Assignment No. V, Monday
Computer Organisation - CS220

---

1. This assignment aims to create a DRAM architecture as discussed in the class. The objective is to create 64 KB DRAM architecture using one 64 bit channel, 16 Kb ×4 DRAM chips, 16 rows, 16 banks, and burst length 8. The address format for the addressing is given as:

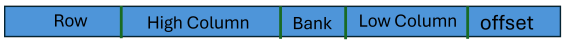

Figure 1: Address format for DRAM addressing

- Decide the width of each domain in the address format
- Write the code of a single BANK using multiplexers, decoders, and flipflops. Deploy 2-level decoding as discussed in the class
- Construct the entire DRAM architecture using structural style with the bank module as your base. The DRAM will take one address as input and will produce 64-bit data as output in each clock cycle. Note that as the burst length is 8, eight consecutive 64 bits of output in eight consecutive cycles should be produced.
- Note that as we can not implement bit line and word line on the FPGA directly, we are developing each bank like an SRAM module of the last assignment.

You need to verify the design on the simulation using the Vivado tool (using post-implementation simulation)

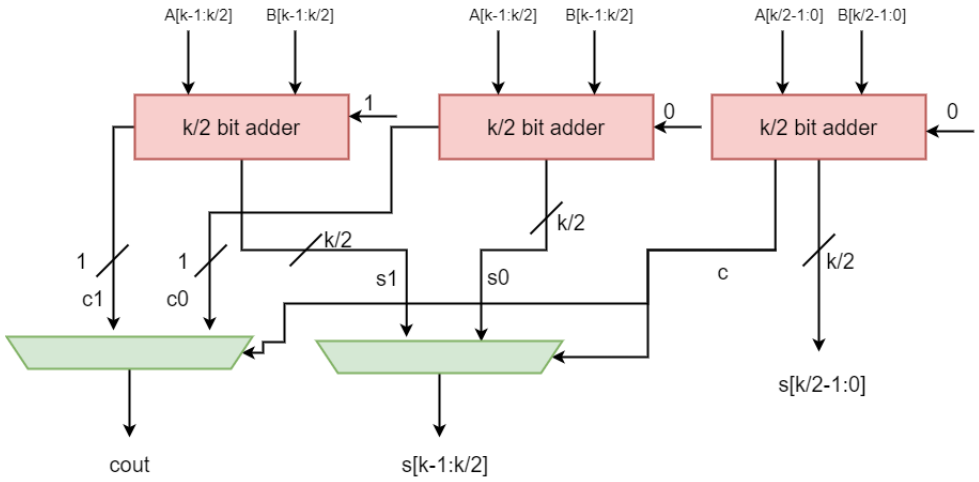

Figure 2: Architecture of "Mjolnir" Adder Circuit

2. Suppose we have a new adder circuit named "Mjolnir" as shown in Figure 2. Implement k-bit "Mjolnir" adder where $k = 16,\ 32,\ 64,\ 128$ using $k/2$-bit fast adder of assignment 2. Use Verilog parameters for k. For $k = 128$, compare the timing and area overhead of this adder with the normal ripple carry adder and verify your result on the FPGA by developing the corresponding IP. Note that you need to create the IP for both the normal ripple carry adder and your design so that you can compare the area and timing performance.