Lecture #6: Lab 3 Game-Plan and Arithmetic in Logic

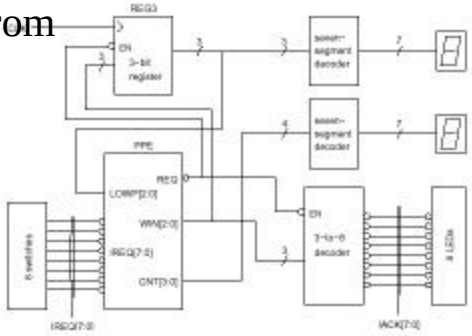
Paul Hartke
Phartke@stanford.edu
Stanford EE121
January 24, 2001

Public Service Announcement

- Computer Forum Job Fair ***TODAY!!!***
  - 9:00 a.m. - 4:00 p.m.
  - In a tent at the lawn between Gates CS and David Packard EE Buildings
- Below is a full list of companies that will be participating:
  - BEA Systems, Inc.; Cisco Systems; Compaq Computer; Corporation; DeepHaven Capital Management; D.E. Shaw; DoCoMo, USA Labs; E.piphany ; Google, Inc.; Honda R&D Americas, Inc.; IBM Corporation; Intel; internalDrive - ID Tech Camps; Microsoft Corporation; NEC USA, Inc.; Oracle Corporation; Rambus, Inc.; SAP Labs, Inc.; Siebel Systems, INc.; Symantec Corporation; TRW; VMware, Inc.

Lab 3 Game Plan

- First, Questions on Lab 3?
  - Everyone is started right? ☺
- How do we go from this diagram to working circuit?



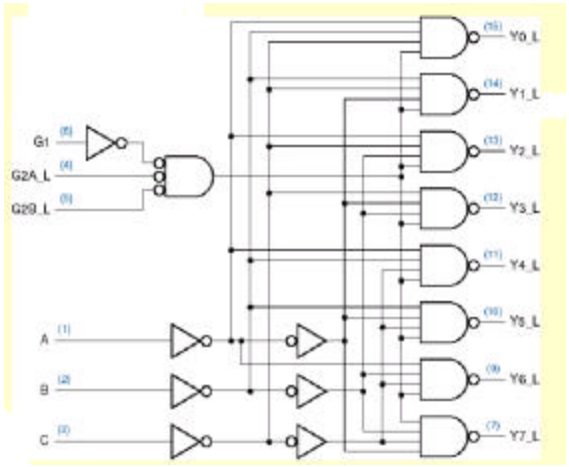
Hierarchical Design

- With Divide and Conquer
- Make basic blocks for 7seg decoder, 3-8 decoder, and Priority Encoder
- Test individually with *separate* verification script
  - Increases confidence that when you integrate the pieces the whole thing will work.

3-8 Decoder

- Implement exactly as we talked about in class last time.
  - Page 358 in book.
  - How do you know you got the wiring correct?
    - Hard to make all those straight lines
  - What about the bubbles?
- What if you want to redesign the block for smaller size/higher performance.

