyte DMA address)

2.2.1 Extend DMA address (connected to DUT) from 32KB to 1MB

a) Open your project qsys file, double click that "MEM" IP, in right side property window, you need to change "Total memory size" from 32768(32KB) to 1048576(1MB)



System: pcie_ed Path: MEM

Use	Name	Description
	flr_ctrl	Conduit
	bus_master_enable	Conduit
	hip_ctrl	Conduit
	hip_pipe	Conduit
	hip_serial	Conduit
	config_tl	Conduit
☑	⊟ resetIP	Reset Release Intel FPGA IP
	ninit_done	Conduit
☑	⊟ MEM	On-Chip Memory (RAM or ROM) Inte
	s1	Avalon Memory Mapped Slave
	s2	Avalon Memory Mapped Slave
	clk1	Clock Input
	reset1	Reset Input
☑	⊟ BAR_INTERPRETER	PCIe 512bit Bar Interpreter
	clk	Clock Input
	reset	Reset Input
	bri_slave	Avalon Memory Mapped Slave
	bri_master	Avalon Memory Mapped Master
	bri_conduit	Conduit
☑	⊟ DMA_CONTROLLER	PCIe 512bit DMA Controller
	clk	Clock Input
	reset	Reset Input
	dma_slave	Avalon Memory Mapped Slave
	wrdm_conduit	Conduit
	wrdm_desc	Avalon Streaming Source
	wrdm_prio	Avalon Streaming Source
	wrdm_tx	Avalon Streaming Sink
	rddm_conduit	Conduit
	rddm_desc	Avalon Streaming Source
	rddm_prio	Avalon Streaming Source
	rddm_tx	Avalon Streaming Sink

Current filter: All Interfaces

System Messages

Type	Path	Message
⚠	4 Component Instantiation Warnings	
⚠	pcie_ed.BAR_INTERPRETER	Project Setting information doesn't match IP Variant file. Double-click to open System Info tab.
⚠	pcie_ed.DMA_CONTROLLER	Project Setting information doesn't match IP Variant file. Double-click to open System Info tab.
⚠	pcie_ed.MEM	Project Setting information doesn't match IP Variant file. Double-click to open System Info tab.
⚠	pcie_ed.resetIP	Project Setting information doesn't match IP Variant file. Double-click to open System Info tab.
⚠	1 System Connectivity Warning	
⚠	pcie_ed.DUT	DUT.flr_ctrl must be exported, or connected to a matching conduit as it has unconnected inpu

tivity: 7 Errors, 1 Warning

System: pcie_ed Path: MEM

HDL entity name: pcie_ed_MEM IP file: ip/pcie_ed/pcie_ed_MEM.ip

Any changes here will be written out to disk when the system is saved.

On-Chip Memory (RAM or ROM) Intel FPGA IP
altera_avalon_onchip_memory2

Memory type

Type: RAM (Writable)

☑ Dual-port access

☐ Single clock operation

Read During Write Mode: DONT_CARE

Block type: AUTO

Tightly Coupled Memory operation require dual port & dual clock sources.

Size

☐ Enable different width for Dual-port access

Slave S1 Data width: 512

Total memory size: 1048576

☐ Minimize memory block usage (may impact fmax)

