EE 457 HW 2
Arithmetic Designs
Redekopp • Puvvada

Name: _____

Due: See Website **Score: _____**

Please post any questions regarding HW problems on Piazza.

Refer to your class notes Unit on Fast Addition. In this class we will count the delay of an XOR gate as two (2) gate-delays. This is based on the SOP (sum of products) expression of X^Y = X•Y' + X'•Y, which requires two AND gates in the first level feeding an OR gate in the second. We are not counting inverters as a level of logic (gate delay). The actual VLSI design of an XOR gate does not use this SOP expression and thus the delay may be less than 2 gates, but is generally more than 1 gate delay.

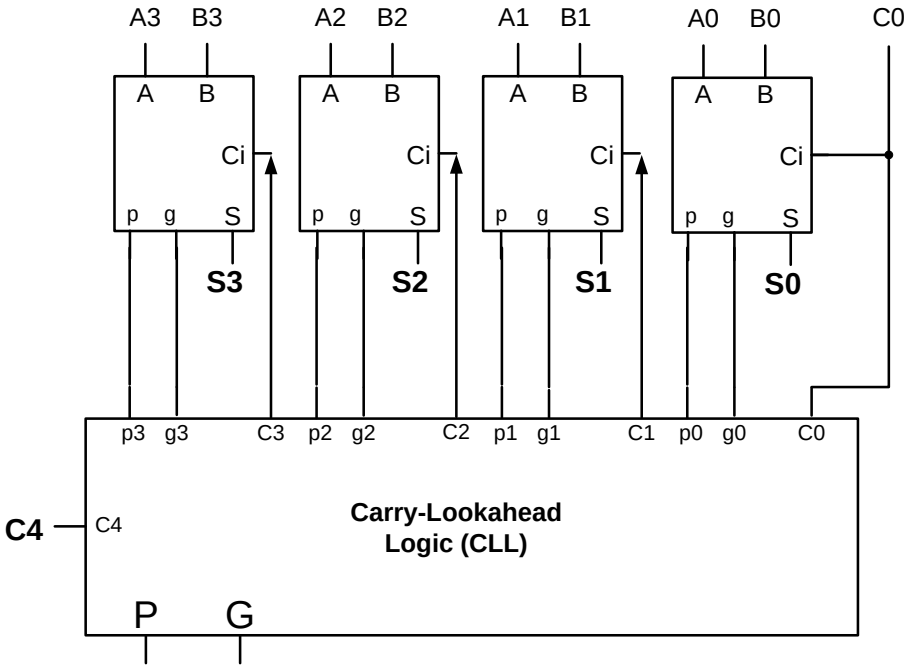
1. **(15 pts.) Fast Adder Design**
   1.1. **(3)** Refer to the 16-bit CLA adder design presented in the 'Fast Adders Slides' on and calculate its delay. Is it 5, 7, or 9 gate delays? Explain how you arrived at your answer.

   1.2. **(4)** If you take four of these 16-bit adders and build a 64-bit adder using an additional level of carry-lookahead logic what would be the delay of such a design?

   1.3. **(2+2)** What would be the complexity of gates needed in the SOP implementation of the carry-lookahead logic which can receive eight pairs of p's and g's and also a carry-in (c0) and produce c1-c8? By complexity of gates, we mean the highest fan-in (number of input pins) warranted for the AND gates and the highest fan-in warranted for the OR gates to implement the design in 2-level AND-OR (SOP) logic.

