CSE 369 QUIZ 3

Name: _Perry_Perfect_____

UWNetID: _perfect_____

Please do not turn the page until 11:40.

Instructions

- This quiz contains 4 pages, including this cover page.
- Show scratch work for partial credit, but put your final answers in the boxes and blanks provided.
- 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 50 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) Decoders	12	12
(2) Shift Registers	10	10
(3) Luhn Algorithm	12	12
Total:	**34**	**34**

Question 1: Decoders [12 pts]

We are building a small **keypad-to-7seg decoder circuit** in two parts. The keypad is 4 buttons arranged in a square. The signals C_1, C_0, R_1, and R_0 are high (1) only if a button in the corresponding column or row is being pushed. The 2-bit bus K (K_1, K_0) represents which key is recognized (in binary). The Valid signal (V) is high when at least one key is being pressed.
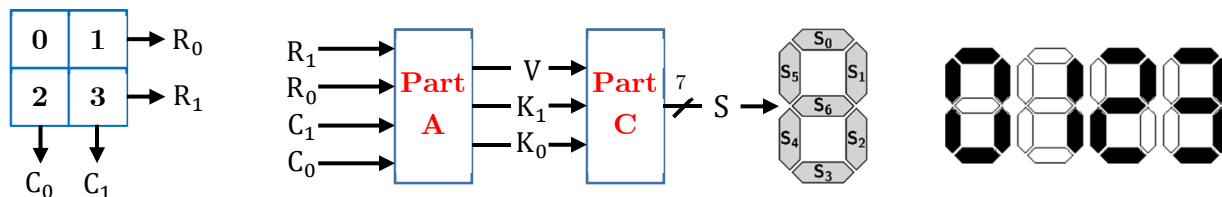
(A) Complete the truth table. We give priority to C_1 and R_1. [4 pt]

| R_1	R_0	C_1	C_0	V	K_1	K_0
0	0	0	0	0	X	X
0	0	0	1	0	X	X
0	0	1	0	0	X	X
0	0	1	1	0	X	X
0	1	0	0	0	X	X
0	1	0	1	1	0	0
0	1	1	0	1	0	1
0	1	1	1	1	0	1
1	0	0	0	0	X	X
1	0	0	1	1	1	0
1	0	1	0	1	1	1
1	0	1	1	1	1	1
1	1	0	0	0	X	X
1	1	0	1	1	1	0
1	1	1	0	1	1	1
1	1	1	1	1	1	1

(B) In the space below, solve for the minimal logical expression (using 2-input gates) for **V**. [4 pt]
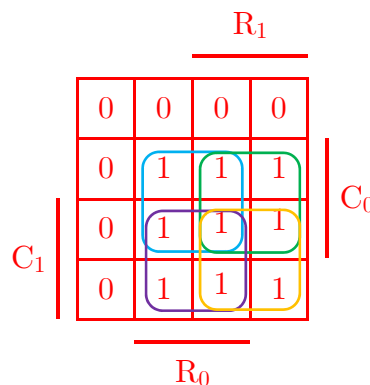
$$V = R_0 C_0 + R_1 C_0 + R_0 C_1 + R_1 C_1 = (R_0 + R_1)(C_0 + C_1)$$

(C) For the 7seg signal numbering and outputs shown above (lit/black = 1), draw the minimal logic for S_4 in terms of V, K_1, and K_0. All signals should be off when $V = 0$. [4 pt]

K_1	K_0	S_4
0	0	1
0	1	0
1	0	1
1	1	0
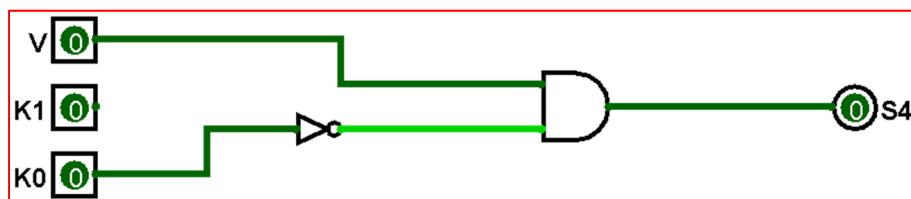
Question 2: Shift Registers [10 pts]

We are using a 3-bit LFSR as a pseudo-random number generator by connecting **q1** and **q0** to a 2-input NOR gate. Assume we start in state **011**.
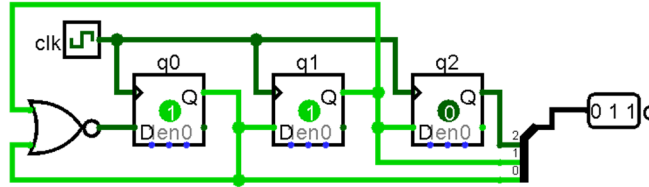
(A) <u>Circle one</u>: This LFSR (look at **Q**) is shifting bits to the **LEFT** / **RIGHT**. [1 pt]

(B) How many distinct states do we end up visiting? What is the length of the cycle we end up in? [4 pt]

$$011 \rightarrow 110 \rightarrow 100 \rightarrow 001 \rightarrow 010$$

States visited:	**5**
Cycle length:	**3**

(C) Modify the LFSR to add **Enable** and **Load** signals and draw the circuit diagram below using *logic gates* and *routing elements* discussed in class. Give priority to Load. [5 pt]

- Make sure you label the corresponding selector bits for ports of routing elements.
- Assume the clock inputs are connected properly for you.