Read latency

Slave s1 Latency: 2

Slave s2 Latency: 1

ROM/RAM Memory Protection

Reset Request: Enabled

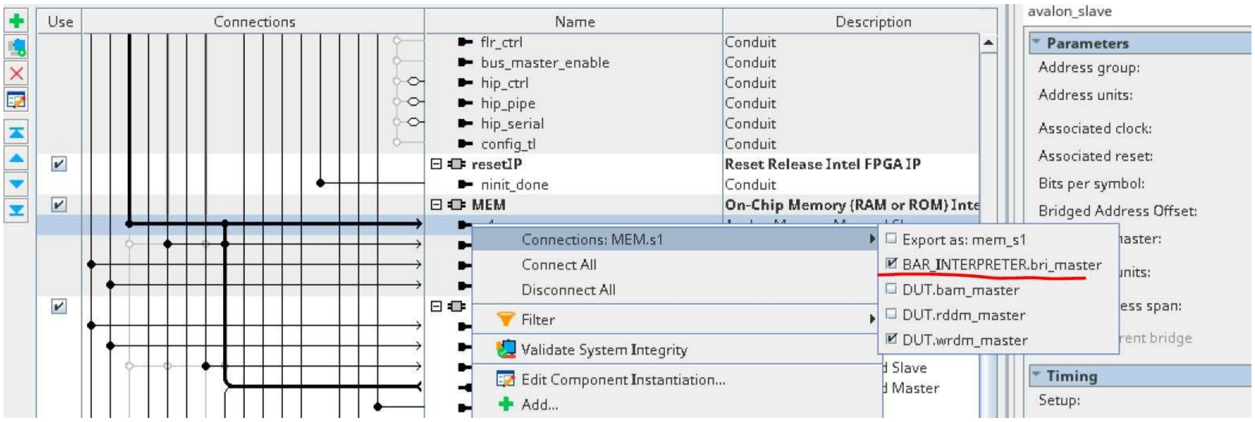
ECC Parameter

Extend the data width to support ECC bits: Disabled

Parameterization Messages

Type	Message
	(No messages)

b) In Qsys system view, on “mem.s1” & “mem.s2” interfaces, dis-connect their connection with BAR_INTERPRETER.bri_master (this is for we don’t need access DMA RAM content through PIO mode, we use small size DMA read to replace PIO polling, & big buffer size easy to get conflict on BAR_INTERPRETER bus address).



Use	Name	Description
	flr_ctrl	Conduit
	bus_master_enable	Conduit
	hip_ctrl	Conduit
	hip_pipe	Conduit
	hip_serial	Conduit
	config_tl	Conduit
☑	⊟ resetIP	Reset Release Intel FPGA IP
	ninit_done	Conduit
☑	⊟ MEM	On-Chip Memory (RAM or ROM) Inte

Connections: MEM.s1

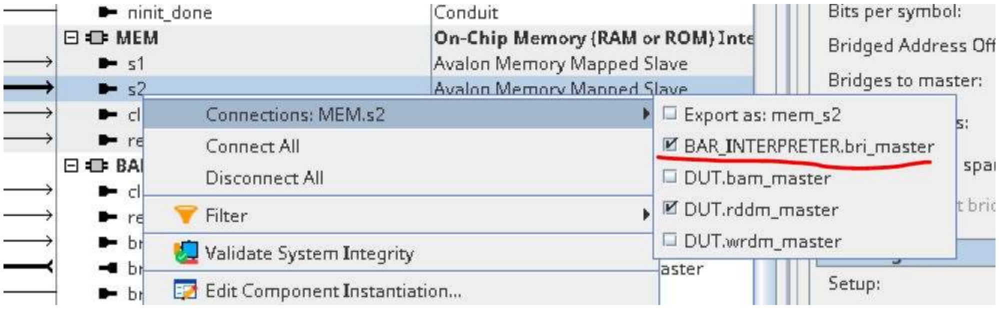
☐ Export as: mem_s1

☑ BAR_INTERPRETER.bri_master

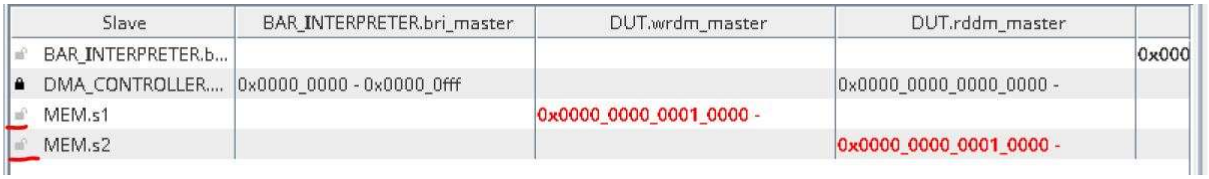
☐ DUT.bam_master

☐ DUT.rddm_master

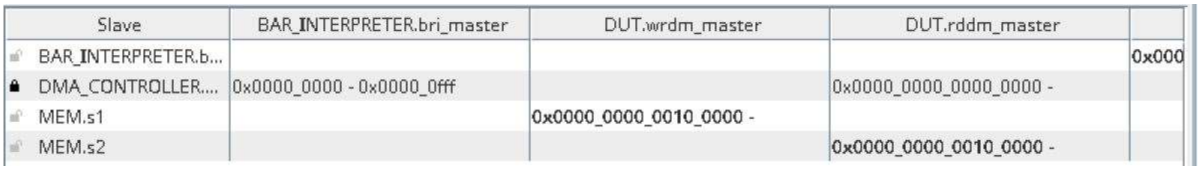
☑ DUT.wrdm_master



c) In "Address Map" window of Qsys, you need to unblock MEM.s1 & MEM.s2 memory map setting at first, then change base address from 0x1_0000 to 0x10_0000 to resolve address map conflict.