   Highest Fan-in of ANDs: _____

   Highest Fan-in of ORs: _____



1.4. **(2+2)** Consider the evaluation of: **B = A63 + A62 + … + A1 + A0**

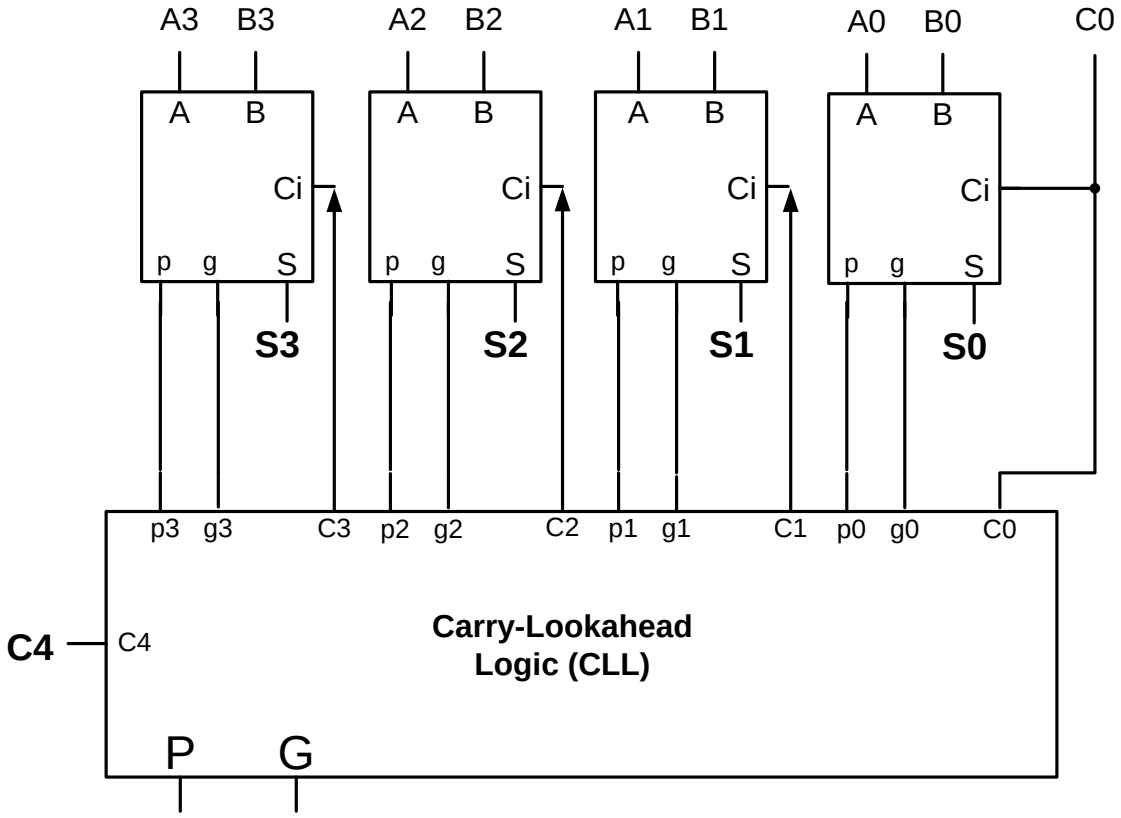
Using 4-input OR gates only what is the number of gate delays required if you arrange OR gates in a linear cascade (one feeding the next and so on)?

What would be the delay if you arrange them in a tree fashion?

You do not have to draw any of the arrangements just work out the number of gate delays required.

