1. Create new project -> create new block design

2. add these blocks:

a. zynq7 processing system

i. run block automation and do the DDR and FIXED\_IO connections

ii. double click PS -- PS-PL configuration -> GP Master AXI Interface -> M AXI GP0 Interface and make sure its checked

b. reset for ps

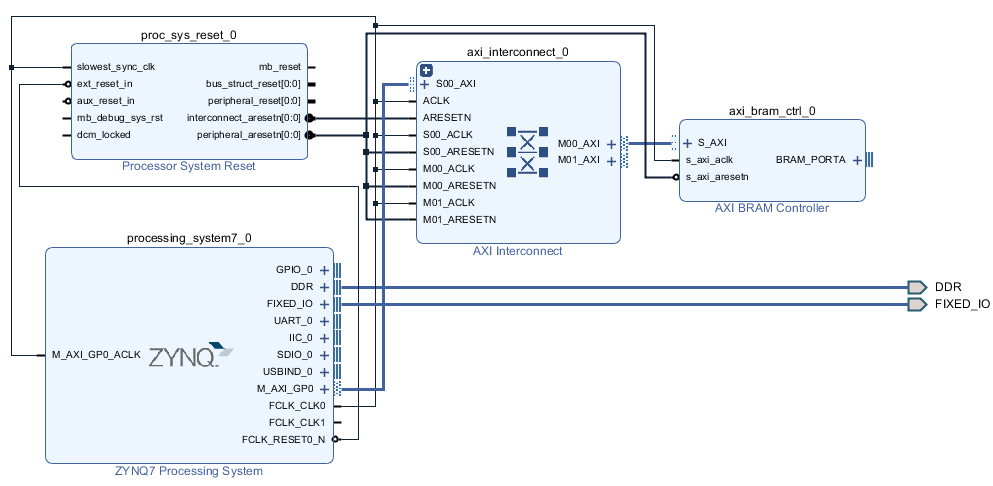
c. axi interconnect

d. axi bram ctrl

i. double click it and changed number of BRAM interfaces to 1 and change to AXI4LITE

3. After all this is added, run connection automation, press regenerate layout to reorder components (\*makes it look neater), press validate design and make sure no errors.

4. You should have this after doing everything above, if not, connect all the wires manually and validate design:



5. Tools -> create and package new IP -> new AXI peripheral -> create a name you will remember -> edit IP on the last step and press finish

6. in the package IP vivado -> add all the design sources

7. add this in, the \_v1\_0.v file, if you are unsure about this step, refer to the github where its uploaded:

a. At line 7:

parameter WIDTH=32, SIZE=512, //WIDTH is bits per word(shouldn't be changed), SIZE is # of WORDS

parameter NUM\_COL = 4,

parameter COL\_WIDTH = 8,

localparam integer LOGSIZE=$clog2(SIZE),

b. at line 49 add: wire ps\_reset;

c. add this inside the instantiation .ps\_reset(ps\_reset)

d. Line 22:

input wire [LOGSIZE:0] bram\_addr,

input wire [WIDTH-1:0] bram\_wrdata,

output wire [WIDTH-1:0] bram\_rddata,

input wire [NUM\_COL-1:0] bram\_we,

e. Line 81:

pipelined\_processor #(.WIDTH(WIDTH), .SIZE(SIZE), .NUM\_COL(NUM\_COL), .COL\_WIDTH(COL\_WIDTH)) dut(.clk(s00\_axi\_aclk), .reset(ps\_reset),

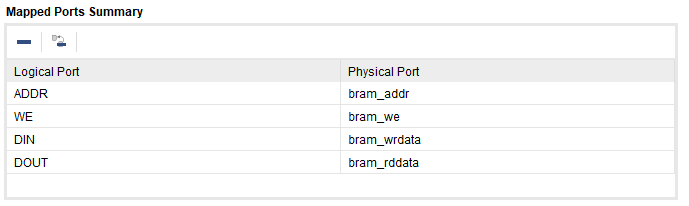
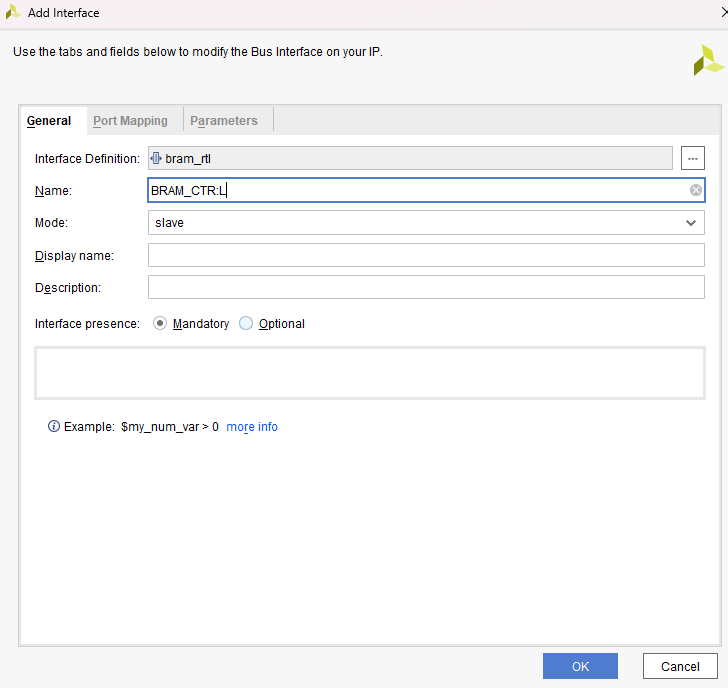
.bram\_din(bram\_wrdata), .shared\_bram\_addr(bram\_addr),

