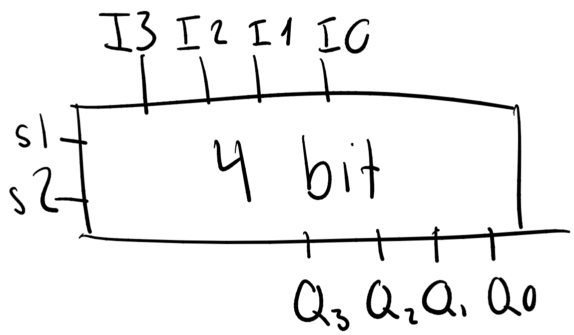
Digital Design Hwk 5

Thursday, November 21, 2019 2:27 PM

4.3 Design a 4-bit register with 2 control inputs s1 and s0; 4 data inputs I3, I2, I1, and I0; and 4 data outputs Q3, Q2, Q1, and Q0. When s1s0=00, the register maintains its value. When s1s0=01, the register loads I3...I0. When s1s0=10, the register clears itself to 0000. When s1s0=11, the register complements itself, so for example, 0000 would become 1111, and 1010 would become 0101. (*Component design problem.*)

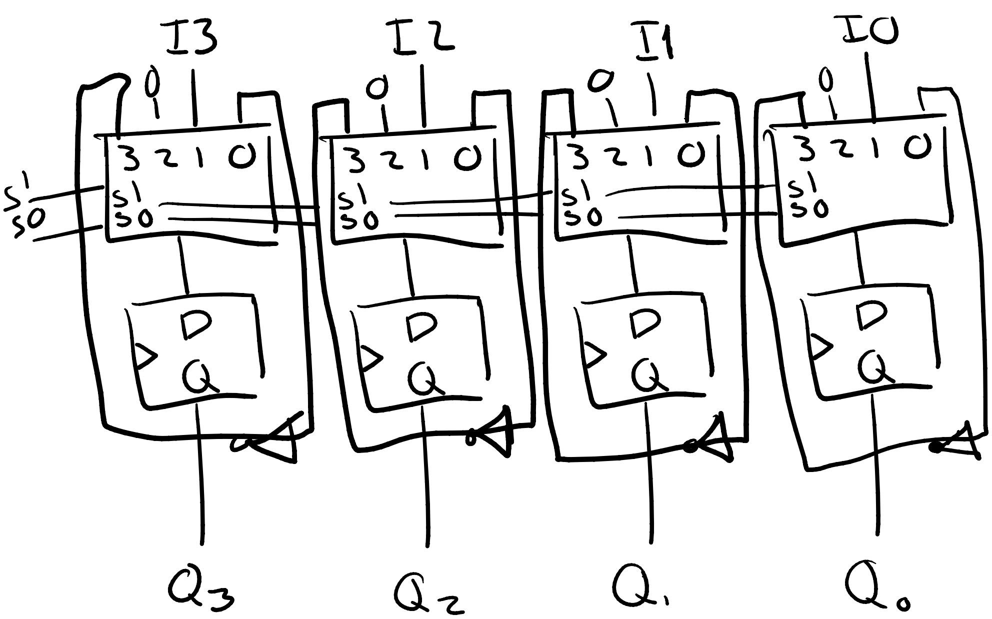


s1s0 = 00 Reg

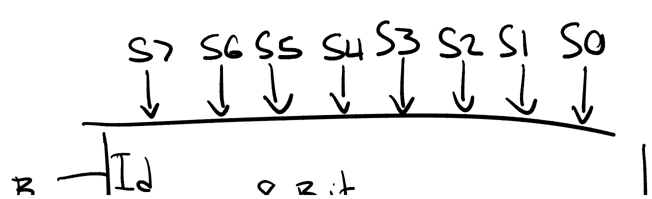
s1s0 = 01 Reg → I3...I0

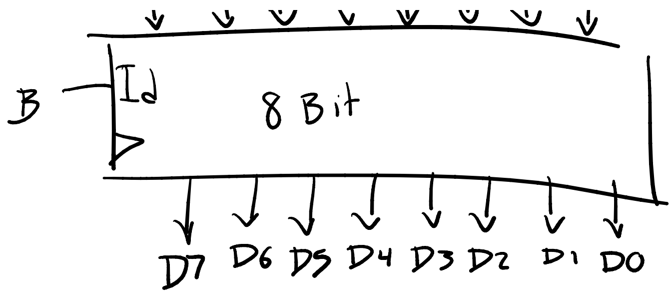
s1s0 = 10 Reg → 0000

s1s0 = 11 Reg If 0000 → 1111



4.6 The radar gun used by a police officer outputs a radar signal and measures the speed of cars as they pass. However, when an officer wants to ticket an individual for speeding, he must save the measured speed of the car on the radar unit. Build a system to implement a speed save feature for the radar gun. The system has an 8-bit speed input *S*, an input B from the save button on the radar gun, and an 8-bit output *D* that will be sent to the radar's gun speed display. (*Component use problem.*)





long time.

