FLIP FLOP CONVERSIONS

- SR to D
- SR to JK
- SR to T
- JK to T
- JK to D
- JK to SR
- D to T
- D to SR
- T to D

PROCEDURE FOR CONVERSION

1. Draw the block diagram of the target flip flop from the given problem.
2. Write truth table for the target flip-flop.
3. Write excitation table for the available flip-flop.
4. Draw k-map for target flip-flop.
5. Draw the block diagram.

SR(Available) to D(Target) Flip flop Conversion

- **Truth table**

Input	Present state	Next state
D	Q n	Qn+1
0	0	0
0	1	0
1	0	1
1	1	1

- **Excitation table**

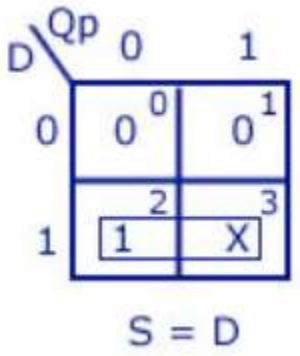
Present state	Next state	Flip flop Inputs	
Q n	Qn+1	S	R
0	0	0	X
0	0	0	1
0	1	1	0
1	1	X	0

SR to D Flip flop Conversion

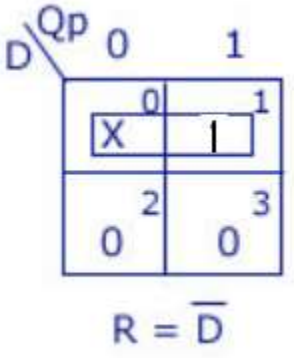
Conversion Table

Input	Present state	Next state	Flip flop Inputs	
D	Q n	Qn+1	S	R
0	0	0	0	X
0	1	0	0	1
1	0	1	1	0
1	1	1	X	0

