

Midterm 1

Daniel Ball

March 27, 2020

Chapter 1

1. A duty cycle can be calculate by dividing the pulse width by the period itself and multiplying that by 100 percent. For the example above the pulse width is.

(a) The pulse width for D_1 seems to be 1 microsec and its period is 2 microseconds. Now if we were to plug it into the duty cycle formula we would get

$$\frac{1}{2} * 100\% = 50\%$$

The duty cycle for D_1 is 50%.

(b) The pulse width for D_2 seems to be 2 microsecs and its period is 4 microseconds. Now if we were to plug it into the duty cycle formula we would get

$$\frac{2}{4} * 100\% = 50\%$$

The duty cycle for D_2 is 50%.

(c) The pulse width for D_3 seems to be 4 microsec and its period is 8 microseconds. Now if we were to plug it into the duty cycle formula we would get

$$\frac{4}{8} * 100\% = 50\%$$

The duty cycle for D_3 is 50%.

(d) The pulse width for D_1 seems to be 1 microsec and its period is 2 microseconds. Now if we were to plug it into the duty cycle formula we would get

$$\frac{8}{16} * 100\% = 50\%$$

The duty cycle for D_3 is 50%.

It is shown that all of the signals have the same duty cycle of 50%. We can also see that each signal is sending out the number 2^n . For example the signal D_0 is sending the binary number 1 every half period. We can see that similar pattern in signal D_1 which is continuously sending out the binary number 2.

2. I added all of the drawn pictures to the back sense I am writing this midterm on Latex. I tried to learn how to draw timing diagrams on Latex and it was confusing me too much. I am sorry for the inconvenience.

3. Since the circuit is in parallel The bitrate will take just as long as the clock frequency. Therefore we can say that the bitrate for the parallel circuit would be $\frac{1}{4MHz} = 4$ microsec.

Now if this were serial we would only be using one line and have to multiple the period we found for the parallel circuit by the amount of bits we are sending though. Therefore the bitrate of a serial circuit would be

$$4(4) = 16\text{microsec}$$

Chapter 2

1. Convert to Binary
 - (a) 1024 (decimal) -> 1000000000
 - (b) 0xB BBBB (hex) -> 101101110111011
 - (c) -2048 (decimal) -> 110000000000 (with a sign bit at the beginning to make the number negative.)
2. Convert to Hex
 - (a) 65535 (decimal) -> 0xFFFF
 - (b) 100010001000 (hex) -> 0X8888
3. Convert to Octal
 - (a) 1024 (decimal) -> 2000
4. Consider the gray code encoder
 - (a) If the shaft rotated 180 degrees it would go through half of its maximum bit changes. Which in this instance would be 8 bit changes.
 - (b) If the coder started at 0000 and rotated 180 degrees the final grey code would be 1100.
 - (c) the angular precision for a 4 bit gray code would be $4/360 = .01$
 - (d) for an 8 bit grey code the angular precision would be $8/360 = .02$.

Chapter 3

1. The gate on the left has the same truth table as an XNOR gate therefore it can be simplified to a single XNOR gate. The simplified logic expression would be

$$(AB) + ((not)A + (not)B)$$

A	B	X
0	0	1
1	0	0
0	1	0
1	1	1

2. Please refer to the last page for the drawn pictures.

3. Will be drawn on last page.

Chapter 4

1. Suppose an investment firm holds stock shares in four different stocks within a portfolio, labeled A through D. The companies corresponding to stocks A through D are labeled inactive or active by the firm, based on information about their productivity. The firm notices that the portfolio output is on (rising) under the following conditions:
 - (a) $X = ((not)A(not)B(not)C(not)D) + ((not)A(not)B(not)C(D)) + (ABCD) + (ABC(not)D)$
 - (b) Drawn in the back
 - (c) Looking at the Karnaugh map it seems as though it does not matter whether account D is active or not. It can be removed from the equation entirely.
2. A circuit contains three main branches leading to one output. The output is observed to fail under the following conditions below
 - (a) Karnaugh map drawn on the back.

