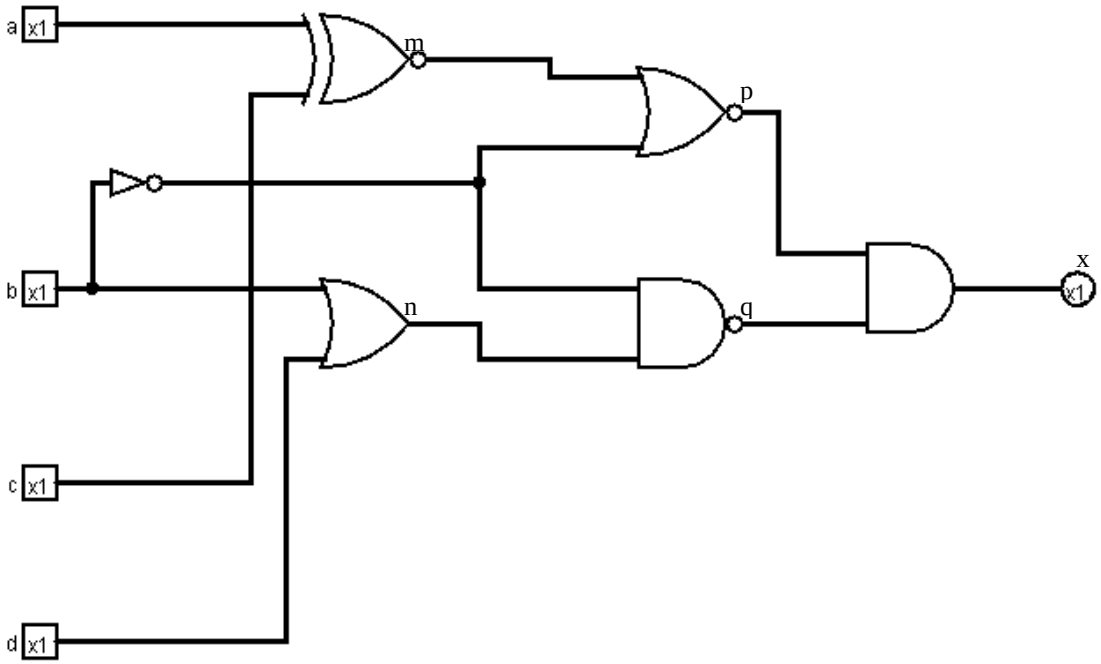
Autumn 2022, Homework 5 Key

Q1. (2 pts) Consider the logic gate circuit shown below.



Q1-1. To derive a Boolean equation for x. You don't need to simply the Boolean equation in this step.

x = ~(~(a XOR c) + ~b) * (~(~b(b + d)))

Grading: the result should be the same as this provided answer.
If the result does not match, -0.5.

Q1-2. Simplify the equation, using the Boolean algebra for output x. The result must be shown in the most simplified format.

x = ((a XOR c) * b) * (b + ~(b + d)) // de Mogen's law
= a XOR c * b * b + a XOR c * b * (~b * ~d) // b*~b must be 0
= a XOR c * b

Grading: the result should be the same as this provided answer. Or, the XOR part could be extended, i.e., the final result is: x = (a~c + ~ac)*b = ab~c + ~abc

If the result does not match, -0.5.
If the progress is not shown, even when the result is correct, -0.5.

Q2. (9pts) Design a combinational circuit system.

Design a combinational circuit with three inputs x, y, z, and three outputs A, B, C. Note that both xyz and ABC represent a three-bit binary number where x and A are the most significant bits, whereas z and C are the least significant bits. When a binary input xyz is 4 (100), 5 (101), 6 (110), or 7 (111), the binary output ABC is two less than the input xyz. When the binary input is 0 (000), 1 (001), 2 (010), or 3 (011), the binary output ABC is two more than the input xyz.
Please answer all of the following sub-questions.

Q2-1. (1pt) Draw a truth table of inputs xyz and outputs ABC.

Inputs			Outputs		
x	y	z	A	B	C
0	0	0	0	1	0
0	0	1	0	1	1
0	1	0	1	0	0
0	1	1	1	0	1
1	0	0	0	1	0
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	0	1

Grading: any mistake, -0.5.

Q2-2. (3pts) Based on the truth table you draw, build a Karnaugh map for each output of A, B, and C. In other words, you have to draw A's K-map, B's K-map, and C's K-map respectively.

A's K-map

z \ xy	00	01	11	10
0	0	1	1	0
1	0	1	1	0

B's K-map

z \ xy	00	01	11	10
0	1	0	0	1
1	1	0	0	1

C's K-map

z \ xy	00	01	11	10
0	0	0	0	0
1	1	1	1	1

Grading: for each table, any mistake, -0.5.

Q2-3. (3pts) Derive (as simple as possible) Boolean equations for A, B, and C using the Karnaugh maps.

A = y
B = ~y
C = z

Grading: for each equation, any mistake, -0.5.

Q2-4. (1pt) Based on the Boolean equations, draw the logical gate diagram (circuit) for this system in Logisim. **Copy and paste the circuit (or capture the diagram) in this submission. (No .circ file is necessary.)**

Grading: If the circuit is not attached, -1.
The correctness of the circuit is evaluated by the next question.

Q2-5. (1pt) Test your circuit with the Logisim simulation and generate the truth table (In Logisim, project-->analyze circuit-->table). **Copy and paste the table (or capture the table) in this submission.**

Grading: If the table is manually drawn, -1.