K- MAP SIMPLIFICATION

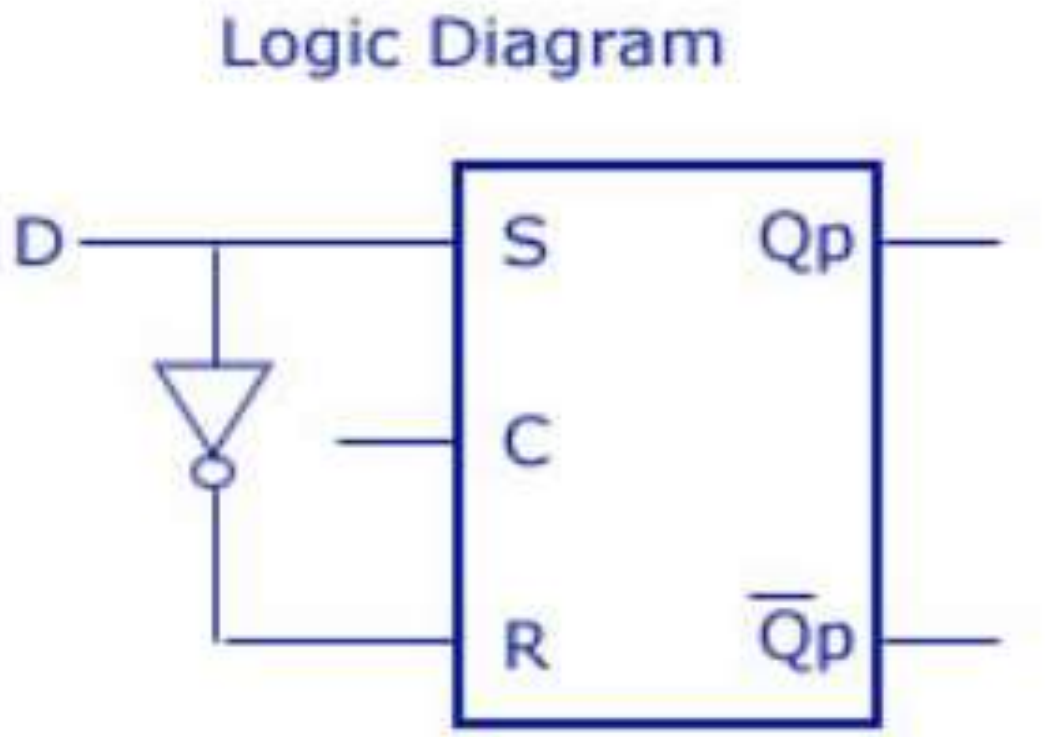


S = D



R = $\overline{D}$

SR to D



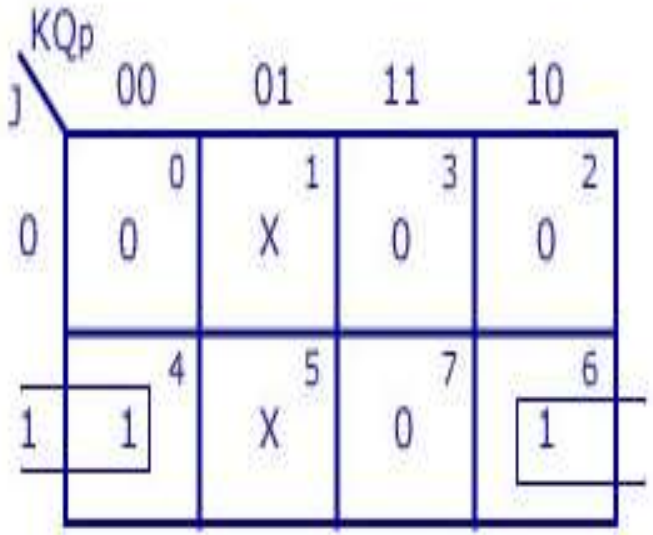
SR(Available) to JK(Target) Flip-Flop

Conversion Table

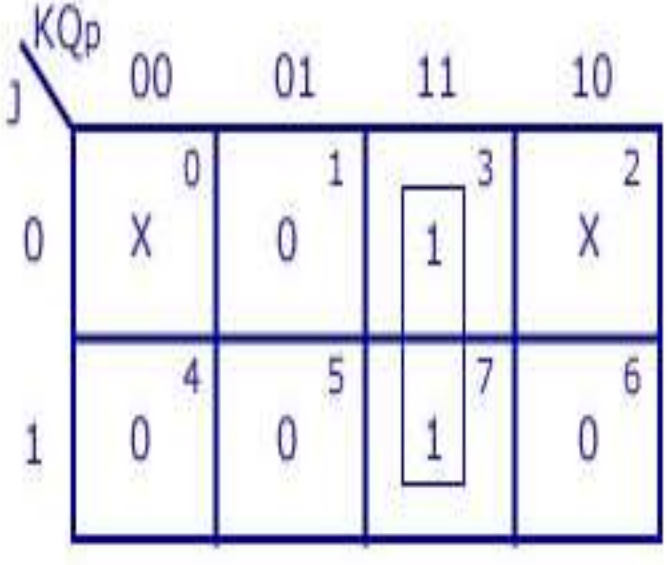
Input		Present State	Next State	Flip-Flop Inputs	
J	K	Q n	Qn+1	S	R
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	0	1
1	0	0	1	1	0
1	0	1	1	X	0
1	1	0	1	1	0
1	1	1	0	0	1

