Lab 3a

CSE141L

Announcements

- Lab 2b “due” today
  - Hardcopy Report
  - Lab Interview

- Lab 3a posted today

Schedule

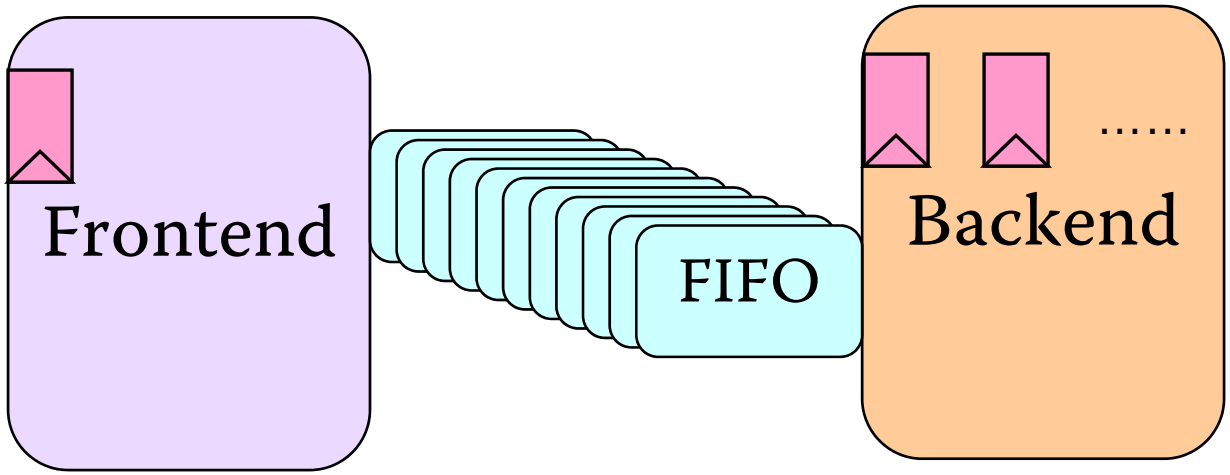
- 5/21 Lab 3a
  - Datapath Design
- 5/28 Lab 3b
  - Complete the Design
    - Control logic
    - >=One Benchmark runs
- 6/4 Lab 3c
  - Final Debugged Design

Lab 3a

- Design the datapath of the processor
  - Revisit the fetch unit
  - Design the datapath of the backend
  - Implement the datapath of the processor using structural Verilog

Big Picture

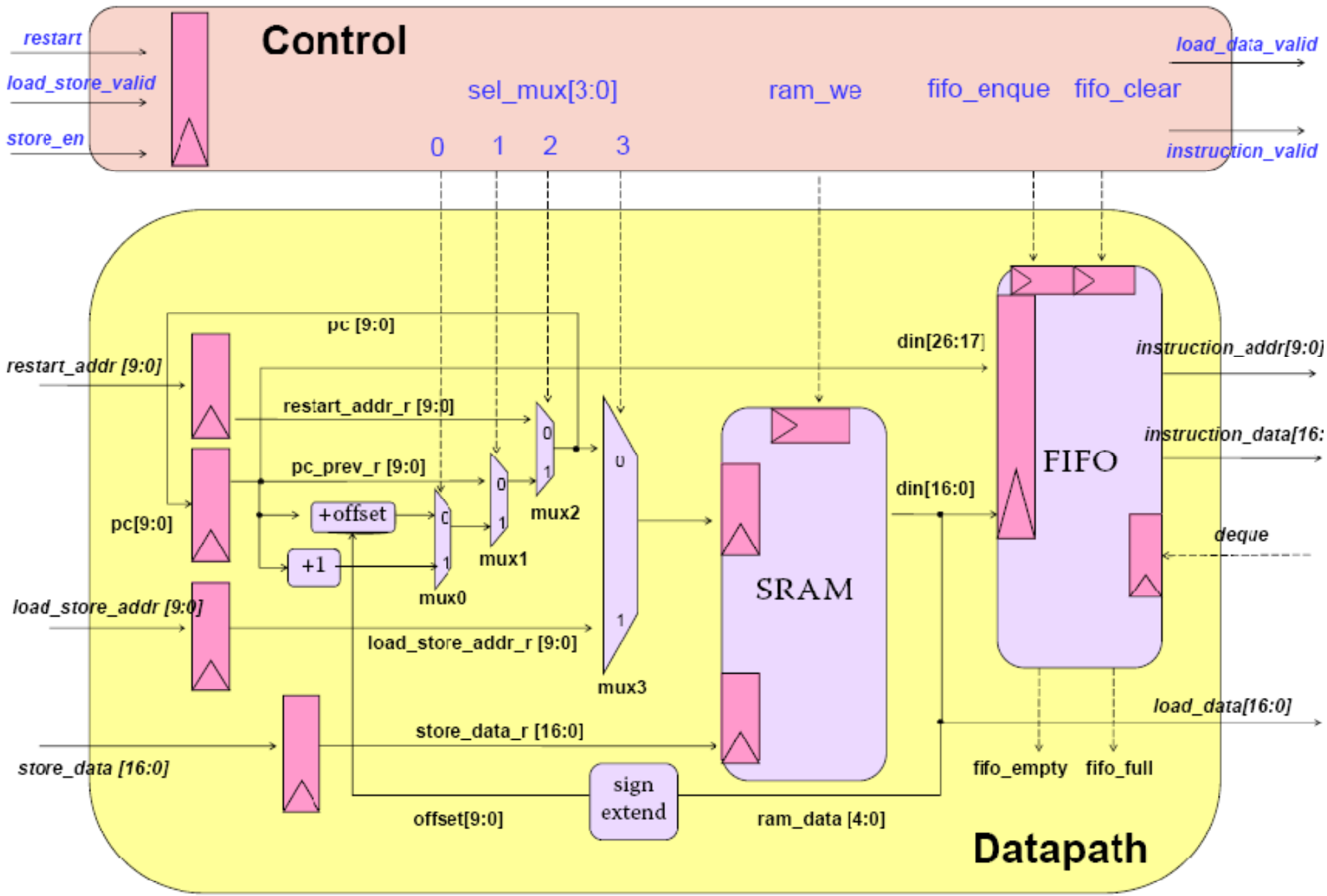
- Decoupled Design
  - At least two stage pipeline
- Backend
  - Single-cycle implementation



