

Midterm 2 for COSC330/PHYS306 - Fall 2021

Dr. Jordan Hanson - Whittier College Dept. of Physics and Astronomy

November 18, 2021

$$\begin{array}{r}
 1111 \\
 +1000 \\
 \hline
 10111 \\
 \text{Car } \Sigma_3 \Sigma_2 \Sigma_1 \Sigma_0
 \end{array}$$

1 Chapter 6 - Functions of Combinational Logic

1. Consider Fig. 1, in which a 4-bit adder is depicted. Assuming a uniform worst-case delay of 8 ns from carry-in to carry-out for each stage, the total delay is 32 ns. (a) At what maximum frequency can this circuit perform additions, if the delay must be less than a clock period? (b) What would the result in (a) be if the adder was extended to 8 bits? (c) Create a timing diagram showing the addition of the numbers in Fig. 1. **Bonus:** include the timing delays, and indicate the clock period.

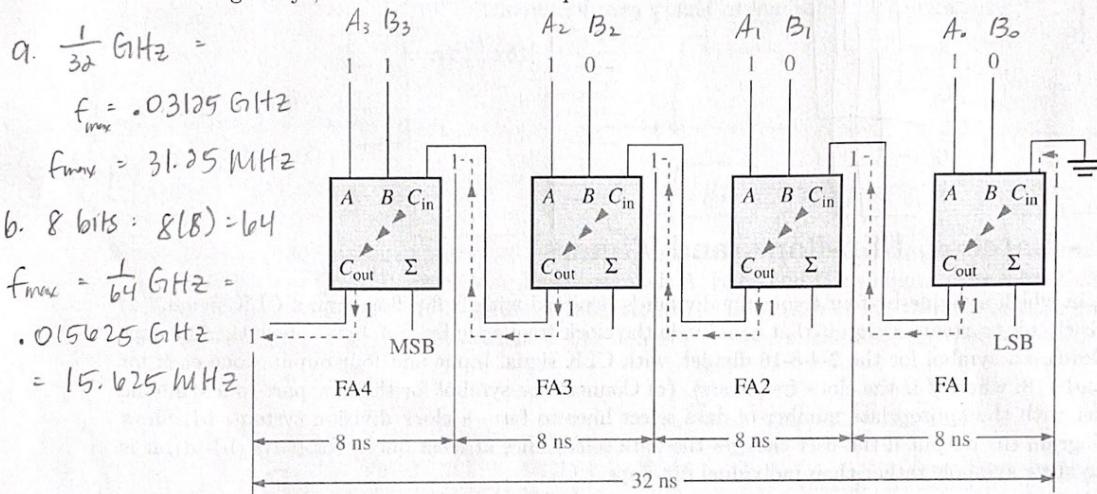
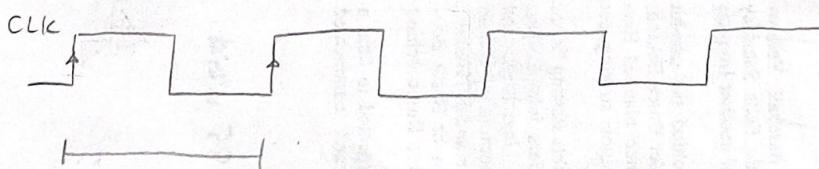
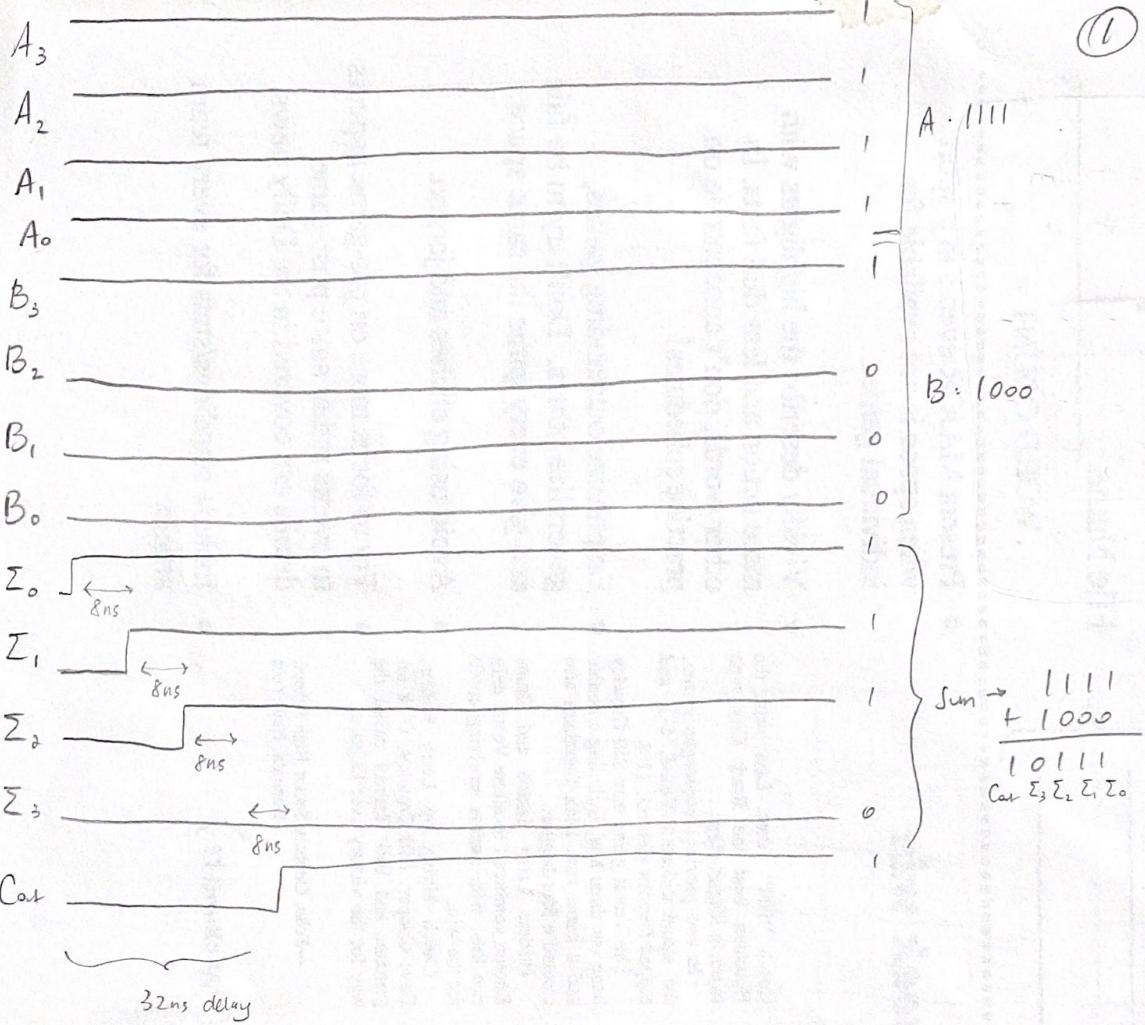