Slave	BAR_INTERPRETER.bri_master	DUT.wrdm_master	DUT.rddm_master	
BAR_INTERPRETER.b...				0x000
DMA_CONTROLLER....	0x0000_0000 - 0x0000_0fff		0x0000_0000_0000_0000 -	
MEM.s1		0x0000_0000_0001_0000 -		
MEM.s2			0x0000_0000_0001_0000 -	



Slave	BAR_INTERPRETER.bri_master	DUT.wrdm_master	DUT.rddm_master	
BAR_INTERPRETER.b...				0x000
DMA_CONTROLLER....	0x0000_0000 - 0x0000_0fff		0x0000_0000_0000_0000 -	
MEM.s1		0x0000_0000_0010_0000 -		
MEM.s2			0x0000_0000_0010_0000 -	

2.3 Apply NPU patch to the Quartus project

1) Go to patch directory

```
cd <project home>/patch
```

2) Apply patch

```
./setup.sh
```

3. Hardware platform setup & Run NPU demo

3.1 host PC selection

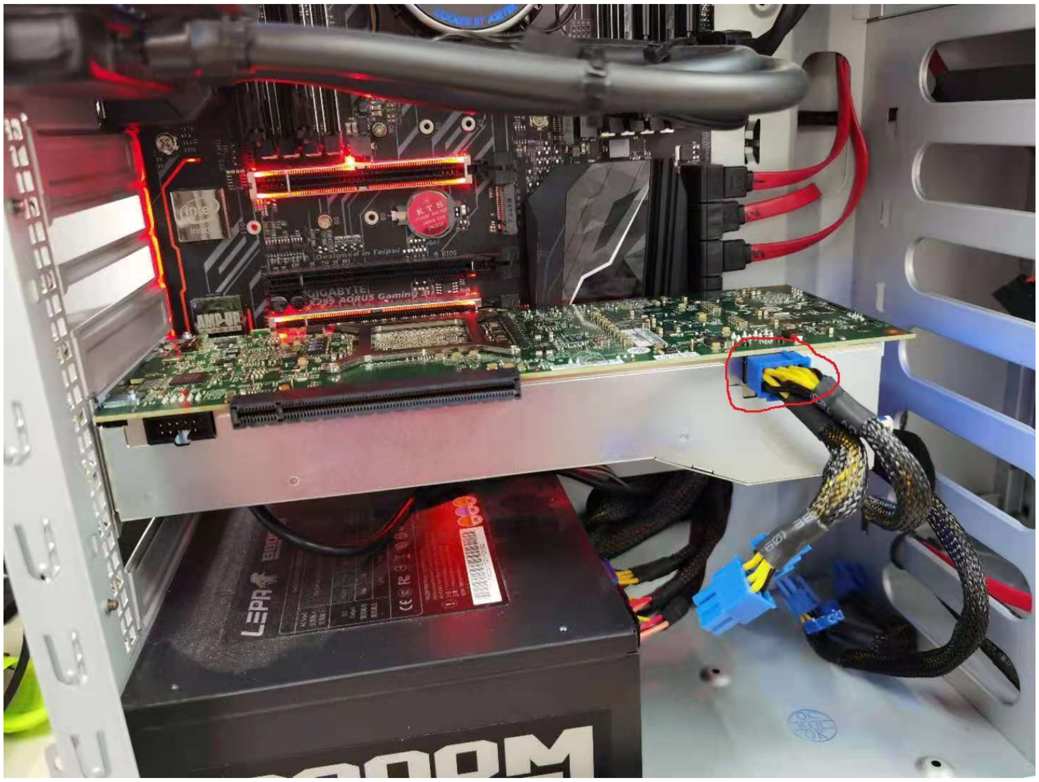
- X86 CPU & can support at least AVX256, better for AVX512 (can use lscpu in Linux to check CPU flags)
- Support Gen3 x16 PCIe slot