2. (10 pts.) Carry-Lookahead Adders

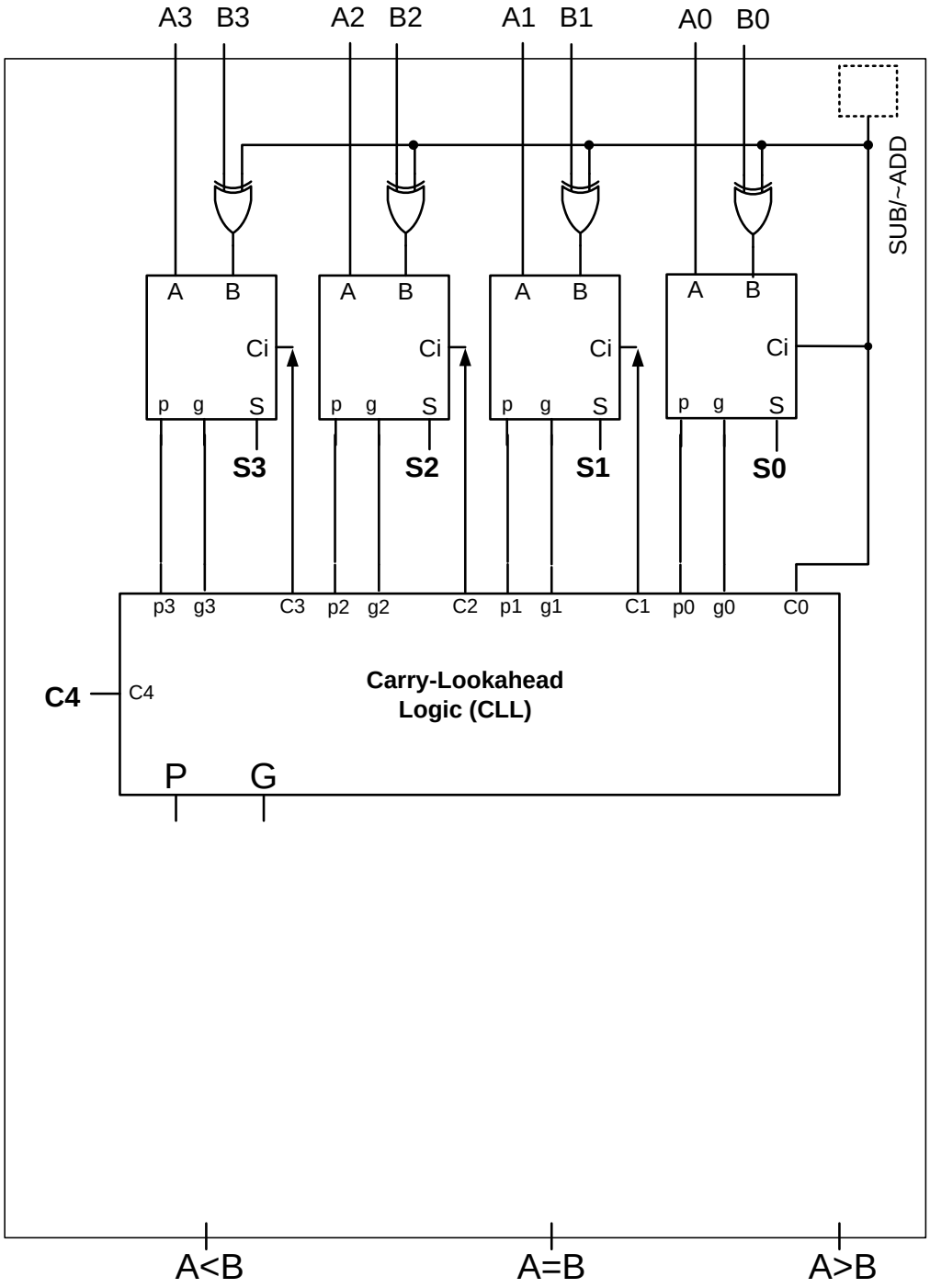
(**3 + 3 + 2 + 2**) Reproduced below is a 4-bit CLA discussed in your class. We have arrived at the delay of this adder as 5 gate delays. Add additional logic to this adder as necessary and produce (i) USAO=Unsigned Addition Overflow output and (ii) SAO=Signed (i.e. 2's complement) Addition Overflow and find the delay for these outputs. Do your best to keep your delay at 5 gates or less.