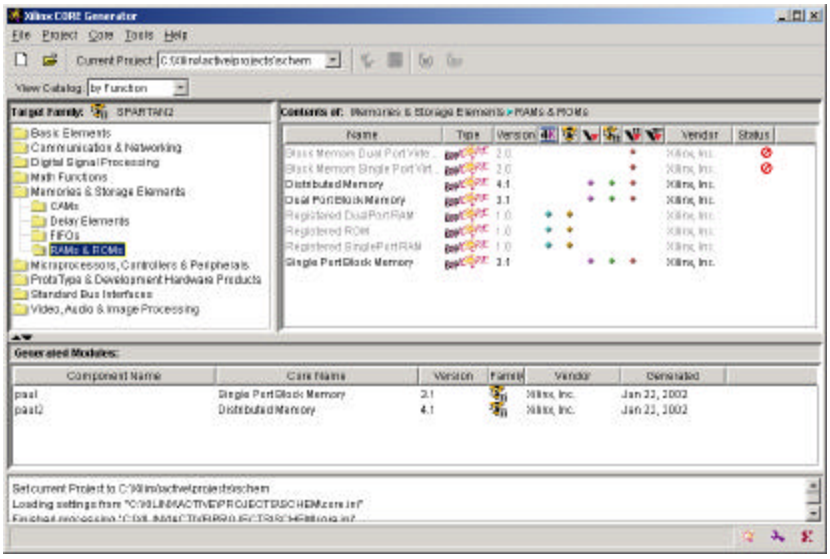
3-8 Decoder in Gates



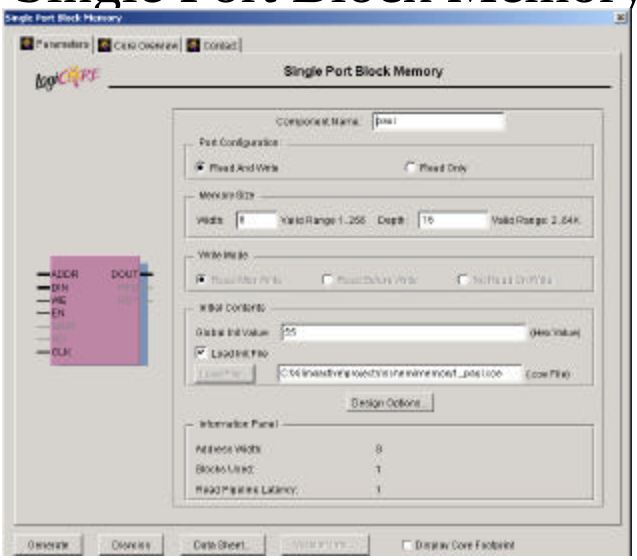
7Segment Decoder

- Could implement with page 373 diagram.
  - We did one part of this last time.
- Or, could use Xilinx Core Generator to make an internal memory element for this subcomponent
- 4 input ROM with 7 outputs

Xilinx Core Generator



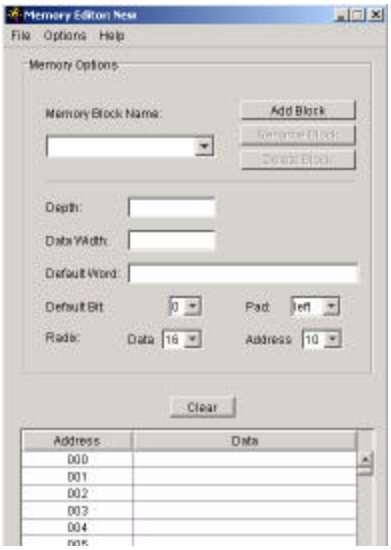
Single Port Block Memory



Initialization File

- Need a memory file to initialize the memory element.
- Learn .coe format from docs on Xilinx website or use the memory editor inside CoreGen

Memory Editor



Block Ram in Xilinx Spartans

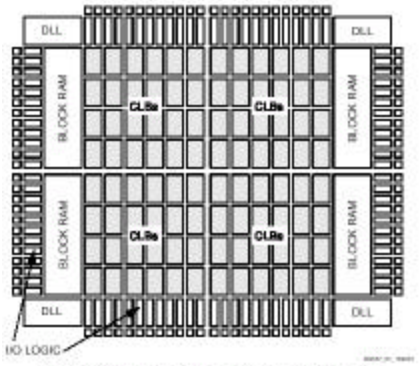


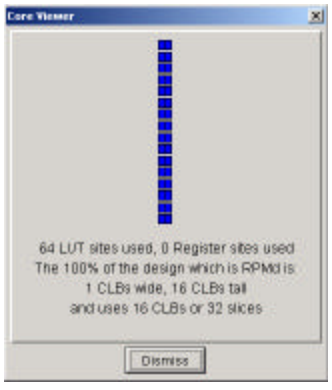
Figure 1: **Basic Spartan-IIE Family FPGA Block Diagram**

Block Ram doesn't work?!?!?!