Figure 1: A schematic of a 4-bit adder with delays marked in nanoseconds.

- MSB: $A_3 A_2$ LSB: $B_3 B_0$
2. Using AND, OR, XNOR, and inverter gates, (a) create a 2-bit comparator that yields True if $A > B$, (b) True if $A = B$, and (c) True if $A < B$. For each circuit, assume both A and B are 2-bit binary positive numbers. (d) Wrap this all into one circuit with three outputs and 4 inputs. Hint: use Karnaugh maps for parts (a)-(c) to simplify the gates.

3. Generate the logic gates that convert 4-bit gray code to 4-bit binary code, and show that it works with a timing diagram.

seperate pgs



$$f_{\max} = \frac{1}{32} \text{ GHz}$$

(2)

	A_1	A_0	B_1	B_0	XNOR	
0	0	0	0	0	=	1
0	0	0	0	1	b	0
0	0	0	1	0	b	0
0	0	0	1	1	b	1
1	0	1	0	0	a	0
1	0	1	0	1	=	1
1	0	1	1	0	b	1
1	0	1	1	1	b	1
2	1	0	0	0	a	0
2	1	0	0	1	a	1
2	1	0	1	0	=	1
2	1	0	1	1	b	1
3	1	1	0	0	a	0
3	1	1	0	1	a	1
3	1	1	1	0	a	1
3	1	1	1	1	=	1

0' 1
0' 2
0' 3
0' 4

$A_1 A_0 \setminus B_1 B_0$	00	01	11	10	$A > B$
00					(a)
01	1				
11		1	1		
10		1	1	1	

$$A_1 \bar{B}_1 + A_0 \bar{B}_1 \bar{B}_0 + A_1 A_0 \bar{B}_0$$

$A_1 A_0 \setminus B_1 B_0$	00	01	11	10	$A < B$
00		1	1	1	
01			1	1	
11				1	
10				1	