SR to JK

- K-map Simplification

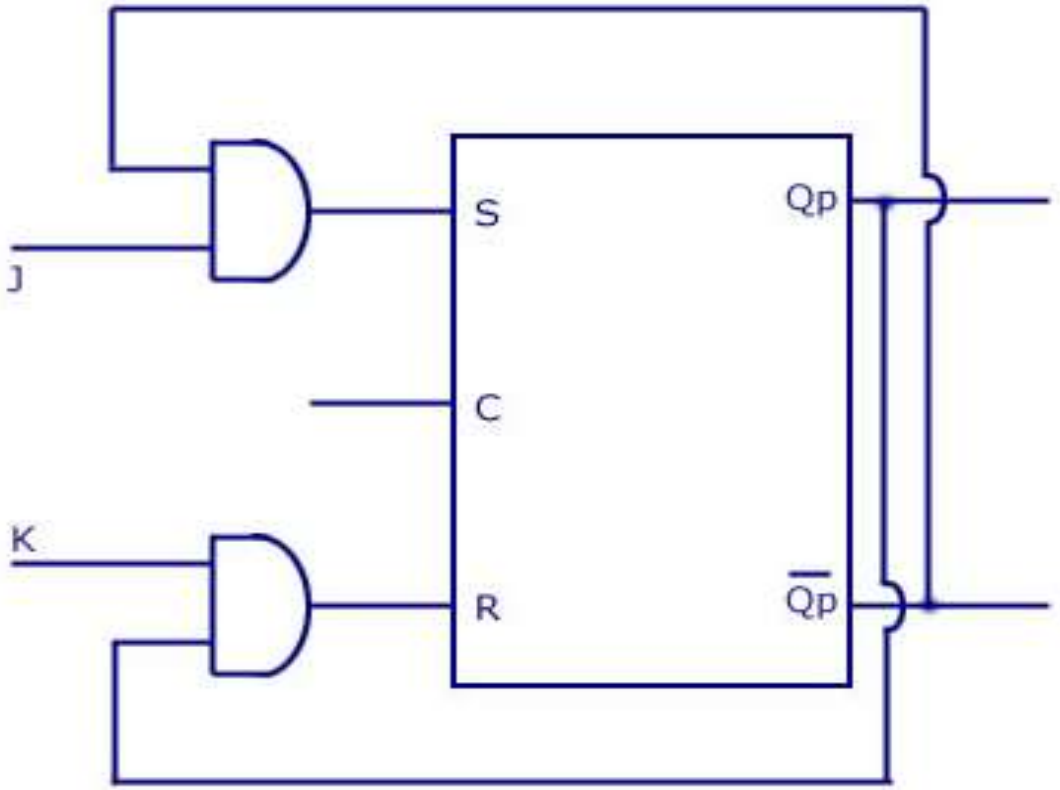


$S = \bar{J}Qp$



$R = KQp$

Logic Diagram (SR to JK)

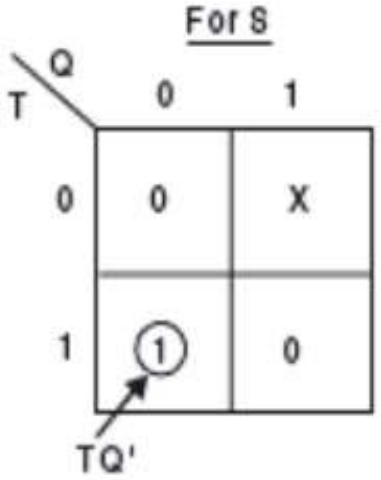


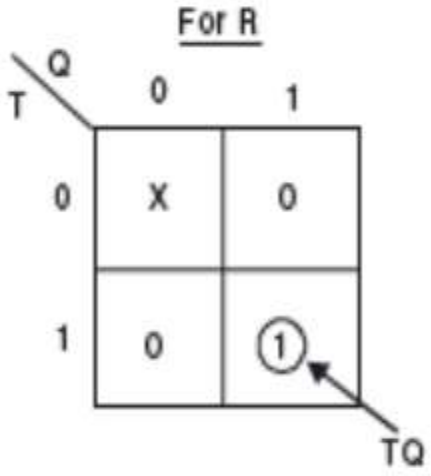
SR(Available) to T(Target)

