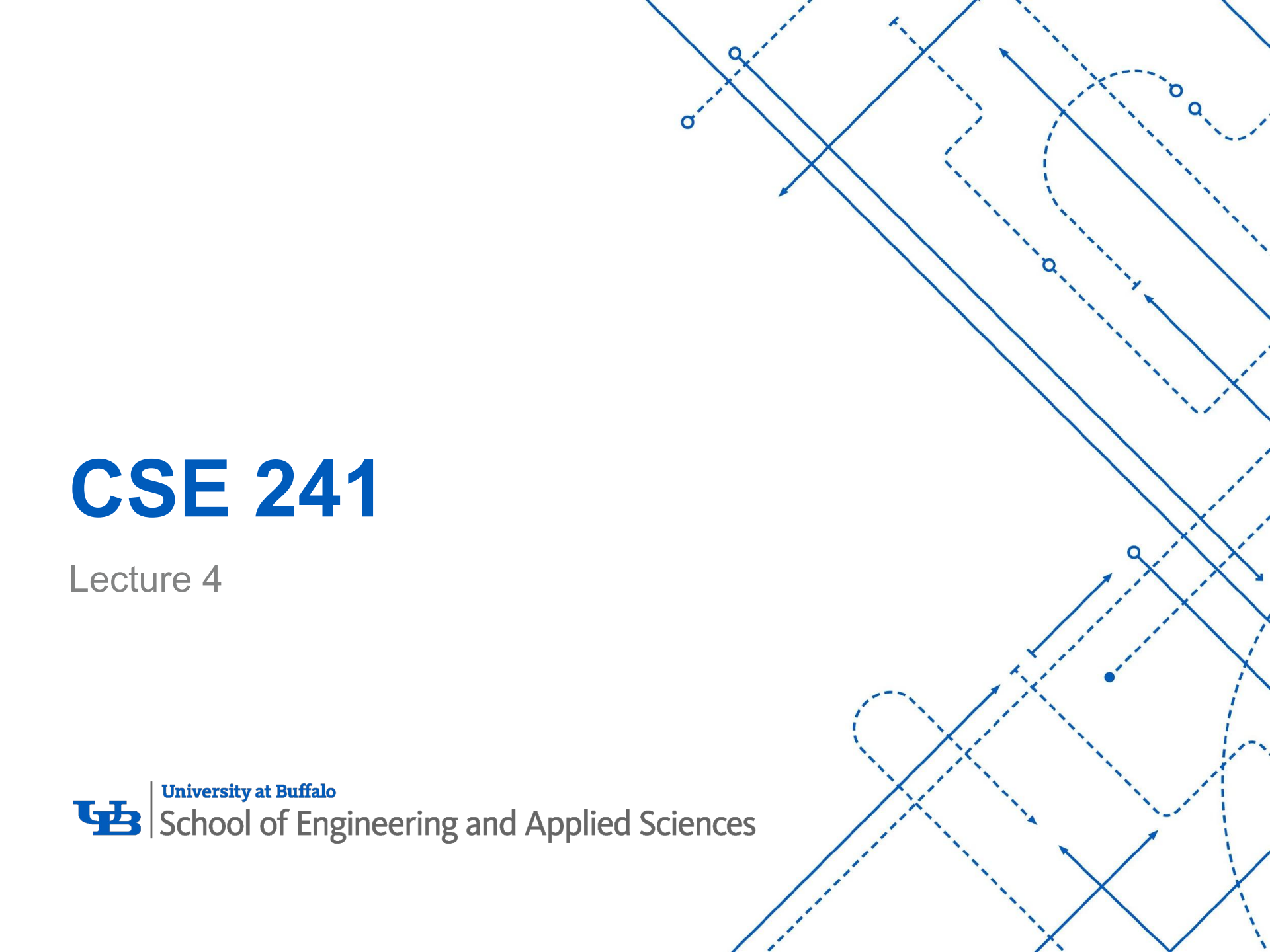


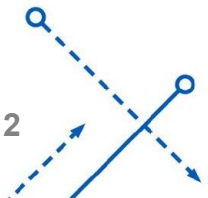
CSE 241

Lecture 4

University at Buffalo
School of Engineering and Applied Sciences

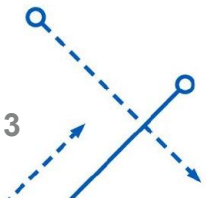
Overview for this lecture

- Reminders
- Introduce some combinational logic - not on exam
- Review

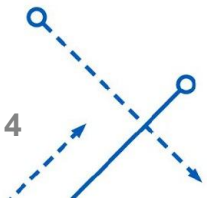


Reminders

- Lab 1 Due June 14 at **5:30 pm**
- HW 3 Due tonight at 11:59 pm
- Exam on Wednesday
- Lab 3 will be set to start Wednesday
  - Due June 20th at 5:30 pm
  - You get 2 lab sections to work on it
