University of Washington – Computer Science & Engineering
Spring 2022 Instructor: Mark Wyse 2022-05-31

CSE 369 QUIZ 3

Name: ______SOLUTIONS______________

Student ID Number: ______123456789_________

Please do not turn the page until 11:30.

Instructions

- This quiz contains 7 pages, including this cover page. You may use the backs of the pages for scratch work.
- Please clearly indicate (box, circle) your final answer.
- The quiz is closed book and closed notes.
- Please silence and put away all cell phones and other mobile or noise-making devices.
- Remove all hats, headphones, and watches.
- You have 60 minutes to complete this quiz.

Advice

- Read questions carefully before starting. Read *all* questions first and start where you feel the most confident to maximize the use of your time.
- There may be partial credit for incomplete answers; please show your work.
- Relax. You are here to learn.

Question	Points	Score
(1) Lightning Round	16	
(2) Analysis & Design	15	
(3) Module Design	15	
Total:	**46**	

Question 1: Lightning Round [16 pts]

(A) True or False: The number of rows in a truth table depends only on the number of inputs to the logic function. [1 pts]

True

(B) Name one of the 2-input "universal" logic gates discussed in class. [1 pts]

NAND or NOR

(C) A K-map is used to perform logic simplification and generates what type of two-level Boolean expression? [1 pts]

Sum of Products

(D) True or False: The input of a positive edge-triggered DFF must remain stable from the positive edge of the clock through the CLK-to-Q delay. [1 pts]

False

(E) True or False: In a Synchronous Digital System (SDS), we only need to know the setup time, hold time, and CLK-to-Q delay constants to ensure correct behavior. [1 pts]

False

(F) True or False: In a circuit where all inputs and outputs are registered, the *critical path* is the path with the longest delay between any two registers in the circuit. [1 pts]

True

(G) True or False: A synchronous reset signal causes the value of the register to be reset when the reset signal is asserted. [1 pts]

False

(H) Sequential logic (e.g., registers) timing must not violate the setup and hold time constraints within a clock period. Write out the inequality used to verify the hold and setup time constraints. [2 pts]

$$t_{hold} \leq t_{input,i} \leq t_{period} - t_{setup}$$

(I) Finite State Machines (FSMs), as studied in this class (Mealy machines) are defined by what three items? [3 pts]

- A set of *states*, S
- An *initial state*, S_0
- A *transition function* that maps from the current input and current state to the next state and output

(J) How many state bits are required to represent a state machine with *S* states? [1 pts]

ceiling(log2(S))

(K) What computational building block or routing element can be used to implement arbitrary N-input logic functions? [1 pts]

Mux / multiplexor

(L) An LFSR (Linear Feedback Shift Register) often uses a two-input logic gate to produce the feedback input in the shift register. How does the choice of logic gate and its input bits affect the output sequence of the LFSR? [2 pts]

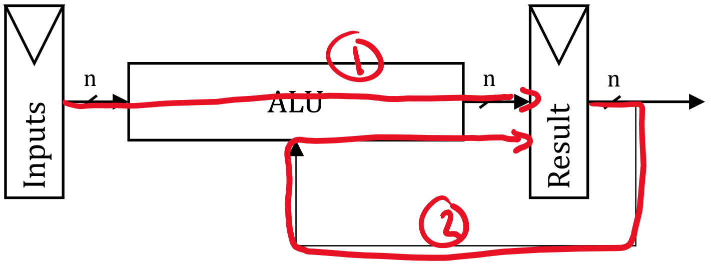
Sub-optimal choice results in an LFSR that produces a shorter and/or less random output sequence.

Question 2: Analysis & Design - TMR [15 pts]

Triple Modular Redundancy (TMR) is a common technique for making a circuit fault tolerant. A TMR-based circuit performs the same computation in three identical and independent circuits before outputting a result that is agreed upon by at least two of the three circuits (i.e., the majority result).