3.2 host OS selection

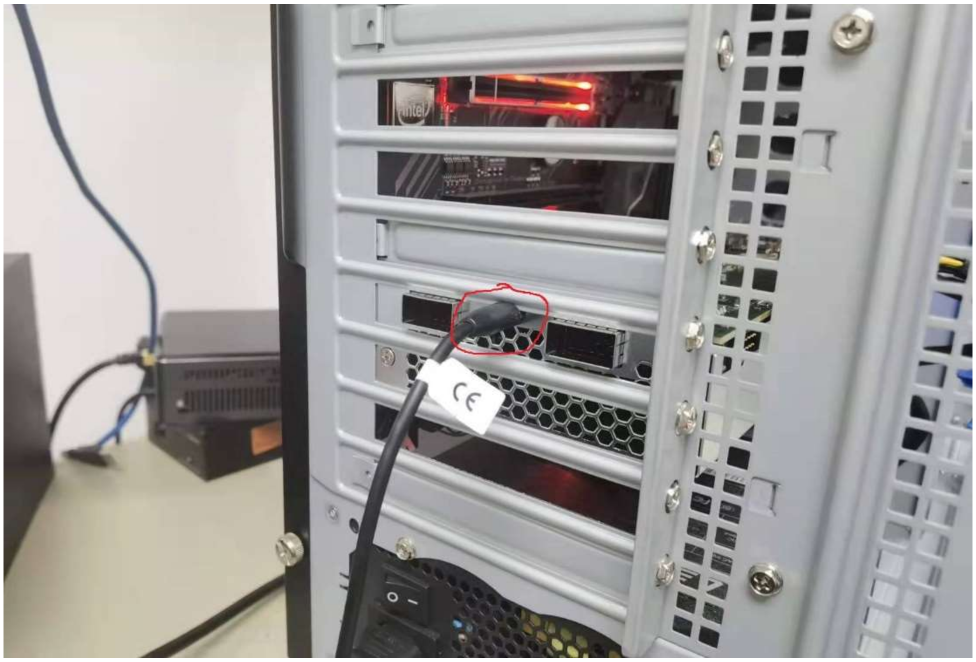
Suggest Ubuntu (> 18.0), or PCIe kernel driver maybe need minor change for kernel mode function support.

3.3 hardware setup

- Connect S10 NX card to one Gen3 x16 PCIe slot on host PC.
- Need to connect ATX power cable to power slot on S10 NX card.



- Connect USB cable between S10 NX card & your control PC (run Quartus programmer & SigTap tool)

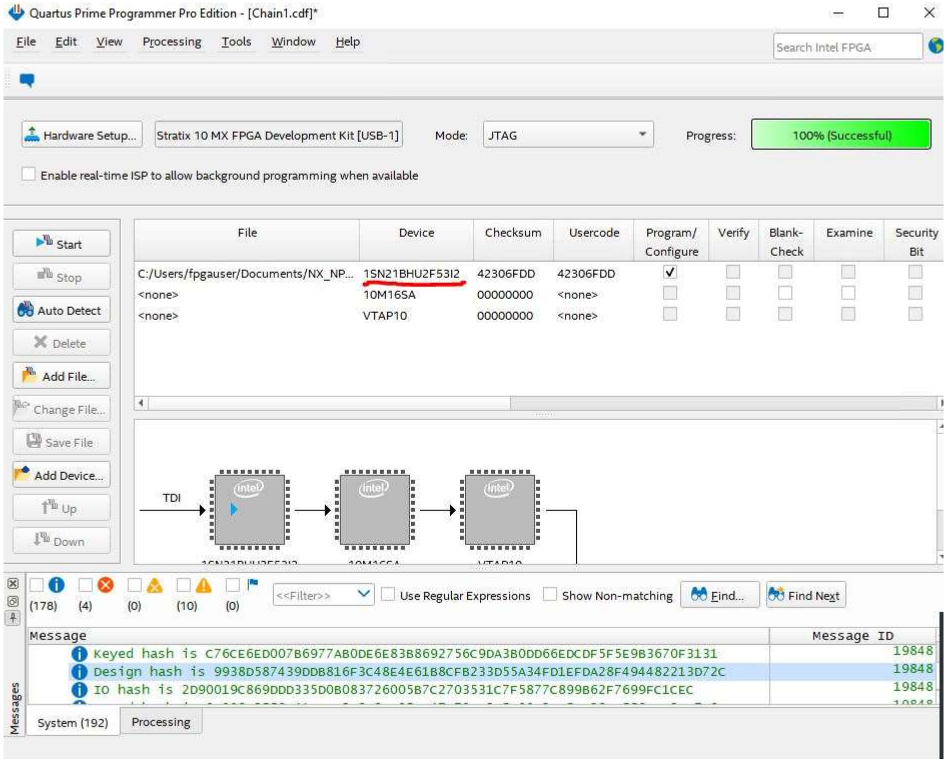


3.4 Control PC tool

please install Quartus program tool (not need full Quartus install, but please use 20.4 version)