- This is the element used for the Literature quotes for Lab 2
- We could not get "Single Port Block Memory" to work.
- Lesson: How did we figure this out?
- Use "Distributed Memory" while we work this out…

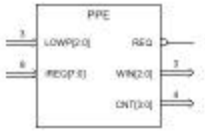
Distributed Memory in Spartans

- Use LUTs to implement memory
  - Not so efficient but fine for small structures.



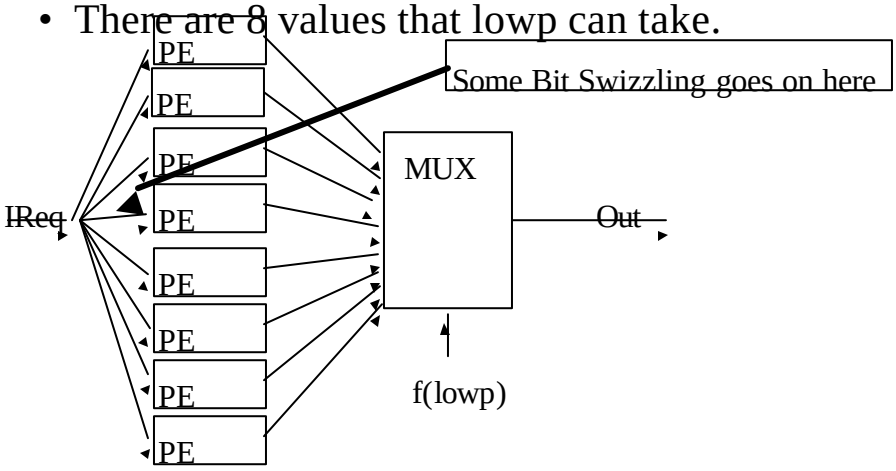
Now the PPE…

- Begin by making macro of basic priority encoder by itself.
  - Test it!!!

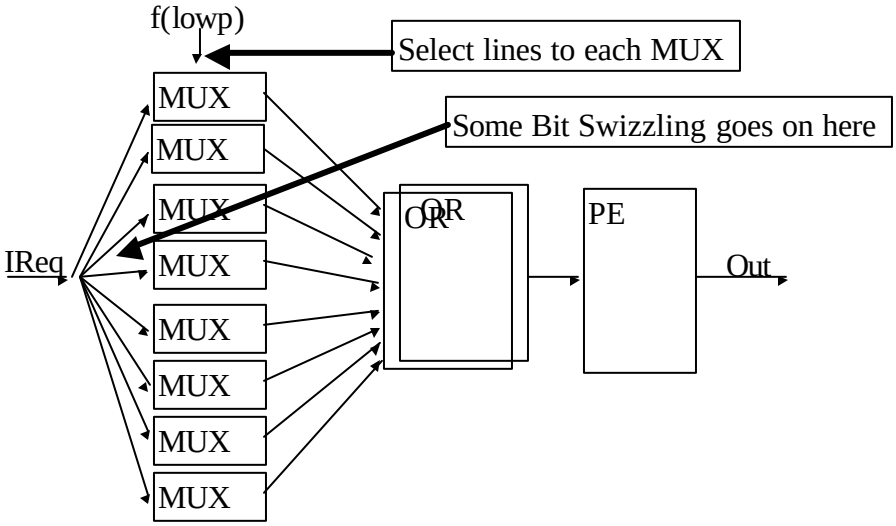


What to do with it?

- There are 8 values that lowp can take.



What to do with it (Part Deux)?



