Logic Design

Lab Experiments

Dr. Manal Tantawi

	Experiment Title
1	Logic gates
2	combination circuit I
3	Combination circuit II
4	Implementing functions using NAND only
5	Implementing functions using NOR only
6	Full adder
7	Full adder using decoder
8	Implementing functions using multiplexers
9	Sequential circuit
10	2 bit counter
11	Pseudo-random generator
12	Frequency division

Lab 1 (logic gates)

Objective: introducing lab equipment and integrated circuits (ICs) to students.

Task to do in lab: verify truth table of logic gates.

OR gate (IC:7432)

X	Y	X+Y
0	0	0
0	1	1
1	0	1
1	1	1

AND gate (IC:7408)

X	Y	X.Y
0	0	0
0	1	0
1	0	0
1	1	1

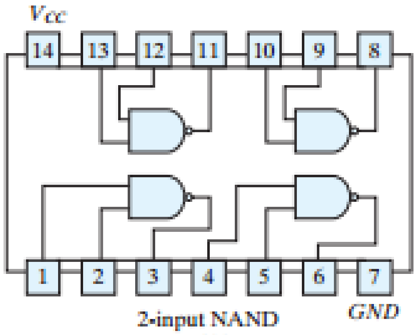
NOR gate (IC:7402)

X	Y	(X+Y)’
0	0	1
0	1	0
1	0	0
1	1	0

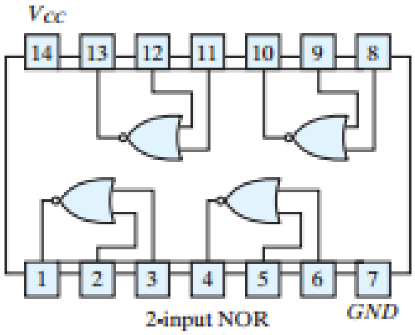
NAND gate (IC:7400)

X	Y	(X.Y)’
0	0	1
0	1	1
1	0	1
1	1	0

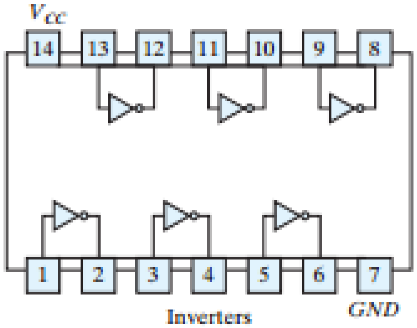
ICs for logic gates



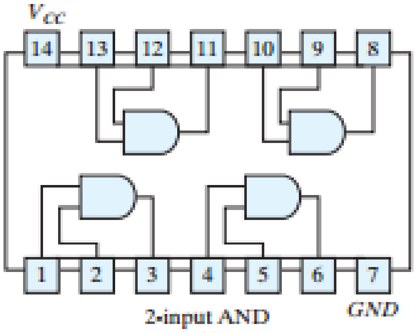
2-input NAND
7400



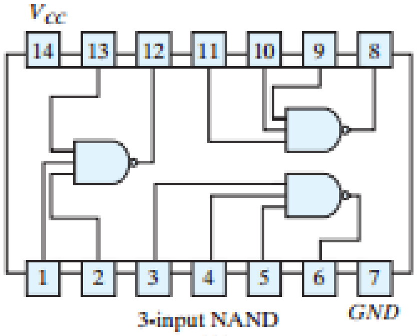
2-input NOR
7402



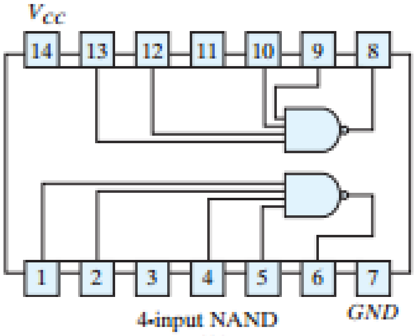
Inverters
7404



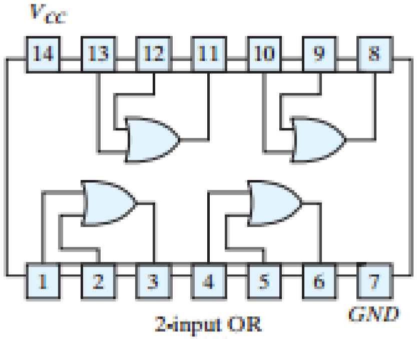
2-input AND
7408



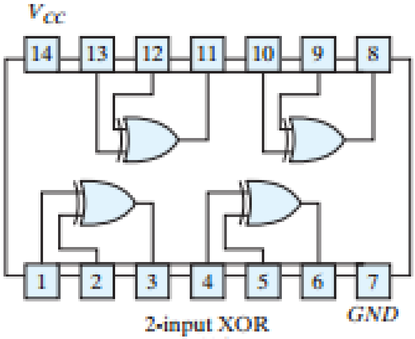
3-input NAND
7410



4-input NAND
7420



2-input OR
7432



2-input XOR
7486

Lab 2 (combinational circuit I)

