

EE 266**Exam III: April 20, 1995**

Complete this page, but do not open the test until you are instructed to do so. You will have sixty (60) minutes to complete the exam. The exam is closed book and notes. No calculators are allowed. Make sure you budget your time well. Strategy statements are not required unless specifically requested (problems 4 and 6). “Basic gates” refers to AND, NAND, OR, NOR, and NOT gates.

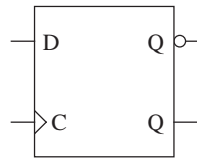
Check which section you are in: 8:30 _____ or 11:30 _____

Write down your seat assignment: Row _____ Number _____

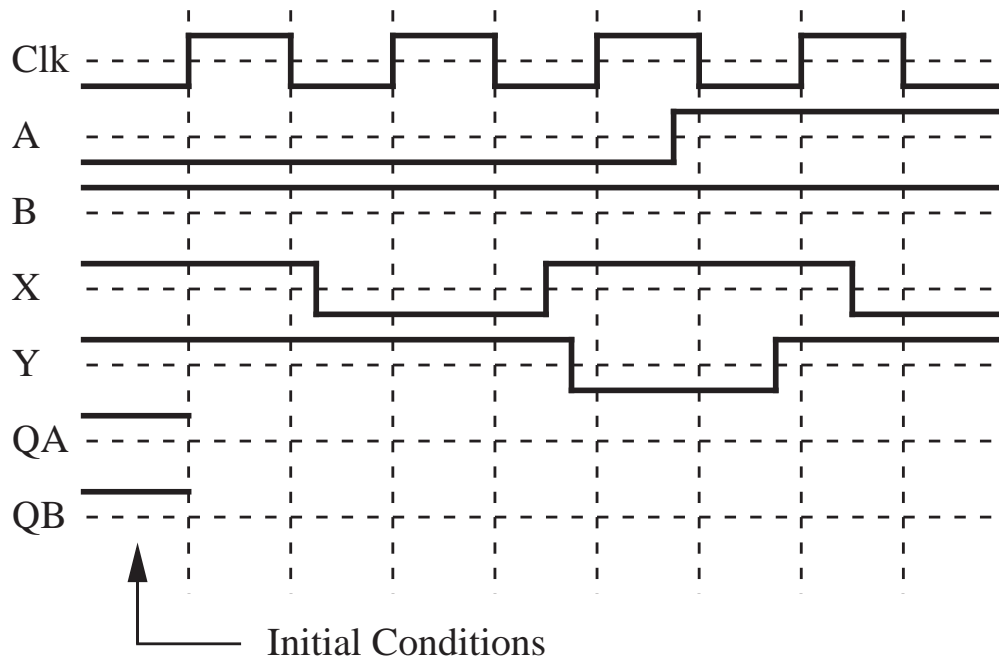
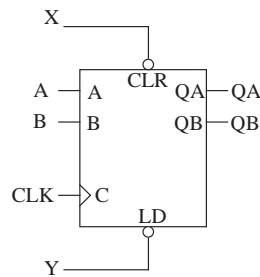
Scores:

Problem	Possible	Attained
1	10	_____
2	10	_____
3a	5	_____
3b	10	_____
3c	5	_____
4	20	_____
5	15	_____
6	25	_____
Total	100	_____

1. (10 pts.) Show how to implement a J-K Flip-Flop starting with a D Flip-Flop and basic gates. Show all of your work.

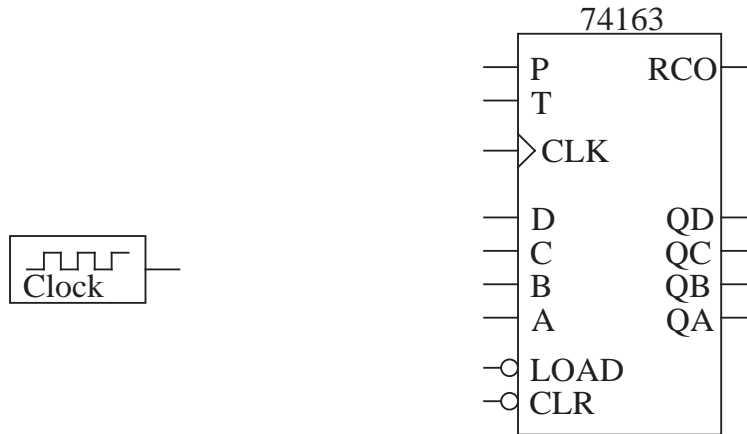


2. (10 pts.) This two bit register features synchronous load (*LD*) and asynchronous clear (*CLR*) inputs. Complete the timing diagram.

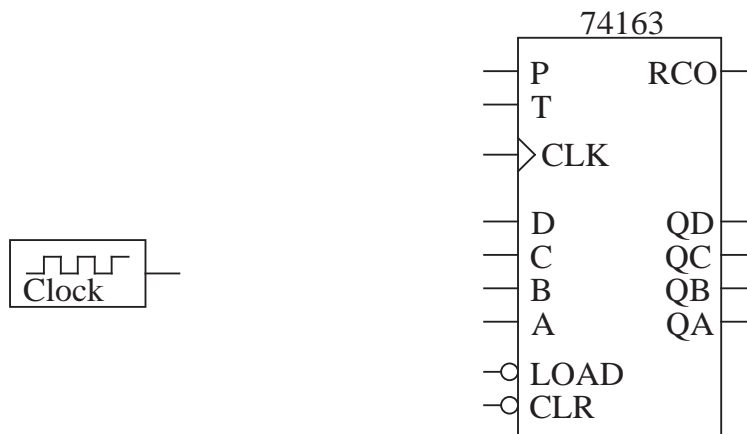


3. The 74163 is a 4-bit binary up-counter with synchronous *LOAD* and *CLEAR*. NOTE: *D* IS THE COUNTER'S MOST SIGNIFICANT BIT.

