EE466

VLSI System Design

Midterm Exam

Nov. 11, 2022. (4:20pm – 5:35pm)

Instructor: Dae Hyun Kim (daehyun@eecs.wsu.edu)

Name:

WSU ID:

Problem	Points	
1	10	
2	10	
3	10	
4	10	
5	20	
6	10	
7	10	
Total	80	

* Allowed: Textbooks, cheat sheets, class notes, notebooks, calculators, watches

* Not allowed: Electronic devices (smart phones, tablet PCs, laptops, etc.) except calculators and watches

Problem #1 (Kogge-Stone Adder, 10 points)

For the 128-bit Kogge-Stone adder, show one of the critical paths to calculate S_{77}. What is the delay of the critical path? Use the following delay values for logic gates.

- 2-input AND, OR: d
- XOR: 2d

$$S_{77} = p_{77} \oplus C_{77}$$

$$C_{77} = g_{76:0} + p_{76:0} \cdot C_0$$

$$g_{76:0} = g_{76:13} + p_{76:13} \cdot g_{12:0}$$

$$g_{76:13} = g_{76:45} + p_{76:45} \cdot g_{44:13}$$

$$g_{76:45} = g_{76:61} + p_{76:61} \cdot g_{60:45}$$

$$g_{76:61} = g_{76:69} + p_{76:69} \cdot g_{68:61}$$

$$g_{76:69} = g_{76:73} + p_{76:73} \cdot g_{72:69}$$

$$g_{76:73} = g_{76:75} + p_{76:75} \cdot g_{74:73}$$

$$g_{76:75} = g_{76} + p_{76} \cdot g_{75}$$

Delay: 2d (for p_{76}) + d (for $p_{76} \cdot g_{75}$) + d (for $g_{76} + p_{76} \cdot g_{75}$)

$g_{76:73}$: +2d, $g_{76:69}$: +2d, $g_{76:61}$: +2d, $g_{76:45}$: +2d, $g_{76:13}$: +2d, $g_{76:0}$: +d, C_{77}: +d, S_{77}: +2d

Answer: 18d

Problem #2 (Kogge-Stone Adder, 20 points)

Count the # following gates required to implement the 16-bit Kogge-Stone adder (including the generation of C_{16}).

- 2-input AND gates:
- 2-input OR gates:
- 2-input XOR gates:

$g_i = A_i \cdot B_i$: one AND → 16 ANDs

$p_i = A_i \oplus B_i$: one XOR → 16 XORs

$C_i = g_{i-1:0} + p_{i-1:0} \cdot C_0$: one OR and one AND → 16 ORs and 16 ANDs

$s_i = p_i \oplus C_i$: one XOR gate → 16 XOR gates

The rest of them will be for the generation of group generation and propagation signals. Notice that generating a $g_{i:k}$ from its parents requires one AND and one OR gates, and generating a $p_{i:k}$ from its parents requires one AND gate.

$g_{i+1:i}$ and $p_{i+1:i}$: 2 ANDs and 1 OR (for i=0 to 14)

$g_{i+3:i}$ and $p_{i+3:i}$: 2 ANDs and 1 OR (for i=0 to 12)

$g_{i+7:i}$ and $p_{i+7:i}$: 2 ANDs and 1 OR (for i=0 to 8)

$g_{i+15:i}$ and $p_{i+15:i}$: 2 ANDs and 1 OR (for i=0)