Conversion Table

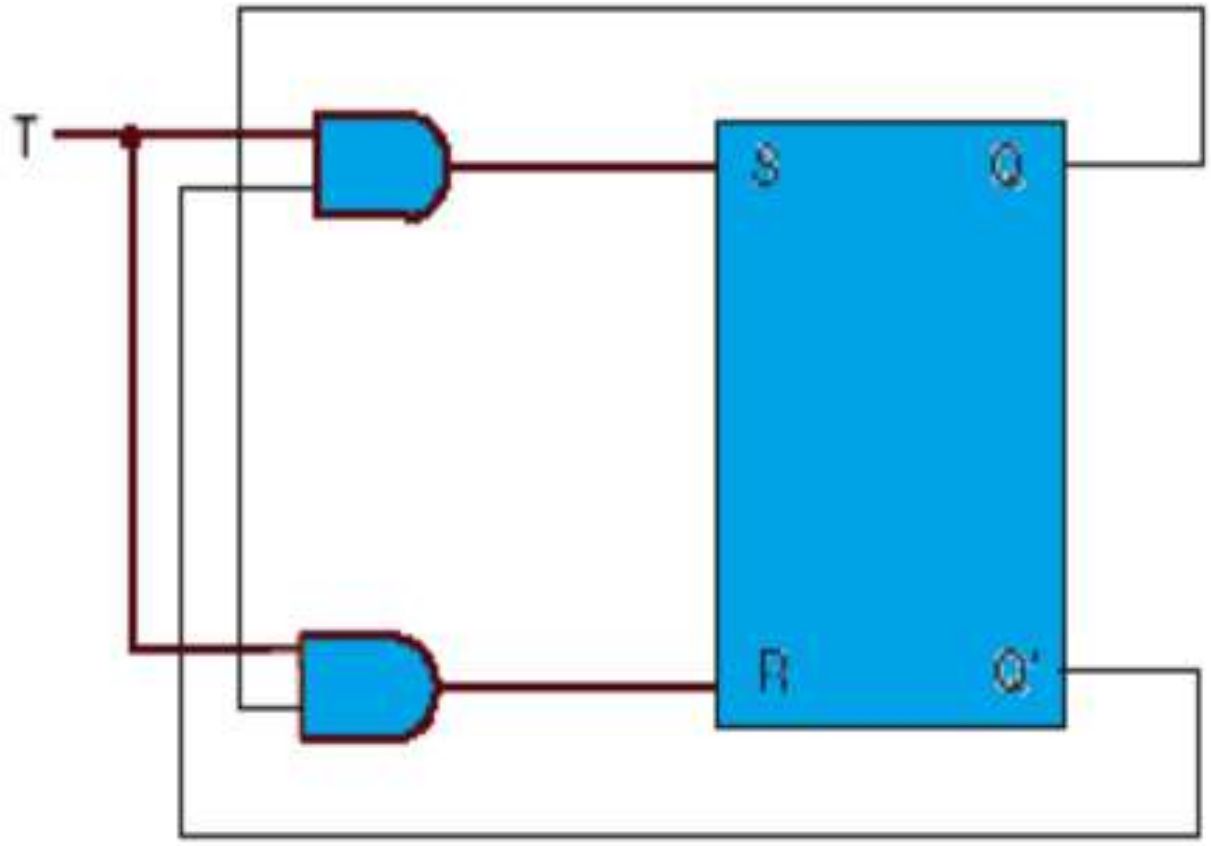
Input	Present state	Next state	Flip flop Inputs	
T	Q n	Qn+1	S	R
0	0	0	0	X
0	1	1	X	0
1	0	1	1	0
1	1	0	0	1

K- MAP SIMPLIFICATION





Logic Diagram (SR to T)



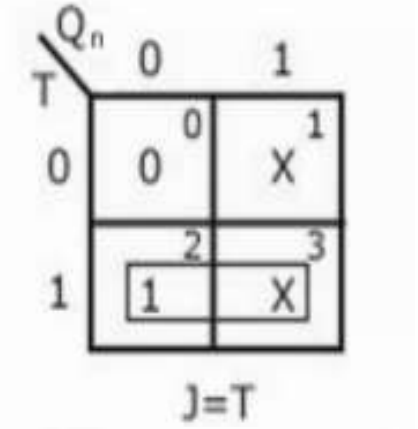
A T flip-flop using S-R flip-flop.