Revisit your fetch unit

- Instruction Memory Size
  - Size
    - 1K Word -> 8K Word
  - Address Space
    - 10 bits -> 13 bits

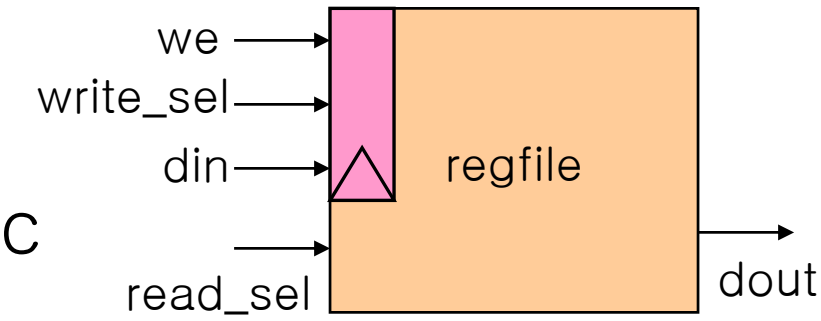
- Fetch unit should be largely identical, but you can freely modify the fetch unit.


Control
restart
load_store_valid
store_en
sel_mux[3:0]
0
1
2
3
ram_we
fifo_enque
fifo_clear
load_data_valid
instruction_valid
pc [9:0]
restart_addr [9:0]
restart_addr_r [9:0]
pc_prev_r [9:0]
pc[9:0]
+offset
+1
mux0
mux1
mux2
mux3
load_store_addr [9:0]
load_store_addr_r [9:0]
store_data [16:0]
store_data_r [16:0]
offset[9:0]
sign extend
ram_data [4:0]
SRAM
din[26:17]
din[16:0]
FIFO
instruction_addr[9:0]
instruction_data[16:
deque
load_data[16:0]
fifo_empty
fifo_full
Datapath