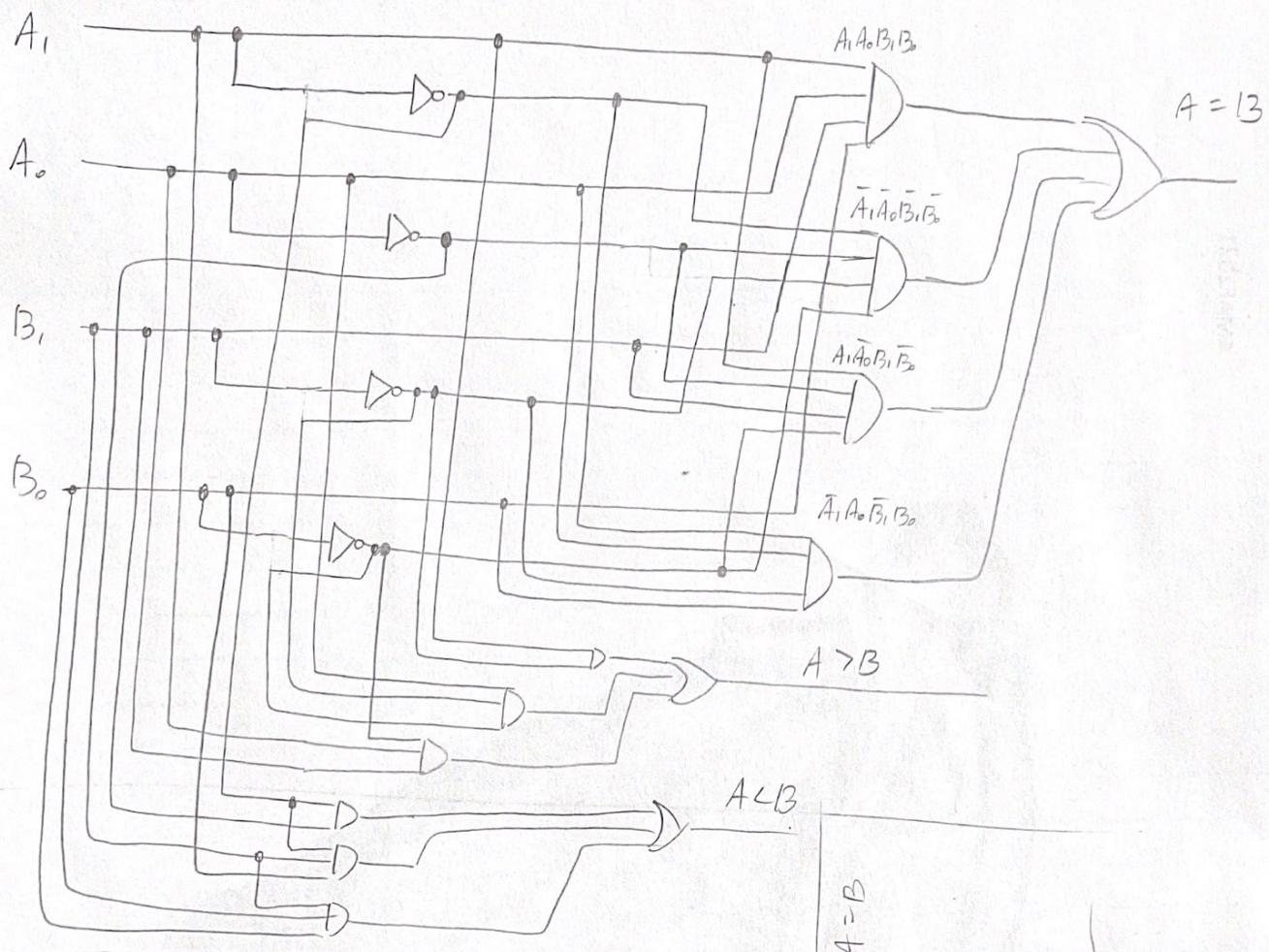
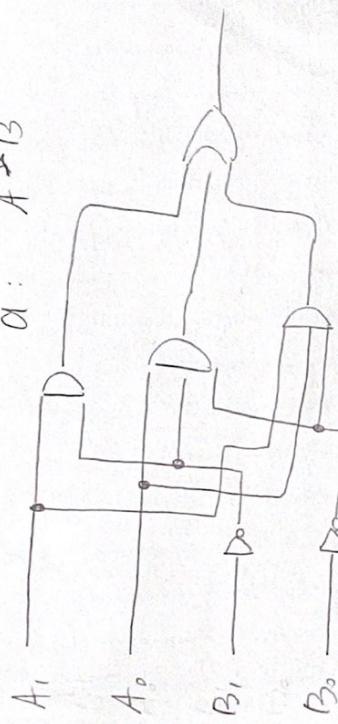
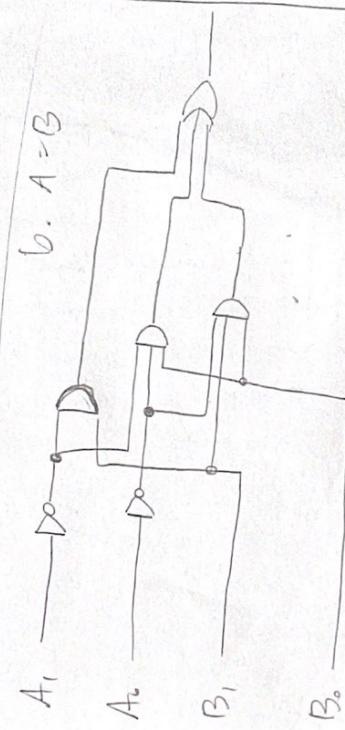
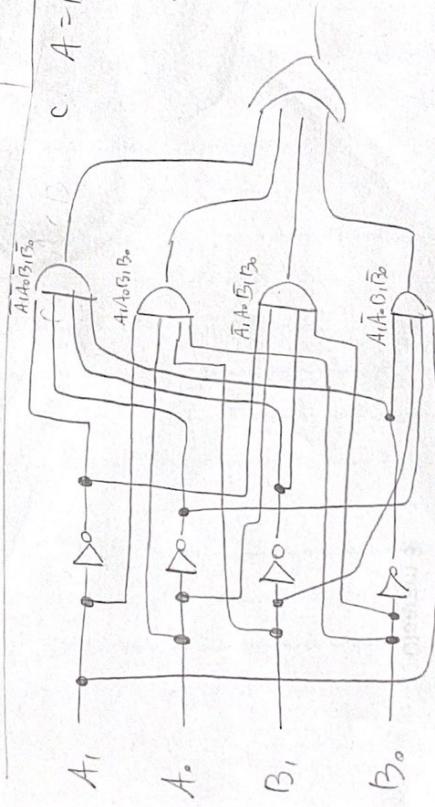
$$\bar{A}_1 B_1 + \bar{A}_1 \bar{A}_0 B_0 + \bar{A}_0 B_1 B_0$$

$A_1 A_0 \setminus B_1 B_0$	00	01	11	10	$A = B$
00	1				
01		1			
11			1		
10				1	