  - It is a short lab

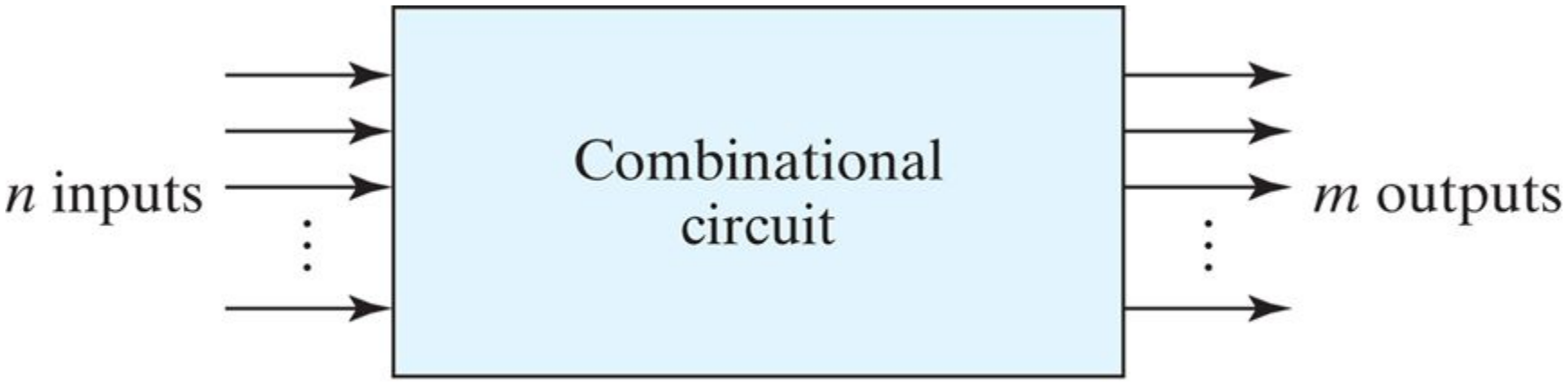


Questions?



What is Combinational Logic?

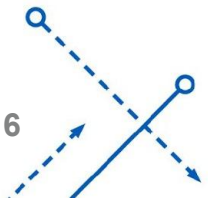
- Circuits fall into one of two categories
  - Combinational
  - Sequential
- These circuits depend on only the current inputs



MSI

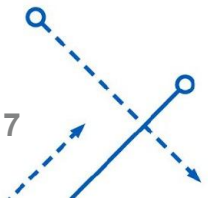
- Medium Scale Integration
- Level of complexity of the combinational logic we will explore
- How Complex Is This?
  - All these circuits can be fabricated to an IC

Anyone recall what IC is short for?