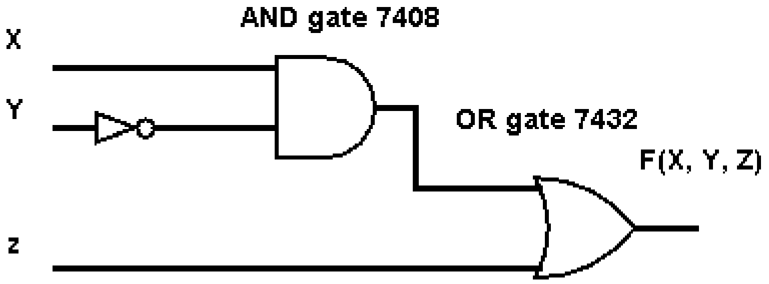
Objective: implementing simple combinational circuit.

Task to do in lab: implementing practically the following function

$$F(X, Y, Z) = XY' + Z$$

Truth table for F

X Y Z	F
0 0 0	0
0 0 1	1
0 1 0	0
0 1 1	1
1 0 0	1
1 0 1	1
1 1 0	0
1 1 1	1



Lab 3 (combinational circuit II)

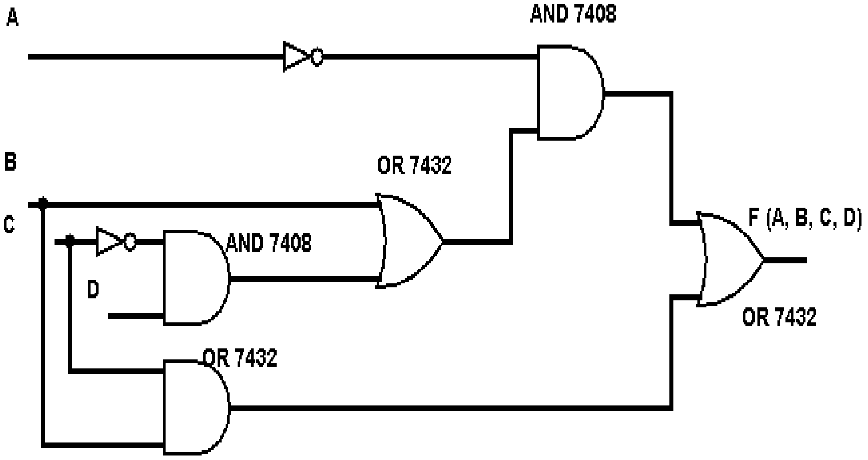
Objective: implementing combinational circuit (long expression)

Task to do in lab: implementing practically the following function

F (A, B, C, D) = A’ (B + C’D) + BC

Truth table for F

A B C D	F
0 0 0 0	0
0 0 0 1	1
0 0 1 0	0
0 0 1 1	0
0 1 0 0	1
0 1 0 1	1
0 1 1 0	1
0 1 1 1	1
1 0 0 0	0
1 0 0 1	0
1 0 1 0	0
1 0 1 1	0
1 1 0 0	0
1 1 0 1	0
1 1 1 0	1
1 1 1 1	1