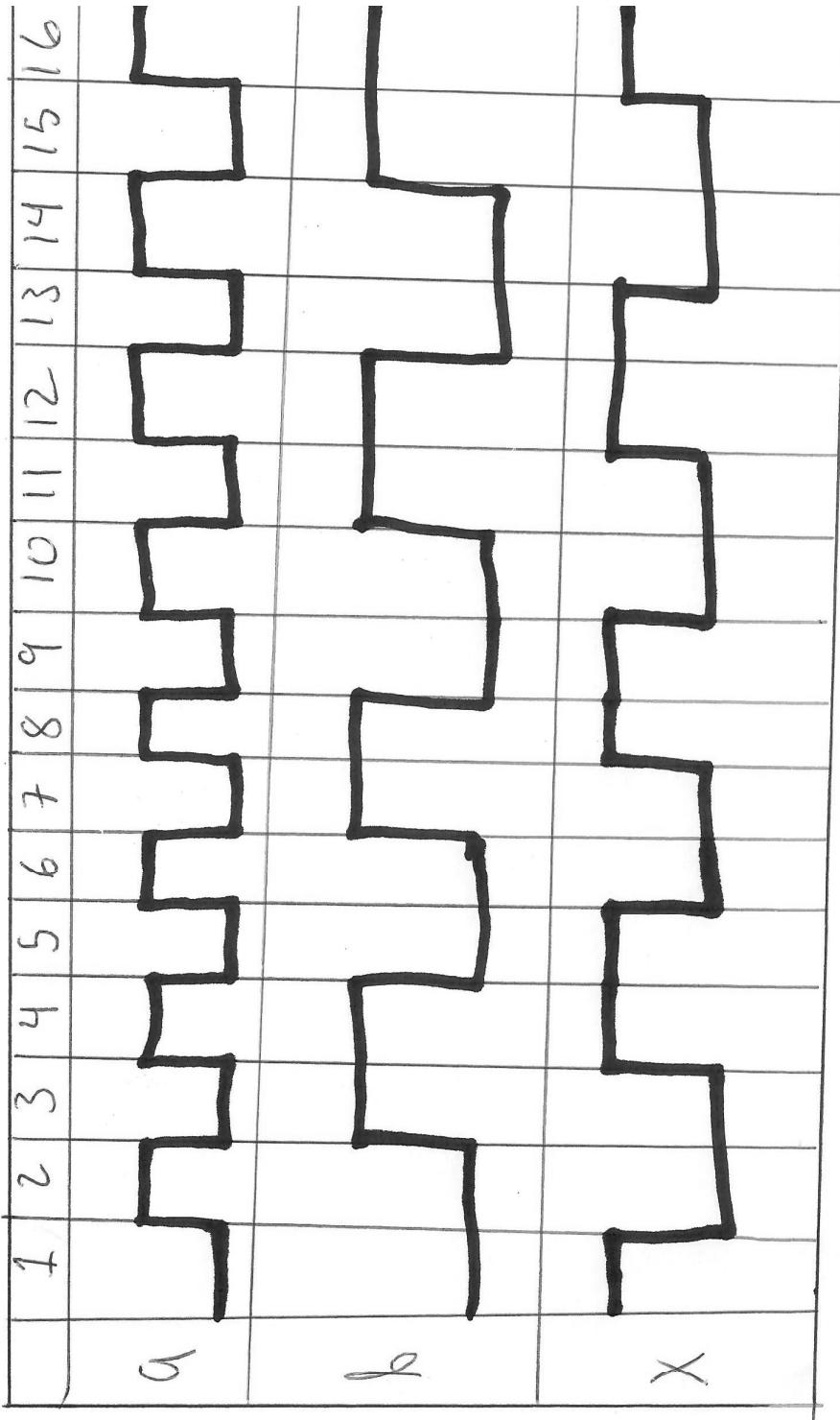
$$X = ((not)A(not)BC) + (A(not)BC)$$

- (b) Circuit drawn on the back

