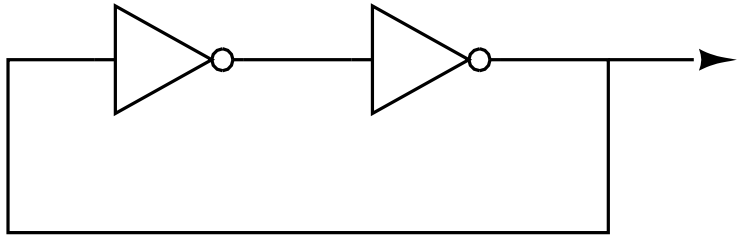
Classical State Machine Design

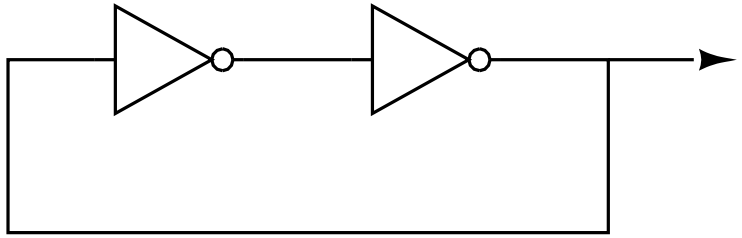
The Basic Memory Cell

Consider this circuit:



The Basic Memory Cell

Consider this circuit:

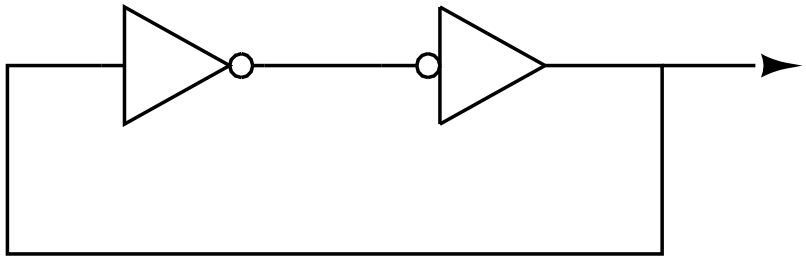


What is the output?

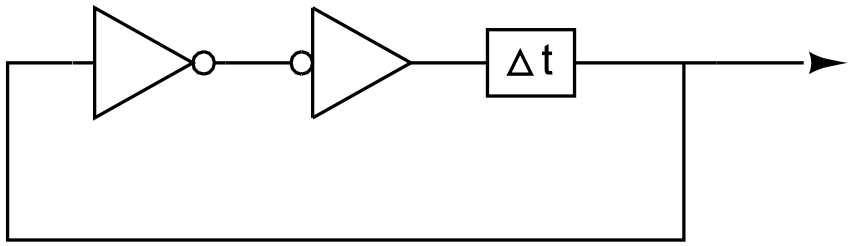
- logic high?
- logic low?
- or something else?

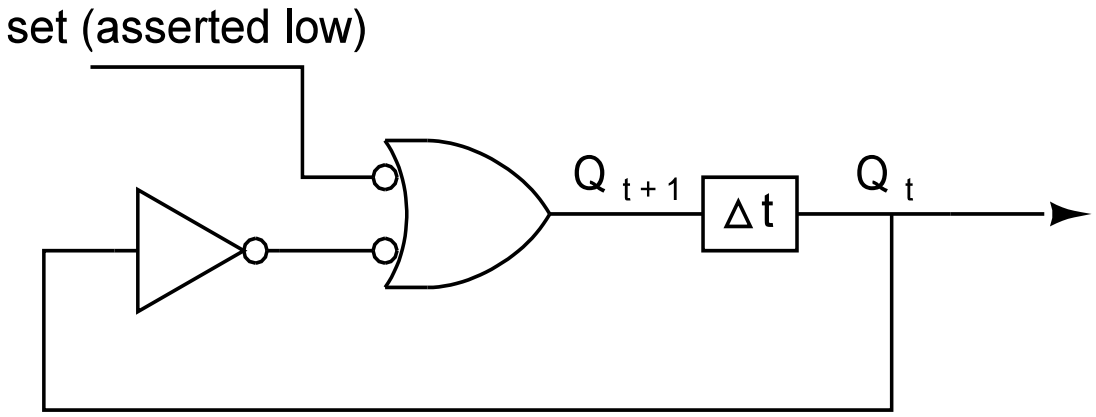
Is this circuit drawn in the best way?