CNT and REQ

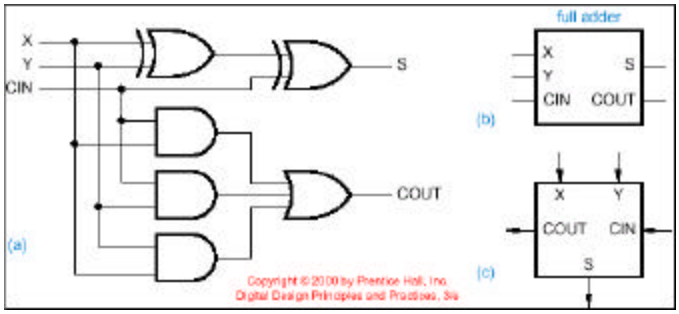
- Don’t forget about the count of active inputs (CNT) and a line (REQ) that indicates at least one input was high?
- How are you going to get the count??

Other Architectures Possible…

- These are just some ideas…

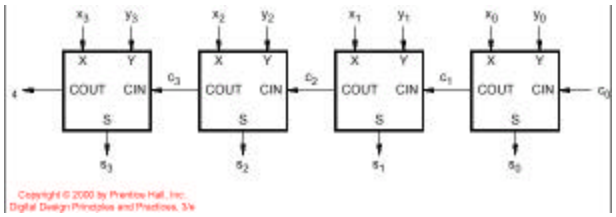
Adders

- Full Adder and Half Adders Same as E40



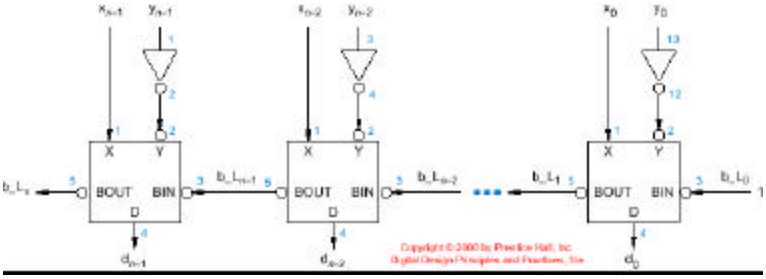
Ripple Adder

- Classic Iterative Architecture
  - Pros/Cons?
    - Low Routing Delay…



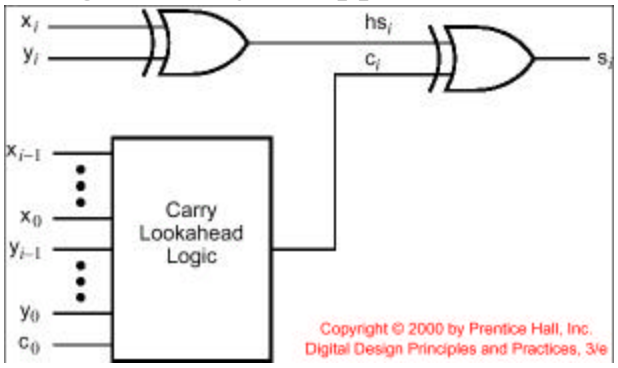
Adder as Subtractor

- As mentioned previously, you can subtract with an adder by inverting the inputs and setting the cin to “1”



Carry Look Ahead

- Directly Calculate the Carry instead of waiting for carry to ripple.



