

## 开发环境

### 设计工具

1. [Vivado 2020.1](#)

2. [Petalinux 2020.1](#)

3. [Vitis2020.1](#)

4. VMware + Ubuntu18.04 LTS

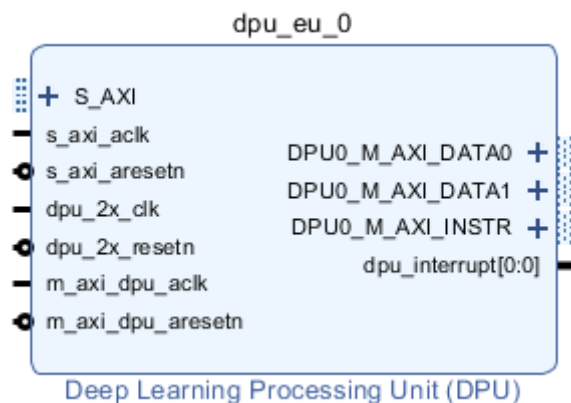
5. Windows10

**环境介绍：** Vivado和Vitis安装在Windows下， Petalinux安装在Ubuntu系统中。 Vivado完成硬件设计， Petalinux完成ZedBoard\_DPU的操作系统设计， Vitis完成应用软件的设计。

**开发板介绍：** 板载芯片是XC7Z020CLG484 -1， DSP资源220， Block RAM资源4.9Mb

## Vivado搭建硬件环境

### 1. 获取DPU IP



### 2. 完成Block Design

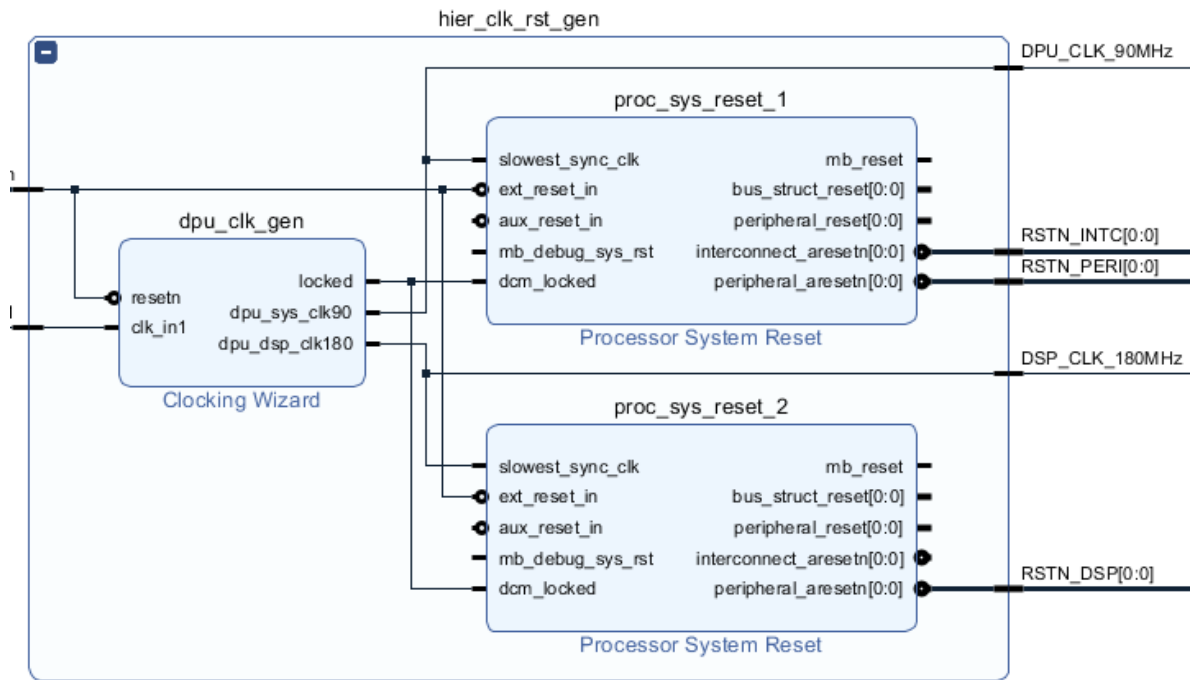
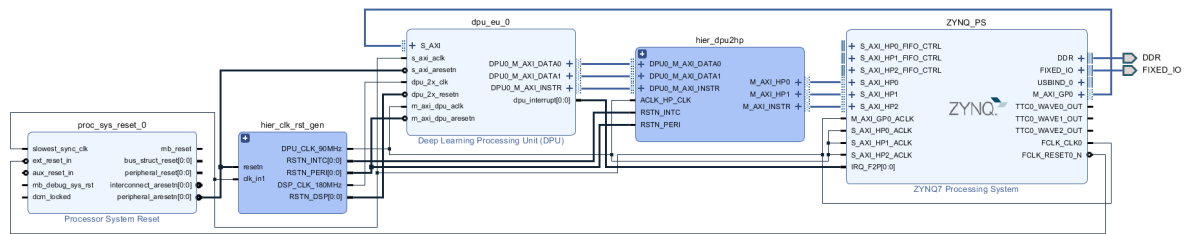
1) 依次添加ZYNQ7 Processing System, DPU IP, Clock Wizard, AXI Interconnect, Processor System Reset模块;