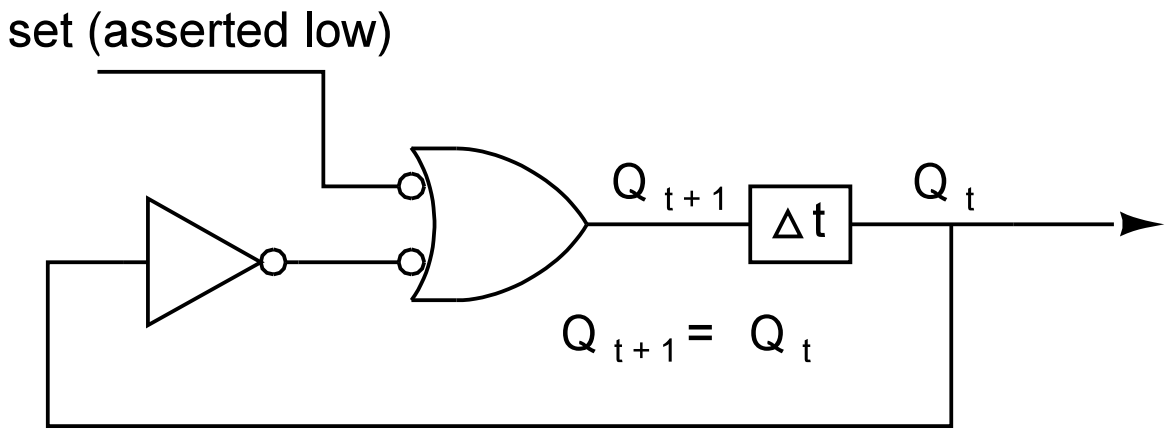
A better circuit diagram:



Recall that it takes time for signals to propagate through a component



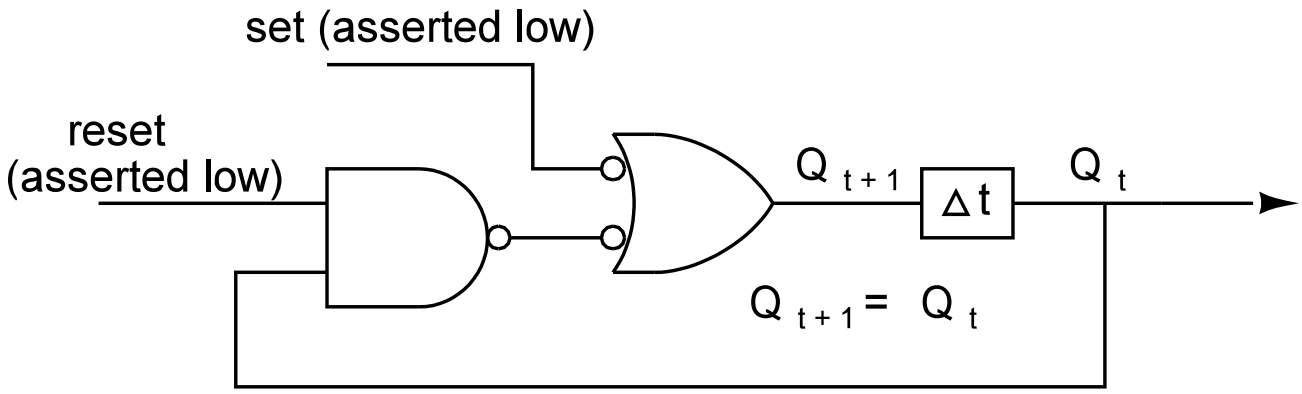

set (asserted low)
Q t + 1
Δ t
Q t



Q_t = Present state

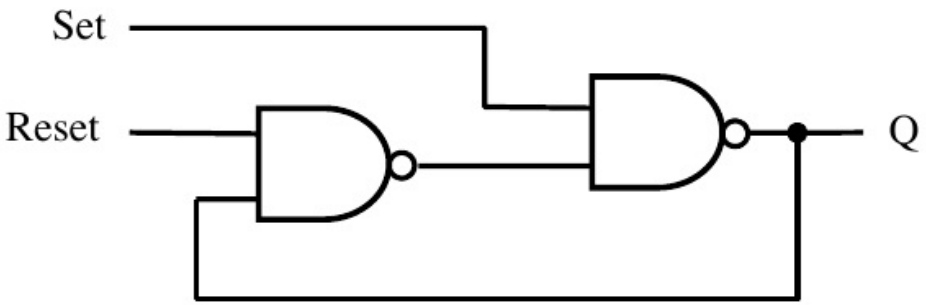
Q_{t+1} = Next state

Set-dominant Basic Cell



NAND-Centered (Set Dominant) Basic Cell

- Inputs are active low (when they are ***asserted***);
- Operation table indicates assertion, not voltage, levels.