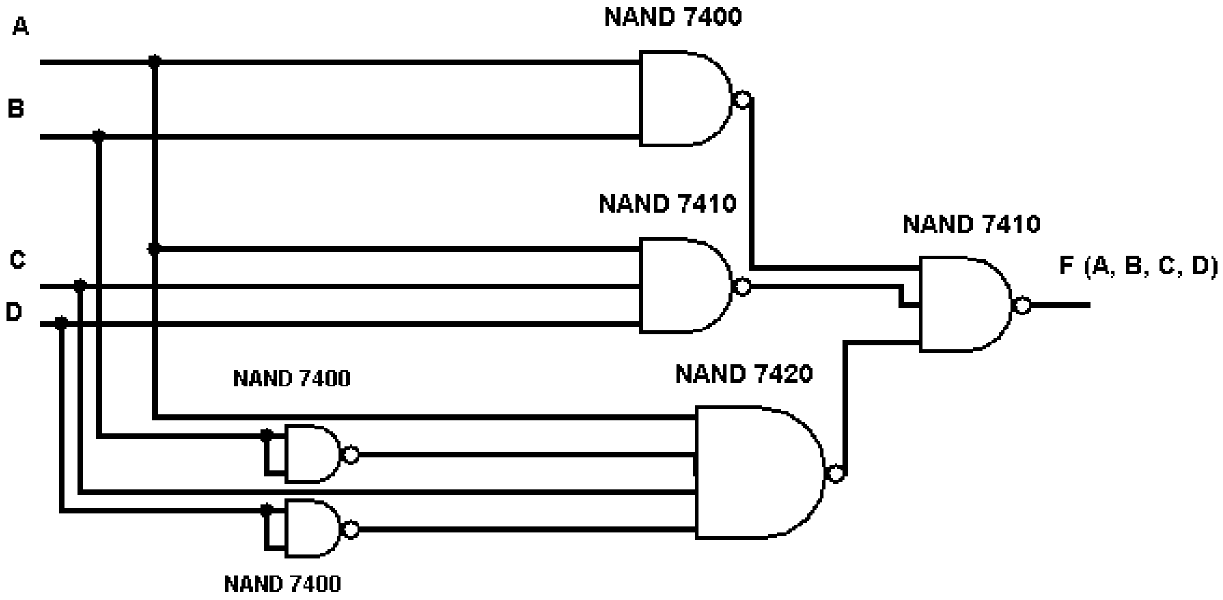


Lab 4 (NAND only)

Objective: implementing Functions using NAND only

Task to do in lab: implementing practically the following function using NAND only

$$F(A, B, C, D) = AB + ACD + AB'CD'$$



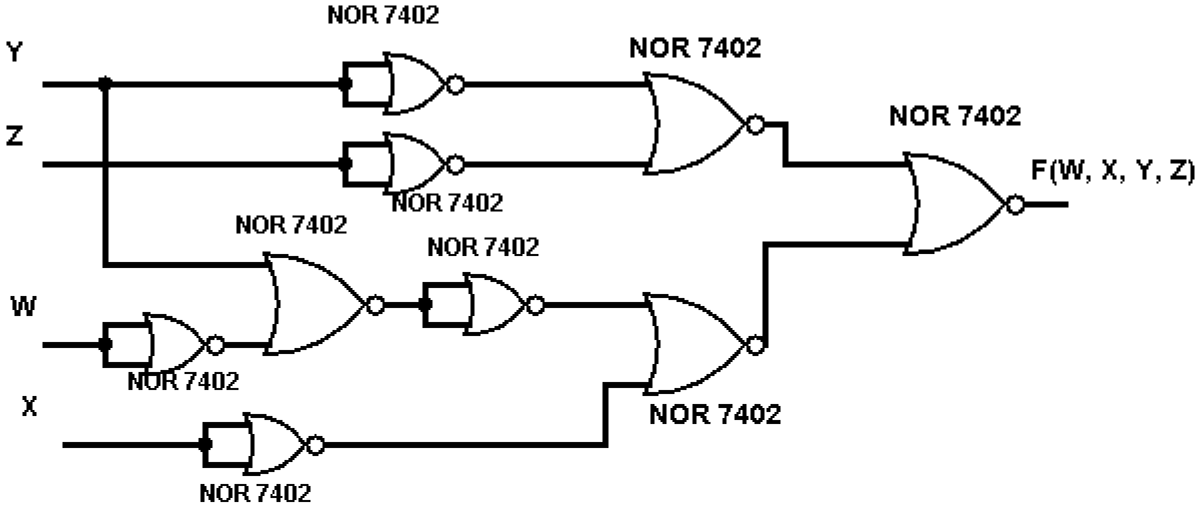
F (A, B, C, D) = $\sum(10, 11, 12, 13, 14, 15)$

Lab 5 (NOR only)

Objective: implementing functions using NOR only.

Task to do in lab: implementing practically the following function using NOR only.