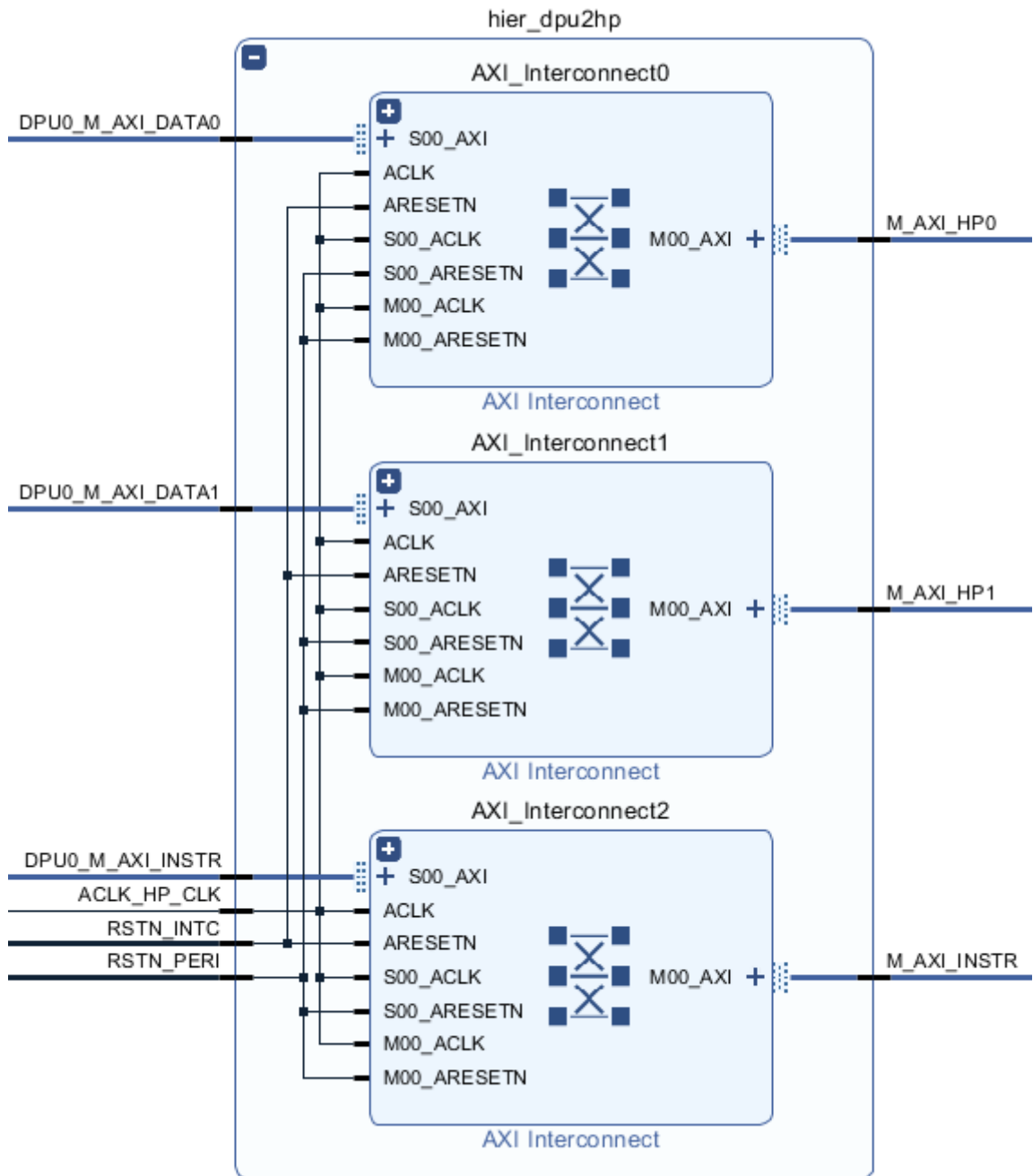
2) 选择PS中的PS-PL Configuration, 展开HP Slave AXI Interface, 勾选HP0到HP2。这些IO在后面会与DPU的DATA0, DATA1, INSTR接口相连接;