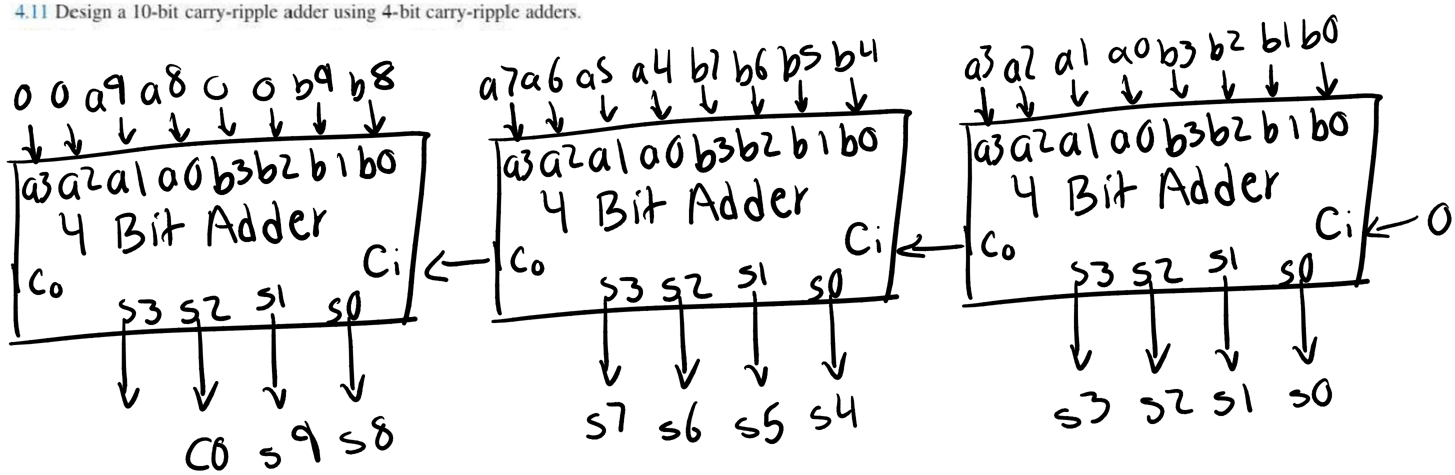
4.9 Assuming all gates have a delay of 1 ns, compute the longest time required to add two numbers using an 8-bit carry-ripple adder.

8 bit carry-ripple adder has 7 full adders and 1 half adder.

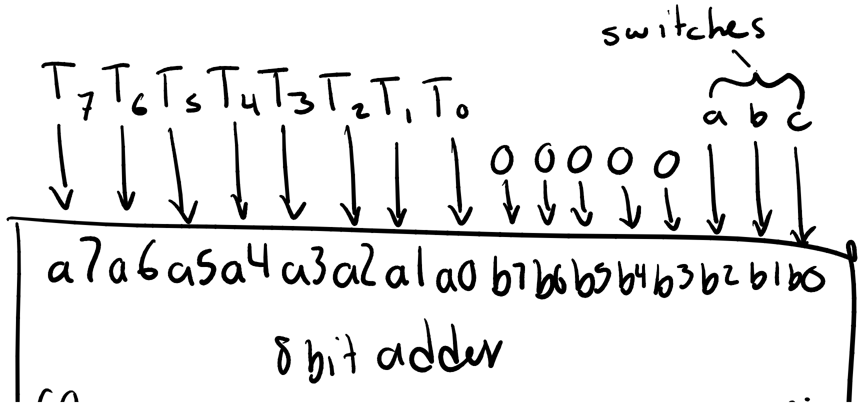
Full adder has 2 gate delays
half adder has 1 gate delay