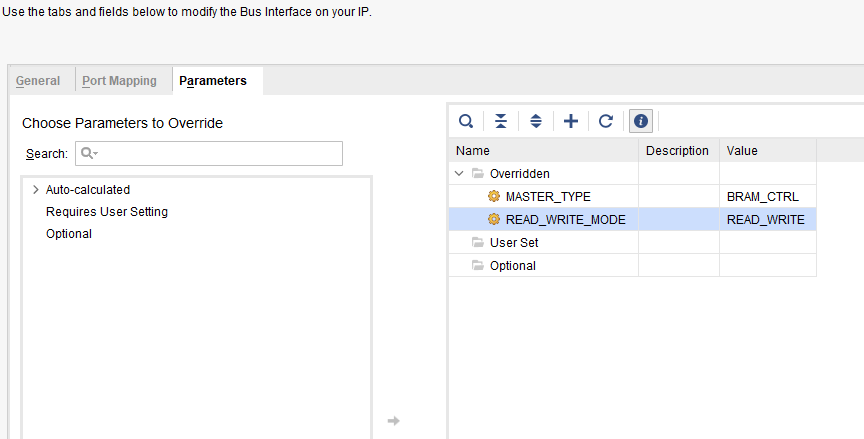
.bram\_wr\_en(bram\_we), .bram\_dout(bram\_rddata));

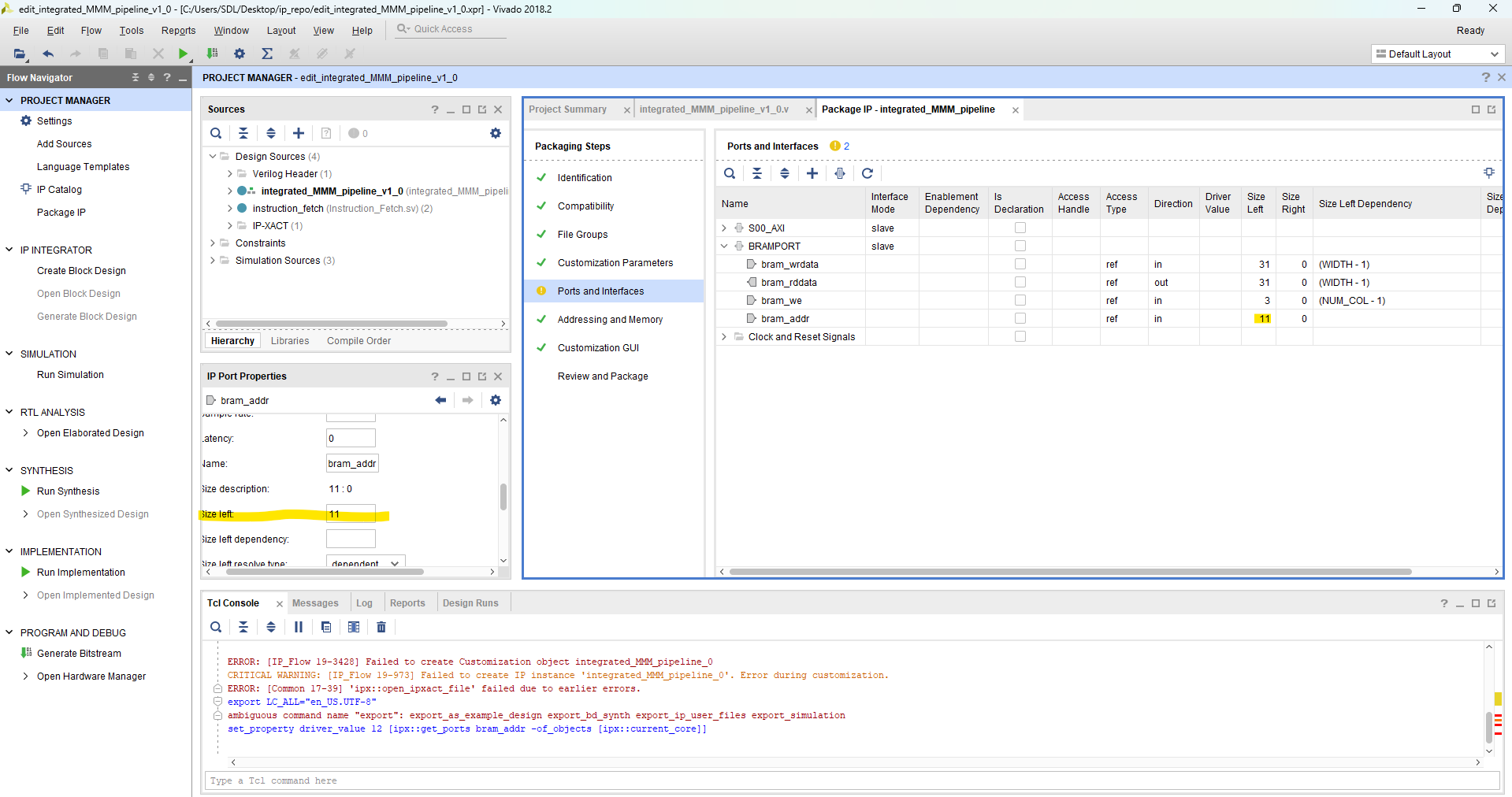
f. inside the v1\_0\_S00\_AXI.v file:

output ps\_reset, -> add at line 18

assign ps\_reset = slv\_reg0; -> add this to line 117

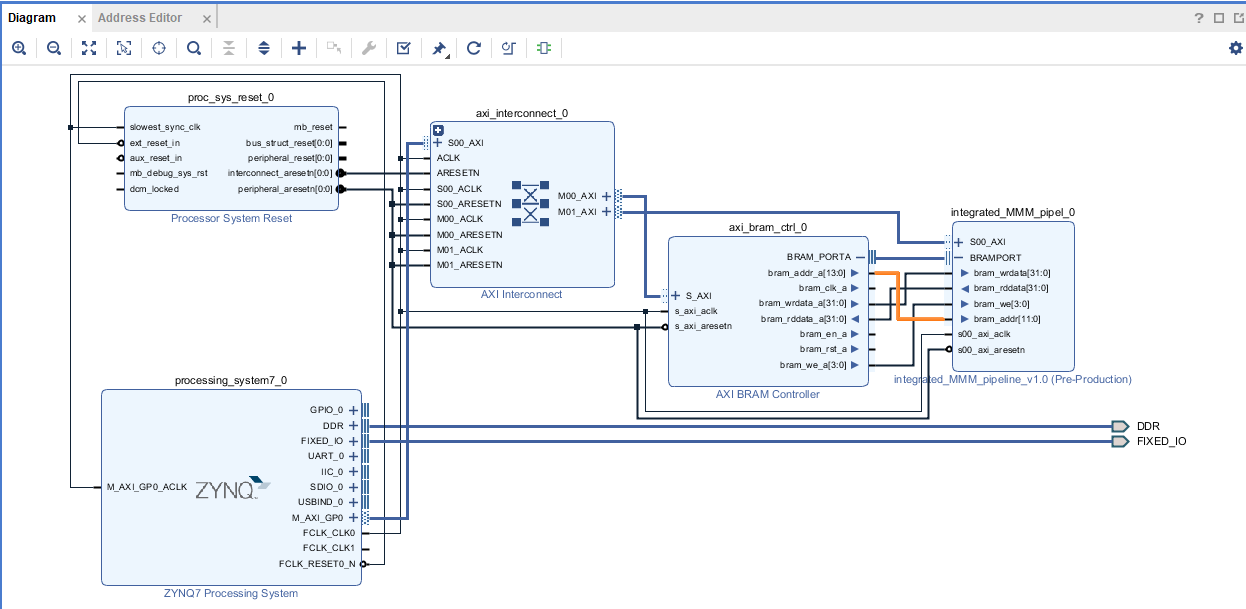






Edit bram\_addr in ip port properties for size left and make it 11.

8. merge changes in the packaging steps tabs and package IP.

9. By now, Block diagram should look like this:  


Validate design to make sure there is no errors.

Lastly, create HDL wrapper

**SDK STUFF**

File -> export hardware -> launch sdk

In the SDK:

Change linker in the application project under the source (src) folder, open lscript.ld (LINKER SCRIPT).

Under, section to memory region mapping, Change all region memory fields from ps7\_DDR\_0 to PS7\_RAM\_0

In case there are weird errors such as an assembly loop or when trying to running the program (AXI error):

1. Create new Application project called “FSBL”, choose zynq FSBL project as the template. Hit Finish.
2. Select the FSBL in project explorer on the left side, in top menu toolbar -> Xilinx-> create boot image-> on the bottom (boot image partitions) -> click add -> browse file path -> navigate to original application project (./Project.sdk/application\_project\_name/debug/application\_name.elf) -> open elf and hit OK
3. Create image
4. With FSBL still highlighted, program and run. Then run application project and proceed as normal.

**For UART and SDK testing**

RUN THE PYTHON AND THEN THE C FILE IN THE SDK