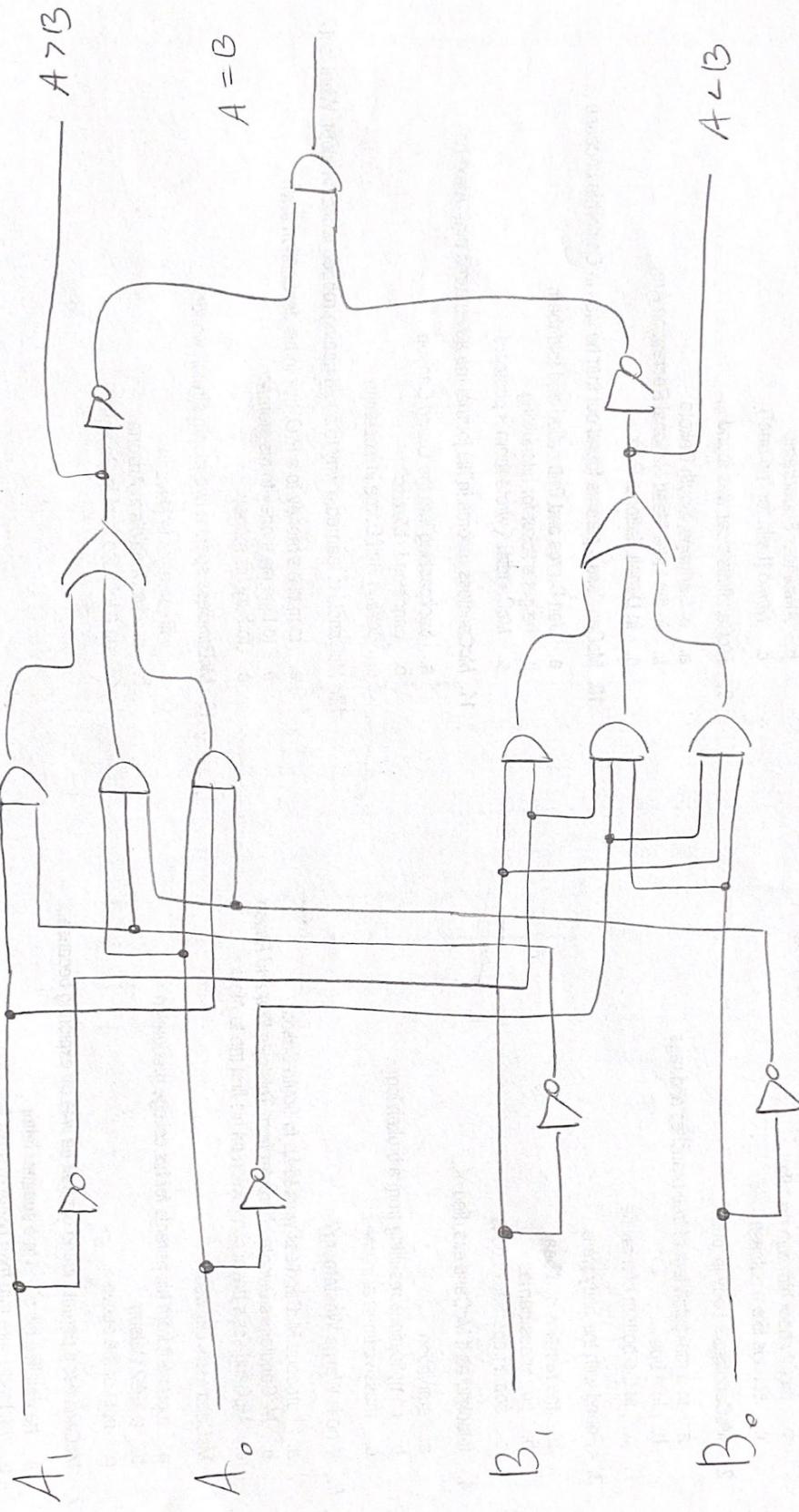
$$\bar{A}_1 \bar{A}_0 B_1 \bar{B}_0 + \bar{A}_1 A_0 \bar{B}_1 B_0 + A_1 \bar{A}_0 B_1 \bar{B}_0 + A_1 A_0 B_1 B_0$$

	A	B	\oplus OR	NOR	XNOR	XOR
0	0	0	0	1	1	0
0	1	1	1	0	0	1
1	0	1	0	0	1	1
1	1	1	0	0	1	0

(2)

a : $A > B$ b. $A = B$ c. $A = B$ 

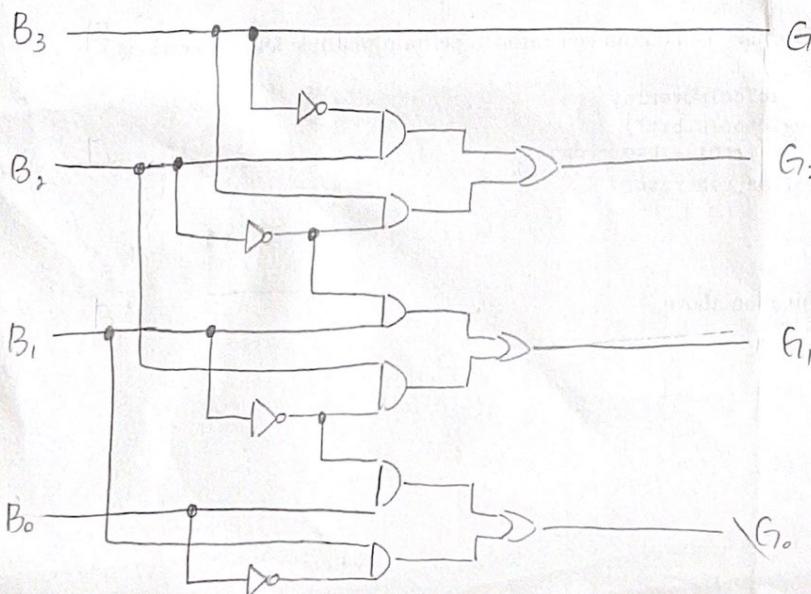
②



B_3	B_2	B_1	B_0	G_3	G_2	G_1	G_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

$$B_3 \cdot \overline{B_3}B_2 + B_3\overline{B}_2 + \overline{B}_2B_1 + B_2\overline{B}_1 + \overline{B}_1B_0 + B_1\overline{B}_0$$