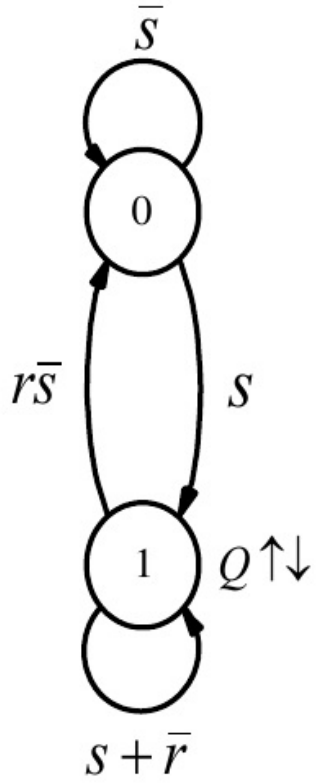
Circuit

S	R	Action	Q_{N+1}
0	0	hold	Q_N
0	1	reset	0
1	0	set	1
1	1	set	1

Operation Table

$Q_n \rightarrow Q_{n+1}$	Inputs S	R
$0 \rightarrow 0$	0	ϕ
$0 \rightarrow 1$	1	ϕ
$1 \rightarrow 0$	0	1
$1 \rightarrow 1$	1	ϕ
	ϕ	0

Excitation Table



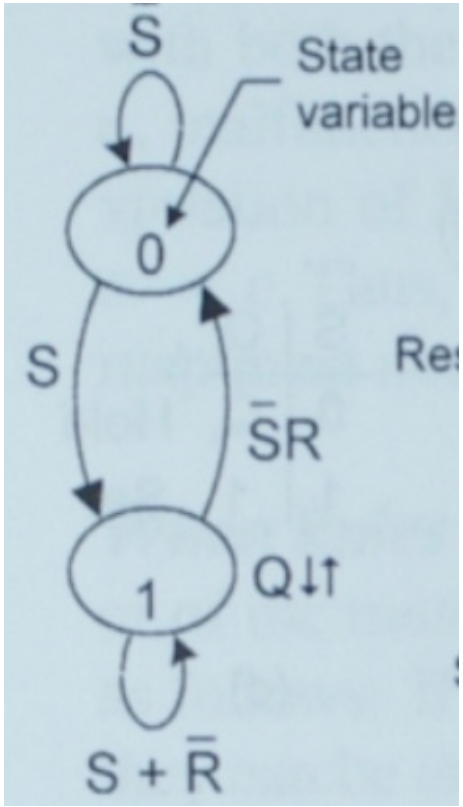
State Diagram

S	R	Action	Q_{N+1}
0	0	hold	Q_N
0	1	reset	0
1	0	set	1
1	1	set	1

Operation Table

$Q_n \rightarrow Q_{n+1}$	Inputs S	R
$0 \rightarrow 0$	0	ϕ
$0 \rightarrow 1$	1	ϕ
$1 \rightarrow 0$	0	1
$1 \rightarrow 1$	1	ϕ
	ϕ	0

Excitation Table



Combined Form of the Basic Cell

$Q_n \rightarrow Q_{n+1}$	Inputs S	R
$0 \rightarrow 0$	0	ϕ
$0 \rightarrow 1$	1	ϕ
$1 \rightarrow 0$	0	1
$1 \rightarrow 1$	1	ϕ
	ϕ	0

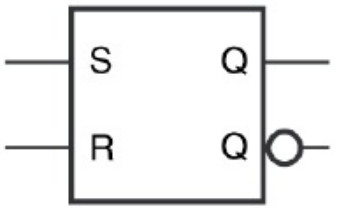
Excitation Table NAND-centered

$Q_n \rightarrow Q_{n+1}$	Inputs S	R
$0 \rightarrow 0$	0	ϕ
	ϕ	1
$0 \rightarrow 1$	1	0
$1 \rightarrow 0$	ϕ	1
$1 \rightarrow 1$	ϕ	0