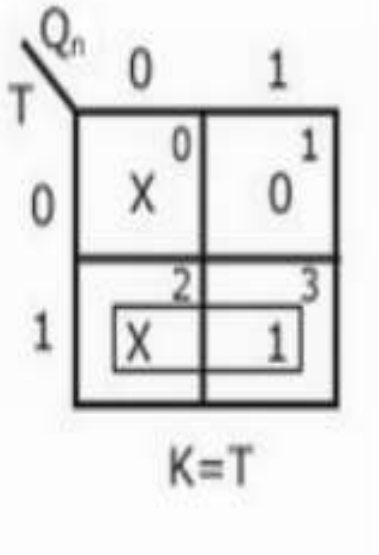
JK(Available) to T (Target)Conversion

Conversion Table

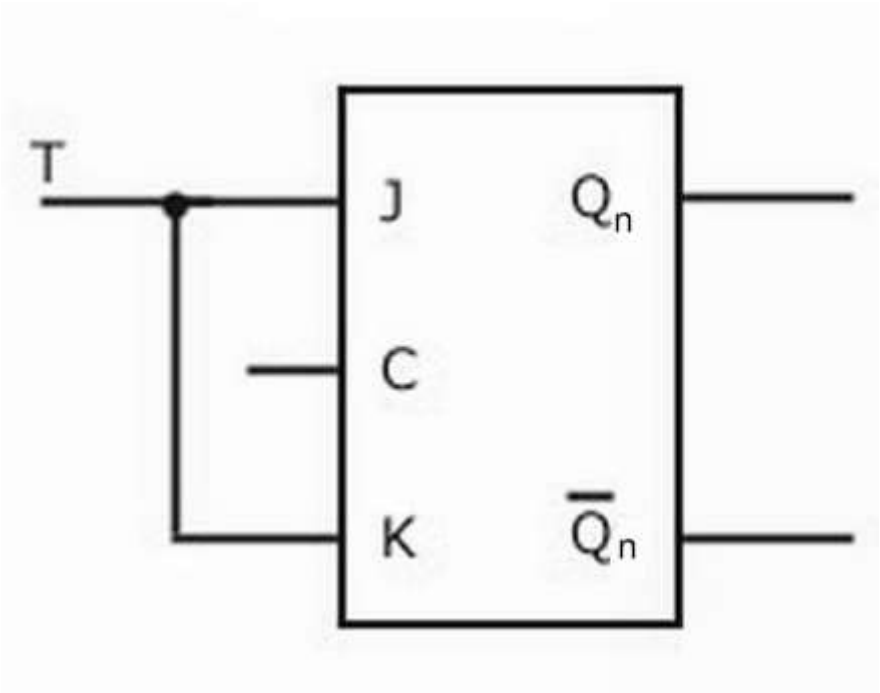
Input	Present state	Next state	Flip flop Inputs	
T	Q n	Qn+1	J	K
0	0	0	0	X
0	1	1	X	0
1	0	1	1	x
1	1	0	x	1

K- MAP SIMPLIFICATION





Logic Diagram (JK to T)

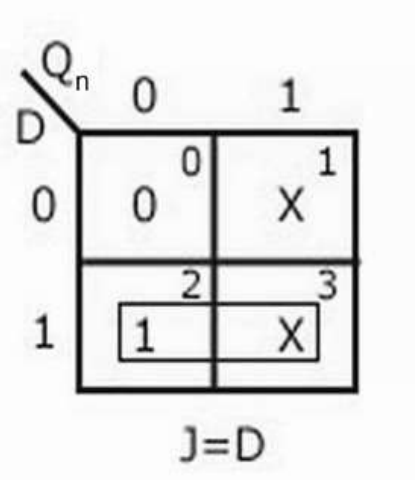


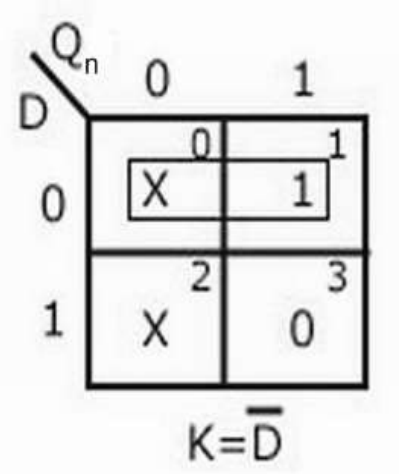
JK(Available) to D(Target)Flip-flop Conversion