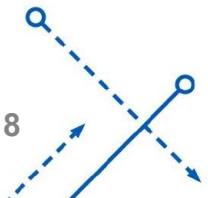
Some key things about Combinational Logic Design

- No Feedback Paths
- No Memory Elements
- Can have multiple inputs *and* outputs



Design Procedure

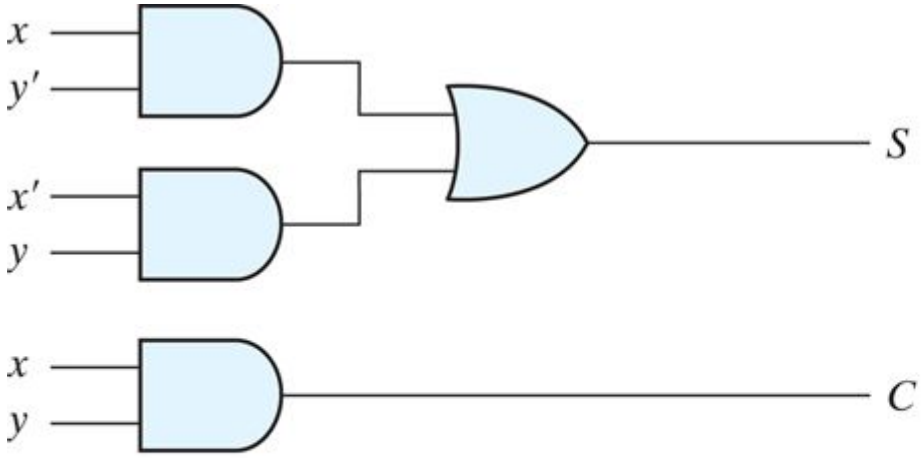
1. Determine the number of inputs and outputs- Assign a symbol to each
2. Derive the Truth Table
3. Obtain the simplified Boolean functions for each output as a function of the inputs
4. Draw the logic diagram and verify correctness (manually or by simulation)