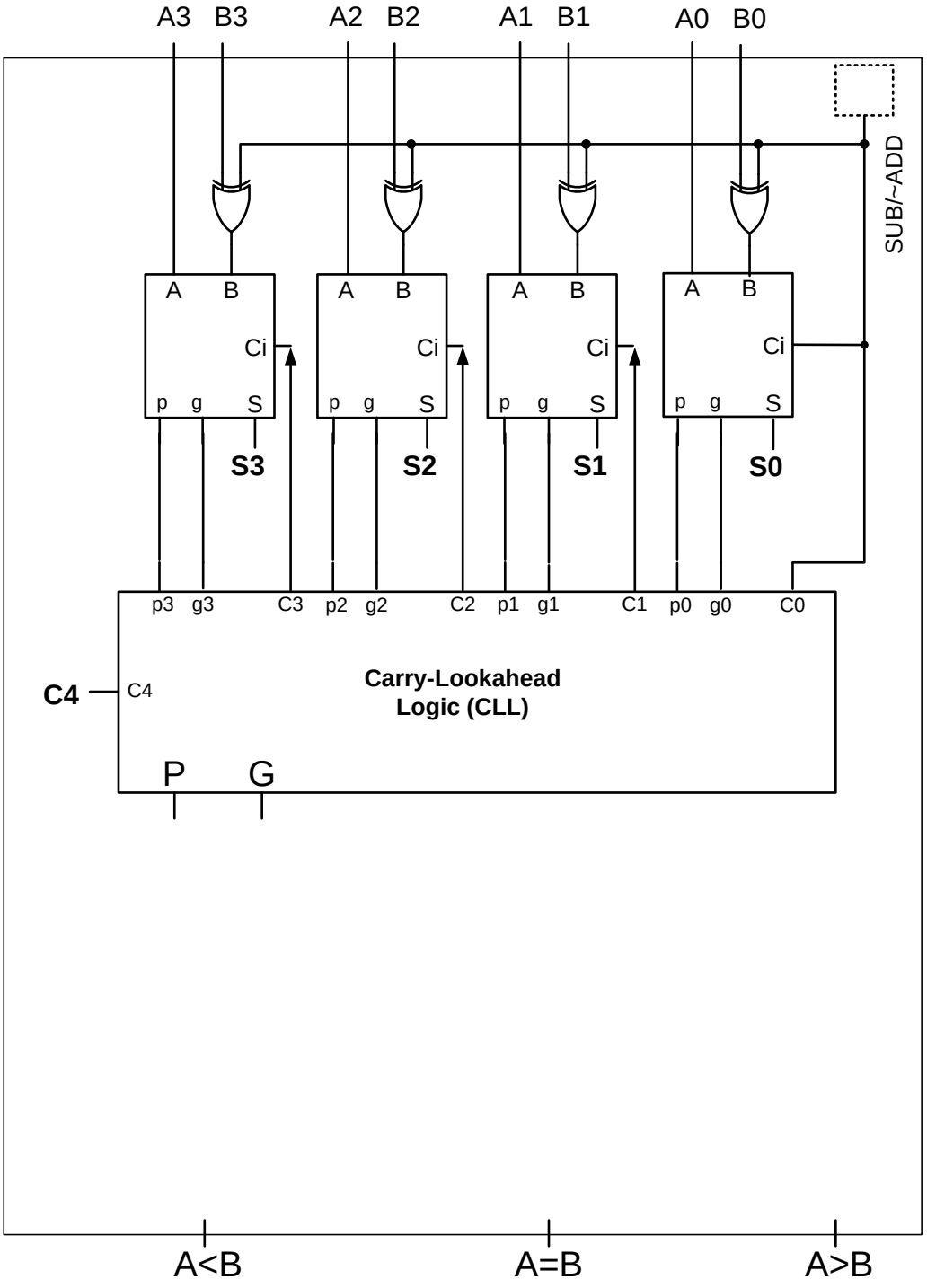
DELAY for USAO=_____ **DELAY for SAO=_____**

3. (30 pts.) Unsigned and Signed Comparison and Overflow

3.1. **(5 + 5 + 5)** Design a 4-bit comparator to compare two 4-bit UNSIGNED numbers A[3:0] and B[3:0] using a 4-bit adder-subtractor given below and additional gates as needed. Hint: Consider the overflow in ***unsigned*** subtraction (as the MIPS SLTU instruction might do) [You may not have covered the MIPS instruction set yet, in which case you can ignore the parenthetical comment].



