

CSE140 Homework 2, Due 11:59PM, Mon 4/22/19

Homework 2 covers the CMOS design, combinational logic specification, and implementation. For the first problem, we construct a gate using CMOS designs. For the next three problems, we practice more on the specification, in particular, when the number of input bits is not small, and the implementation may take multiple levels of logic gates. For the last two problems, we practice design minimization using Karnaugh maps in sum of products and product of sums formats.

1 CMOS Designs

Design a CMOS gate with function $y(a, b, c, d) = [a(b + cd)]'$. Draw the schematic diagram using NMOS and PMOS transistors as the basic elements.

- 20 points for correct CMOS design
- 2 points deducted for each minor error

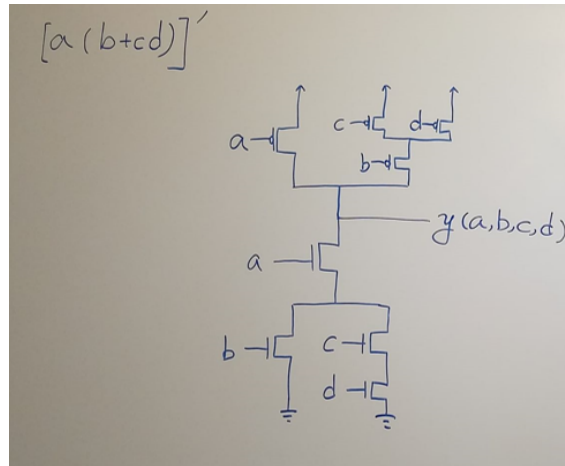


Figure 1: (i) Circuit Diagram

2 Voting Machine

A voting machine reads six binary bits $(a_5, a_4, a_3, a_2, a_1, a_0)$ and outputs a binary number (y_2, y_1, y_0) so that the output represents the number of input bits which are 1 instead of 0. For example, when input $(a_5, a_4, a_3, a_2, a_1, a_0) = (1, 1, 1, 1, 1, 0)$, output $(y_2, y_1, y_0) = (1, 0, 1)$. When input $(a_5, a_4, a_3, a_2, a_1, a_0) = (0, 1, 1, 0, 0, 1)$, output $(y_2, y_1, y_0) = (0, 1, 1)$.

- 10 points, graded by completion

A. Write the truth table of the machine.

a_5	a_4	a_3	a_2	a_1	a_0	y_2	y_1	y_0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	1
0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	0	1
0	1	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	1
0	0	0	0	1	1	0	1	0
0	0	0	1	0	1	0	1	0
0	0	1	0	0	1	0	1	0
0	1	0	0	0	1	0	1	0
1	0	0	0	0	1	0	1	0
0	0	0	1	1	0	0	1	0
0	0	1	0	1	0	0	1	0
0	1	0	0	1	0	0	1	0
1	0	0	0	1	0	0	1	0
0	0	1	1	0	0	0	1	0
0	1	0	1	0	0	0	1	0
1	0	0	1	0	0	0	1	0
0	1	1	0	0	0	0	1	0
1	0	1	0	0	0	0	1	0
1	1	0	0	0	0	0	1	0

a_5	a_4	a_3	a_2	a_1	a_0	y_2	y_1	y_0
0	0	0	1	1	1	0	1	1
0	0	1	0	1	1	0	1	1
0	1	0	0	1	1	0	1	1
1	0	0	0	1	1	0	1	1
0	0	1	1	0	1	0	1	1
0	1	0	1	0	1	0	1	1
1	0	0	1	0	1	0	1	1
0	1	1	0	0	1	0	1	1
1	0	1	0	0	1	0	1	1
1	1	0	0	0	1	0	1	1
0	0	1	1	1	0	0	1	1
0	1	0	1	1	0	0	1	1
1	0	0	1	1	0	0	1	1
0	1	1	0	1	0	0	1	1
1	0	1	0	1	0	0	1	1
1	1	0	0	1	0	0	1	1
0	1	1	1	0	0	0	1	1
1	0	1	1	0	0	0	1	1
1	1	0	1	0	0	0	1	1
1	1	1	0	0	0	0	1	1