$$F(W, X, Y, Z) = (Y'+Z')(W'+X'+Y)$$



$$F(W, X, Y, Z) = \sum(0, 1, 2, 4, 5, 6, 8, 9, 10, 14)$$

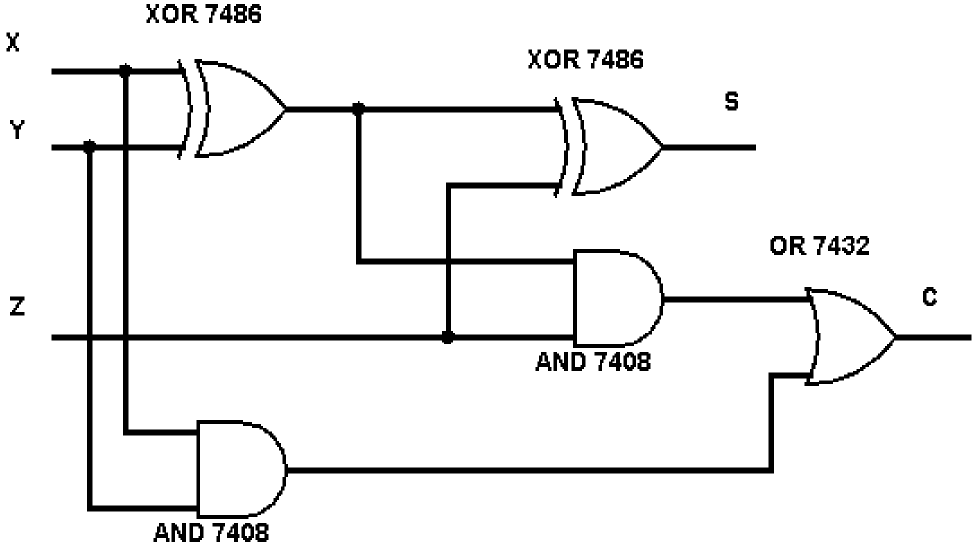
Lab 6 (Full adder)

Objective: implementing Full Adder.

Task to do in lab: implementing practically the Full Adder.

Truth table for Full Adder

X Y Z	C S
0 0 0	0 0
0 0 1	0 1
0 1 0	0 1
0 1 1	1 0
1 0 0	0 1
1 0 1	1 0
1 1 0	1 0
1 1 1	1 1



Lab 7 (Decoders)

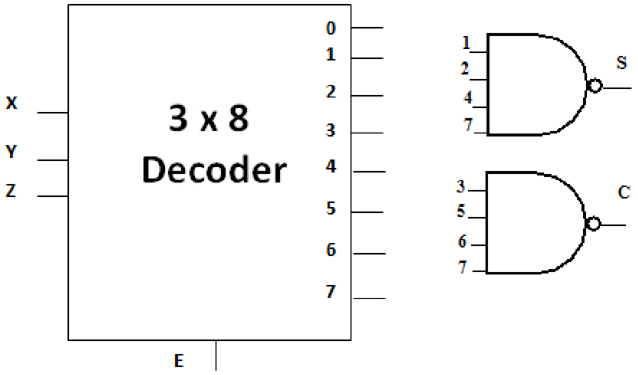
Objective: implementing functions using decoder.

Task to do in lab: implementing practically the Full Adder using suitable decoder.

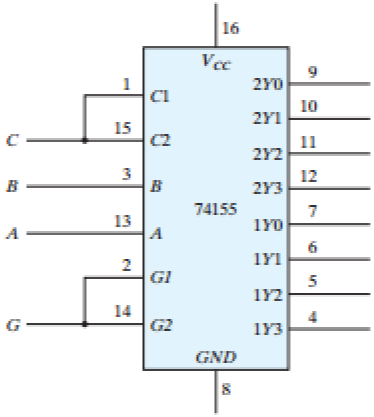
Truth table for Full Adder

X	Y	Z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

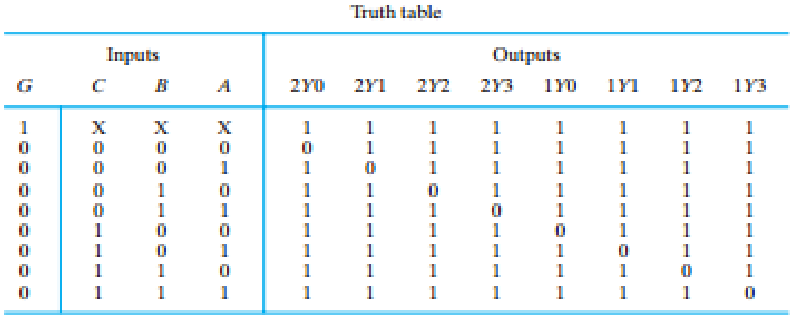
Full adder using active low decoder



Pin assignment for available Active low decoder IC: 74155



Truth table for IC: 74155



Truth table

Inputs				Outputs							
G	*C*	*B*	*A*	2*Y*0	2*Y*1	2*Y*2	2*Y*3	1*Y*0	1*Y*1	1*Y*2	1*Y*3
1	X	X	X	1	1	1	1	1	1	1	1
0	0	0	0	0	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1	1
0	0	1	1	1	1	1	0	1	1	1	1
0	1	0	0	1	1	1	1	0	1	1	1
0	1	0	1	1	1	1	1	1	0	1	1
0	1	1	0	1	1	1	1	1	1	0	1
0	1	1	1	1	1	1	1	1	1	1	0

Lab 8 (Multiplexers)

Objective: implementing functions using multiplexer.

