Review Problem 53

- If we only had 8x2 memories available, how could we make an 8x6 RAM?
Verilog Memories

module memory16x6 (data_out, data_in, addr, we, clk);
    output logic [5:0] data_out;
    input logic [5:0] data_in;
    input logic [3:0] addr;
    input logic we, clk;

    logic [5:0] mem [15:0];

always_ff @(posedge clk) begin
    data_out <= mem[addr];
    if (we)
        mem[addr] <= data_in;
end

endmodule
Field Programmable Gate Arrays (FPGAs)

- Readings: B.6-B.6.5

Logic cells imbedded in a general routing structure

Logic cells usually contain:

- 6-input Boolean function calculator
- Flip-flop (1-bit memory)

All features electronically (re)programmable
Using an FPGA

Verilog → FPGA CAD Tools → Bitstream → Simulation
FPGA Programming

Bitstream

⊕ = 1 memory cell (stores 1 bit of info)
FPGA Combinational Logic

- How can we use Muxes and Programming bits to compute combinational binary function $F(A, B, C)$?

- Creates a "LUT" or lookup table.

DE1-SOC: 6 LUT
FPGA Sequential Logic

- How do we put DFF’s onto LUT outputs only when we need them?

- Creates a “LE” or logic block
FPGA Local Routing

- How do we combine LE’s to build larger functions?

- This is an Altera “LAB”.

In1

In2

In3

In4

Out1

Out2