a_5	a_4	a_3	a_2	a_1	a_0	y_2	y_1	y_0
0	0	1	1	1	1	1	0	0
0	1	0	1	1	1	1	0	0
1	0	0	1	1	1	1	0	0
0	1	1	0	1	1	1	0	0
1	0	1	0	1	1	1	0	0
1	1	0	0	1	1	1	0	0
0	1	1	1	0	1	1	0	0
1	0	1	1	0	1	1	0	0
1	1	0	1	0	1	1	0	0
1	1	1	0	0	1	1	0	0
0	1	1	1	1	0	1	0	0
1	1	1	0	1	0	1	0	0
1	1	0	1	1	0	1	0	0
1	0	1	1	1	0	1	0	0
1	1	1	1	0	0	1	0	0

a_5	a_4	a_3	a_2	a_1	a_0	y_2	y_1	y_0
0	1	1	1	1	1	1	0	1
1	0	1	1	1	1	1	0	1
1	1	0	1	1	1	1	0	1
1	1	1	0	1	1	1	0	1
1	1	1	1	0	1	1	0	1
1	1	1	1	1	0	1	0	1
1	1	1	1	1	1	1	1	0

B. Express the function in a minimal Boolean expression.

- $y_0 = a_5 \oplus a_4 \oplus a_3 \oplus a_2 \oplus a_1 \oplus a_0$
- $y_1 = (a_5 a_4 + a_3(a_5 \oplus a_4)) \oplus (a_2 a_1 + a_0(a_2 \oplus a_1)) \oplus ((a_5 \oplus a_4 \oplus a_3)(a_2 \oplus a_1 \oplus a_0))$
- $y_2 = (a_5 a_4 + a_3(a_5 \oplus a_4))(a_2 a_1 + a_0(a_2 \oplus a_1)) + ((a_5 \oplus a_4 \oplus a_3)(a_2 \oplus a_1 \oplus a_0))((a_5 a_4 + a_3(a_5 \oplus a_4)) \oplus (a_2 a_1 + a_0(a_2 \oplus a_1)))$

C. Use a minimal number of full adders (no other gates) to implement the machine. Depict your design with a schematic diagram.

3 Majority Function

A majority function f reads five binary bits (a_4, a_3, a_2, a_1, a_0) and produces a binary output $f = 1$ when the majority of the input bits are true, otherwise the output $f = 0$. For example, $f(1, 1, 1, 0, 0) = 1$, and $f(1, 1, 0, 0, 0) = 0$.

- 10 points, graded by completion

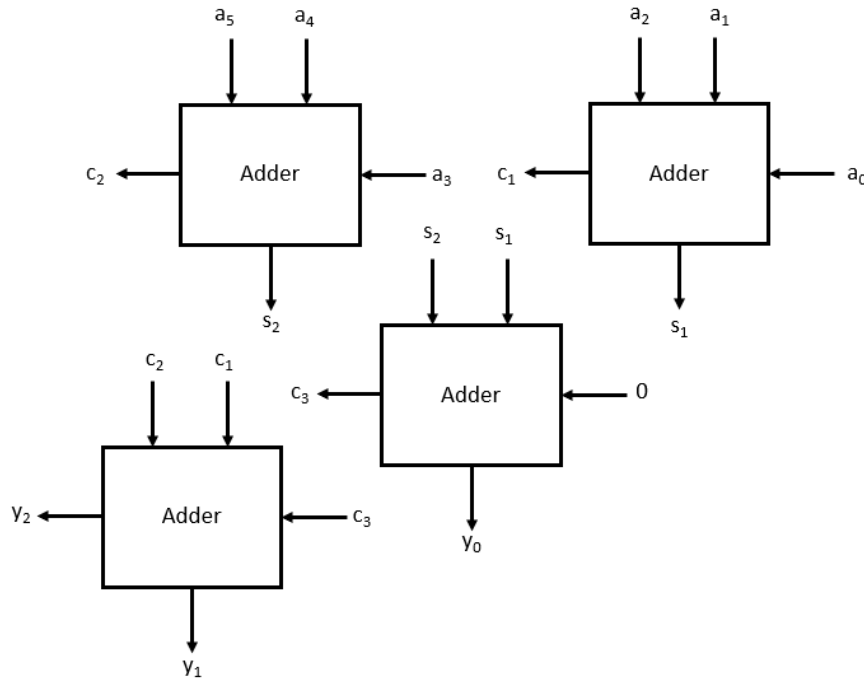


Figure 2: Voting Machine Schematic Diagram

A. Express the function in a minimal Boolean expression.

$$f(a_4, a_3, a_2, a_1, a_0) = a_4a_3a_2 + a_4a_3a_1 + a_4a_3a_0 + a_4a_2a_1 + a_4a_2a_0 + a_4a_1a_0 + a_3a_2a_1 + a_3a_2a_0 + a_3a_1a_0 + a_2a_1a_0$$

B. Implement the function using a minimal number of full adders (no other gates). Depict your design with a schematic diagram.