Major Components of the Backend

- Register File
- Data Memory
- Others
  - ALU
  - Sign extender
  - Mux
  - ……

Register File



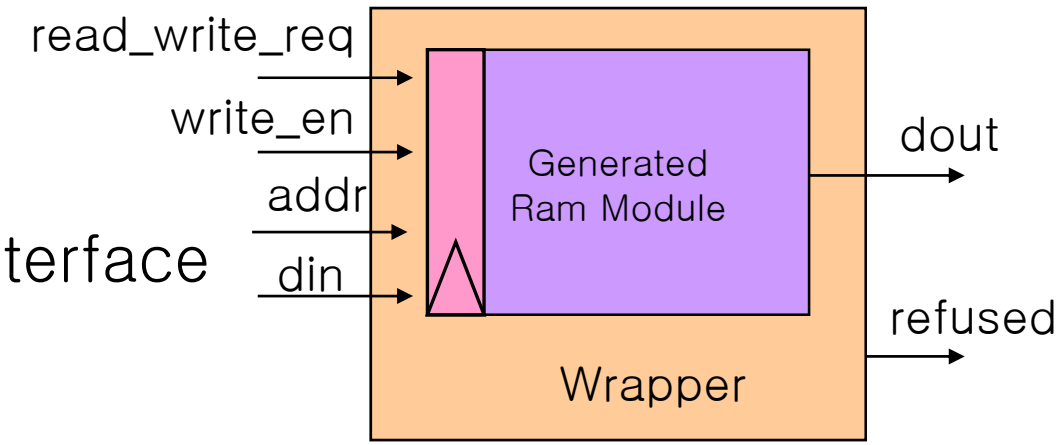
- Read : combinational logic
- Write : sequential logic
- Easily implemented by using flip-flops

```
module regfile#(parameter SEL_WIDTH = 4, D_WIDTH = 34)
{
  input  clk,
  input  we,
  input  [SEL_WIDTH-1 : 0] read_sel,
  input  [SEL_WIDTH-1 : 0] write_sel,
  input  [D_WIDTH-1 : 0]   din,
  output [D_WIDTH-1 : 0]   dout
}
```

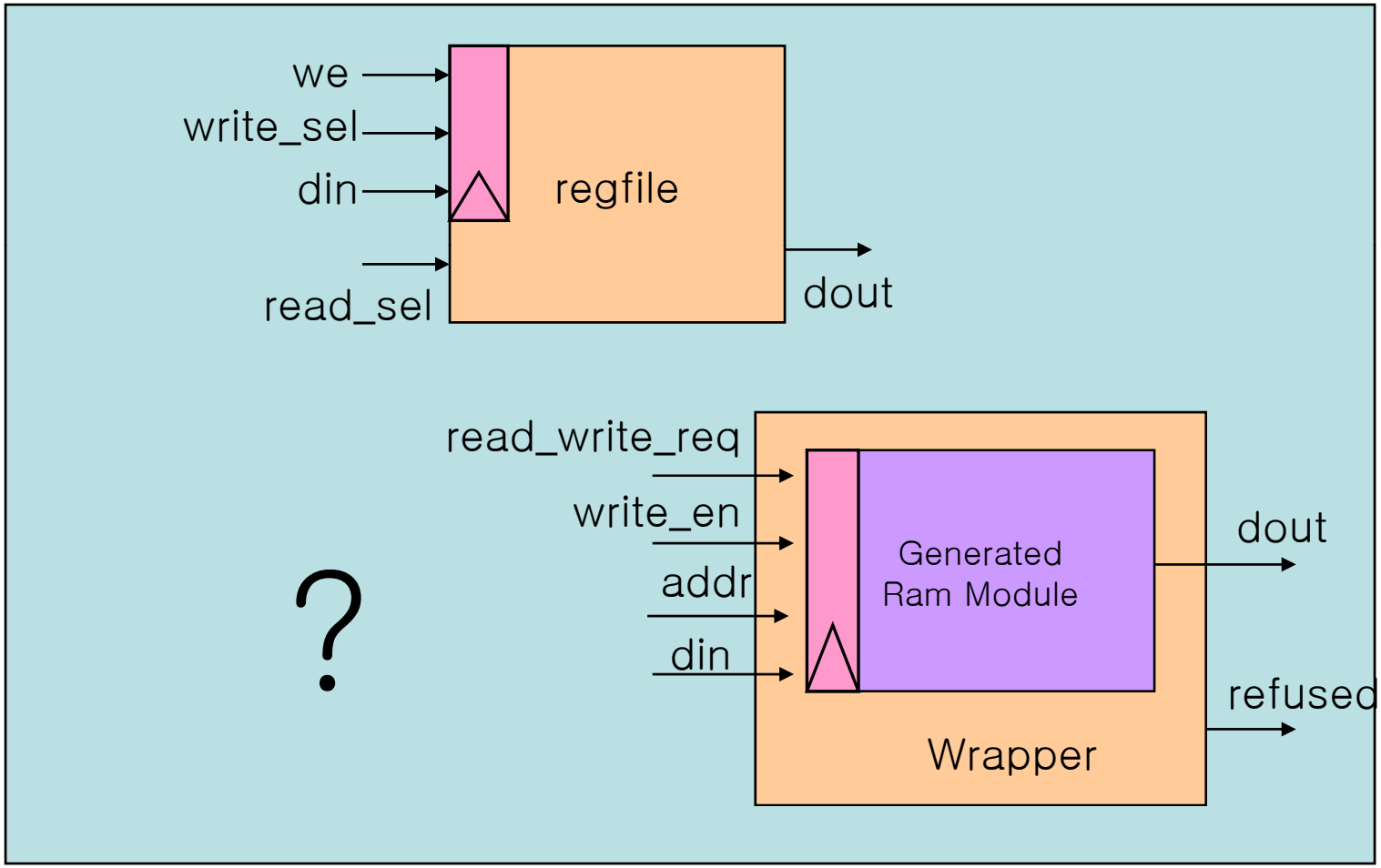
Data Memory

- Use ‘Core Generator’
  - optimized implementation
  - easy to generate
  - initialize with your ‘*.coe’ file
  - 34 bit width, 8K word size