$g_{2:0}$ and $p_{2:0}$: 2 ANDs and 1 OR

$g_{4:0}$ and $p_{4:0}$: 2 ANDs and 1 OR

$g_{5:0}$ and $p_{5:0}$: 2 ANDs and 1 OR

$g_{6:0}$ and $p_{6:0}$: 2 ANDs and 1 OR

$g_{8:0}$ and $p_{8:0}$: 2 ANDs and 1 OR

$g_{9:0}$ and $p_{9:0}$: 2 ANDs and 1 OR

$g_{10:0}$ and $p_{10:0}$: 2 ANDs and 1 OR

$g_{11:0}$ and $p_{11:0}$: 2 ANDs and 1 OR

$g_{12:0}$ and $p_{12:0}$: 2 ANDs and 1 OR

$g_{13:0}$ and $p_{13:0}$: 2 ANDs and 1 OR

$g_{14:0}$ and $p_{14:0}$: 2 ANDs and 1 OR

AND: 16 + 16 + 15*2 + 13*2 + 9*2 + 1*2 + 11*2 = 130

OR: 16 + 15*1 + 13*1 + 9*1 + 1*1 + 11*1 = 65

XOR: 32

Problem #3 (Carry-Lookahead Adder, 10 points)

For the 128-bit Carry-lookahead adder, show one of the critical paths to calculate S_{77}. What is the delay of the critical path? Use the following delay values for logic gates.

- 2-, 3-, 4-input AND, OR: d
- XOR: 2d

$$S_{77} = p_{77} \oplus C_{77}$$

$$C_{77} = g_{76} + p_{76} \cdot C_{76}$$

$$C_{76} = g_{75:72} + p_{75:72} \cdot g_{71:68} + p_{75:72} \cdot p_{71:68} \cdot g_{67:64} + p_{75:72} \cdot p_{71:68} \cdot p_{67:64} \cdot C_{64}$$

$$C_{64} = g_{63:0} + p_{63:0} \cdot C_0$$

$$g_{63:0} = g_{63:48} + p_{63:48} \cdot g_{47:32} + p_{63:48} \cdot p_{47:32} \cdot g_{31:16} + p_{63:48} \cdot p_{47:32} \cdot p_{31:16} \cdot g_{15:0}$$

$$g_{15:0} = g_{15:12} + p_{15:12} \cdot g_{11:8} + p_{15:12} \cdot p_{11:8} \cdot g_{7:4} + p_{15:12} \cdot p_{11:8} \cdot p_{7:4} \cdot g_{3:0}$$

$$g_{3:0} = g_3 + p_3 \cdot g_2 + p_3 \cdot p_2 \cdot g_1 + p_3 \cdot p_2 \cdot p_1 \cdot g_0$$

$$p_3 = A_3 \oplus B_3$$

Delay: 2d (for p_3) + 2d (for $g_{3:0}$) + 2d (for $g_{15:0}$) + 2d (for $g_{63:0}$) + d (for C_{64}) + 2d (for C_{76}) + 2d (for C_{77}) + 2d (for S_{77})

Answer: 15d

Problem #4 (Carry-Lookahead Adder, 20 points)

Count the # following gates required to implement the 16-bit Carry-Lookahead adder (including the generation of C_{16}).

- 2,3,4-input AND gates (i.e., # 2-input ANDs + # 3-input ANDs + # 4-input ANDs):
- 2,3,4-input OR gates:
- 2-input XOR gates:

$g_i = A_i \cdot B_i$: one AND → 16 ANDs

$p_i = A_i \oplus B_i$: one XOR → 16 XORs

$s_i = p_i \oplus C_i$: one XOR → 16 XORs

In each level-1 carry-lookahead unit (there are four L-1 units, i=0,4,8,12)

- $g_{i+1:i} = g_{i+1} + p_{i+1} \cdot g_i$
- $p_{i+1:i} = p_{i+1} \cdot p_i$
- $g_{i+2:i} = g_{i+2} + p_{i+2} \cdot g_{i+1} + p_{i+2} \cdot p_{i+1} \cdot g_i$
- $p_{i+2:i} = p_{i+2} \cdot p_{i+1} \cdot p_i$
- $g_{i+3:i} = g_{i+3} + p_{i+3} \cdot g_{i+2} + p_{i+3} \cdot p_{i+2} \cdot g_{i+1} + p_{i+3} \cdot p_{i+2} \cdot p_{i+1} \cdot g_i$
- $p_{i+3:i} = p_{i+3} \cdot p_{i+2} \cdot p_{i+1} \cdot p_i$
- $C_{i+1} = g_i + p_i \cdot C_i$
- $C_{i+2} = g_{i+1:i} + p_{i+1:i} \cdot C_i$
- $C_{i+3} = g_{i+2:i} + p_{i+2:i} \cdot C_i$

