EECS 150 Fall 2003

Lab Lecture 12
Memory & ROM
11/14/2003

Greg Gibeling

Today

- BlockRAM Models
- Making ROMs
- Vector Intensity
- Resource Limits
- Project Test Patterns
- Final Project Checkoffs

BlockRAM Models (1)

- Use the UNISIM Library
  - RAMB4_S16_S16 most likely
  - BUFGs are included
- The Simulation Library:
  - C:\ModelTech5.7d\xilinx\verilog\unisim_ver
- The synthesis files:
  - C:\Xilinx\verilog\src\unisims
- But what about the glbl.GSR stuff?

BlockRAM Models (2)

- We need to implement glbl.GSR
  - Simply add these lines to your testbench
    reg GSR;
    assign glbl.GSR = GSR;
    glbl glbl();
  - This will allow the unisim library to simulate properly
  - What is glbl.GSR?

BlockRAM Models (3)

- glbl.GSR is the global reset
  - This is not normal logic signal reset
  - It is a global reset, just like the BUFGs create global clocks
  - Build your own resets

BlockRAM Models (4)

- What kinds of blockRAMs are there?
  - RAMB4_S1, RAMB4_S2, ... RAMB4_S16
  - RAMB4_S1_S1, ... RAMB4_S16_S16
  - RAMB4_S<WidthA>_S<WidthB>
    - Width: 1, 2, 4, 8, 16
    - WidthA <= WidthB
    - If widthA and widthB are not the same you need to figure out what bits you get at what address (check the datasheet)
BlockRAM Models (5)
- Two separate ports
  - Read one, write to the other
  - Or you can use them together, RAMB4_S16_S16 can be used to fake RAMB4_S32 (which doesn’t exist)
  - Remember they can be clocked differently!

ROM (1)
- How do we build ROM?
  - A big case statement
  - Initialize a BlockRAM
  - Big Case
    - Plus: Asynchronous Read
    - Minus: Painfully inefficient if large
  - BlockRAM
    - Much more elegant

ROM (2)
- Just instantiate memory
  - WE should probably always be 0
  - Initialize the memory
    - The memory takes 16 lines x 256b of data

RAMB4_S8_S8 ROMModule(.WE(1'b0), ...);
defparam ROMModule.INIT_00 = 256'h87 ... F1;
...
defparam ROMModule.INIT_0F = 256'hD2 ... 0E;
...

ROM (3)
- This example:
  - ROMModule(ADDR = 0) = 0xF1
  - ROMModule(ADDR = 511) = 0xD2

RAMB4_S8_S8 ROMModule(.WE(1'b0), ...);
defparam ROMModule.INIT_00 = 256'h87 ... F1;
...
defparam ROMModule.INIT_0F = 256'hD2 ... 0E;
...

Vector Intensity
- Two main schemes
  - Intensity and Direction based on SubCell
    - 81 Direction, Intensity values in ROM
    - ROM indexed by which of 81 subcells has min error
  - Intensity: Error, Direction: SubCell
    - Direction as above
    - Intensity is some linear function of error

Resource Limits
- 100 BlockRAMs (or so)
  - 160 is hard limit
- 60 States in an FSM
  - Anything above maybe 20 states might be too many
- 37ns Clock Period
  - If you’re above 35ns, think about adding registers
  - Remember pipelining?
Test Patterns (1)
- Three test inputs for the project
  - Horizontally scrolling block
  - Vertically scrolling block
  - Spinning Wedges
  - LIVE VIDEO!
- The test pattern generator is on the website (read the README)

Test Patterns (2)
- Blocks should produce obvious arrows
- Wedges will be weird
  - Think hard about the MAE algorithm
  - The wedges have gradients inside them
  - They will not produce the originally expected spinning arrows
- Live video is the real test

Final Project Checkoff (1)
- You must resynthesize and recompile
  - We're watching to see errors, etc
  - We'll have some questions
- You will submit verilog and a bitfile
  - We will load your bitfile for you to demo it
  - Verilog will be read and graded
  - We will run diff on all projects!!

Final Project Checkoff (2)
- We will be picky
  - Fix your reset problems
    - (At least mostly)
  - Watch the borders of the picture
  - Make sure the arrows look good
- Our demo is the preliminary standard
  - You can do better
  - You should do at least as well.

Final Project Checkoff (3)
- Only one checkoff per group
  - If you want early and 30Hz you must have 30Hz on the early checkoff day
- Extra Credit
  - Get the basics working first
  - Extra credit won't make up for a broken project

Next...
- We’ll talk about the report next week
  - COME TO LAB LECTURE
- More final checkoff details
  - Early checkoff next Friday
  - Details will be posted online