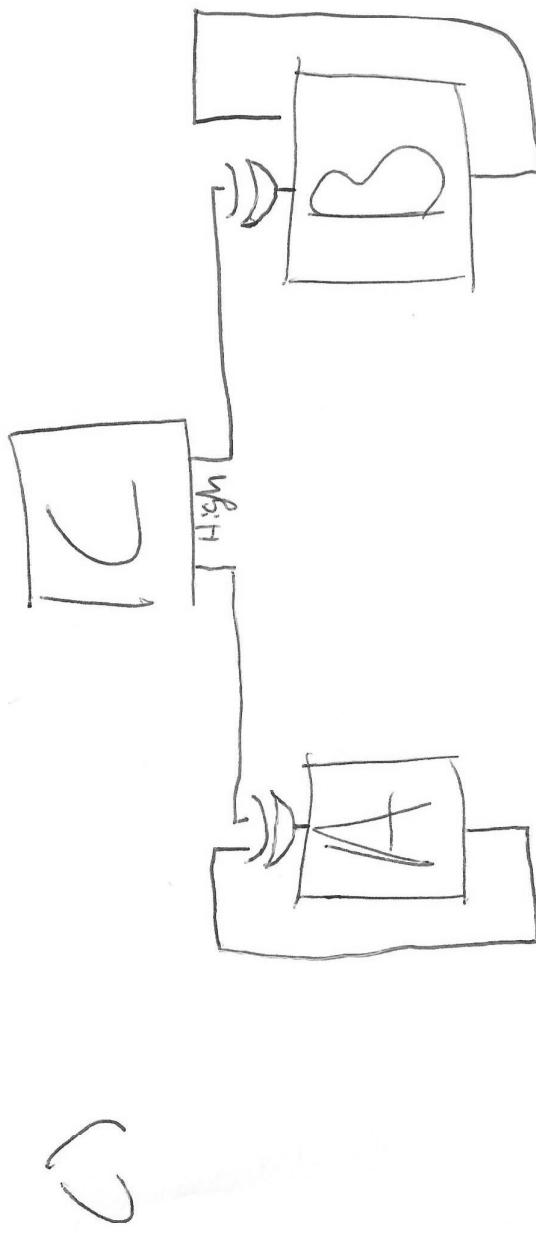
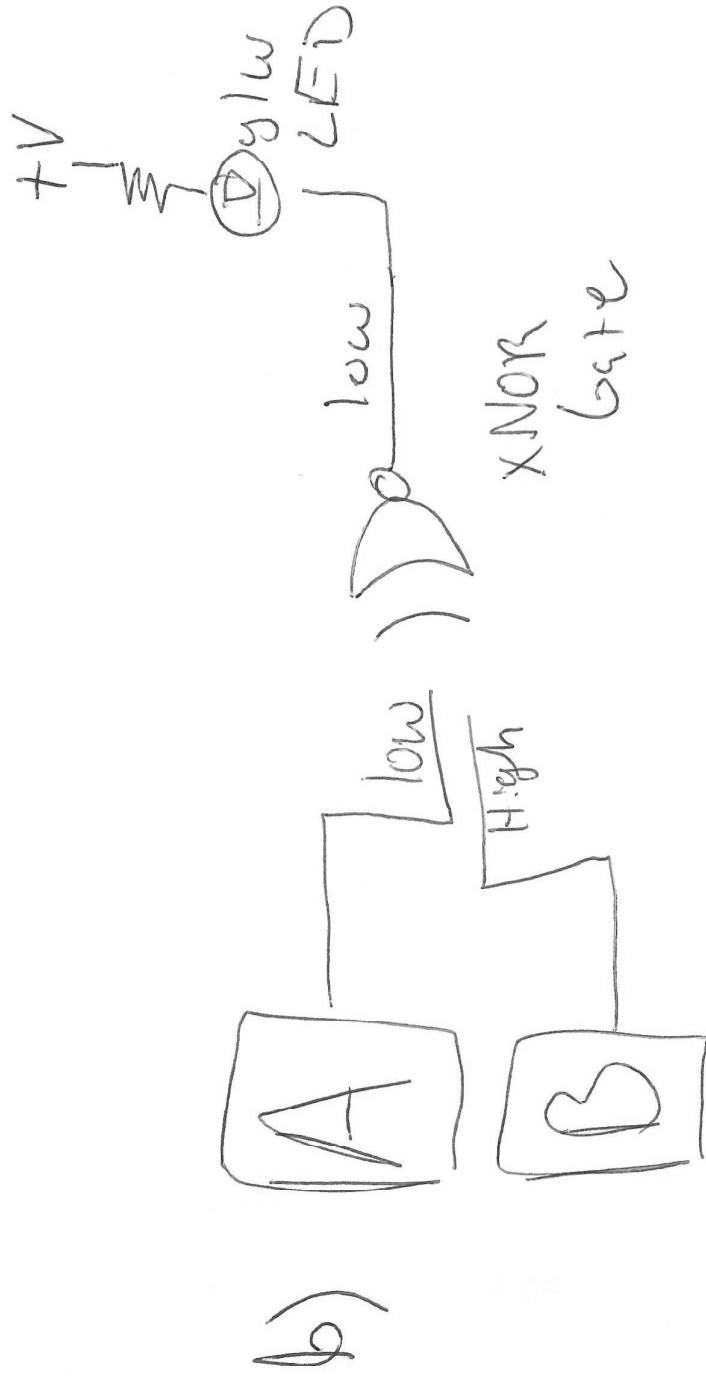
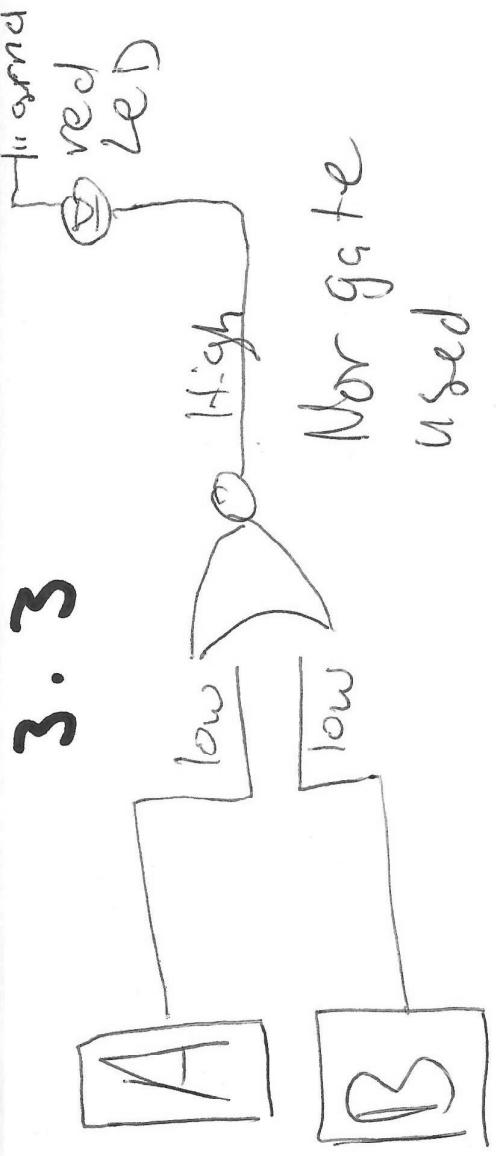
Chapter 5