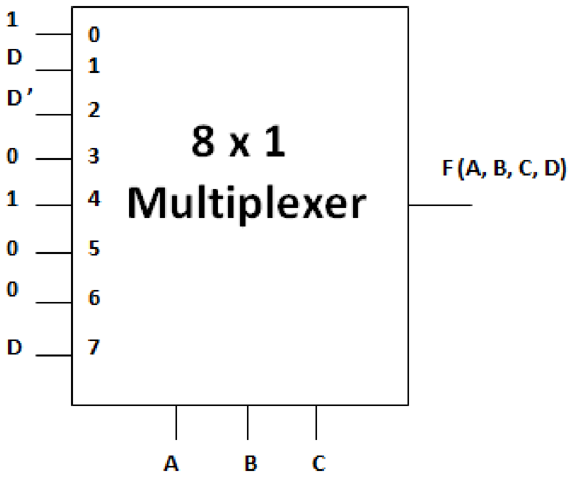
Task to do in lab: implementing practically the Following function using multiplexer.

$$F(A, B, C, D) = \sum(0, 1, 3, 4, 8, 9, 15)$$

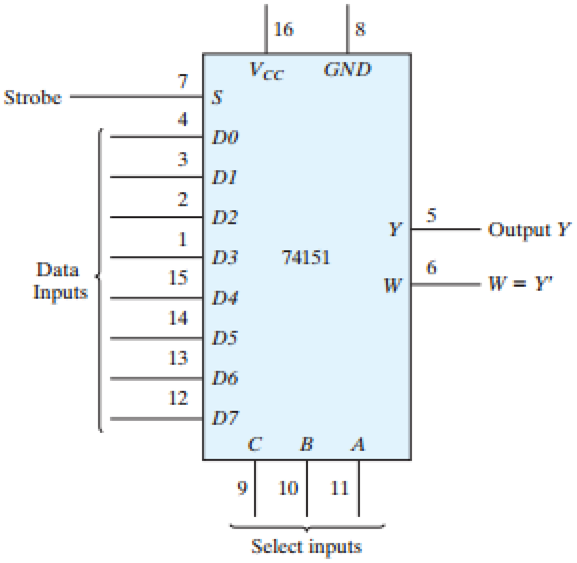
Truth table for F

A B C D	F
0 0 0 0	1
0 0 0 1	1
0 0 1 0	0
0 0 1 1	1
0 1 0 0	1
0 1 0 1	0
0 1 1 0	0
0 1 1 1	0
1 0 0 0	1
1 0 0 1	1
1 0 1 0	0
1 0 1 1	0
1 1 0 0	0
1 1 0 1	0
1 1 1 0	0
1 1 1 1	1

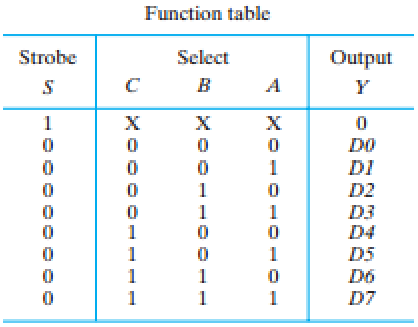
Implementation using multiplexer



Pin assignment for available multiplexer IC: 74151



Function table for IC: 74155



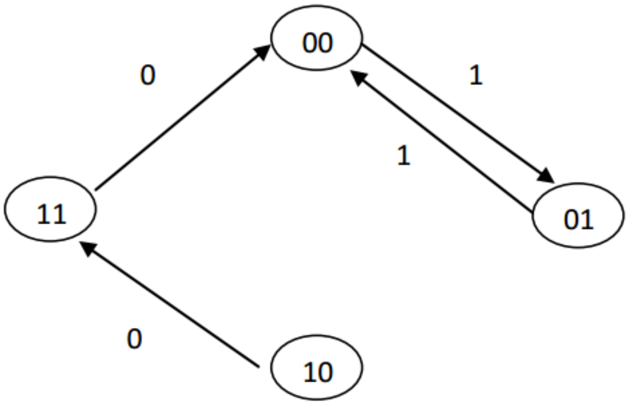
Function table

Strobe	Select			Output
S	C	B	A	Y
1	X	X	X	0
0	0	0	0	D0
0	0	0	1	D1
0	0	1	0	D2
0	0	1	1	D3
0	1	0	0	D4
0	1	0	1	D5
0	1	1	0	D6
0	1	1	1	D7

Lab 9 (Sequential circuits)

Objective: implementing sequential circuits using different flip flops.