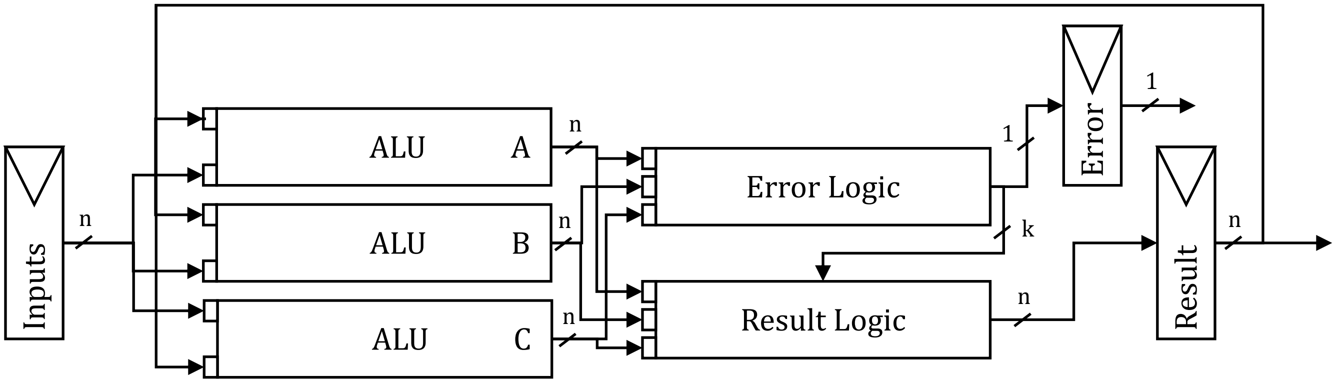
__Part 1__ What is the minimum clock period achievable by the non-TMR circuit below? The Inputs and Results blocks are n-bit registers. [3 pts]

Assume $t_{setup} = 10\ ns$, $t_{hold} = 5\ ns$, $t_{C2Q} = 8\ ns$, and $t_{ALU} = 50\ ns$.



68 ns = (8 + 10 + 50)

__Part 2__ The circuit diagram below shows the circuit above with TMR added. The Error Logic block takes n-bit ALU results A, B, and C as inputs and produces two outputs: a 1-bit Error signal that is captured by a register and a k-bit signal that is sent to the Result Logic block. The Result Logic block takes the k-bit signal from the Error Logic block and ALU results A, B, and C as inputs and produces the n-bit Result signal that is captured into a register.



What is the minimum clock period achievable by the TMR circuit above? The Inputs and Results blocks are registers. [4 pts]

Assume $t_{setup} = 10\ ns$, $t_{hold} = 5\ ns$, $t_{C2Q} = 8\ ns$, $t_{ALU} = 50\ ns$, $t_{Error} = 12\ ns$, $t_{Result} = 15\ ns$.

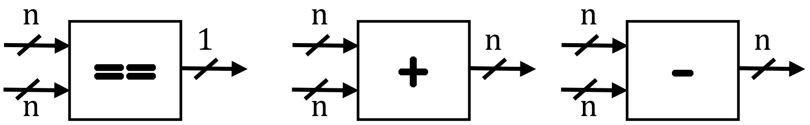
$t_{C2Q} + t_{ALU} + t_{Error} + t_{Result} + t_s$

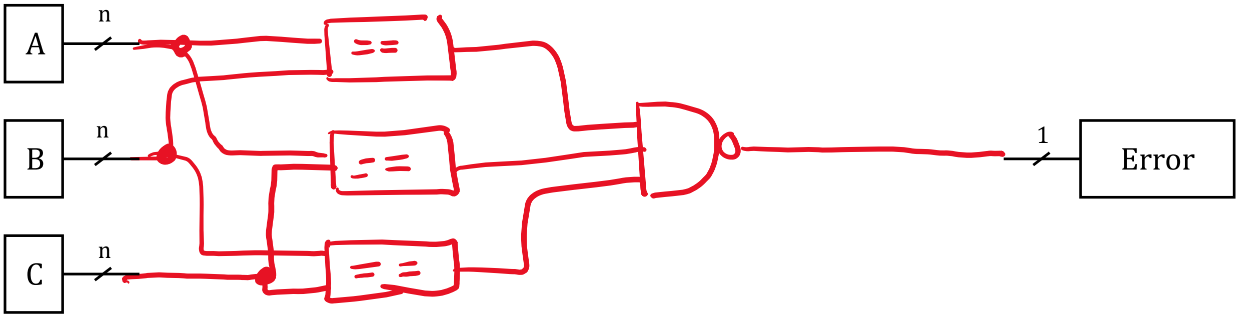
$= 8 + 50 + 12 + 15 + 10 = 95$

95 ns

<u>Part 3</u> The TMR-based design in Part 2 has an Error Logic block to detect a failure in one of the ALUs. The block takes ALU results A, B, and C as inputs and outputs a **1-bit Error signal**, which has a value of 1 if the result of any ALU does not match the result of any other ALU. Assume the ALUs produce **n-bit** results. ***Implement the Error Logic block's Error Bit functionality from the TMR circuit in Part 2 (draw the circuit diagram).*** [8 pts]

You can freely use 2:1 muxes, 1:2 demuxes, 2- or 3-input logic gates, the constants **0** and **1**, and the following logic blocks:





Question 3: Module Design – Up/Down Counter [15 pts]

An n-bit Up/Down Counter is a counter that supports both increment and decrement operations.