- Inputs (c_2, c_1, s_2) of the third full adder (Figure 4) represent (2,2,1) votes. Thus, any two of the input bits equal to 1 will have the output of carry-out bit being 1.

4 Priority Encoder

A priority encoder reads a vector of eight binary bits $(a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0)$ and produces a binary number (d_2, d_1, d_0) that presents the lowest index of the input bit which is true. For example, when $(a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0) = (0, 0, 1, 0, 1, 0, 0, 0)$, the output $(d_2, d_1, d_0) = (0, 1, 1)$; $(a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0) = (0, 1, 0, 0, 0, 0, 0, 0)$, the output $(d_2, d_1, d_0) = (1, 1, 0)$; and when $(a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0) = (0, 0, 0, 0, 0, 0, 0, 1)$, the output $(d_2, d_1, d_0) = (0, 0, 0)$. However, when none of the input bits asserts a true value, i.e. $(a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0) = (0, 0, 0, 0, 0, 0, 0, 0)$, the output remains to be $(d_2, d_1, d_0) = (0, 0, 0)$.

- 10 points, graded by completion

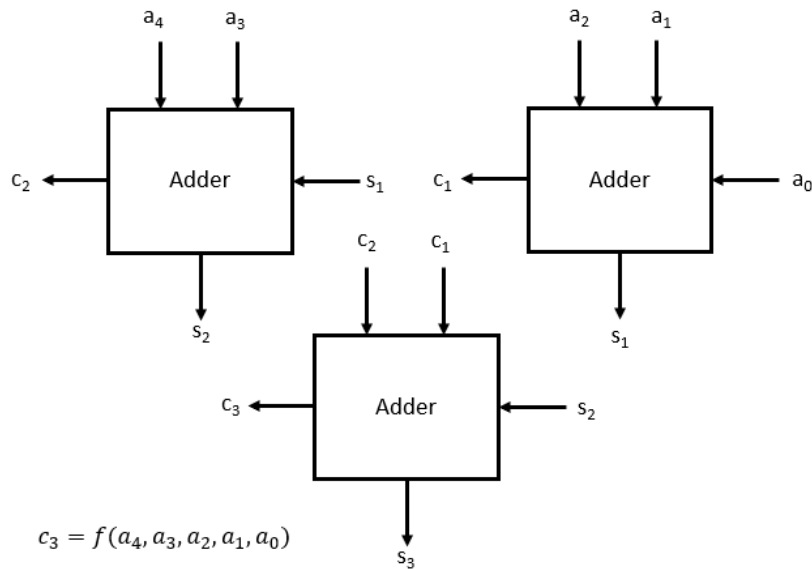


Figure 3: Majority Function Schematic Diagram

A. Derive a Boolean expression of the three output bits (no need to be minimal but try to be concise).

- $d_2 = a'_0 a'_1 a'_2 a'_3 (a_4 + a_5 + a_6 + a_7)$
- $d_1 = a'_0 a'_1 (a_2 + a_3) + a'_0 a'_1 a'_2 a'_3 a'_4 a'_5 (a_6 + a_7)$
- $d_0 = a'_0 (a_1 + a'_2 a_3) + a'_0 a'_1 a'_2 a'_3 a'_4 (a_5 + a'_6 a_7)$

B. Implement the encoder with AND, OR, NOT gates (no need to be minimal but try to be organized).

5 Minimal Sum of Products Expression

Implementation from truth table to sum of products expressions.

1. Use Karnaugh map to simplify function

$$f(a, b, c) = \sum m(0, 4, 5, 7) + \sum d(1, 2).$$

List **all** possible minimal two-level sum of products expressions. Show the switching functions. No need for the schematic diagram.