- (a) Drawn in the back

3.2



3.3



XOR Gates used

CD	00	01	11	10
AB	00	✓		
	01			
	11	✓	✓	
	10			

which simplifies to

$$\overline{A}\overline{B}\overline{C} + ABC$$

A	B	C	Y
0	0	0	1
0	0	1	✓
0	1	0	
0	1	1	✓
1	0	0	
1	0	1	
1	1	0	
1	1	1	1

Which simplifies to

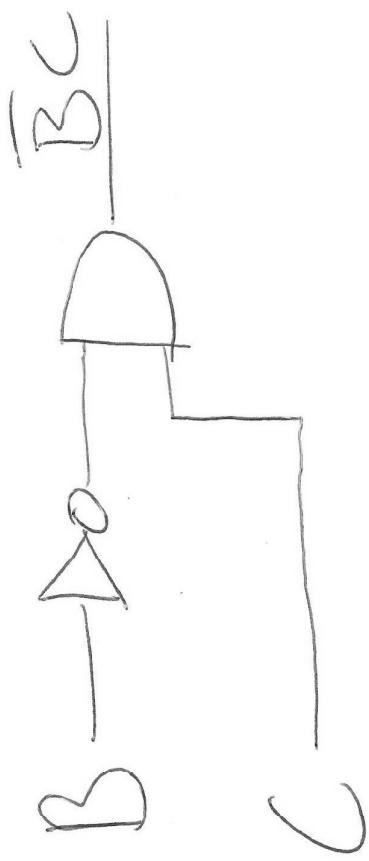
$$\overline{B} C$$

S-SOP form

$$\overline{A} \overline{B} C + A \overline{B} C$$

4.2a.

4. 28.



5.1