In each level-2 carry-lookahead unit (there is one L-2 unit)

- $C_4 = g_{3:0} + p_{3:0} \cdot C_0$
- $C_8 = g_{7:4} + p_{7:4} \cdot g_{3:0} + p_{7:4} \cdot p_{3:0} \cdot C_0$
- $C_{12} = g_{11:8} + p_{11:8} \cdot g_{7:4} + p_{11:8} \cdot p_{7:4} \cdot g_{3:0} + p_{11:8} \cdot p_{7:4} \cdot p_{3:0} \cdot C_0$
- $g_{15:0} = g_{15:12} + p_{15:12} \cdot g_{11:8} + p_{15:12} \cdot p_{11:8} \cdot g_{7:4} + p_{15:12} \cdot p_{11:8} \cdot p_{7:4} \cdot g_{3:0}$
- $p_{15:0} = p_{15:12} \cdot p_{11:8} \cdot p_{7:4} \cdot p_{3:0}$

In a level-3 unit

- $C_{16} = g_{15:0} + p_{15:0} \cdot C_0$

L1: 4*(12 ANDs + 6 ORs) = 48 ANDs + 24 ORs

L2: 10 ANDs + 4 ORs

L3: 1 AND + 1 OR

Answer:

AND: 16 + 48 + 10 + 1 = 75

OR: 24 + 4 + 1 = 29

XOR: 16 + 16 = 32

Problem #5 (Ripple-Carry Adder, 10 points)

For the 128-bit ripple-carry adder, estimate the delay of the critical path to compute S_{127}. Use the following delay values for logic gates.

- 2-input AND, OR: d
- XOR: 2d

Use the following expressions for the computation of each carry signal.

- $g_i = A_i \cdot B_i$
- $p_i = A_i \oplus B_i$
- $C_{i+1} = g_i + p_i \cdot C_i$
- $S_i = p_i \oplus C_i$

p_0: 2d

C_1: +2d

C_2: +2d

…

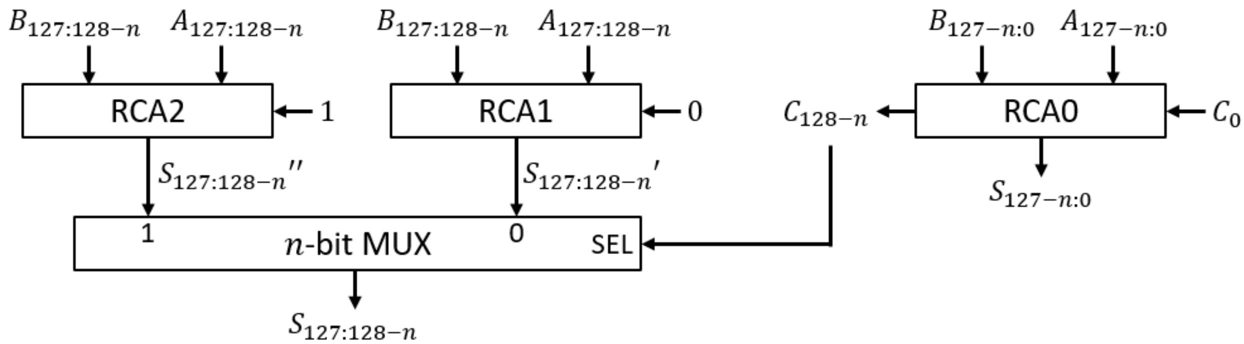
C_{127}: +2d

S_{127}: +2d

Answer: 258d

Problem #6 (Ripple-Carry Adder, 20 points)

We want to design a 128-bit adder as follows. First, we split it into n bits (MSBs) and $128 - n$ bits (LSBs). Then, we add $A_{127-n:0}$, $B_{127-n:0}$, and C_0 using a $(128 - n)$-bit ripple-carry adder (RCA0). At the same time, we add $A_{127:128-n}$, $B_{127:128-n}$, 0 in an n-bit RCA (RCA1), and also add $A_{127:128-n}$, $B_{127:128-n}$, 1 in another n-bit RCA (RCA2). The following shows a schematic of this adder.