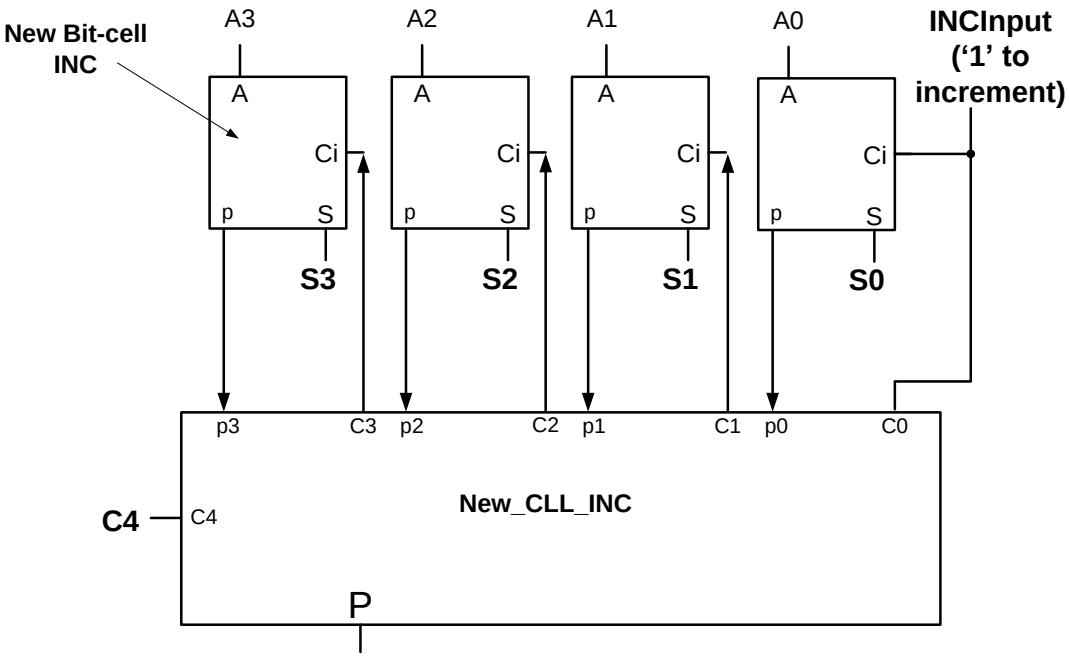
3.2. **(5 + 5 + 5)**Design a 4-bit comparator to compare two 4-bit SIGNED numbers X[3:0] and Y[3:0] using a 4-bit adder-subtractor given below and additional gates as needed. Hint: Consider the overflow in ***signed*** subtraction (as the MIPS SLT instruction might do) [You may not have covered the MIPS instruction set yet, in which case you can ignore the parenthetical comment].



4. (27 pts.) Basic Computer Arithmetic

4.1. An incrementer is a special case of an adder adding the constant 1 (ONE). Instead of using a full CLA adder to increment, a simplified incrementer can be designed as shown below. The ONE to be added can be conveyed at the **INCInput** which is connected to the C0 (carry-in) input. If **INCInput** is 0 the circuit will simply reproduce A (A+0=A).

Note: Si = (Ai XOR Bi) XOR Ci = (Ai XOR 0) XOR Ci = Ai XOR Ci
pi = Ai OR Bi = Ai OR 0 = Ai
gi = Ai AND Bi = Ai AND 0 = 0
Ci = 1 if some earlier adder (FA) generates a carry and all intermediate adders (FAs) propagate the same. Since gi = 0, the Ci can only be true if there is a carry-in (C0) and all intermediate pi signals are true.



4.1.1. **(2+2+2)** Write the Boolean equations for C1, C2, C3 produced by the NEW_CLL_INC block.