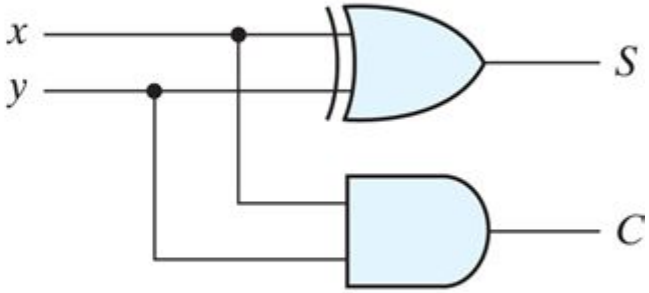
Let's Consider a Half Adder

Table 4.3
Half Adder

x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

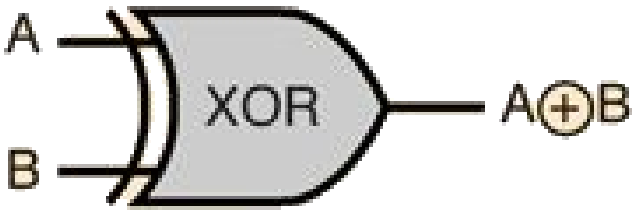


(a) $S = xy' + x'y$
$C = xy$



(b) $S = x \oplus y$
$C = xy$

What is ⊕



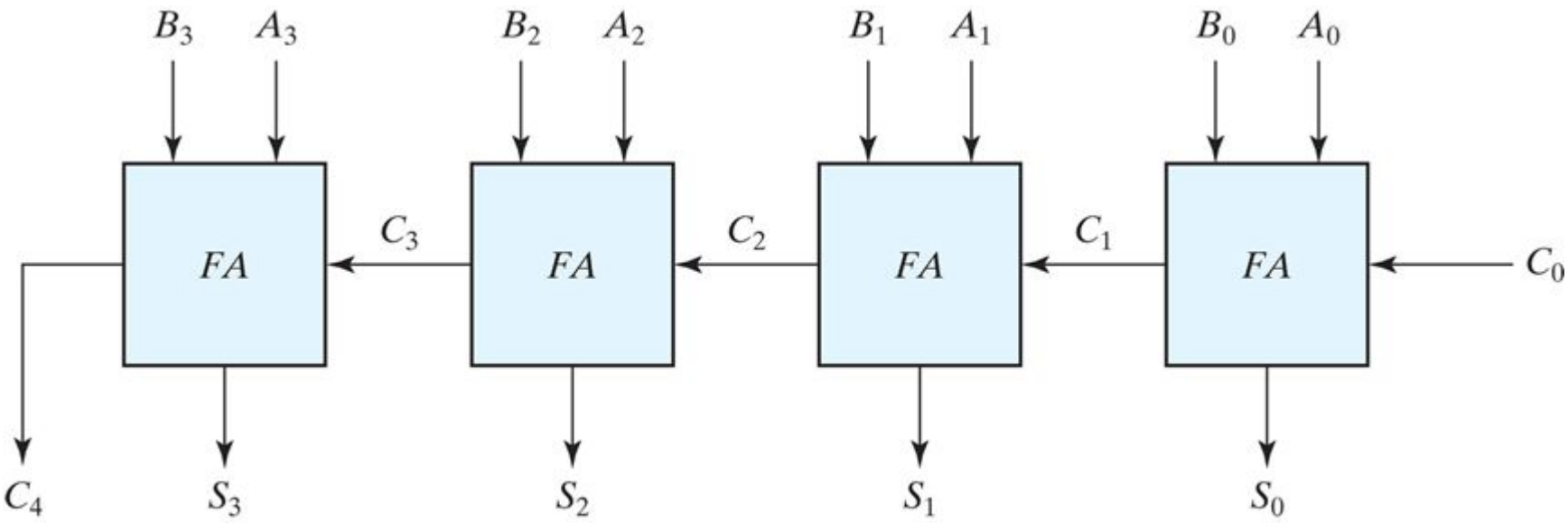
A	B	Out
0	0	0
0	1	1
1	0	1
1	1	0

Let's Walk Through a Bit Slower With a Full Adder

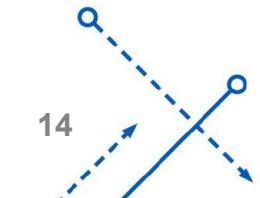
- What is different between a Full Adder and a Half Adder?

What if we wanted a 4 bit adder?

- This is called a ripple carry

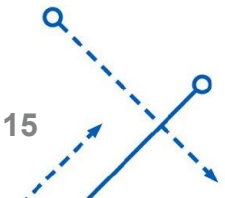


Is there anything we have to consider with this design?



Solution- Fast Adders

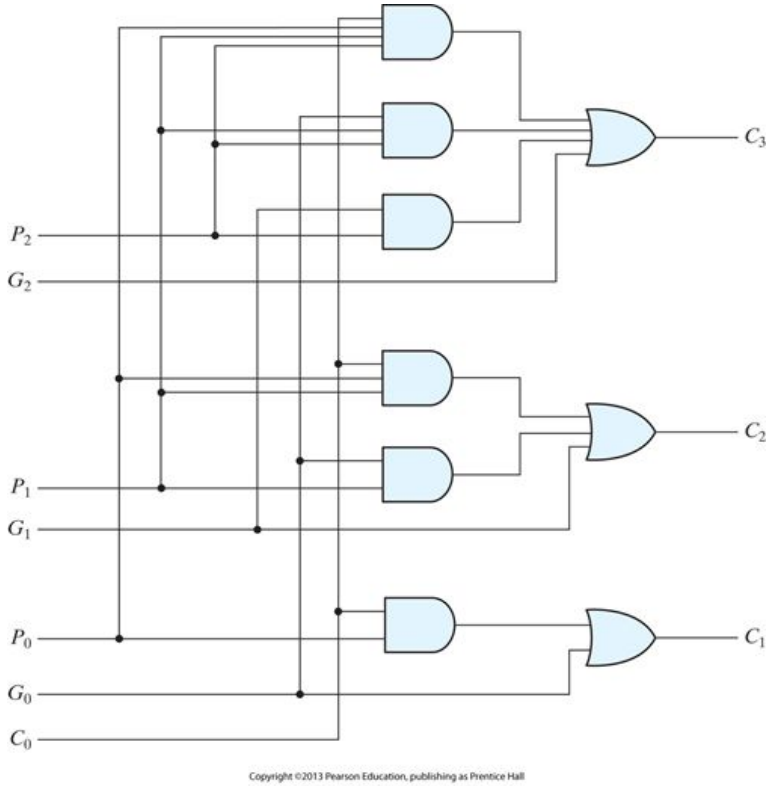
- There are multiple methods to deal with the delay of waiting for the carries to propagate through
- They are more complex
- Here is the 5 minute or so run down