MSB Binary



MSB Gray

G_2		(3)			
B_3/B_2	$/B_1B_0$	00	01	11	10
00					
01		1	1	1	1
11					
10		1	1	1	1

$$G_2 \rightarrow \overline{B}_3B_2 + B_3\overline{B}_2$$

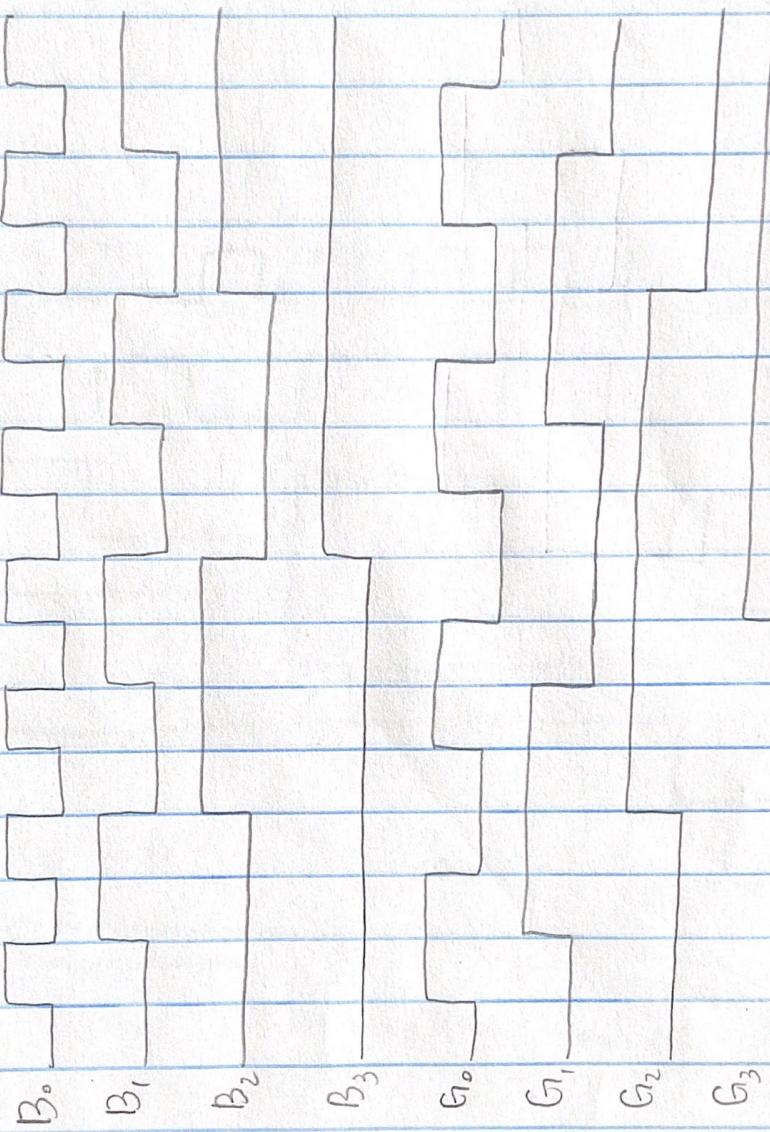
G_1		(3)			
B_3B_2	$/B_1B_0$	00	01	11	10
00					
01		1	1		
11		1	1		
10				1	1

$$G_1 \rightarrow B_2\overline{B}_1 + \overline{B}_2B_1$$

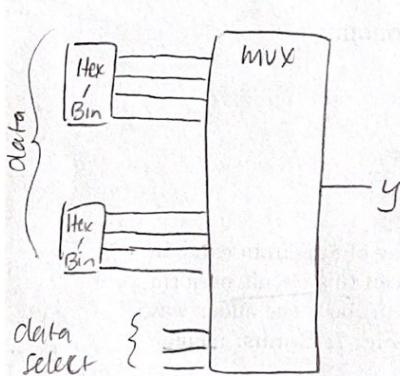
G_0		(3)			
B_3B_2	$/B_1B_0$	00	01	11	10
00					
01					
11					
10					

$$G_0 \rightarrow \overline{B}_1B_0 + B_1\overline{B}_0$$

(B)



4. Consider Fig. 2, in which a decimal to binary conversion circuit is shown. (a) Add logic to the circuit such that it becomes a hexadecimal to binary converter. (b) Draw a logic symbol for this circuit with the correct number of inputs and outputs. (c) Connect two hex-to-bin converters to a symbolic 2-to-1 multiplexer with the correct number of data select line(s) to form a system that can send two-digit hex numbers over one output line.



a

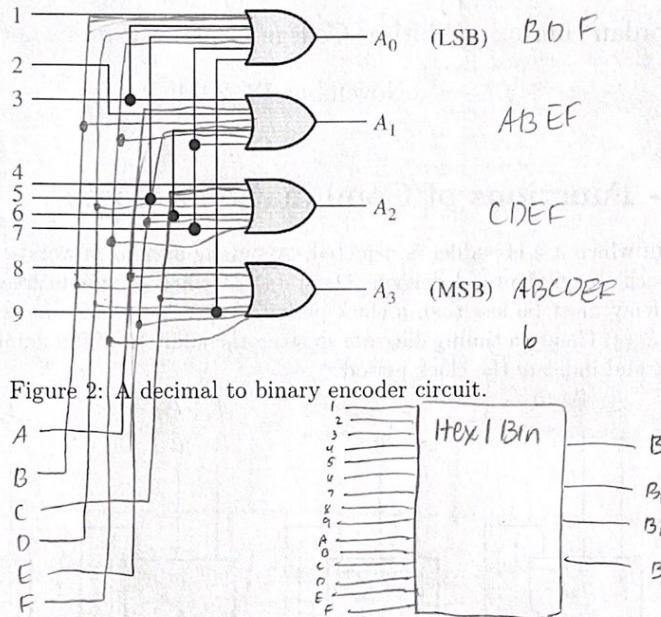
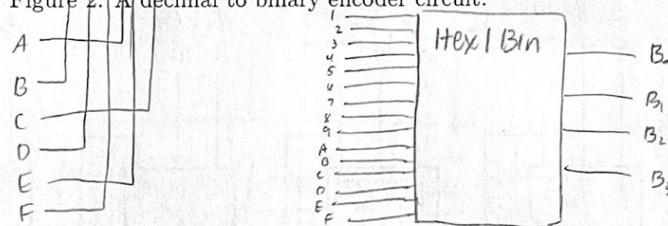


Figure 2: A decimal to binary encoder circuit.