- 20 points for correct answer
- 2 points deducted for each wrong term

Solution:

- $f(a, b, c) = b' + ac$

C_1		C_2S_2		0	1
				0	1
	00			0	0
	01			0	1
	11			1	1
	10			0	1

Figure 4: K-map of Carry-Out Bit

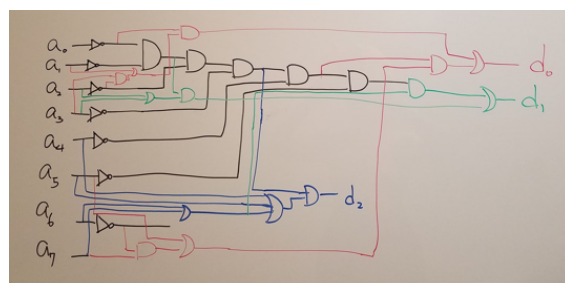


Figure 5: Priority Encoder

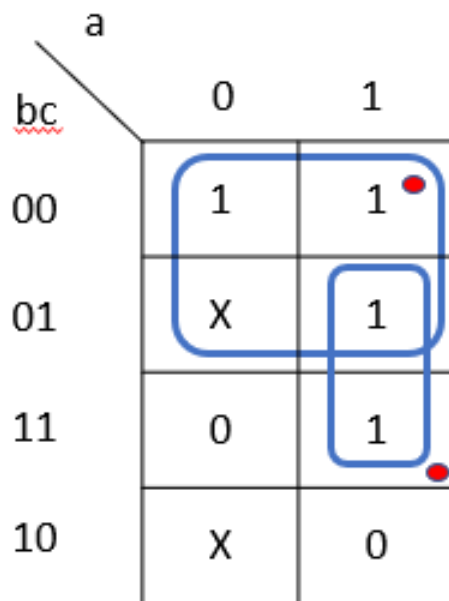


Figure 6: 5.1

2. Use Karnaugh map to simplify function

$$f(a, b, c, d) = \sum m(2, 3, 5, 6, 8, 11, 15) + \sum d(0, 9, 12).$$

List all possible minimal two-level sum of products expressions. Show the switching functions. No need for the schematic diagram.

- 2.5 points, graded by completion

Solution:

- $f(a, b, c, d) = a'bc'd + acd + a'cd' + a'b'c + b'c'd'$
- $f(a, b, c, d) = a'bc'd + acd + a'cd' + a'b'c + ac'd'$
- $f(a, b, c, d) = a'bc'd + acd + a'cd' + a'b'c + ab'c'$
- $f(a, b, c, d) = a'bc'd + acd + a'cd' + b'cd + b'c'd'$
- $f(a, b, c, d) = a'bc'd + acd + a'cd' + b'cd + ac'd'$
- $f(a, b, c, d) = a'bc'd + acd + a'cd' + b'cd + ab'c'$

3. Use Karnaugh map to simplify function

$$f(a, b, c, d, e) = \sum m(6, 7, 8, 10, 12, 13, 15, 16, 21, 22, 24, 26, 31) + \sum d(0, 2, 9, 18, 28, 29)$$

Write **one** minimal two-level sum of products expression. Show the switching functions. No need for the schematic diagram.

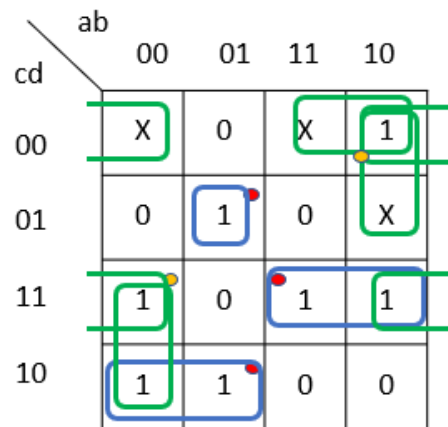


Figure 7: 5.2

- 2.5 points, graded by completion

Solution:

- $f(a, b, c, d, e) = c'e' + acd'e + b'de' + bce + a'cde + bcd'$
- $f(a, b, c, d, e) = c'e' + acd'e + b'de' + bce + a'cde + bd'e'$
- $f(a, b, c, d, e) = c'e' + acd'e + b'de' + bce + a'b'cd + bcd'$
- $f(a, b, c, d, e) = c'e' + acd'e + b'de' + bce + a'b'cd + bd'e'$

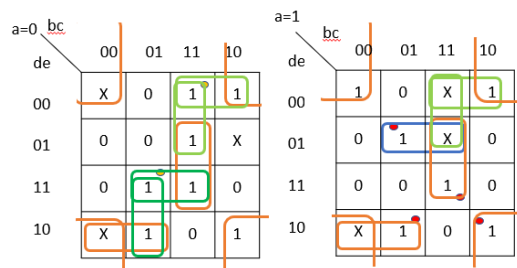


Figure 8: 5.3

6 Minimal Product of Sums Expression

Implementation from truth table to product of sums expressions.