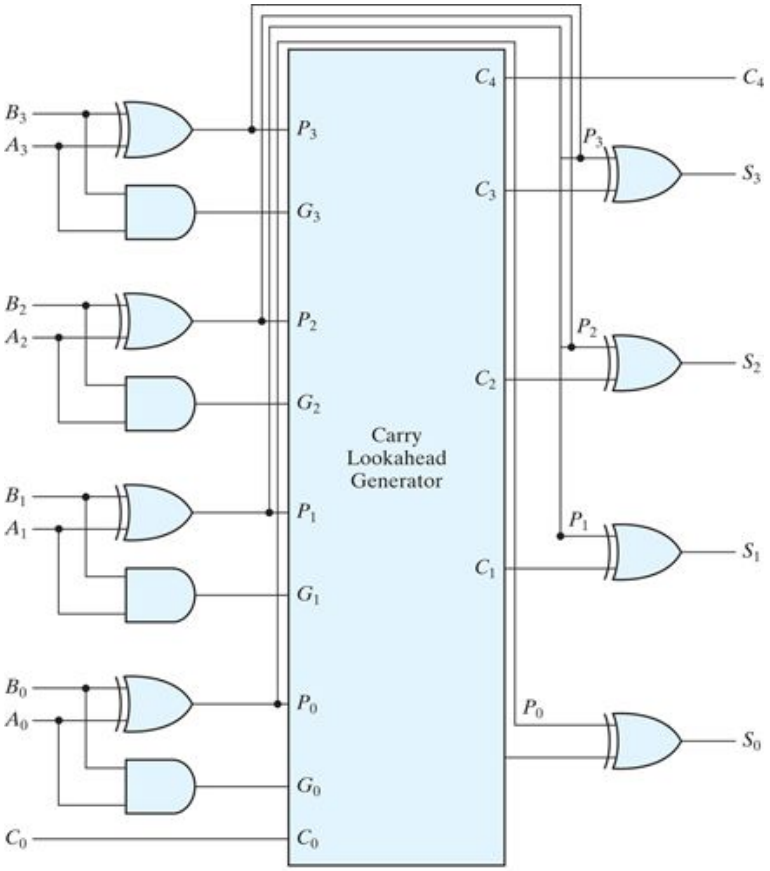
Carry Lookahead Adder

- Evaluate carries first, then crank out the sum.
- Faster, but gets really complex as number of inputs goes up.
- In a little more detail:
- We calculate a generate and a propagate function for each addition.
  - $c_i+1 = g_i + p_i c_i$
  - $g_i = x_i y_i$, $p_i = x_i + y_i$
  - $c_{i+1} = g_i + p_i g_{i-1} + p_i p_{i-1} c_{i-1}$
  - $c_{i+1} = g_i + p_i g_{i-1} + p_i p_{i-1} g_{i-2} + \ldots + p_i p_{i-1} \ldots p_2 p_1 p_0 g_0 + p_i p_{i-1} \ldots p_2 p_1 p_0 c_0$
- p_i is the carry propagate and g_i is the carry generate term

What does the carry generator part look like for 3 carries?



Putting it all together



Hierarchical Carry Lookahead Adder

- Break up larger adds into blocks, and do an initial calculation of the carry between the blocks
- Think divide and conquer approach

Now

- A Short Break
- Then your practice exam
- If you finish early you can take a break until recitation
  - Recitation starts at 5:30
  - Others may be using the room before then
- If you finished your lab last time you don't need to go to recitation today