a



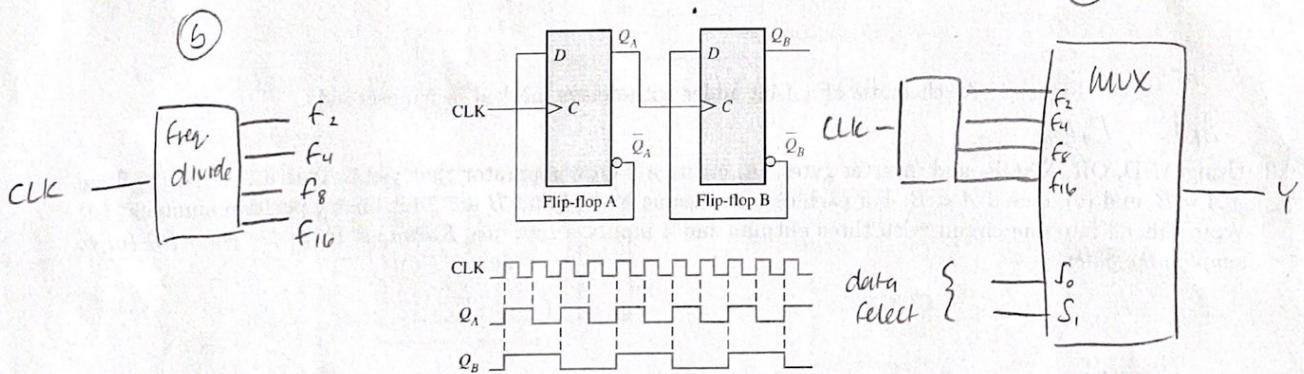
b

CDEF

ABCDEF

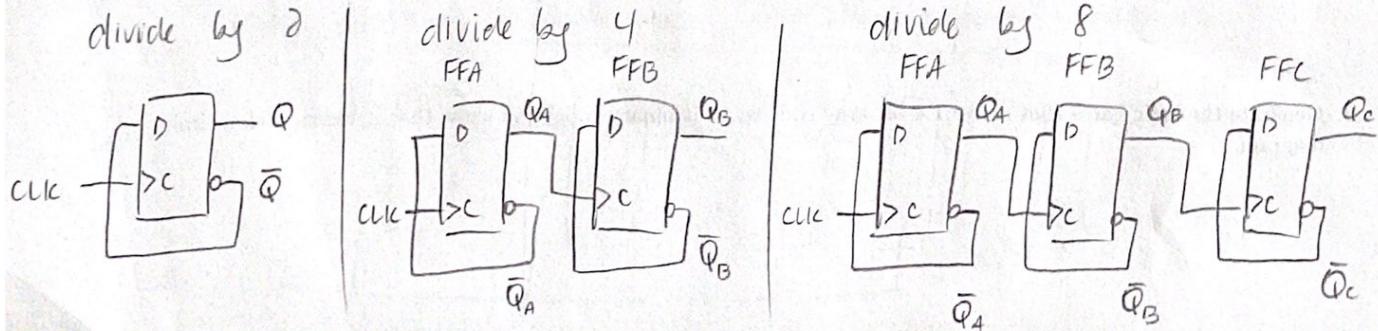
2 Chapter 7 - Latches, Flip-flops, and Timers

1. Consider Fig. 3, in which a divide-by-four frequency divider is depicted with D flip-flops and a CLK signal. (a) Elaborate on this circuit to create a circuit that can divide the clock frequency by 2, 4, or 8. Show the flip-flops explicitly. (b) Develop a symbol for the 2-4-8-16 divider, with CLK signal input and four outputs (one each for $f/2$, $f/4$, $f/8$, and $f/16$, where f is the clock frequency). (c) Connect the symbol for the new part to a symbolic 4-to-1 multiplexer with the appropriate number of data select lines to form a clock division system. (d) Show with a timing diagram the output if the user changes the data select lines at least once. For parts (b)-(d), it is only necessary to show symbols rather than individual flip-flops.



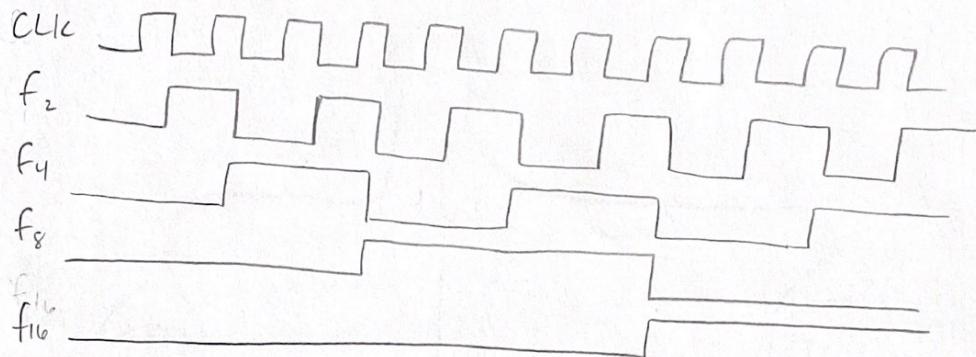
(a)

Figure 3: Two D flip-flops forming a frequency divider.