Excitation Table NOR-centered

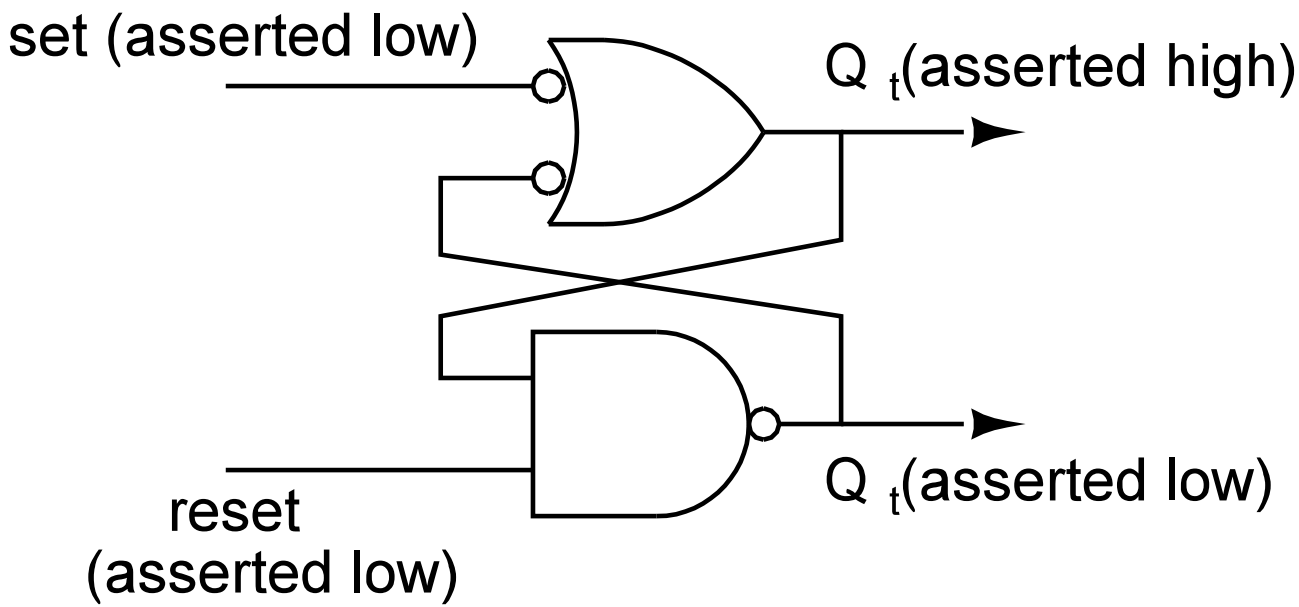
$Q_n \rightarrow Q_{n+1}$	Inputs S	R
$0 \rightarrow 0$	0	ϕ
$0 \rightarrow 1$	1	0
$1 \rightarrow 0$	0	1
$1 \rightarrow 1$	ϕ	0