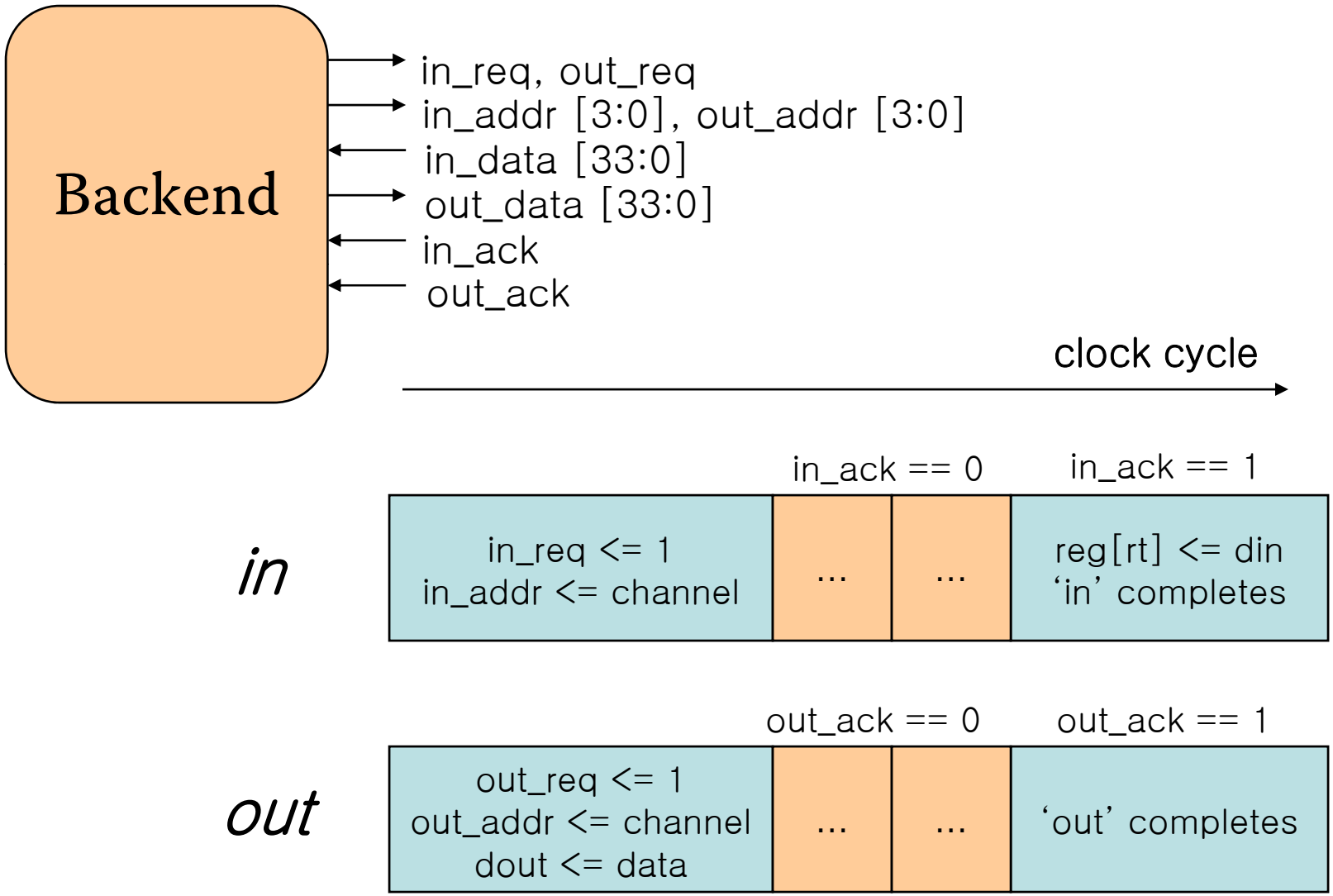
- Wrapper
  - provide general interface
  - ‘refused’ signal



The Backend Datapath



I/O Interface



Deliverables

- Implemented Leaf Modules
  - register file
  - memory module
  - other leaf modules

- The Datapath Schematic for the Processor
  - with explanations on how each instruction work
  - datapath implementation in structural Verilog

Any Questions?