3.5 How to run NPU on FPGA board

a) Download pcie_ed.sof generated in <project directory>/DUT_example_design/ to S10 NX (make sure image is using this device name)
   a. Click "Auto Detect"
   b. Click "Add file" to add the pcie_ed.sof bitstream
   c. Remove the redundant entry, Use "Up" button to make the sequence matching the example image shown below.



Stratix 10 MX FPGA Development Kit [USB-1]   Mode: JTAG   Progress: 100% (Successful)

□ Enable real-time ISP to allow background programming when available

File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine	Security Bit
C:/Users/fpgauser/Documents/NX_NP...	1SN21BHU2F53I2	42306FDD	42306FDD	☑	□	□	□	□
<none>	10M16SA	00000000	<none>	□	□	□	□	□
<none>	VTAP10	00000000	<none>	□	□	□	□	□

□ (178) □ (4) □ (0) □ (10) □ (0) <<Filter>> □ Use Regular Expressions □ Show Non-matching Find... Find Next

```
Message                                                                          Message ID
  Keyed hash is C76CE6ED007B6977AB0DE6E83B8692756C9DA3B0DD66EDCDF5F5E9B3670F3131      19848
  Design hash is 9938D587439DDB816F3C48E4E61B8CFB233D55A34FD1EFDA28F494482213D72C     19848
  IO hash is 2D90019C869DDD335D0B083726005B7C2703531C7F5877C899B62F7699FC1CEC         19848
```

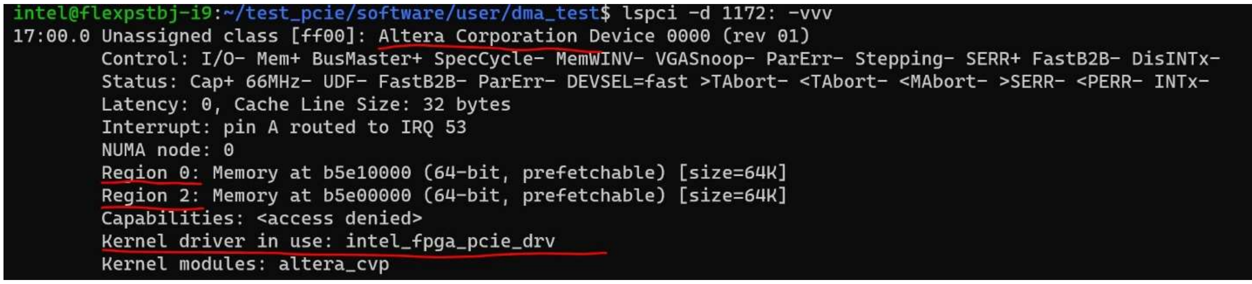
b) You need to reboot host PC (Note: it is "reboot", can't be "power down")

c) Copy <project home>/DUT_example_design/software folder to host PC

d) Go to "kernel/linux". For the first time, compile the source files by typing "make".

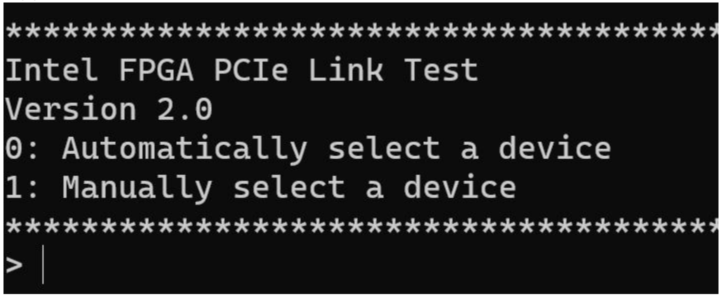
e) Load the driver by typing "sudo ./load"

f) Type "lspci -d 1172: -vvv" to make sure you can find that PCIe device, BAR2 enabled & driver load