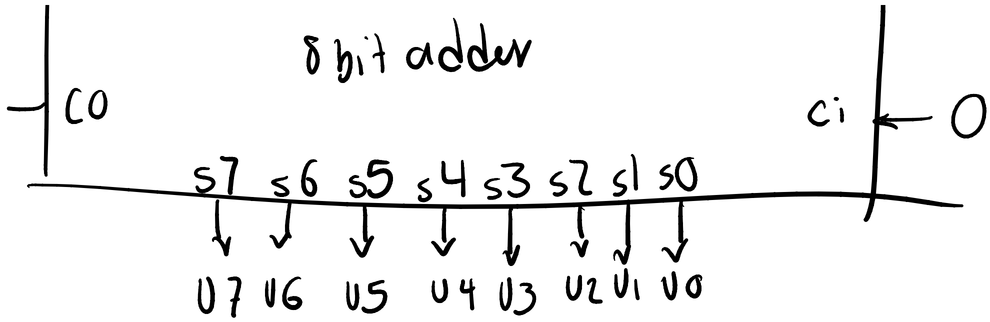
$((7\cdot 2)+(1\cdot 1)) = 15$ ns

4.11 Design a 10-bit carry-ripple adder using 4-bit carry-ripple adders.



4.14 Design a digital thermometer system that can compensate for errors in the temperature sensing device's output *T*, which is an 8-bit input to the system. The compensation amount can be positive only and comes to the system as a 3-bit binary number c, b, and a (a is the least significant bit), which come from a 3-pin DIP switch. The system should output the compensated temperature on an 8-bit output *U*. *(Component use problem.)*





problem.)

4.18 Design a 4-bit carry-ripple-style magnitude comparator that has two outputs, a greater-than or equal-to output `gte`, and a less-than or equal-to output `lte`. Be sure to clearly show the equations used in developing the individual 1-bit comparators and how they are connected to form the 4-bit circuit. *(Component design problem.)*

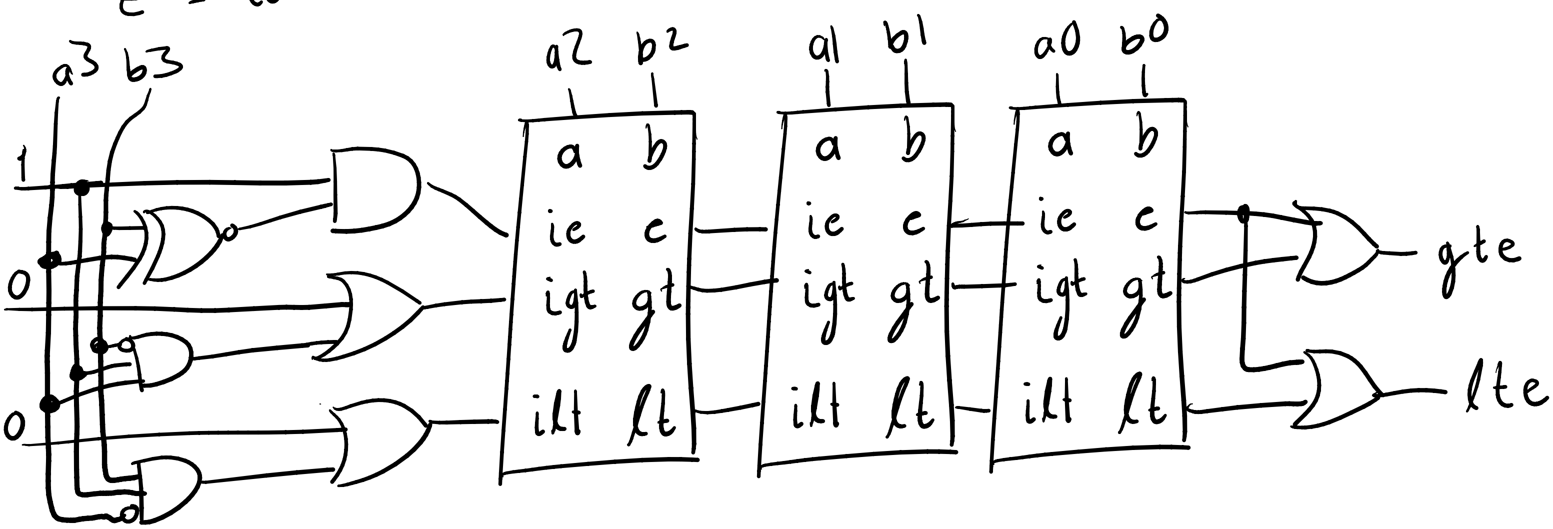
gte → $a \geq b$ $\qquad$ $a \cdot b' \rightarrow a > b$

lte → $a \leq b$ $\qquad$ $a' \cdot b \rightarrow a < b$

gt = igt + ((a XNOR b) $a \cdot b'$)

lt = ilt + ((a XNOR b) $a' \cdot b$)

e = ie · (a XNOR b)