- (a) (5 pts.) Use it and basic gates to design a circuit that counts up from 0011 to 1011 and repeats.



- (b) (10 pts.) Use it and basic gates to design a circuit that counts from 0000 to 0100, then jumps to 1000, counts up to 1101 and repeats.



- (c) (5 pts.) Is the counter in b) a self starting counter? _____ Why?

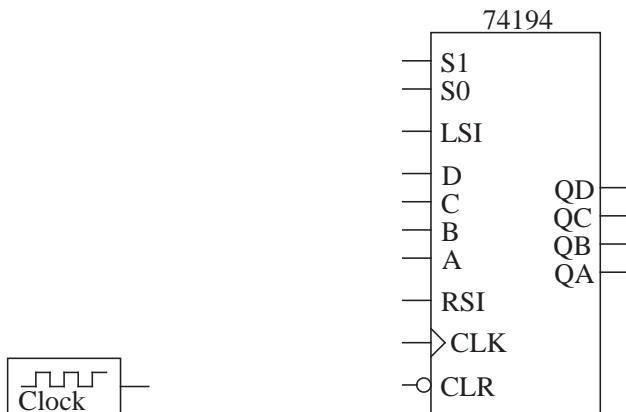
4. (20 pts.) Use the 74194 shift register and basic gates to implement a Johnson counter. HINT: THE COUNTER SEQUENCE IS {0000, 1000, 1100, 1110, 1111, 0111, 0011, 0001, 0000, ...}.

74194 Function Table

<i>Clr</i>	<i>S</i> ₁	<i>S</i> ₀	<i>Clk</i>	<i>LSI</i>	<i>RSI</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>Q</i> _A	<i>Q</i> _B	<i>Q</i> _C	<i>Q</i> _D
<i>L</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>L</i>	<i>L</i>	<i>L</i>	<i>L</i>
<i>H</i>	<i>H</i>	<i>H</i>	↑	<i>X</i>	<i>X</i>	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>
<i>H</i>	<i>L</i>	<i>H</i>	↑	<i>X</i>	<i>H</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>H</i>	<i>Q</i> _a	<i>Q</i> _b	<i>Q</i> _c
<i>H</i>	<i>L</i>	<i>H</i>	↑	<i>X</i>	<i>L</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>L</i>	<i>Q</i> _a	<i>Q</i> _b	<i>Q</i> _c
<i>H</i>	<i>H</i>	<i>L</i>	↑	<i>H</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>Q</i> _b	<i>Q</i> _c	<i>Q</i> _d	<i>H</i>
<i>H</i>	<i>H</i>	<i>L</i>	↑	<i>L</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>Q</i> _b	<i>Q</i> _c	<i>Q</i> _d	<i>L</i>
<i>H</i>	<i>L</i>	<i>L</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>X</i>	<i>Q</i> _a	<i>Q</i> _b	<i>Q</i> _c	<i>Q</i> _d

<i>Q</i> _A	<i>Q</i> _B	<i>Q</i> _C	<i>Q</i> _D
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

Strategy: (6 pts.)



5. (15 pts.) Using the transition table below, complete the ASM chart. Simplify conditional boxes as much as possible. (It may be helpful to sketch the state diagram first.)

X	Y	A	B	A^+	B^+	Z
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	1	0	1
0	0	1	1	1	1	0
0	1	0	0	0	0	0
0	1	0	1	1	1	0
0	1	1	0	1	0	1
0	1	1	1	0	1	0
1	0	0	0	0	1	0
1	0	0	1	1	1	0
1	0	1	0	1	0	1
1	0	1	1	0	1	1
1	1	0	0	0	1	0
1	1	0	1	1	1	0
1	1	1	0	1	0	1
1	1	1	1	0	1	1

00

10

01

11

6. Purdue University would like to put a car counter in the new parking garage that they are building. They have given you the finite state machine portion of the design. Your job is to build a counter that will increment by one every time a car enters the garage and decrement by one every time a car leaves the garage.

- (a) (22 pts.) Design a synchronous 3-bit FSM that will perform this task. Assume that you have inputs CAR and IN/\overline{OUT} where $CAR = 1$ when a car is either entering or leaving and $IN/\overline{OUT} = 1$ only when a car is entering. Assume that you will never have more cars leave than enter, that no more than seven cars will ever be in the garage at one time, and that two cars cannot enter and leave at the same time. Use a D Flip-Flop for the most significant bit, a J-K Flip-Flop for the middle bit, and a T Flip-Flop for the least significant bit. Go no further than the Flip-Flop input table. Next state equations and an actual circuit implementation are *not* required. HINT: AN EXTERNAL CLOCK SIGNAL MAY NOT BE NEEDED.

Strategy: (8 pts.)

- (b) (3 pts.) How many Flip-Flops would be required to design a FSM that could handle the parking limit of the new garage if the new garage has 1,500 parking spaces?