Comp-arch Projects:
Choose structural or behavioral according to the ease of programming.

1. Write an 8085 assembly to compute a 3x3 matrix multiplication. Design an 8085 processor in Verilog to fetch, decode and execute this program. Assume that the 8085 assembly program for matrix multiplication is stored in memory. Your decode unit will have to decode only these set of instructions needed for the multiplication calculation. You do not have design a decode unit for all 8085 instructions.
2. Write a MIPS assembly to compute a 3x3 matrix multiplication (You may use the MIPS assembler to convert from C to assembly). Design a MIPS non-pipelined processor in Verilog to run this program. Your decode unit will have to decode only these set of instructions needed for the multiplication calculation. You do not have design a decode unit for all MIPS instructions.
3. Write a MIPS assembly to compute a 3x3 matrix multiplication (You may use the MIPS assembler to convert from C to assembly). Design a MIPS pipelined processor in Verilog to run this program. Implement stalls/NOPs when there are dependencies.
4. In Verilog - design a direct-mapped cache to hold 256 blocks, 16 words per block (word = 32 bits). Assume a 32 bit address. Implement an LRU scheme for replacement. Generate random addresses to read/write from/to and demonstrate the miss rate with this LRU scheme. How would the miss rate change with varying word size per block.