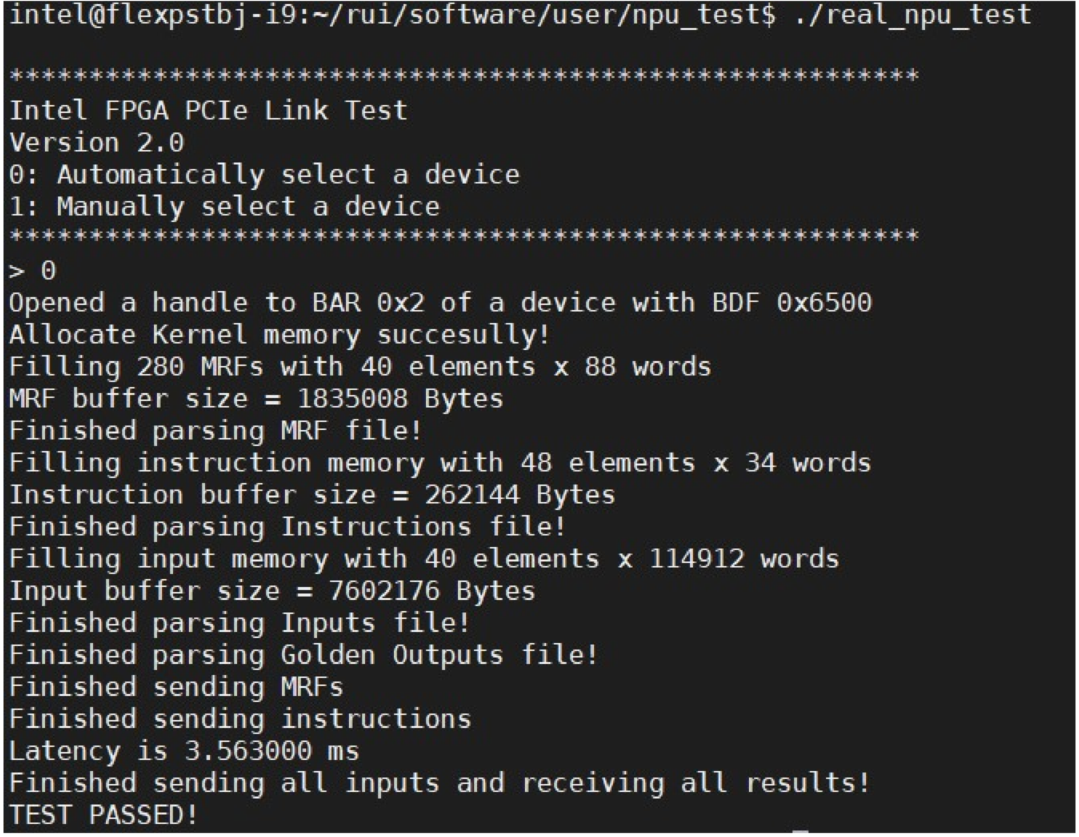


```
intel@flexpstbj-i9:~/test_pcie/software/user/dma_test$ lspci -d 1172: -vvv
17:00.0 Unassigned class [ff00]: Altera Corporation Device 0000 (rev 01)
        Control: I/O- Mem+ BusMaster+ SpecCycle- MemWINV- VGASnoop- ParErr- Stepping- SERR+ FastB2B- DisINTx-
        Status: Cap+ 66MHz- UDF- FastB2B- ParErr- DEVSEL=fast >TAbort- <TAbort- <MAbort- >SERR- <PERR- INTx-
        Latency: 0, Cache Line Size: 32 bytes
        Interrupt: pin A routed to IRQ 53
        NUMA node: 0
        Region 0: Memory at b5e10000 (64-bit, prefetchable) [size=64K]
        Region 2: Memory at b5e00000 (64-bit, prefetchable) [size=64K]
        Capabilities: <access denied>
        Kernel driver in use: intel_fpga_pcie_drv
        Kernel modules: altera_cvp
```

g) Go to “user/npu_test”. For the first time, compile the source files by typing “make”.
h) Copy benchmark data. E.g., to run mlp_batch4104, type “cp ./mlp_batch4104/* ./”
i) Type “./real_npu_test” to run the test
j) Type 0 for automatic PCIe device find



```
*************************************
Intel FPGA PCIe Link Test
Version 2.0
0: Automatically select a device
1: Manually select a device
*************************************
> 
```