Use the following delay values for logic gates.

- 2-input AND, OR: d
- XOR: 2d
- MUX: 3d

Use the following expressions for the computation of each carry signal.

- $g_i = A_i \cdot B_i$
- $p_i = A_i \oplus B_i$
- $C_{i+1} = g_i + p_i \cdot C_i$
- $S_i = p_i \oplus C_i$

(1) Calculate the delay of C_{128-n}. (Note: It should include n.)

p_0: 2d

C_1: +2d

…

C_{128-n}: +2d

Answer: 2d + 2d*(128-n)=$258d - 2dn$

(2) Calculate the delay of S_{127}'. (Note: It should include n.)

p_{128-n}: 2d

C_{129-n}: +2d

…

C_{127}: +2d

S_{127}: +2d

Answer: 2d + 2d*(n-1) + 2d = $2d + 2dn$

(3) Calculate the delay of S_{127} for $n = 64$. (Note: the delay of S_{127}'' is equal to that of S_{127}'.)

Delay of C_{64}: 130d

Delay of S_{127}': 130d

Answer: 130d + 3d (MUX) = 133d

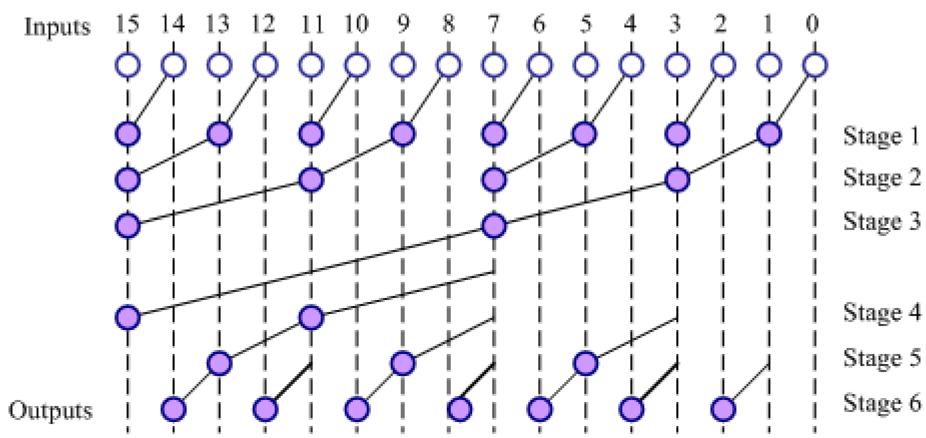
(4) Calculate the delay of S_{127} for $n > 64$. (Note: It should include n.)

If n>64, S_{127}' is the critical path, so the delay is $2d + 2dn + 3d = 5d + 2dn$.

Problem #7 (Prefix Adder, 10 points)

Count the # following gates required to implement the 16-bit Brent-Kung adder shown below (including the generation of C_{16}).

- 2-input AND gates:
- 2-input OR gates:
- 2-input XOR gates:



$g_i = A_i \cdot B_i$: one AND → 16 ANDs

$p_i = A_i \oplus B_i$: one XOR → 16 XORs

$C_i = g_{i-1:0} + p_{i-1:0} \cdot C_0$: one OR and one AND → 16 ORs and 16 ANDs

$s_i = p_i \oplus C_i$: one XOR gate → 16 XOR gates

Each purple circle generates $g_{i:k}$ and $p_{i:k}$ from two group generation and propagation signals. Thus, each purple circle has two ANDs and 1 OR (one AND for $p_{i:k}$, and one AND and one OR for $g_{i:k}$).

There are 26 purple circles => 26*(2 ANDs and 1 OR) = 52 ANDs and 26 ORs

AND: 16 + 16 + 52 = 84

OR: 16 + 26 = 42

XOR: 32