3) 配置DPU IP, 选择Arch界面, DPU的大小设置为B1152, RAM Usage的大小为High, 关闭DepthWiseConv, ReLU Type为ReLU+LeakyReLU+ReLU6, 与ZYNQ7000使用时, DPU不支持SoftMax的计算, 这个得放在ARM CPU里面算(问题不大)。选择Advanced界面, S-AXI Clock Mode这里可选Common with M-AXI Clock也可以选Independent, 这里我们选Independent。配置完按OK; **Tips:这么配置后DSP Slice Count = 194, Block-RAM Count = 139.5。**

4) 配置Clock Wizard输出两路时钟, 频率分别为90MHz和180MHz, 90MHz的时钟是DPU的系统时钟, 180MHz的时钟是DDR-DSP的时钟将这两路时钟依次与DPU相连接。DPU上S\_AXI的时钟由ZYNQ模块的FCLK\_CLK0提供。**时钟与复位信号的搭配关系是: 相同时间域的复位信号连接到一起。**

5) 将DPU0\_M\_AXI\_DATA0-1, DPU0\_M\_AXI\_INSTR与AXI Interconnect连接到一起(可能是起到缓冲的作用, 应该是可以直接与HP0-2相连)。





### 3.生成Bitstream，导出XSA文件

1) 通过Address Editor分配DPU的内存，貌似自动分配就行了

Diagram x Address Editor x						
<input checked="" type="checkbox"/> Assigned (4) <input checked="" type="checkbox"/> Unassigned (0) <input checked="" type="checkbox"/> Excluded (0) <input type="button" value="Hide All"/>						
Name	Interface	Slave Segment	Master Base Address	Range	Master High Address	
Network 0						
ZYNQ_PS						
ZYNQ_PS/Data (32 address bits : 0x40000000 [ 1G ])						
/dpu_eu_0	S_AXI	reg0	0x4000_0000	16M	0x40FF_FFFF	
Network 1						
/dpu_eu_0						
/dpu_eu_0/DPU0_M_AXI_DATA0 (40 address bits : 1T)						
ZYNQ_PS	S_AXI_HP0	HP0_DDR_LOWOCM	0x00_0000_0000	512M	0x00_1FFF_FFFF	
/dpu_eu_0/DPU0_M_AXI_DATA1 (40 address bits : 1T)						
ZYNQ_PS	S_AXI_HP1	HP1_DDR_LOWOCM	0x00_0000_0000	512M	0x00_1FFF_FFFF	
/dpu_eu_0/DPU0_M_AXI_INSTR (40 address bits : 1T)						
ZYNQ_PS	S_AXI_HP2	HP2_DDR_LOWOCM	0x00_0000_0000	512M	0x00_1FFF_FFFF	

2) 点击Validate Block Design。

3) 点击Generate Bitstream

4) 点击`Export hardware`, 注意勾选`include bitstream`

**到这里Vivado上的工作就做完了**