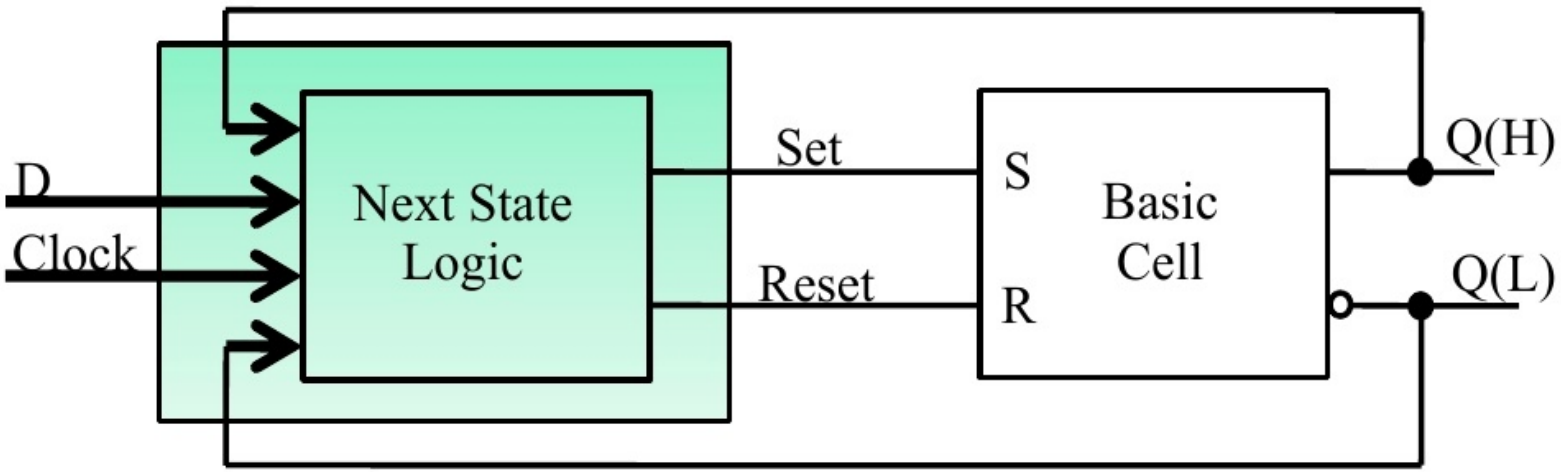
Excitation Table Combined Form

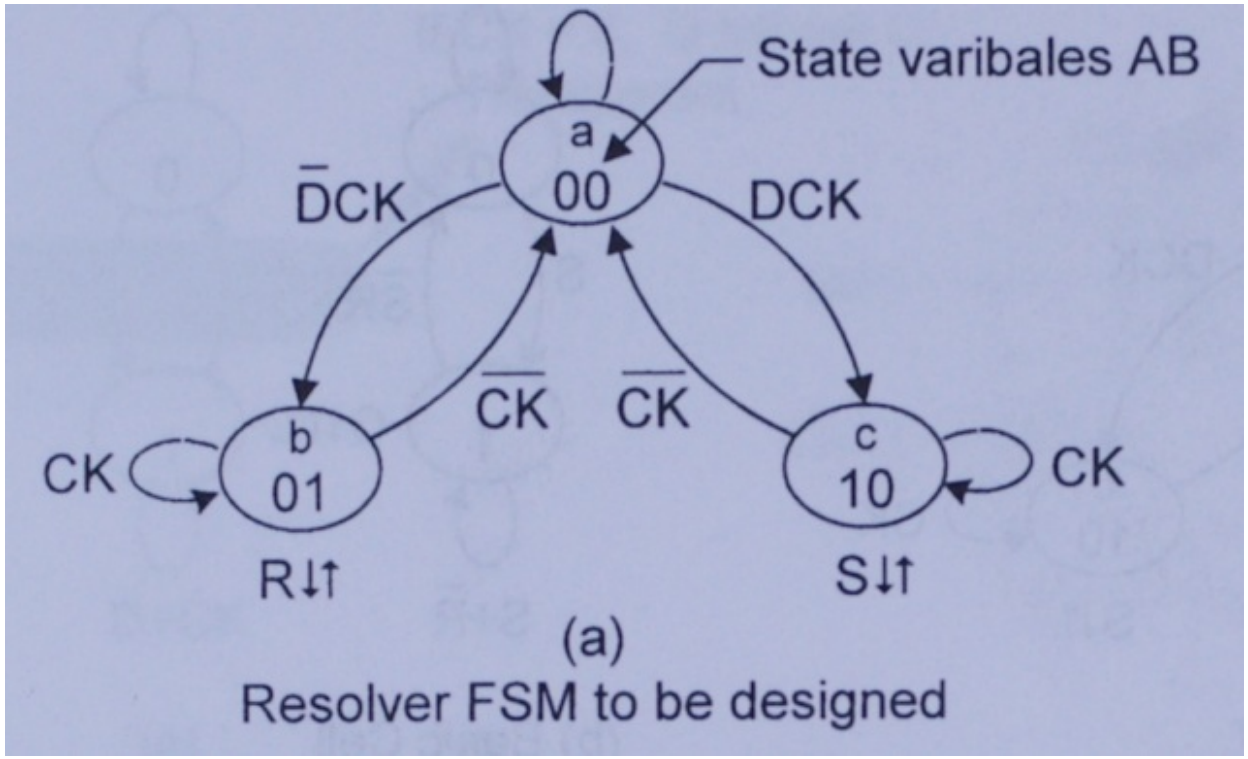


Basic Cell

Set dominant basic cell