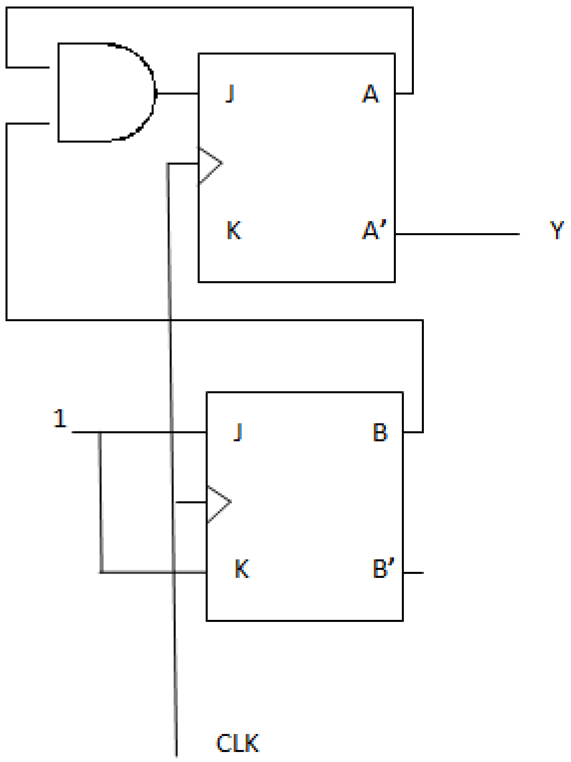
Task to do in lab: design and implement a sequential circuit that follows the following state diagram with one external output using T flip flops.



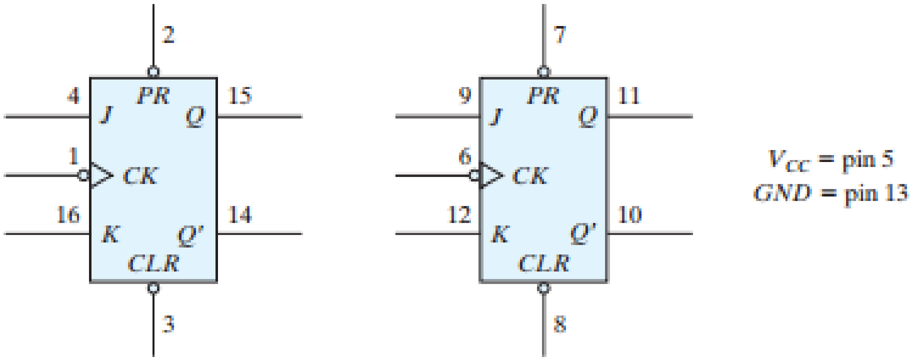
A_n	B_n	A_{n+1}	B_{n+1}	T_A	T_B	Y
0	0	0	1	0	1	1
0	1	0	0	0	1	1
1	0	1	1	0	1	0
1	1	0	0	1	1	0

T_A = AB T_B = 1 Y= A'

Circuit Logic diagram using JK flip flops (j and k are connected to form T flip flop)



Pin assignment for available JK flip flop IC: 7476



Function table

Inputs					Outputs	
Preset	Clear	Clock	*J*	*K*	*Q*	*Q'*
0	1	X	X	X	1	0
1	0	X	X	X	0	1
0	0	X	X	X	1	1
1	1	⎍	0	0	No change	
1	1	⎍	0	1	0	1
1	1	⎍	1	0	1	0
1	1	⎍	1	1	Toggle	

Lab 10 (Counters)

Objective: implementing counter circuits using different flip flops.

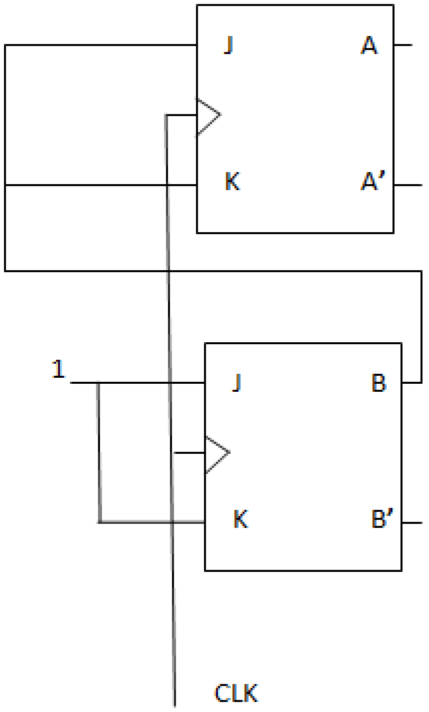
Task to do in lab: design and implement a two bit counter using JK flipflops