Q3. (9pts) Design a 2-bit by 2-bit multiplier.
In the class, we discussed about how to build a 2-bit by 2-bit multiplier using two half adders. (See slide deck 10. CpuComponentAluDesign.pptx). To eliminate a propagation delay with a ripple carry, design the same multiplier through KMaps, thus without cascading two half adders.

Q3-1. (1pt) Fill out the following truth table.

A	*	B	= C	A_1	A_0	B_1	B_0	C_3	C_2	C_1	C_0
0		0	0	0	0	0	0	0	0	0	0
0		1	0	0	0	0	1	0	0	0	0
0		2	0	0	0	1	0	0	0	0	0
0		3	0	0	0	1	1	0	0	0	0
1		0	0	0	1	0	0	0	0	0	0
1		1	1	0	1	0	1	0	0	0	1
1		2	2	0	1	1	0	0	0	1	0
1		3	3	0	1	1	1	0	0	1	1
2		0	0	1	0	0	0	0	0	0	0
2		1	2	1	0	0	1	0	0	1	0
2		2	4	1	0	1	0	0	1	0	0
2		3	6	1	0	1	1	0	1	1	0
3		0	0	1	1	0	0	0	0	0	0
3		1	3	1	1	0	1	0	0	1	1
3		2	6	1	1	1	0	0	1	1	0
3		3	9	1	1	1	1	1	0	0	1

Grading: any mistake in this table, -0.5.

Q3-2. (2pts) From the truth table, derive the Boolean equations, each representing C_0 and C_3
Their Boolean equations are too simple to use a KMap.

$C_0 = \sim A_1A_0\sim B_1B_0 + \sim A_1A_0B_1B_0 + A_1A_0\sim B_1B_0 + A_1A_0B_1B_0$, or, $C_0 = A_0\ B_0$
$C_3 = A_1A_0B_1B_0$

Grading: any mistake in each equation, -0.5.

Q3-3. (2pt) Draw a KMap for C2. Derive the Boolean equation to represent C2.

A_1A_0 \ B_1B_0	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	0	0	0	1
10	0	0	1	1

$C_2 = A_1{\sim}A_0B_1 + A_1B_1{\sim}B_0 = A_1\ B_1\ (\sim A_0 + \sim B_0)$
Or, $C_2 = A_1\ B_1\ \sim(A_0B_0)$

Grading: any mistake in the table, -0.5. Any mistake in the equation, -0.5.

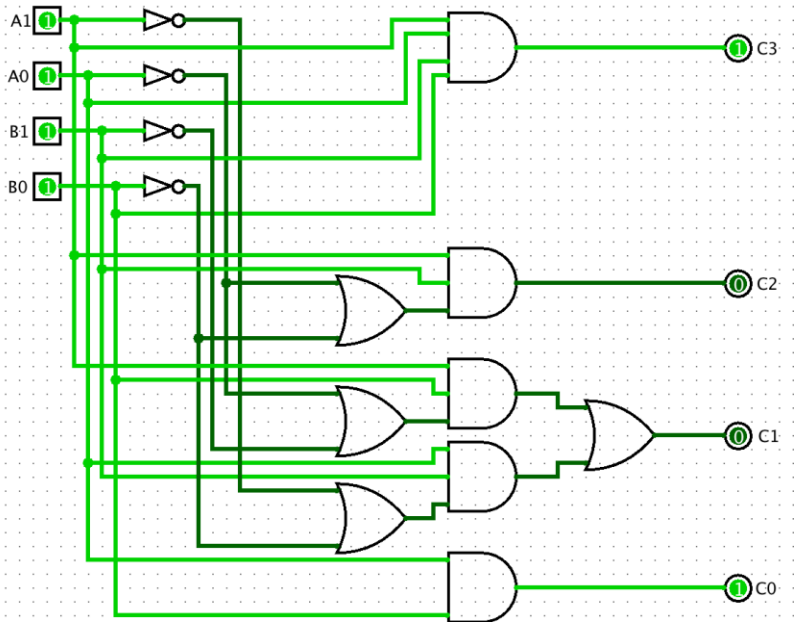
Q3-4. (2pt) Draw a KMap for C1. Derive the Boolean equation to represent C1.

A_1A_0 \ B_1B_0	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	0	1	0	1
10	0	1	1	0

$C_1 = A_1{\sim}B_1B_0 + A_1{\sim}A_0B_0 + {\sim}A_1A_0B_1 + A_0B_1{\sim}B_0$
$= A_1\ B_0(\sim A_0 + \sim B_1) + A_0B_1(\sim A_1 + \sim B_0)$
Or, $C_1 = A_1\ B_0{\sim}(A_0B_1) + A_0B_1{\sim}(A_1B_0) = (A_0B_1)$ XOR (A_1B_0)

Grading: any mistake in the table, -0.5. Any mistake in the equation, -0.5.

Q3-5. (1pt) Based on the Boolean equations, draw this multiplier logic in Logisim. **Copy and paste the circuit (or capture the diagram) in your submission. (No .circ file is necessary.)**



Grading: If the circuit is not attached, -1.
The correctness of the circuit is evaluated by the next question.

Q3-6. (1pt) Test your circuit with the Logisim simulation and generate the truth table. **Copy and paste the table (or capture the table) in your submission.**

a	b	c	d	x	y	z	u
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	1	0	0	0	1
0	1	1	0	0	0	1	0
0	1	1	1	0	0	1	1
1	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0
1	0	1	0	0	1	0	0
1	0	1	1	0	1	1	0
1	1	0	0	0	0	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	1	1	0
1	1	1	1	1	0	0	1

Grading: If the table is manually drawn, -1. Any mistake in this table, -0.5.