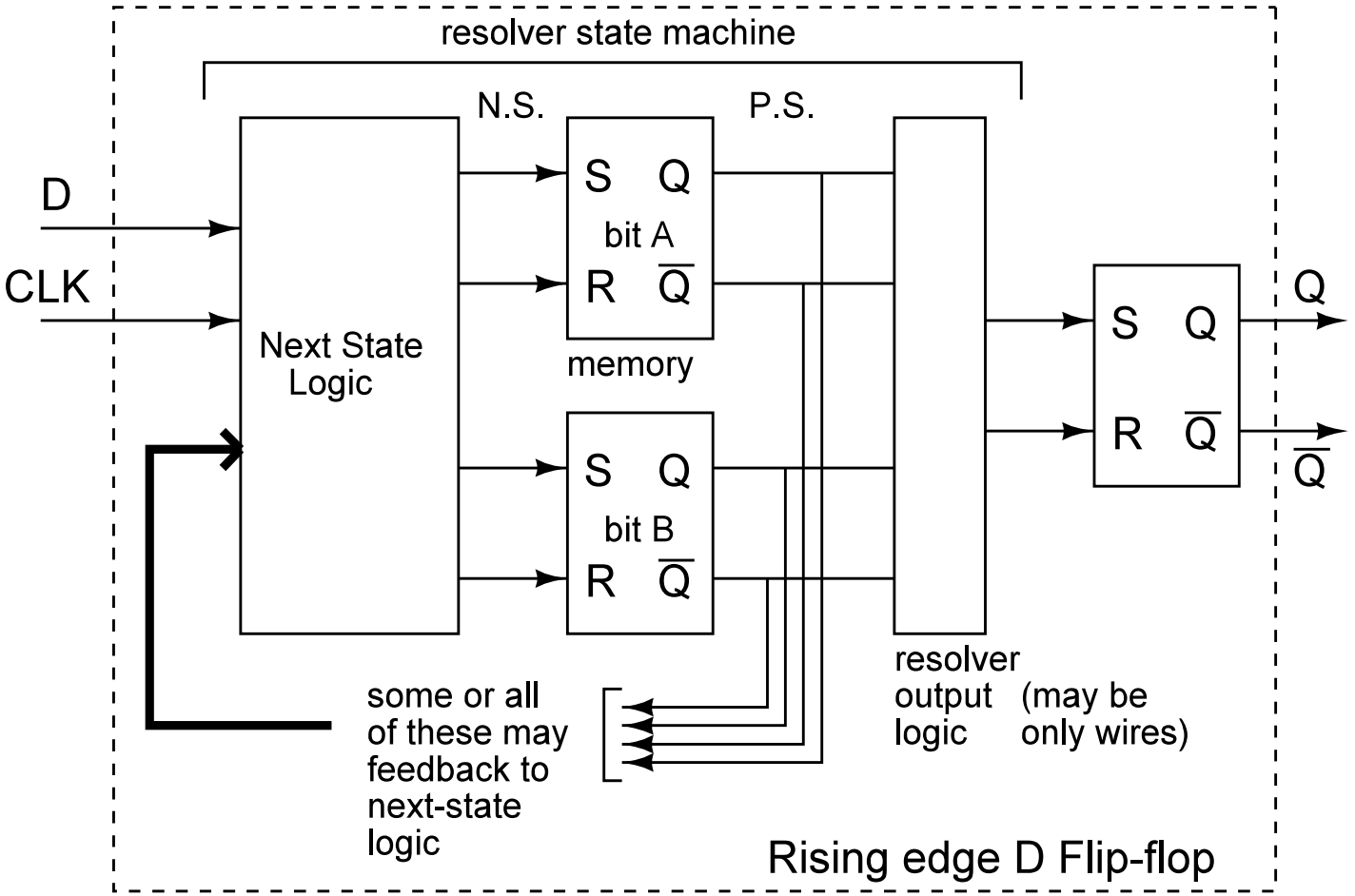


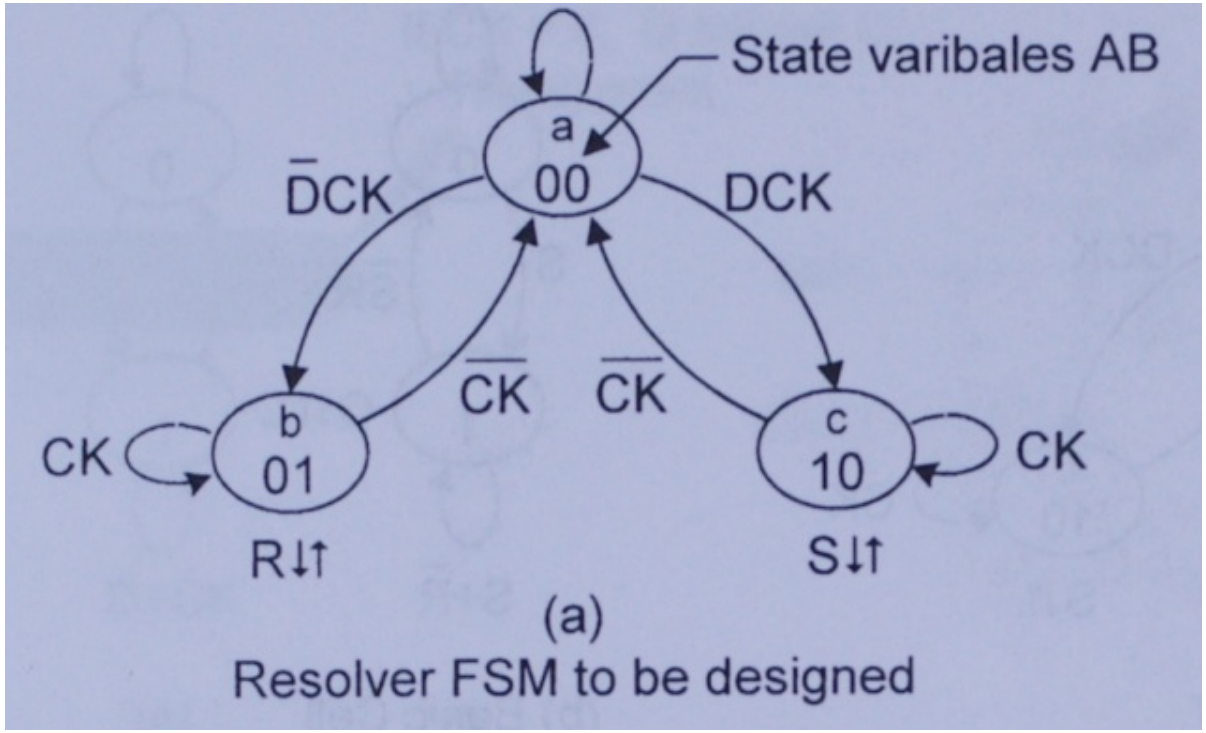


(a)

Resolver FSM to be designed

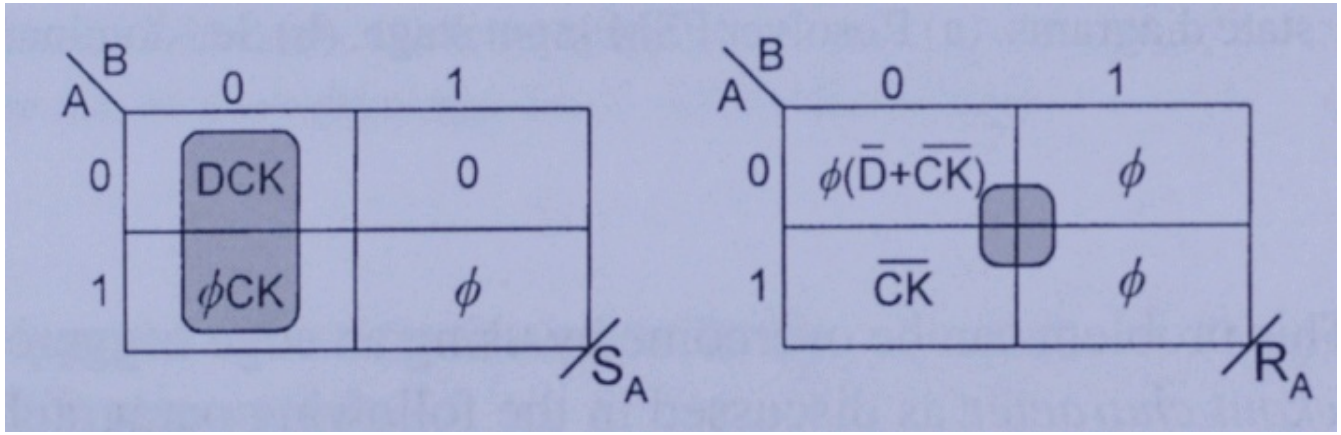
Note that the resolver state diagram has three states and thus two state bits are required. Each state bit will be implemented with a Set-Reset (SR) basic cell, sometimes called an SR-Latch.