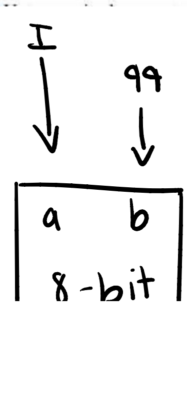


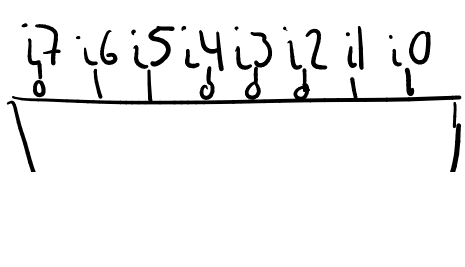
4.19 Design a circuit that outputs `1` if the circuit's 8-bit input equals 99:

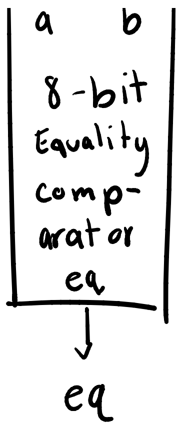
(a) using an equality comparator,

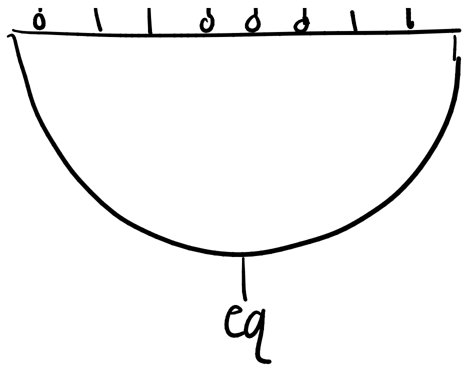
(b) using gates only.

Hint: In the case of (b), you need only 1 AND gate and some inverters. *(Component use problem.)*





