ELEC 4200 Lab#6
Parameterized VHDL Modeling & Synthesis of Register File with Testbench

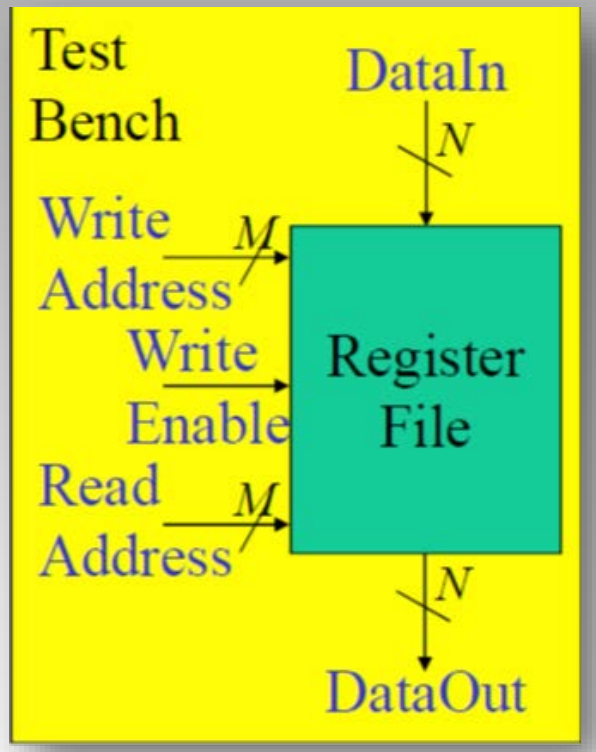
AUBURN
UNIVERSITY

SAMUEL GINN
COLLEGE OF ENGINEERING

- References you may need:
  - **Overview of FPGA Editor.pdf**
  - **Adding Probes in FPGA Editor.pdf**

Specifications

- Write a parameterized VHDL model for a register file and a test-bench for simulation
- A register file is essentially a small RAM
- Specification for the parameterized register file
  - 2^M registers of N-bits each
  - M write address bits & an active high Write Enable
    - » On the rising edge of Write Enable, contents of register selected by write address = Data inputs
  - M read address bits
    - » Data outputs = contents of register at selected read address



Pre-lab Assignment

- Write a VHDL model for the parameterized register file on the previous slide
- Write a parameterized test-bench to instantiate your register file component and apply to it the memory test algorithm on the following slide.
- Determine the FPGA pin numbers for inputs/outputs
  - Assuming M=2 and N=4
    - » Address and Data Inputs → Switches
    - » Data Outputs → LEDs
    - » Write Enable → Push Button**

    ** Since data is to be written on the rising edge of *Write Enable*, that signal is effectively a “clock”. Therefore, *Write Enable* should be supplied via a clock-capable FPGA pin.
      - Pin P17 is clock-capable and is connected to push button BTNL on the Nexys4 DDR board.

Memory Testing Algorithm

1. ⇳ (write address as data)
2. ⇑ (read address, write inverted address as data)
3. ⇓ (read inverted address, write address as data)
4. ⇑(read address)

Notes:

This is a standard notation for memory test algorithms where each step above consists of one or more read and/or write operations to the complete memory address space to detect faults (or in your case design errors) in the memory.

⇑ = indicates ascending addresses (from 0 to N-1)

⇓ = indicates descending address (from N-1 to 0)

⇳ = indicates addressing in either direction

Lab Exercise

- Simulate your VHDL test bench and model and verify your design using Active-HDL.
  - Use two sets of parameters (M=3, N=3 and M=2, N=4) to verify the design via simulation.
- Synthesize and implement your register file *(but NOT the test bench)* for the Artix-7 FPGA (on the Nexys4 board).
  - Use M=2, N=4 for synthesis
  - Record the Slices, LUTs, and FF/latches from the implementation report.
  - Note how the memory cells were implemented, i.e. were they implemented with latches, flip-flops, LUTs, or block RAMs?
- Demonstrate your working circuit to the GTA

Report Guidelines

- Be sure to include all sections required by the lab manual guidelines. In addition be sure your report includes the following:
  - Verified VHDL models
  - Annotated screenshots of your Aldec Active-HDL simulation results for different values of M & N
  - Implementation results (LUTs, FFs, slices, etc), including components used to implement the memory cells.
  - Answers to the following questions…

1. How difficult was it to parameterize your model? Do you feel that the time saved in debugging made up for it?

2. Do you think this approach would apply to a larger project?

3. Do you think it is possible to over-parameterize a model?