3
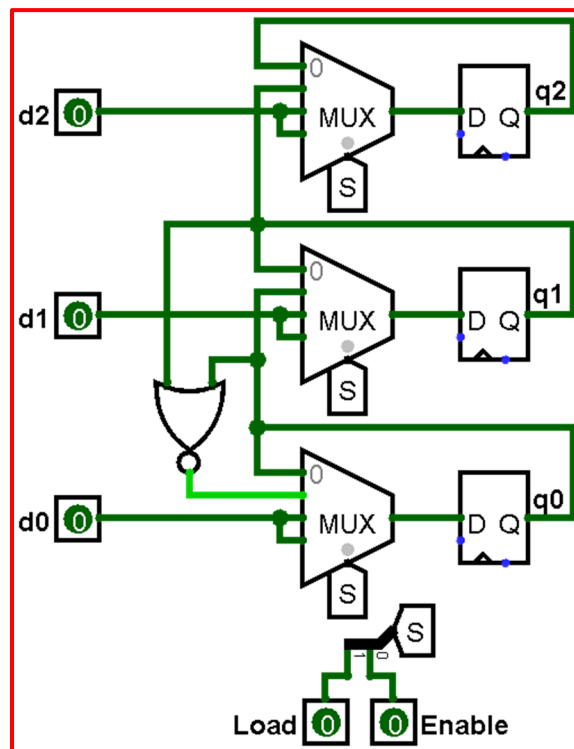
Question 3: Luhn Algorithm [12 pts]

The Luhn algorithm is a "mod 10" formula used for validation of decimal numbers (like credit cards) that uses a "check digit" in the rightmost position. The validation algorithm is as follows:

1. From right to left, double every *second* digit.
2. Sum all of the *digits*.
3. The number is valid if the sum % 10 = 0, invalid otherwise.

Example:

Digits:	$d_3 = 9$	$d_2 = 7$	$d_1 = 6$	check = 1
Double every other:	**18**	7	**12**	1
Sum digits:	**9**	7	**3**	1

= 20, so 9761 is *valid* ☑

Here we will implement a Luhn verifier for 4-digit numbers. You can freely add **registers**, **constants**, and the following **logic blocks** (with specified combinational delays):

$t_{add} = 50$ ps $t_{mult} = 90$ ps $t_{div} = 90$ ps $t_{shift} = 40$ ps
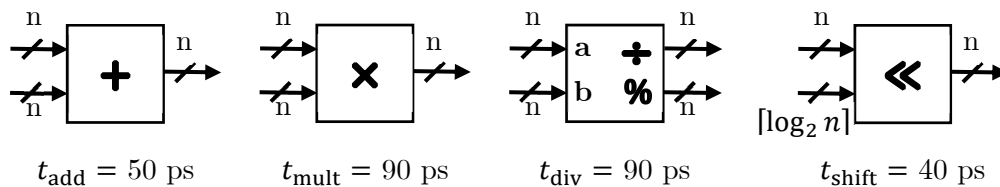
(A) Using these parts, complete the *fastest* 4-digit Luhn verification circuit below. Assume the clock inputs are connected properly for you. [7 pts]
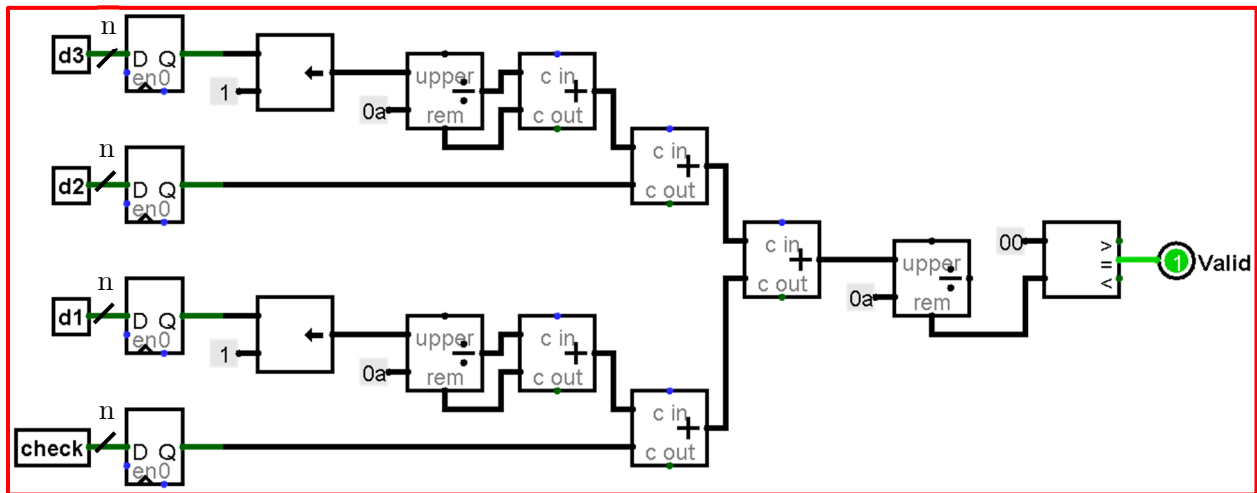
(B) What is the minimum value of the bus width **n** for this circuit? [2 pt]
max sum = 9+9+9+9 = 1+8+9+1+8+9 = 36, ceil($\log_2$ 36) = 6.

n_{min} = **6 bits**

(C) Assume $t_{hold} = 10$ ps, $t_{setup} = 40$ ps, $t_{C2Q} = 70$ ps and $t_{comparator} = 55$ ps. How long after a clock trigger does your circuit from Part A take to compute the final value of **Valid**? [3 pt]

$t_{C2Q} + t_{shift} + t_{div} + 3 * t_{add} + t_{div} + t_{comparator} = 495$.

495 ps

4