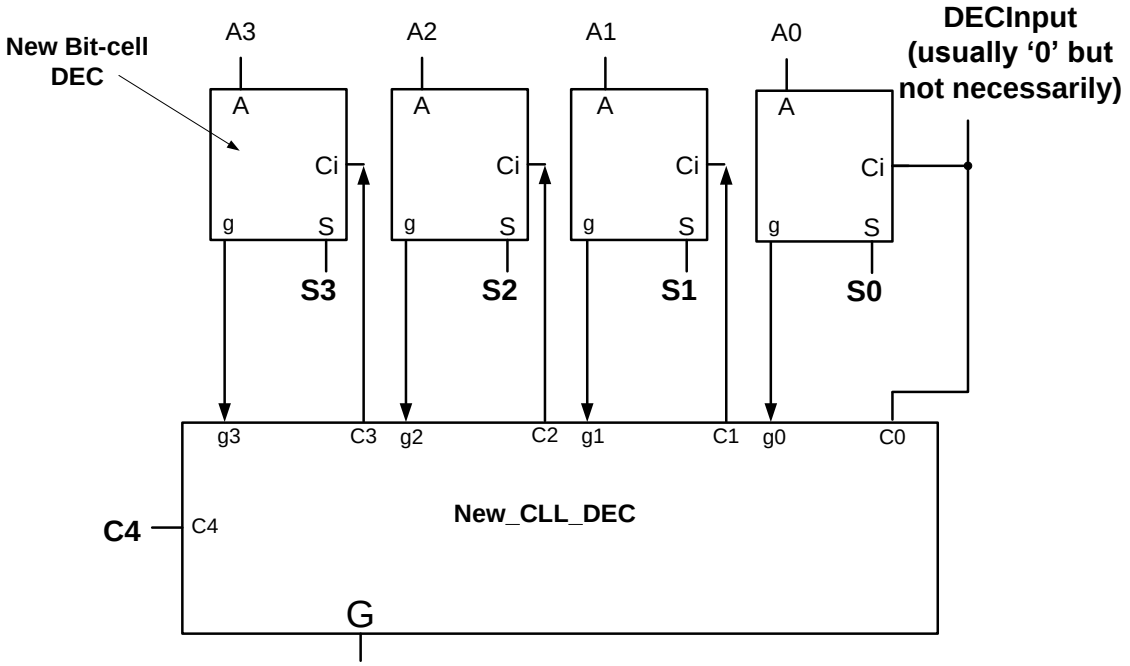
C1 =

C2 =

C3 =

4.1.2. **(3)** What is the gate delay of the 4-bit incrementer? Count XOR as a 2-gate delay device. Explain how you arrived at your answer?

4.2. Similar to an incrementer, a decrementer is a special case of an adder adding the constant -1 (MINUS ONE). In a 4-bit 2's complement, -1 = 1111 (all ones). Design a simplified (area efficient, speed efficient) decrementer by simplifying a CLA adder that adds 1111. Do not assume that C0 is a zero but instead treat it as a variable.



Note: Si = (Ai XOR Bi) XOR Ci = (Ai XOR 1) XOR Ci = ~Ai XOR Ci
= Ai XNOR Ci

4.2.1. **(1+1+1)** Complete the following lines:

pi = Ai OR Bi = Ai OR 1 = _____

gi = Ai AND Bi = Ai AND 1 = _____

Ci = 1 if some earlier adder (FA) generates a carry and all intermediate adders (FAs) propagate the earlier carry. Since all pi are _____ in this design, the Ci can be true if _____

4.2.2. **(1+2+2+2)** Write the Boolean equations for C1, C2, C3 and *group generate, G,* signal produced by the NEW_CLL_DEC.

C1 =

C2 =

C3 =

G =

Consider a 64-bit decrementer using three levels of the above NEW_CLL_DEC.