A_n	B_n	A_{n+1}	B_{n+1}	J_A	K_B	J_B	K_B
0	0	0	1	0	x	1	x
0	1	1	0	1	x	x	1
1	0	1	1	x	0	1	x
1	1	0	0	x	1	x	1

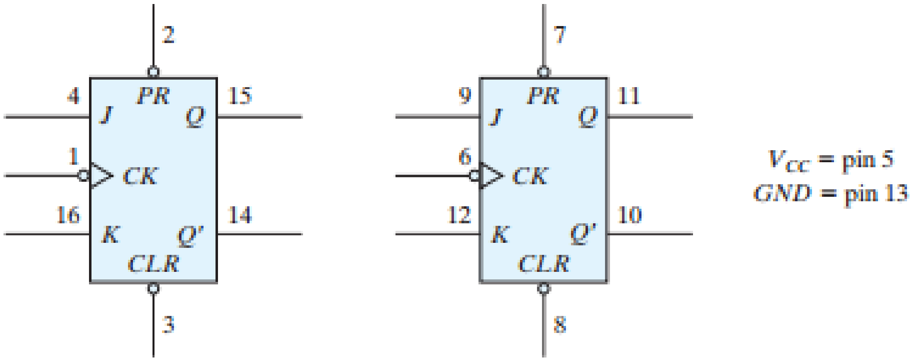
$J_A = K_A = B$

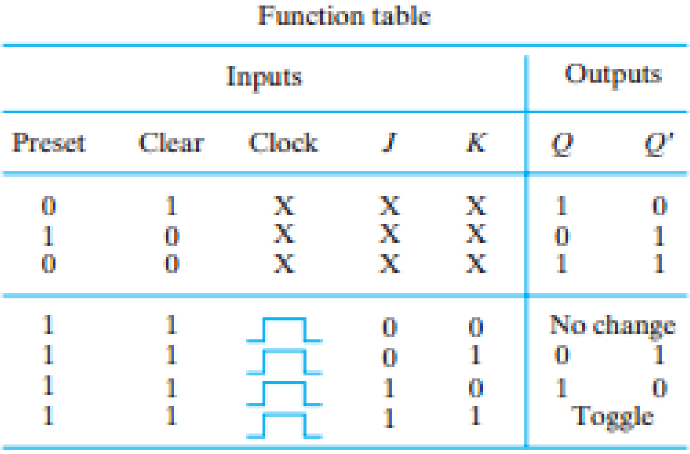
$J_B = K_B = 1$

Circuit Logic diagram using JK flip flops



Pin assignment for available JK flip flop IC: 7476





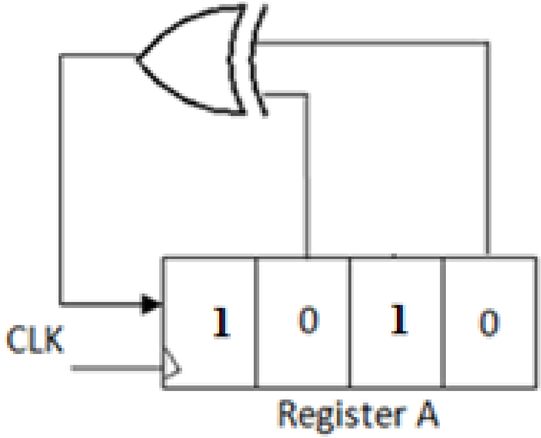
Function table

Inputs					Outputs	
Preset	Clear	Clock	*J*	*K*	*Q*	*Q'*
0	1	X	X	X	1	0
1	0	X	X	X	0	1
0	0	X	X	X	1	1
1	1	⎍	0	0	No change	
1	1	⎍	0	1	0	1
1	1	⎍	1	0	1	0
1	1	⎍	1	1	Toggle	

Lab 11 (Registers)

Objective: implementing pseudorandom generator using 4 bit shift register.