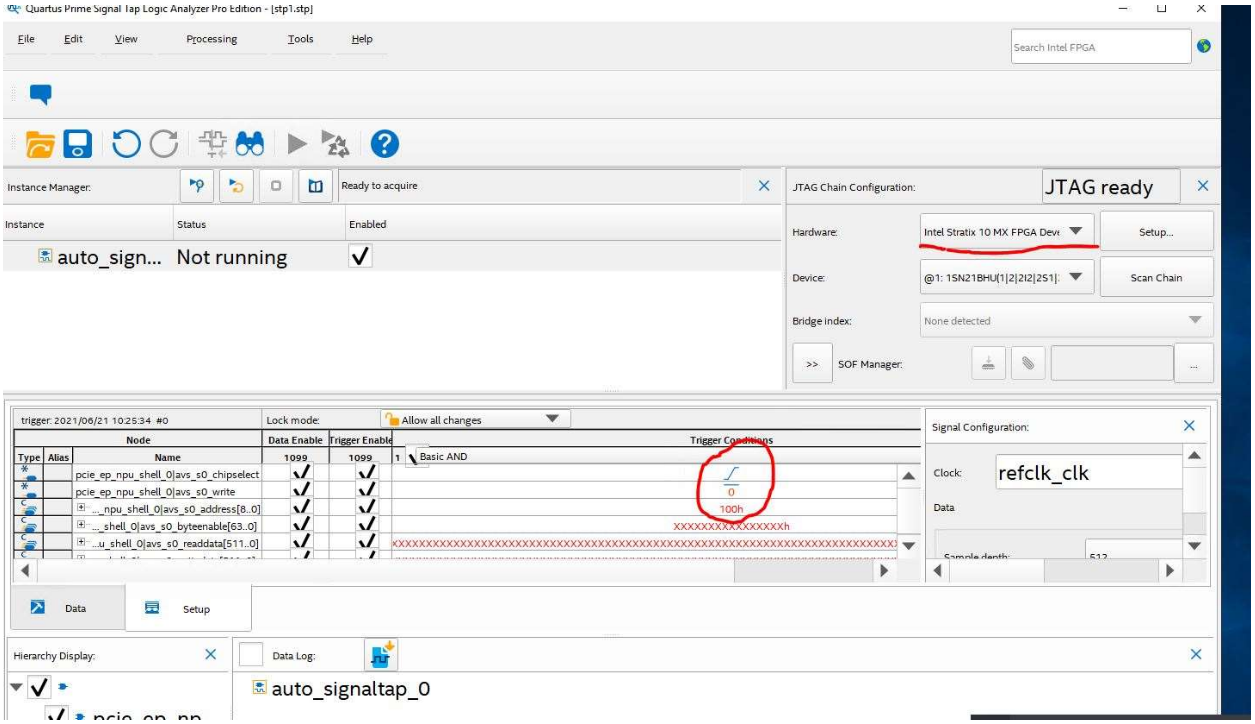


```
intel@flexpstbj-i9:~/rui/software/user/npu_test$ ./real_npu_test

*********************************************************
Intel FPGA PCIe Link Test
Version 2.0
0: Automatically select a device
1: Manually select a device
*********************************************************
> 0
Opened a handle to BAR 0x2 of a device with BDF 0x6500
Allocate Kernel memory succesully!
Filling 280 MRFs with 40 elements x 88 words
MRF buffer size = 1835008 Bytes
Finished parsing MRF file!
Filling instruction memory with 48 elements x 34 words
Instruction buffer size = 262144 Bytes
Finished parsing Instructions file!
Filling input memory with 40 elements x 114912 words
Input buffer size = 7602176 Bytes
Finished parsing Inputs file!
Finished parsing Golden Outputs file!
Finished sending MRFs
Finished sending instructions
Latency is 3.563000 ms
Finished sending all inputs and receiving all results!
TEST PASSED!
```

3.6 [Optional] Sigtap debugging on FPGA

- new one *stp file with debug signals you want, & include it into FPGA image build
- Copy that *.stp to control PC
- Use SigTap tool to open this *.stp
- Select correct hardware & device in tool
- Set trigger condition in that “setup” tab.