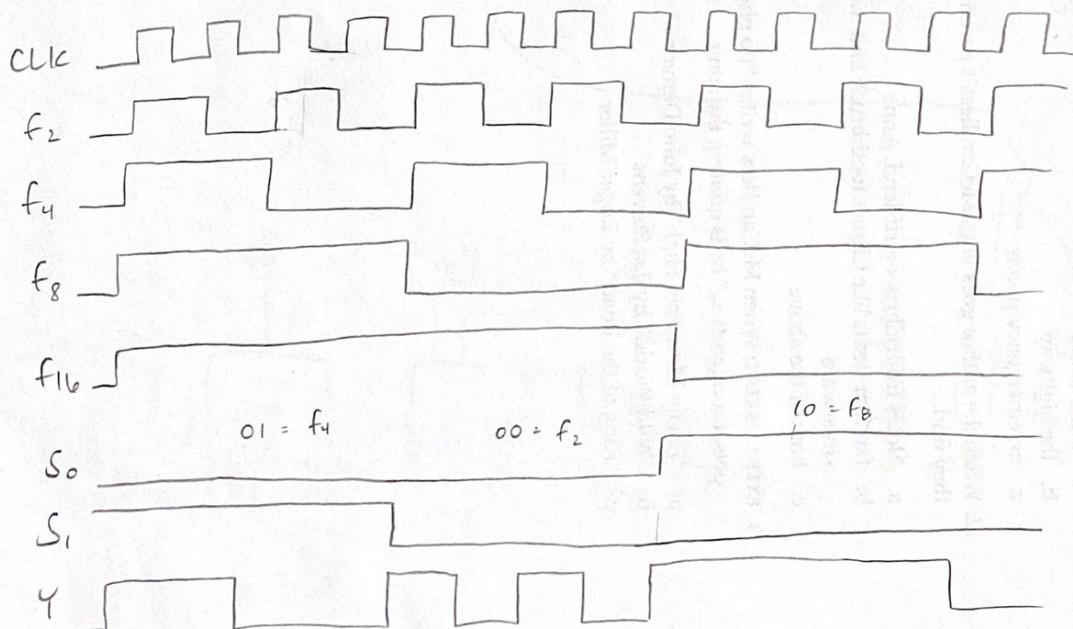


S_1, S_0	<u>Input Selected</u>
00	f_2
01	f_4
10	f_8
11	f_{16}

2-1



S_1, S_0	<u>Input Selected</u>
00	f_2
01	f_4
10	f_8
11	f_{16}



3 Chapters 8 and 9 - Shift Registers and Counters

1. Consider the timing diagram in Fig. 4. Using any combination of *shift registers* and supporting gates, create a circuit with 8-outputs that produces this timing diagram.

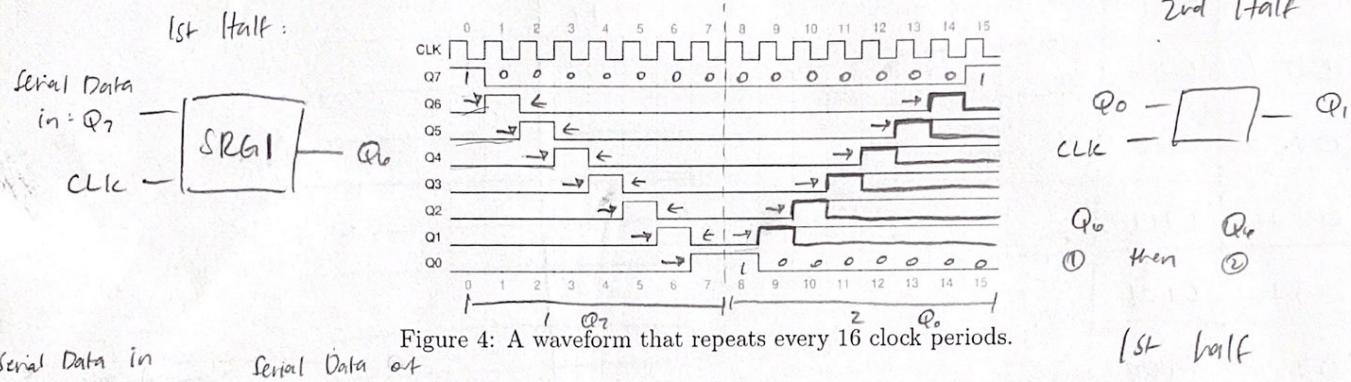
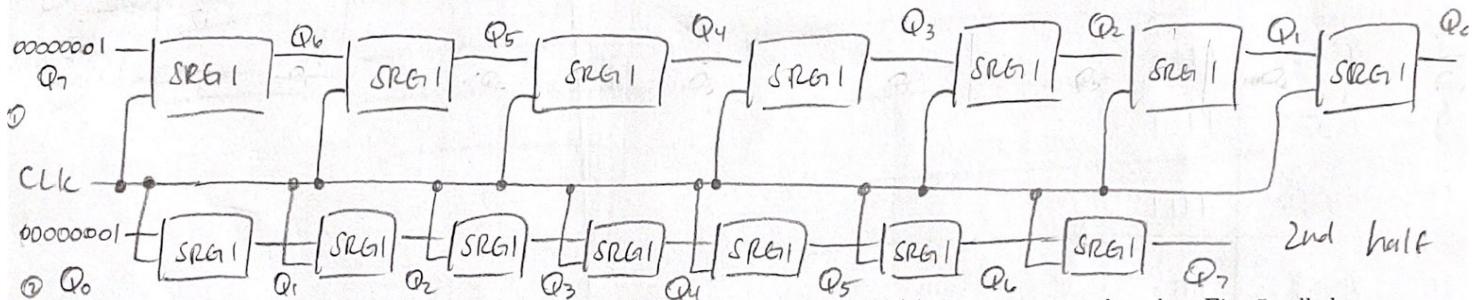


Figure 4: A waveform that repeats every 16 clock periods.

Serial Data In Serial Data Out



2. Consider Fig. 5, in which 4 and 5-bit Johnson counters are depicted. (a) Create a circuit based on Fig. 5 called a *combinatorial trigger*. Imagine four digital channels A, B, C, and D, as inputs. The output of the circuit should be True if *any two* of the four channels is True *within 100 ns of each other*. Develop any necessary logic symbols, and use the appropriate clock frequency. The Johnson counter(s) can have any number of bits.