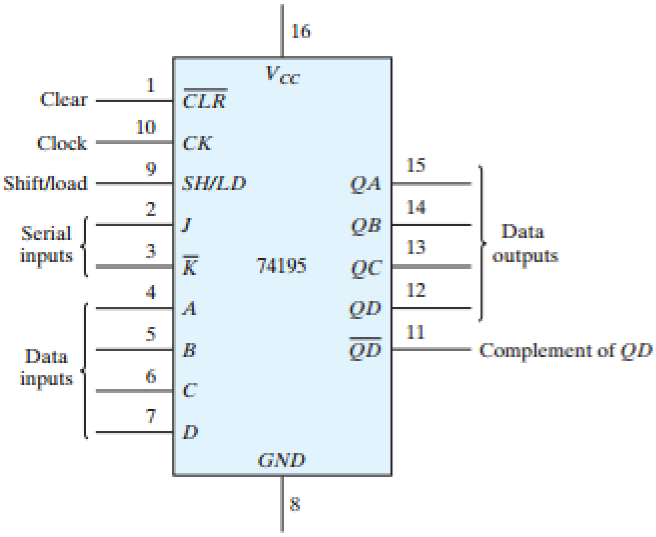
Task to do in lab: use 4 bit shift register with XOR to generate random numbers.



Serial input (SI)	Register Content
0	1010
0	0101
0	0010
1	0001
0	1000
1	0100
	1010

Cycle length = 6

Pin assignment for available 4 bit shift register IC: 74195



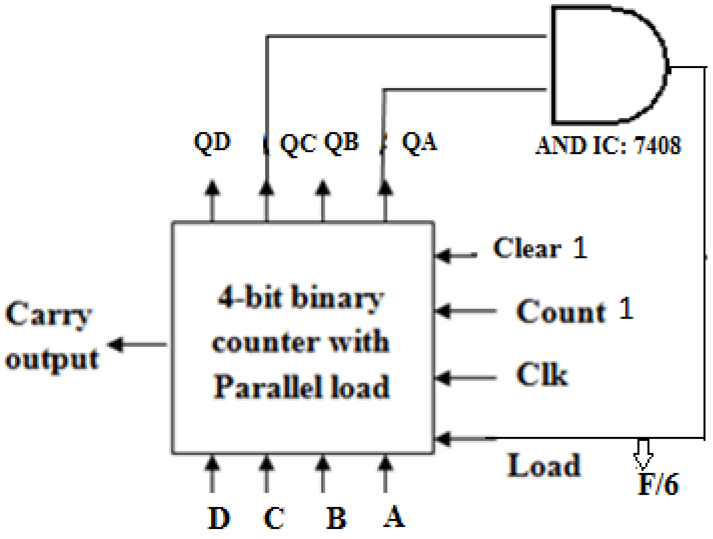
Function table

Clear	Shift/ load	Clock	J	$\overline{K}$	Serial input	Function
0	X	X	X	X	X	Asynchronous clear
1	X	0	X	X	X	No change in output
1	0	↑	X	X	X	Load input data
1	1	↑	0	0	0	Shift from QA toward QD, $QA = 0$
1	1	↑	1	1	1	Shift from QA toward QD, $QA = 1$

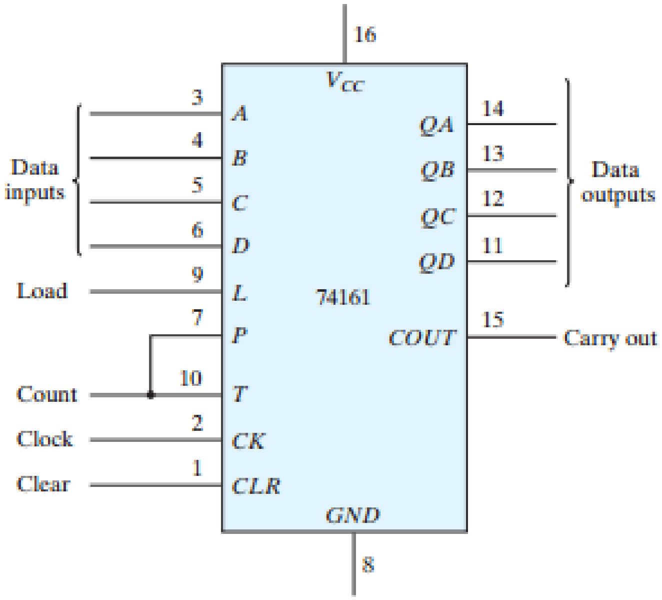
Lab 12 (Programmable counters)

Objective: implementing frequency divider.

Task to do in lab: implement frequency divider using 4 bit programmable counter. Ex: F/6



Pin assignment for available 4 bit programmable counter IC: 74161



Function table

Clear	Clock	Load	Count	Function
0	X	X	X	Clear outputs to 0
1	↑	0	X	Load input data
1	↑	1	1	Count to next binary value
1	↑	1	0	No change in output