1. Use Karnaugh map to simplify function

$$f(a, b, c) = \sum m(0, 7) + \sum d(1, 2, 5).$$

List all possible minimal two-level product of sums expressions. Show the switching functions. No need for the schematic diagram.

- 20 points for correct answer
- 2 points deducted for each wrong term

Solution:

- $f(a, b, c) = (a' + c)(a + c')$
- $f(a, b, c) = (a' + c)(a + b')$

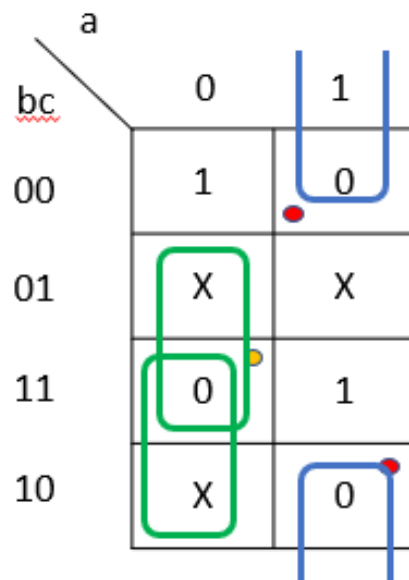


Figure 9: 6.1

2. Use Karnaugh map to simplify function

$$f(a, b, c, d) = \sum m(1, 2, 4, 9, 12, 15) + \sum d(3, 11, 14).$$

List all possible minimal two-level product of sums expressions. Show the switching functions. No need for the schematic diagram.

- 2.5 points, graded by completion

Solution:

- $f(a, b, c, d) = (b + c + d)(b' + c + d')(a + b' + c')(a' + c' + d)$
- $f(a, b, c, d) = (b + c + d)(b' + c + d')(a + b' + c')(a' + b + c')$
- $f(a, b, c, d) = (b + c + d)(b' + c + d')(a + b' + c')(a' + b + d)$

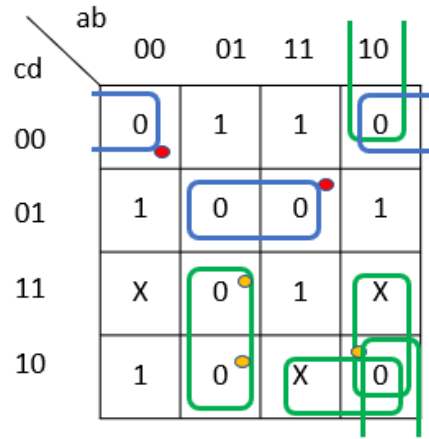


Figure 10: 6.2

3. Use Karnaugh map to simplify function

$$f(a, b, c, d, e) = \sum m(1, 3, 4, 11, 12, 13, 14, 19, 20, 22, 27, 28, 30) + \sum d(7, 9, 23, 24, 26).$$

Write **one** minimal two-level product of sums expression. Show the switching functions. No need for the schematic diagram.

- 2.5 points, graded by completion

Solution:

- $f(a, b, c, d, e) = (c + e)(a' + d + e')(b + c' + e')(c' + d' + e')(a + b + d' + e)$
- $f(a, b, c, d, e) = (c + e)(a' + d + e')(b + c' + e')(c' + d' + e')(a + b + c' + d')$

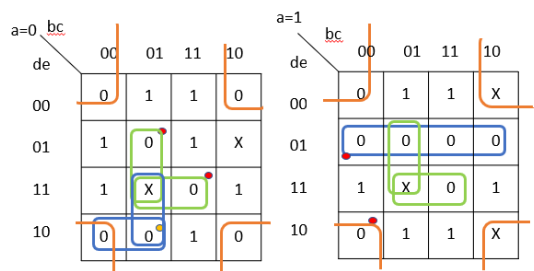


Figure 11: 6.3