CLA Propagate and Generate

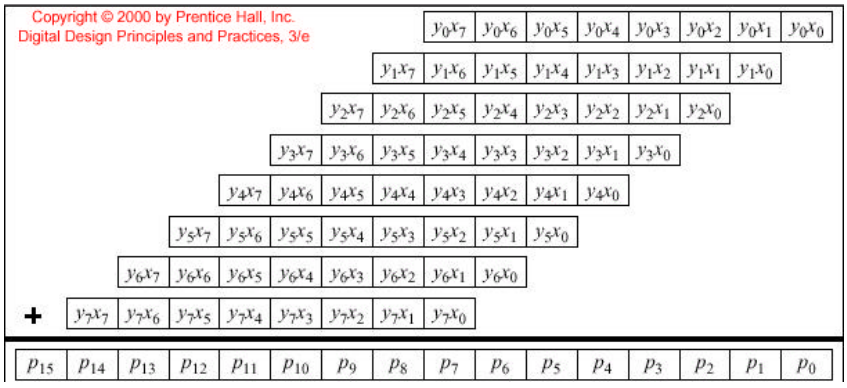
- Gi = xi * yi
- Pi = xi + yi
- C1 = g0 + p0 * c0
- Iterate
- Three levels of logic to calculate the carry
  - What about fan-in/fan-out of gates?
  - Wiring Delay?

Lots of other architectures

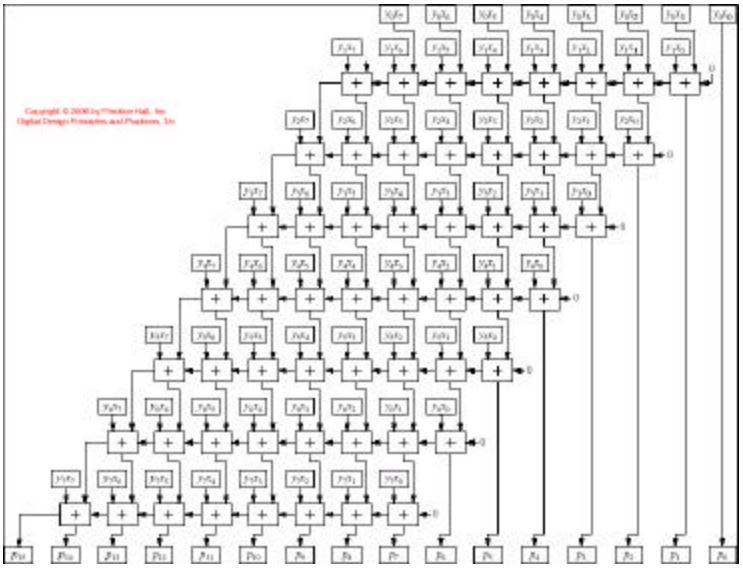
- Carry Save, Carry Skip, Carry Select, etc
- Often combine techniques.
- We won't talk about them but there is so particular magic to them.

Multipliers

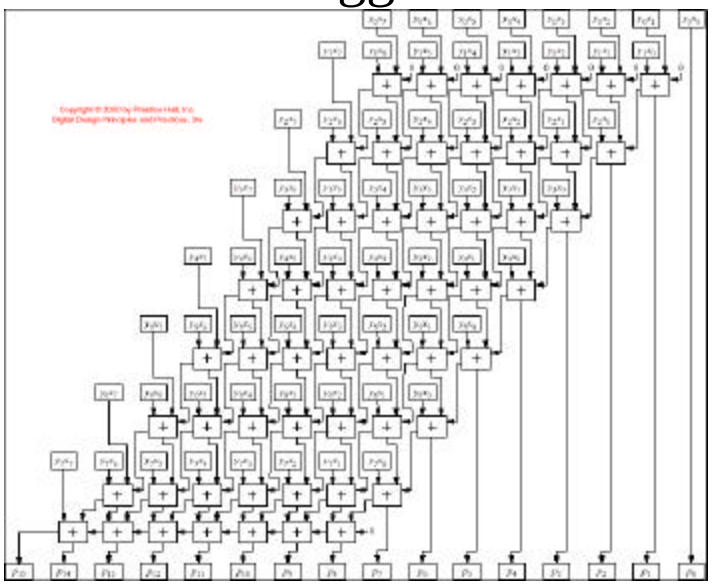
- Repeated Addition



Initial Architecture



More Aggressive



Multipliers Summary

- Lots of clever architectures out there. They all do the same thing—multiply!
- Consider routing delay in addition to logic delay.

Next Time

- State Machine Design