The Up/Down Counter has the following interface:

- ***up*** (input, 1-bit): increments counter by 1 when raised
- ***down*** (input, 1-bit): decrements counter by 1 when raised
- ***count*** (output, n-bits): current value of counter
- ***clock*** (input): clock signal for sequential logic
- ***reset*** (input, 1-bit): synchronous reset, resets counter to 0 when raised

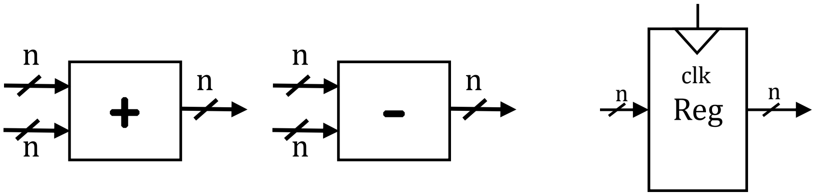
reset has priority over ***up*** and ***down***, and the ***up*** and ***down*** signals may be raised at the same time.

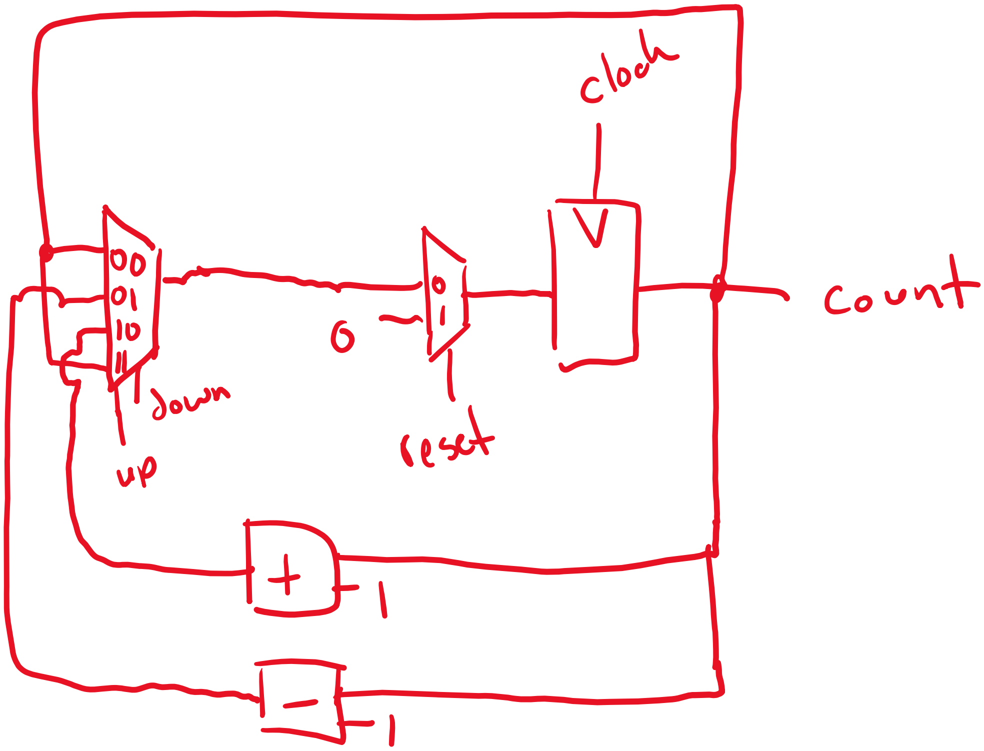
Part 1 Complete the following table describing the functionality of the Up/Down Counter. ***next count*** is the next value of the counter. The current count is given by the variable C. You may use the values 0, 1, and X (logical don't care), along with the variable C in expressions in the table. [5 pts]

Note: There may be more rows than are required to fully specify the functionality.

count	**reset**	**up**	**down**	**next count**
C	1	X	X	0
C	0	0	0	C
C	0	0	1	C-1
C	0	1	0	C+1
C	0	1	1	C
C				
C				
C				

<u>Part 2</u> ***Implement the Up/Down Counter described above (draw the circuit diagram)***. You can freely use 4:1 and 2:1 muxes, 1:2 and 1:4 demuxes, 2-input logic gates, the constants **0** and **1**, and the logic blocks given below. The Reg block is a collection of **n** 1-bit DFF's that share the same clock and collectively hold an n-bit vector. *Clearly label all five of the module's interface signals in your circuit diagram.* [10 pts]


n
n
+
n
n
n
-
n
clk
Reg
n
n


clock
00
01
10
11
down
up
0
0
1
reset
count
+
1
-
1