Conversion Table

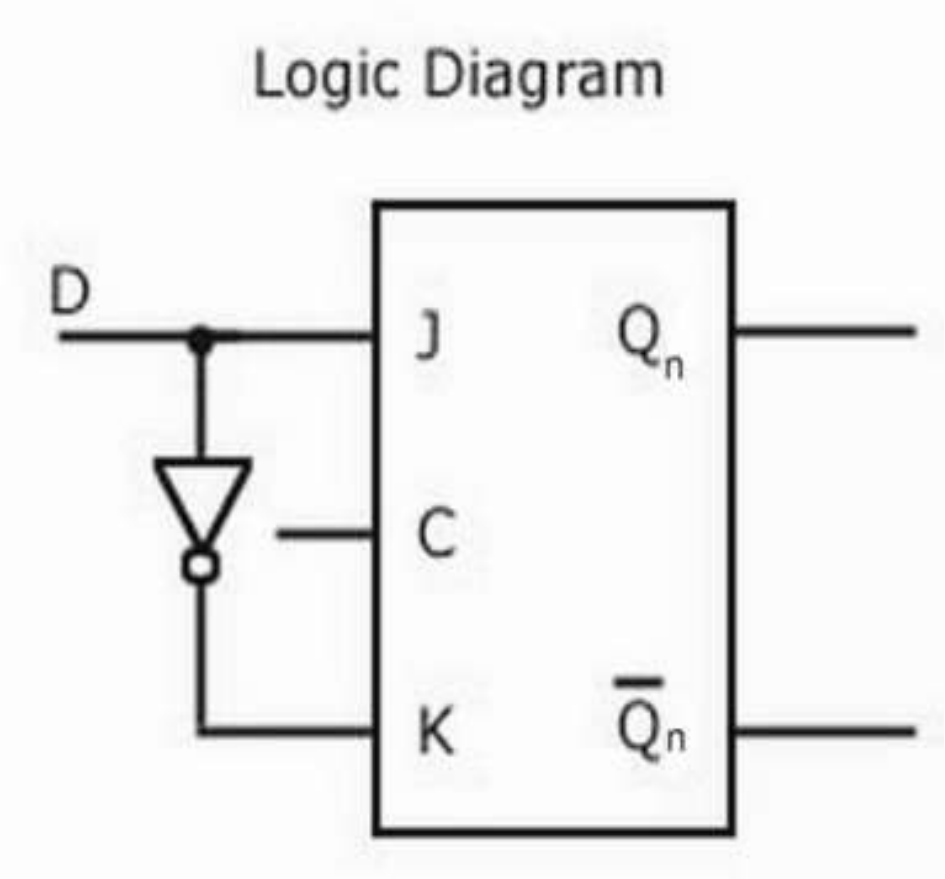
Input	Present state	Next state	Flip flop Inputs	
D	Q n	Qn+1	J	K
0	0	0	0	X
0	1	0	X	1
1	0	1	1	x
1	1	1	x	0

K- MAP SIMPLIFICATION





Logic Diagram (JK to D)

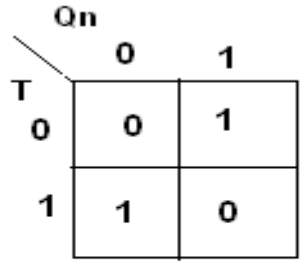


D(Available) to T(Target)Flip-Flop

Conversion Table

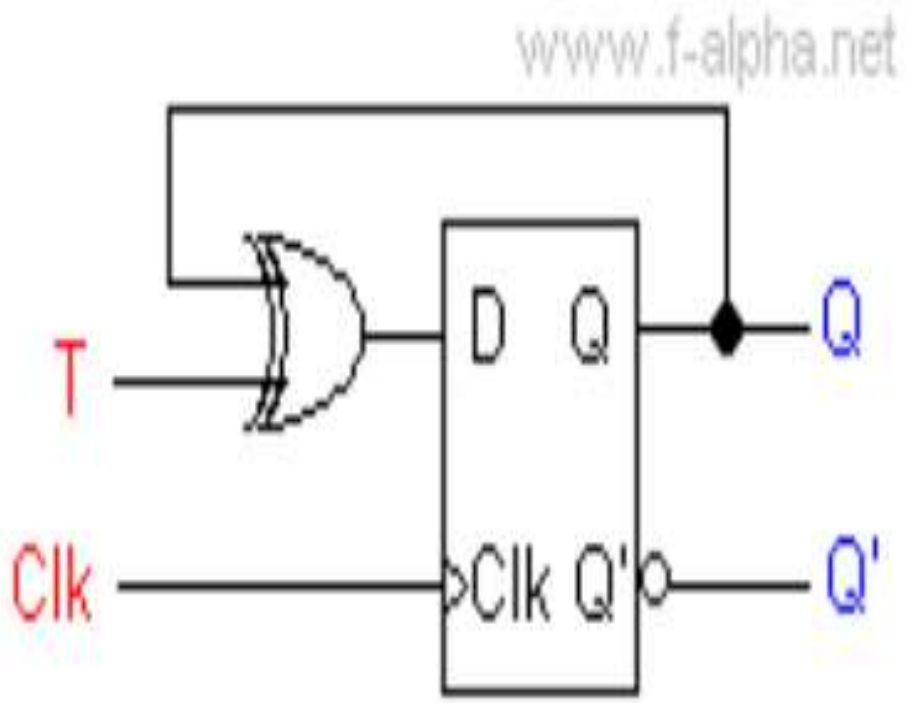
Input	Present state	Next state	Flip flop Inputs
T	Q n	Qn+1	D
0	0	0	0
0	1	1	1
1	0	1	1
1	1	0	0

