ECE385
Digital System Laboratory

IP Modules

AMD Xilinx offer a variety of Intellectual Property (IP) modules to facilitate the design and development of complex systems using their FPGA and ACAP platforms. These modules are pre-designed FPGA components (such as interfaces, memory controllers, computation accelerators, etc.) which can be licensed from AMD Xilinx or third parties and instantiated into an FPGA design. Typically, these can be configured graphically through a GUI, or through code via SystemVerilog parameters. For Experiment 5, you will configure an on-chip memory IP called ***Block Memory Generator*** to use as your main memory.

To start, open your existing Experiment 5 project that you have created. From ***Flow Navigator - IP INTEGRATOR***, select ***IP Catalog***. Search for ***Block Memory Generator*** and select it from the search results following image 1.

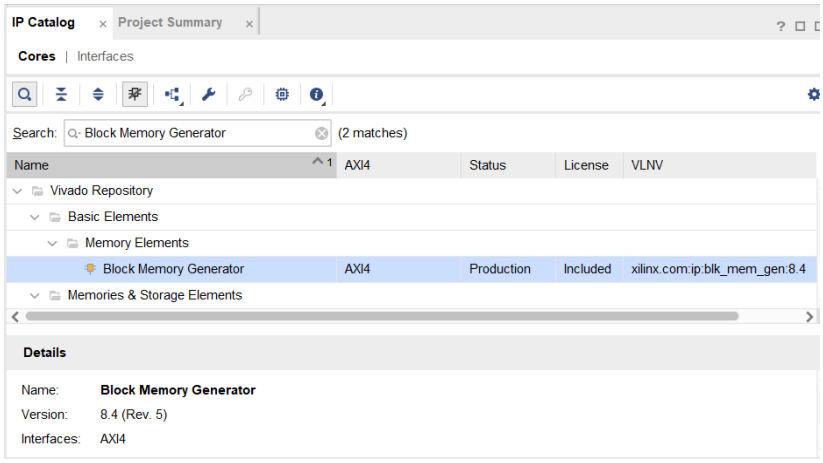

Image 1

You will then be presented with the following GUI tool. Populate the following according to Image 2-4, leave the other settings as default and click ***OK***. Click ***Generate***.

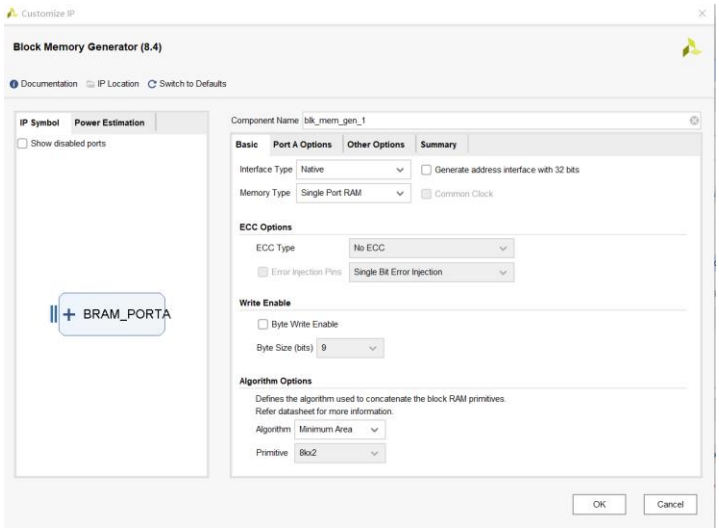

Image 2

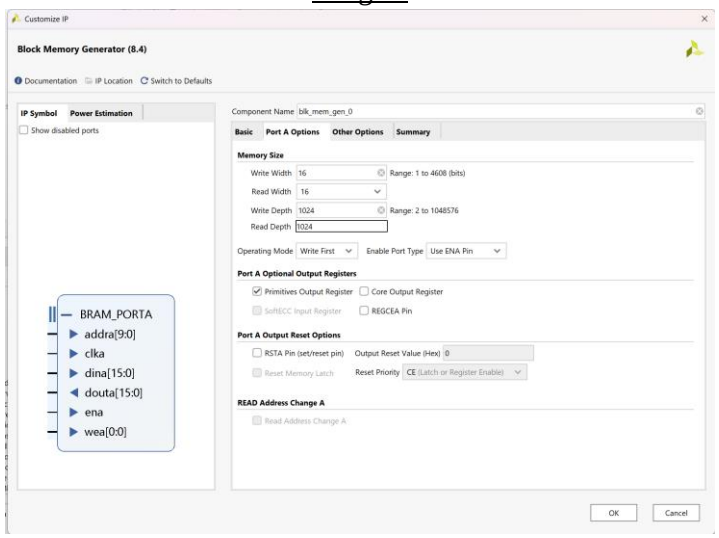

Image 3

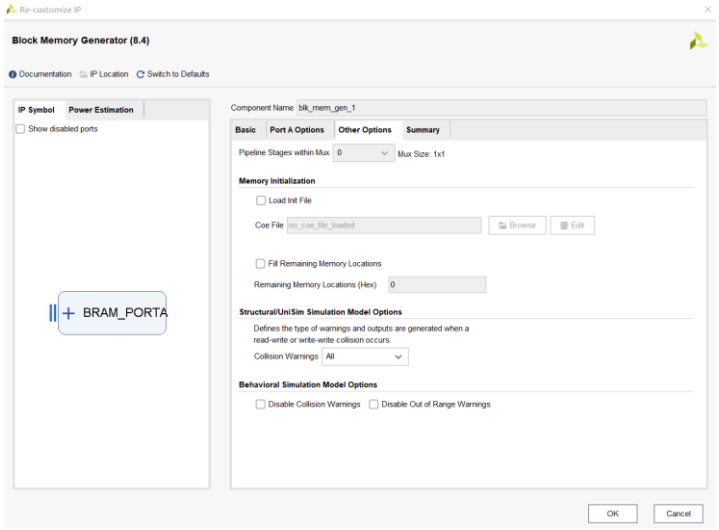

Image 4

Note that the ***ena*** bit is the enable bit for both read and write to the memory block whereas the ***wea*** bit is used for write enable only. But because you configured the memory ***write first***, it will perform writing instead of reading if both bits are turned on. Please be careful when you set up the state machine for the lab.

Also notice that we are only provisioning our BRAM with 1024 (16-bit) words. This does not occupy the full memory space of our SLC-3 CPU, so the memory address is zero padded with 6 leading bits. Still, this is more than enough to contain and run our test programs (note that none of the test programs require dynamic memory, although the Sort test has a pre-allocated data buffer containing the data to be sorted).

Although it is possible to populate the memory contents from the GUI, we provide a module called instantiateram.sv which initializes the on-chip memory from the FPGA logic on reset. This is so the memory contents can be restored during a reset, rather than requiring a full FPGA reprogramming to restore.