Instance Manager: Ready to acquire

Instance	Status	Enabled
auto_sign...	Not running	✓

JTAG Chain Configuration: JTAG ready

Hardware: Intel Stratix 10 MX FPGA Deve

Device: @1: 1SN21BHU(1\|2\|2I2\|2S1\|:

Bridge index: None detected

SOF Manager:

trigger: 2021/06/21 10:25:34 #0

Lock mode: Allow all changes

Type	Alias	Name	Data Enable	Trigger Enable	Trigger Conditions
			1099	1099	1 Basic AND
		pcie_ep_npu_shell_0|avs_s0_chipselect	✓	✓	∕
		pcie_ep_npu_shell_0|avs_s0_write	✓	✓	0
		..._npu_shell_0|avs_s0_address[8..0]	✓	✓	100h
		..._shell_0|avs_s0_byteenable[63..0]	✓	✓	XXXXXXXXXXXXXXXXh
		...u_shell_0|avs_s0_readdata[511..0]	✓	✓	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Signal Configuration:

Clock: refclk_clk

Data

Sample depth: 512

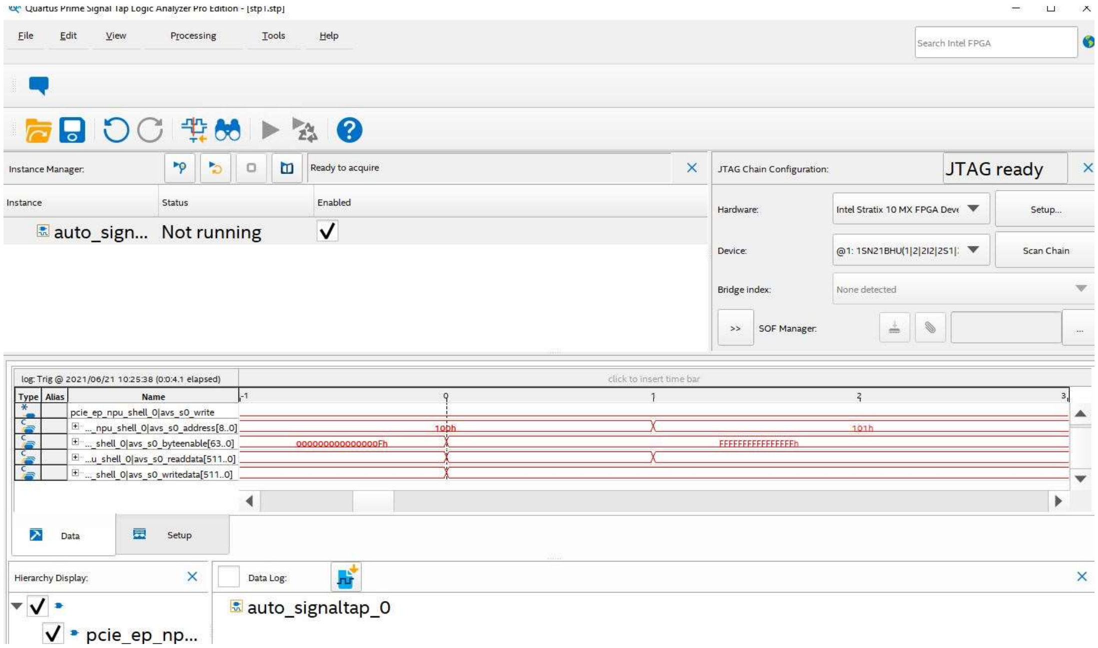
Hierarchy Display:

✓ pcie_ep_np...

Data Log:

auto_signaltap_0

- Then kick off "capture" button, it will stop & show waveform when trigger condition hit.



Instance Manager: Ready to acquire

Instance	Status	Enabled
auto_sign...	Not running	✓

JTAG Chain Configuration: JTAG ready

Hardware: Intel Stratix 10 MX FPGA Deve

Device: @1: 1SN21BHU(1\|2\|2I2\|2S1\|:

Bridge index: None detected

SOF Manager:

log: Trig @ 2021/06/21 10:25:38 (0:0:4.1 elapsed)

Type	Alias	Name
		pcie_ep_npu_shell_0|avs_s0_write
		..._npu_shell_0|avs_s0_address[8..0]
		..._shell_0|avs_s0_byteenable[63..0]
		...u_shell_0|avs_s0_readdata[511..0]
		..._shell_0|avs_s0_writedata[511..0]

Hierarchy Display:

✓ pcie_ep_np...

Data Log:

auto_signaltap_0