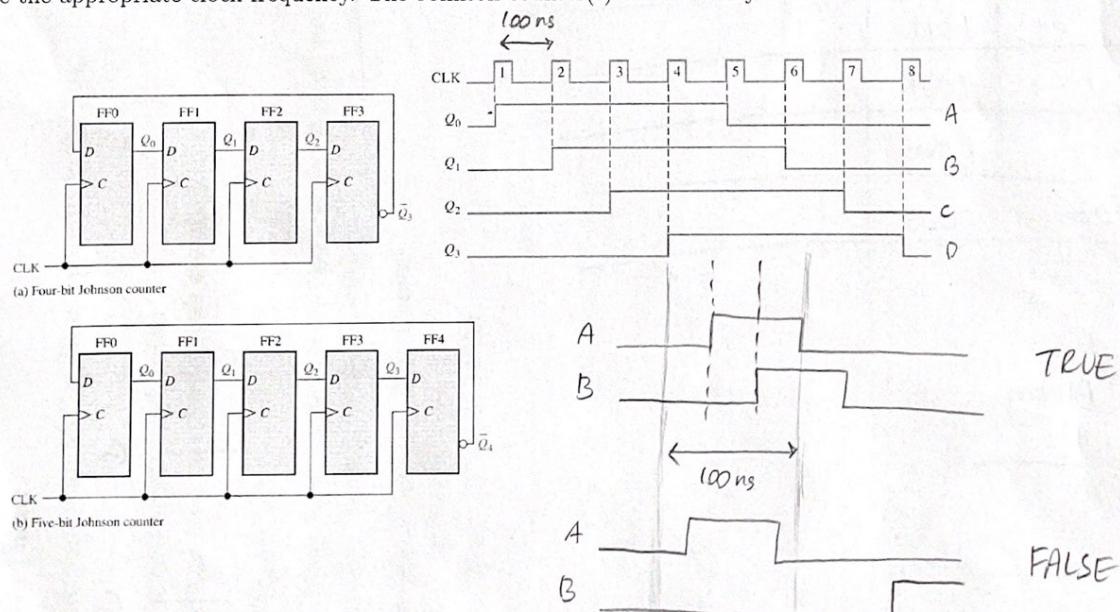
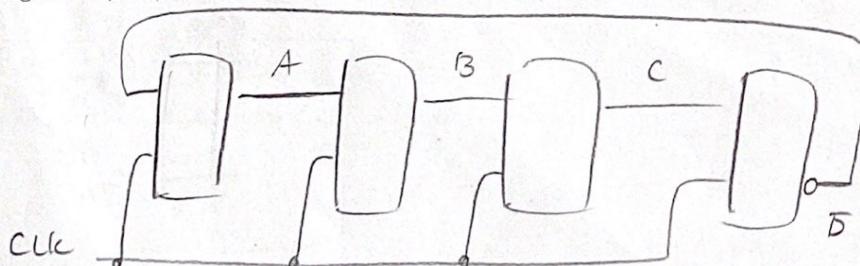
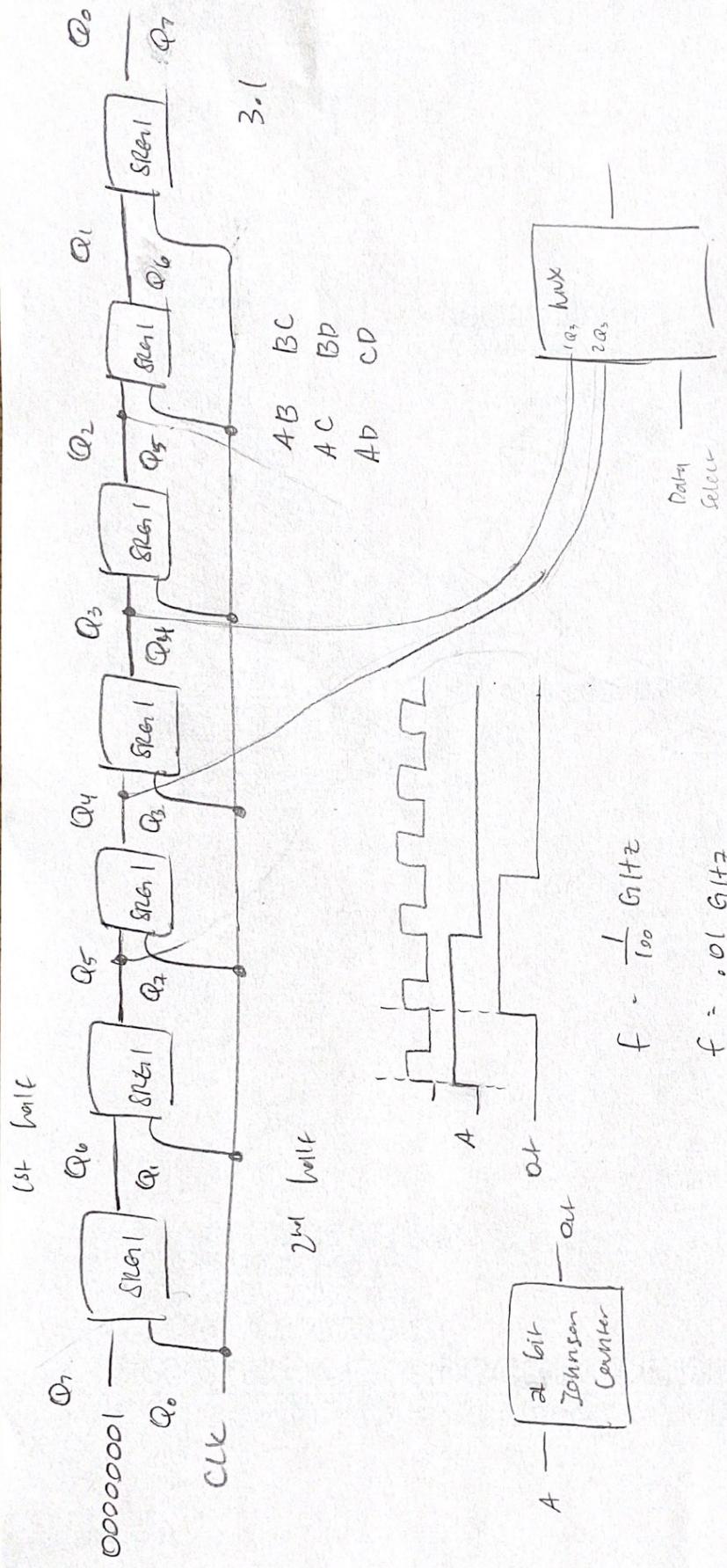


Figure 5: (Left) 4 and 5-bit Johnson counters. (Right) The timing diagram for the 4-bit case.





Do Data Select = 0 for
1st 8 clk cycles, 1 for next

Do for all Q₀ - Q₇

3.2