K- MAP SIMPLIFICATION



D=T'Qn+TQn'

Logic Diagram(D to T)

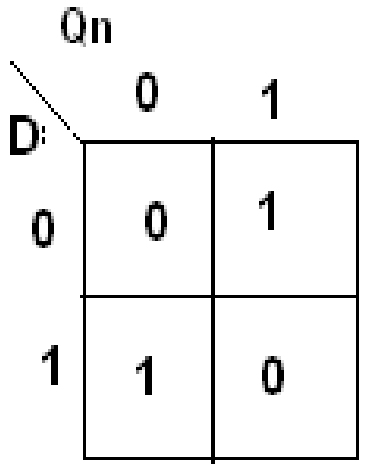


T (Available) to D(Target) Flip-flop Conversion

Conversion Table

Input	Present state	Next state	Flip flop Inputs
D	Q n	Qn+1	T
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

K- MAP SIMPLIFICATION



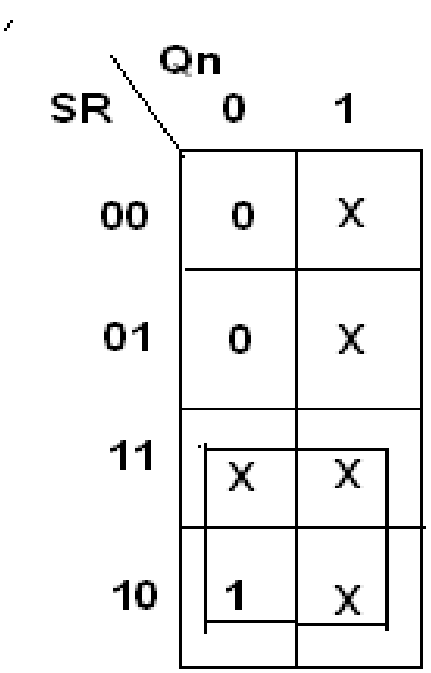
$T = DQn' + D'Qn$

JK(Available) to **SR**(Target)Flip-flop conversion

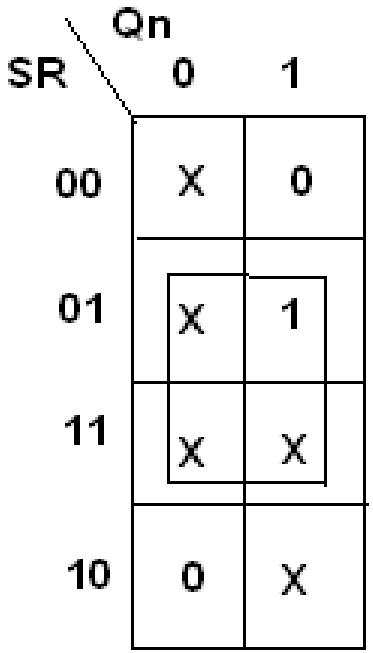
Conversion Table

Input		Present State	Next State	Flip-Flop Inputs	
S	R	Q n	Qn+1	J	K
0	0	0	0	0	X
0	0	1	1	X	0
0	1	0	0	0	X
0	1	1	0	X	1
1	0	0	1	1	X
1	0	1	1	X	0
1	1	0	X	X	X
1	1	1	X	X	X