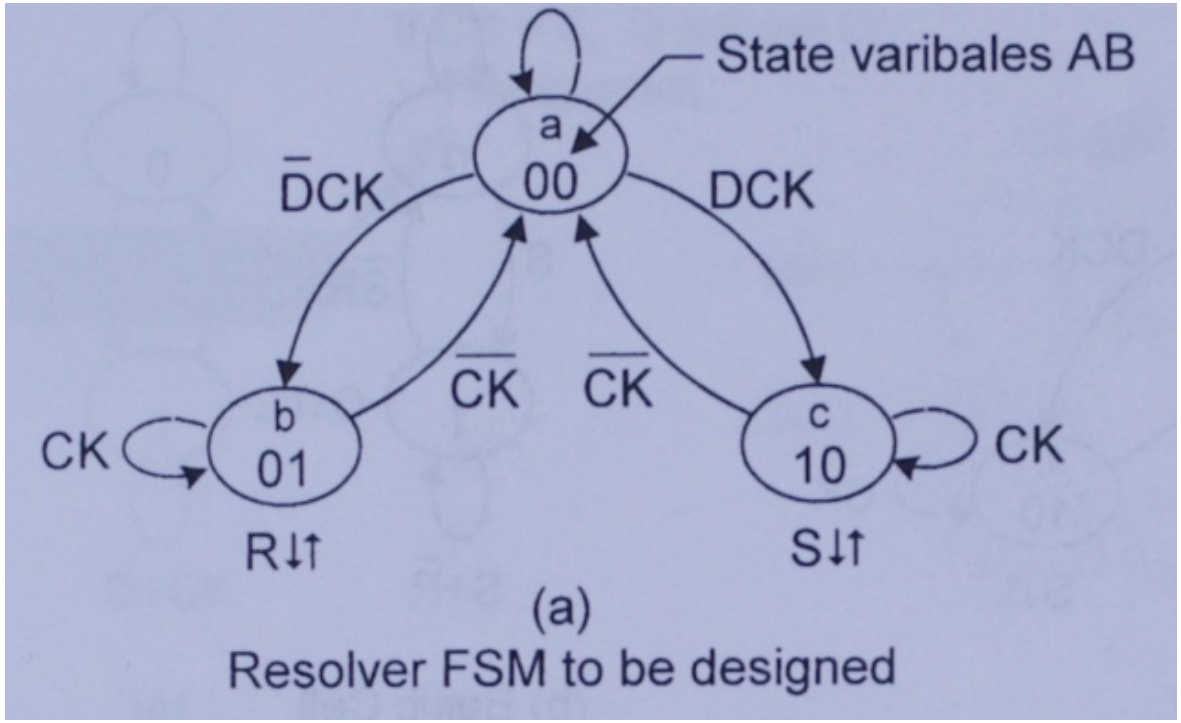




$Q_n \rightarrow Q_{n+1}$	Inputs S	R
$0 \rightarrow 0$	0	ϕ
$0 \rightarrow 1$	1	0
$1 \rightarrow 0$	0	1
$1 \rightarrow 1$	ϕ	0

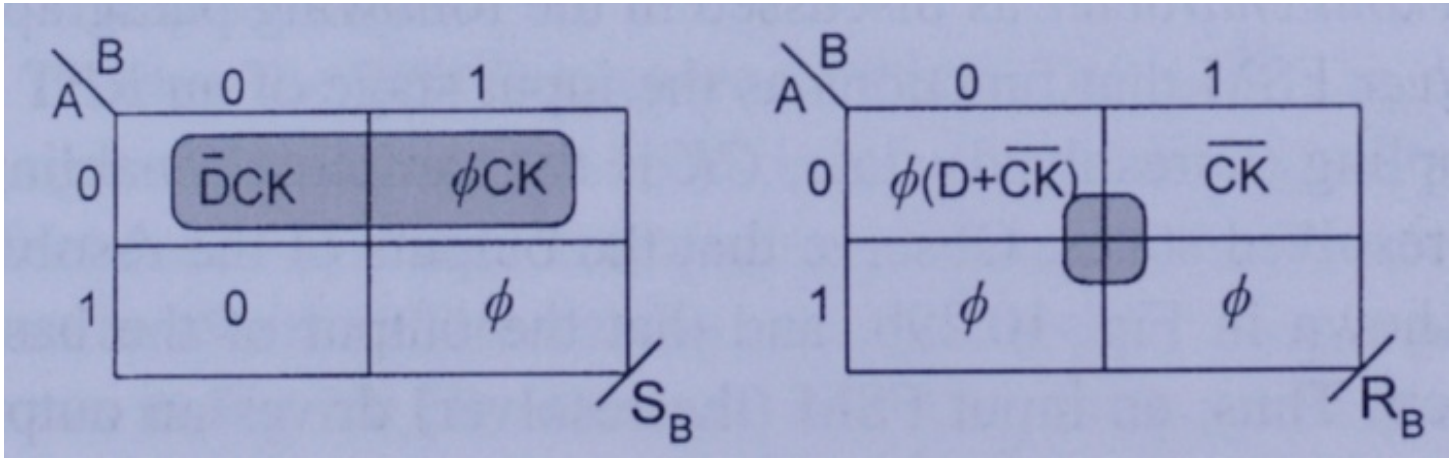
EV-Kmap to document and reduce the needed Next State logic, bit A





$Q_n \rightarrow Q_{n+1}$	Inputs S	R
$0 \rightarrow 0$	0	ϕ
$0 \rightarrow 1$	1	0
$1 \rightarrow 0$	0	1
$1 \rightarrow 1$	ϕ	0

EV-Kmap to document and reduce the needed Next State logic, bit B



blank page