4.2.3. **(2)** Number of NEW_CLL_DECs needed: _____

4.2.4. **(2)** Number of New Bit-Cell DECs needed: _____

4.2.5. **(2)** Delay of the 64-bit decrementer in gates (counting XOR/XNOR as 2 gate delays): _____

4.2.6. **(2)** The 64-bit decrementer is good to decrement **(circle the correct answer)**:

- **Unsigned numbers only**
- **Signed 2's complement numbers only**
- **Both unsigned and signed**

5. (18 pts.) Ripple Carry Adder and Carry Lookahead Adder Delays

5.1. Delay of the 2-bit RCA (design 1 below) is _____ gate delays.

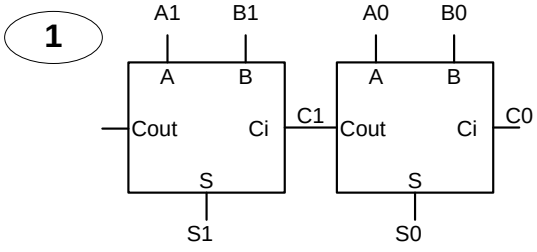
5.2. Delay of the 2-bit CLA (design 2 below is _____ gate delays

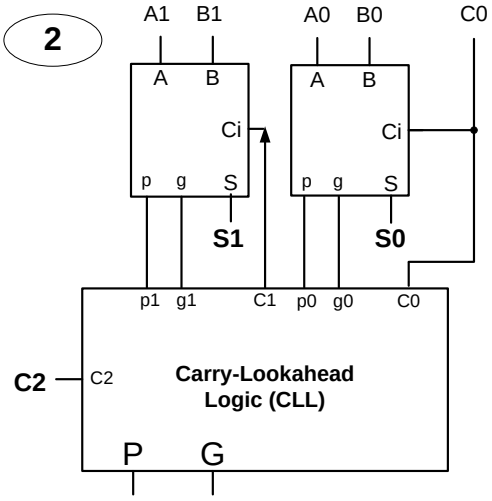
5.3. Delay of a 4-bit RCA (shown in class) is _____ gate delays.

5.4. Delay of a 4-bit CLA (shown in class) is _____ gate delays.

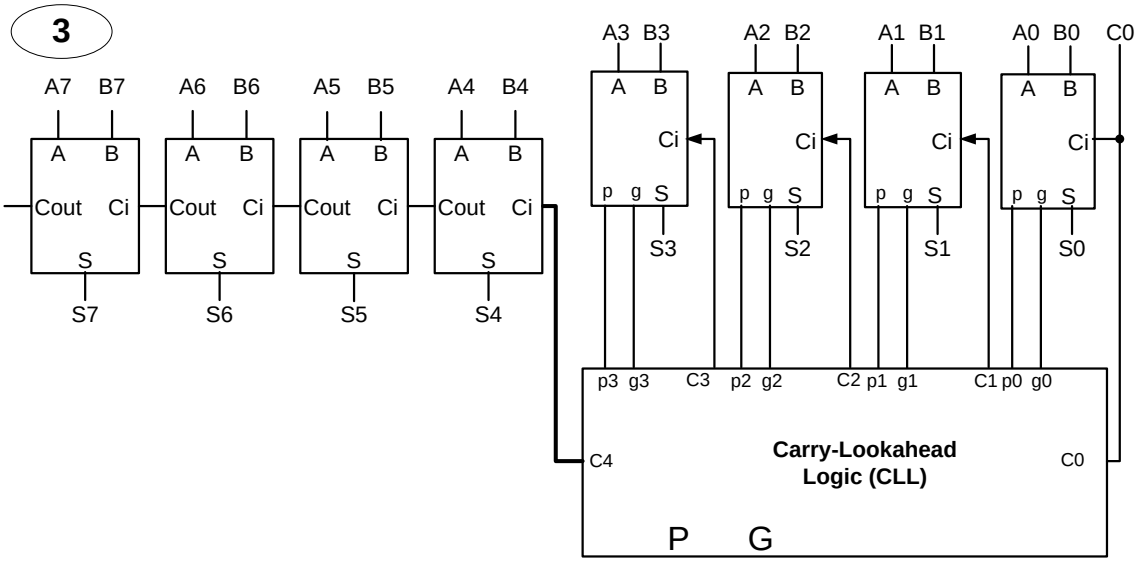
5.5. Delay of the 8-bit adder in design 3 on the next pg. is _____ gate delays.

5.6. Delay of the 8-bit adder in design 4 on the next pg. is _____ gate delays.





3



4