JK(Available) to **SR**(Target)Flip-flop conversion



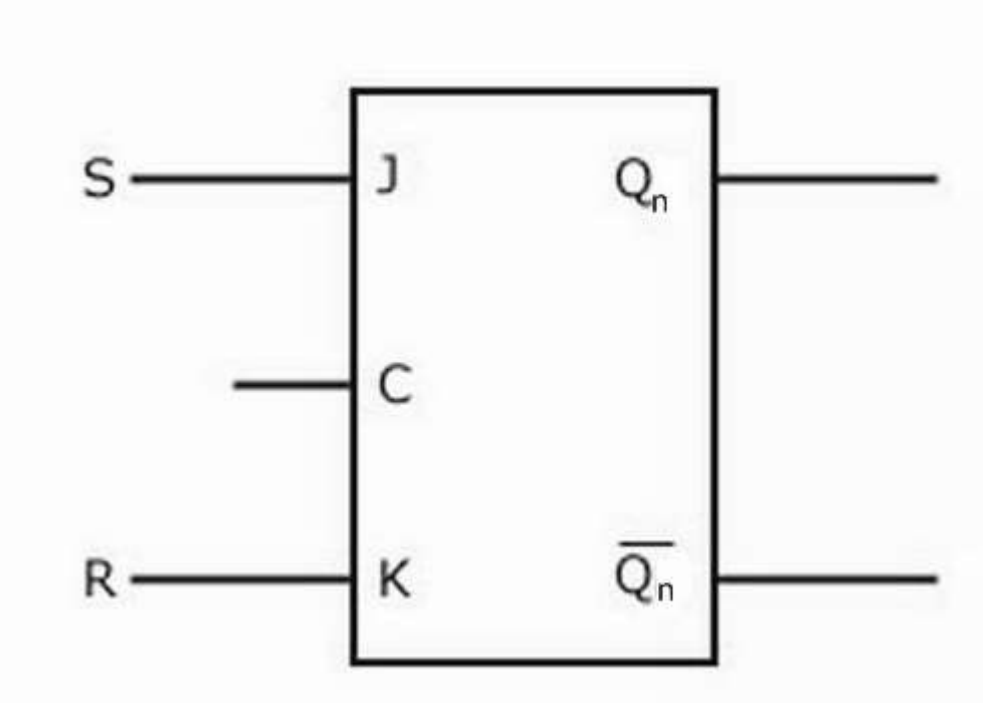
J=S



K=R

JK to SR

- Logic Diagram



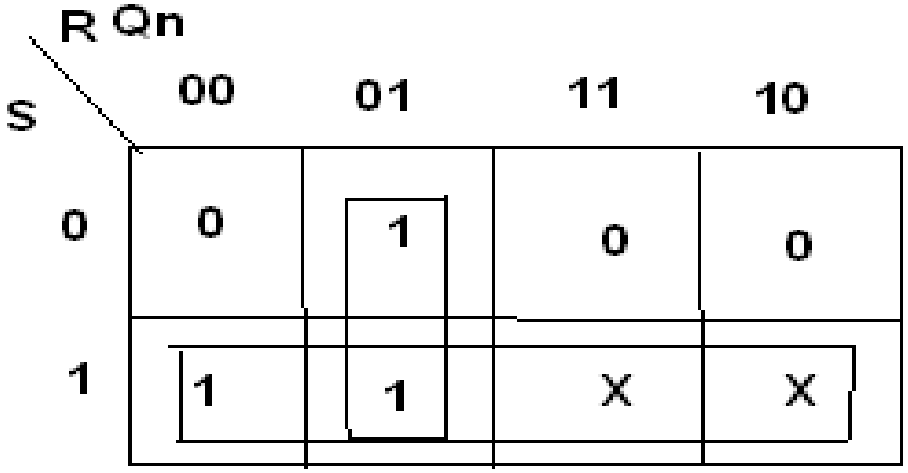
D(Available) to **SR**(Target) Flip-Flop Conversion

Conversion Table

Input		Present State	Next State	Flip-Flop Inputs	
S	R	Q n	Qn+1	J	K
0	0	0	0	0	0
0	0	1	1	1	1
0	1	0	0	0	0
0	1	1	0	0	0
1	0	0	1	1	1
1	0	1	1	1	1
1	1	0	X	X	X
1	1	1	X	X	X

D to SR

K- MAP SIMPLIFICATION



D=R'Qn+S

Logic Diagram For D to SR