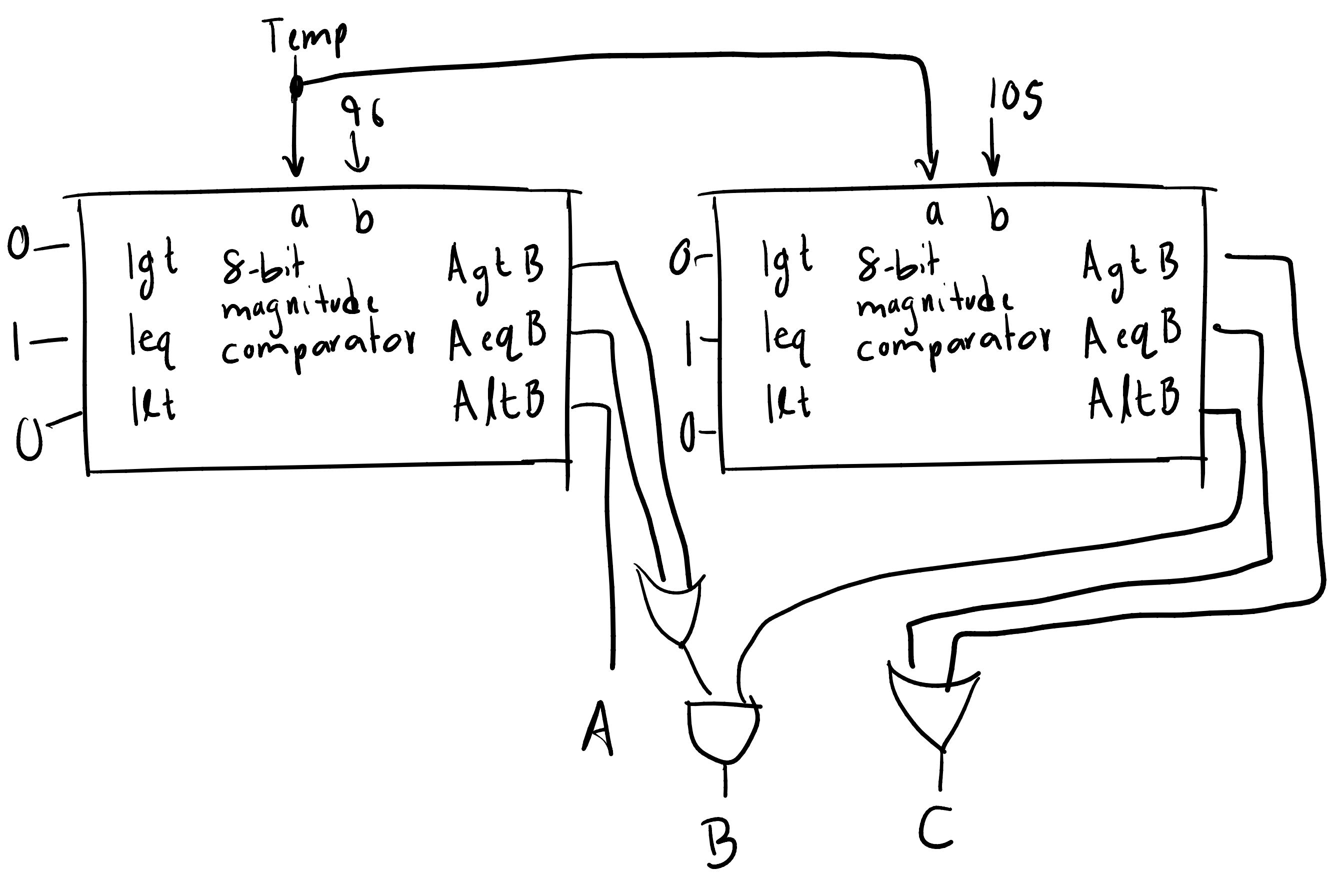




4.23 Design a human body temperature indicator system for a hospital bed. Your system takes an 8-bit input representing a person's body temperature, which can range from 0 to 255. If the measured temperature is 95 or less, set output A to 1. If the temperature is 96 to 104, set output B to 1. If the temperature is 105 or above, set output C to 1. Use 8-bit magnitude comparators and additional logic as required. *(Component use problem.)*

$A < 96 \qquad C \geq 105$

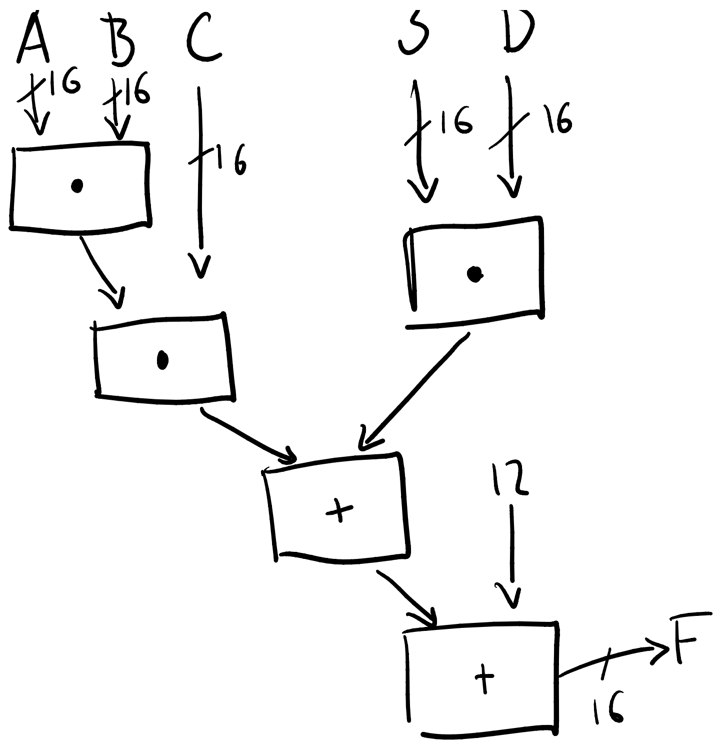
$96 \leq B < 105$



4.27 Design a circuit to compute $F=(A*B*C)+3*D+12$. A, B, C, and D are 16-bit inputs, and F is a 16-bit output. Use 16-bit multiplier and adder components, and ignore overflow issues.

A B C 3 D

/16 /16


A
B
C
S
D
16
16
16
16
16
12
16
F