

CS M51A

Logic Design of Digital Systems

Winter 2021

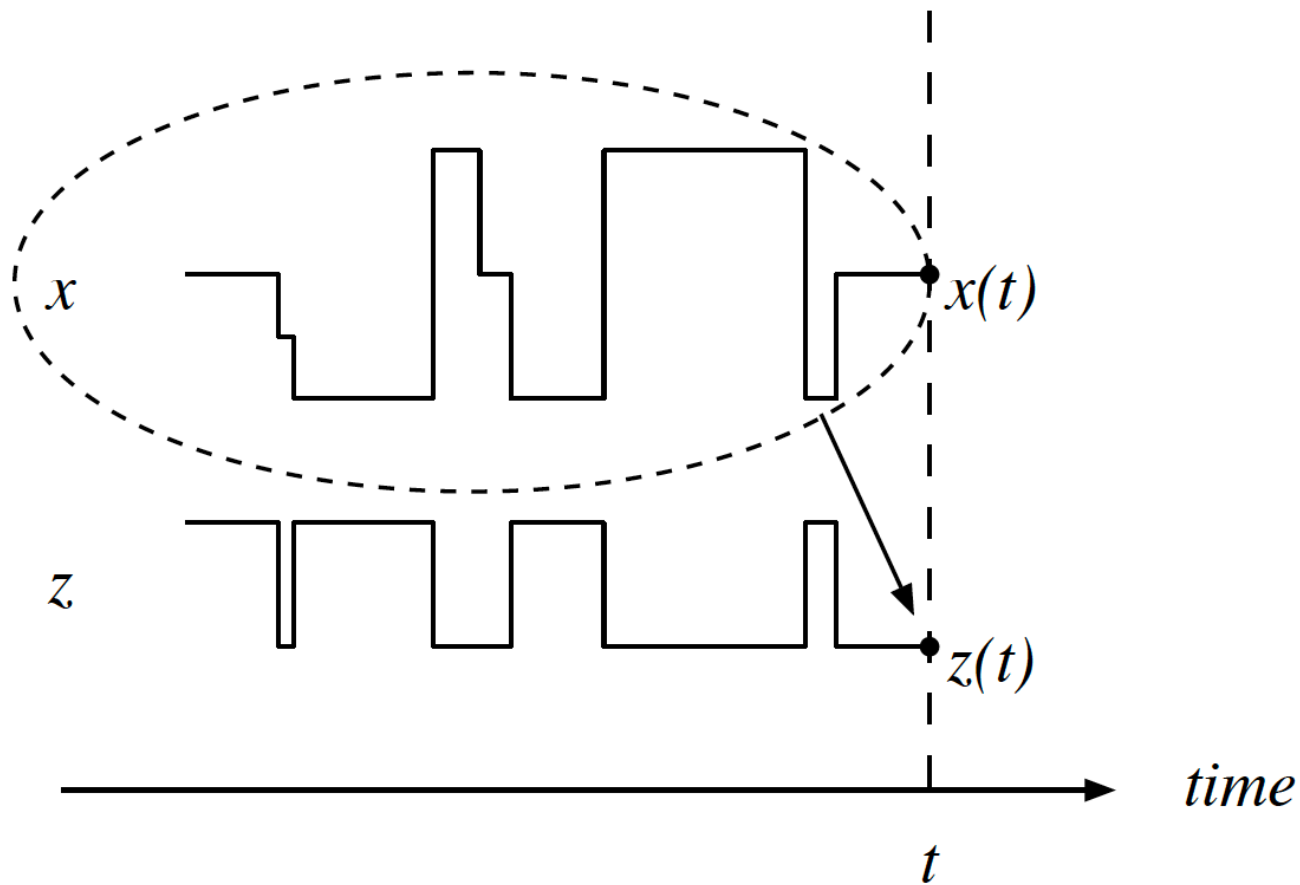
Some slides borrowed and modified from:

M.D. Ercegovic, T. Lang and J. Moreno, Introduction to Digital Systems.

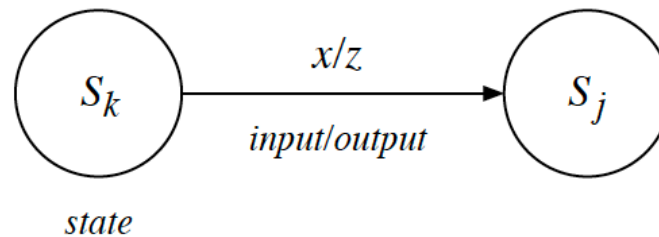
SEQUENTIAL SYSTEMS

DEFINITION

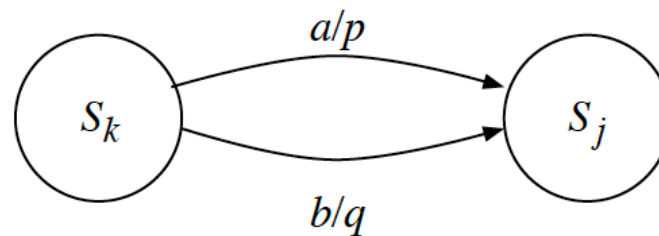
$$z(t) = F(x(0, t))$$



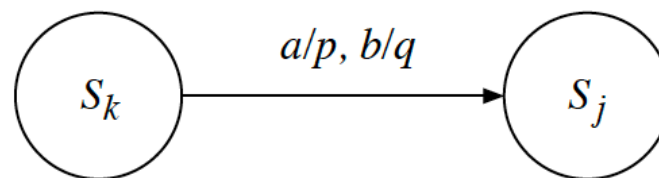
REPRESENTATION OF STATE-TRANSITION AND OUTPUT FUNCTIONS WITH STATE DIAGRAM



(a)



Complete state diagram



Simplified state diagram

(b)

What is the state diagram for this example?

Input: $x(t) \in \{a, b, c\}$

Output: $z(t) \in \{0, 1\}$

State: $s(t) \in \{S_0, S_1, S_2, S_3\}$

Initial state: $s(0) = S_0$

Functions: Transition and output functions:

PS	Input			
	a	b	c	
S_0	S_0	S_1	S_1	0
S_1	S_2	S_0	S_1	1
S_2	S_2	S_3	S_0	1
S_3	S_0	S_1	S_2	0
	NS			Output

Reduction of the State Set

- k-DISTINGUISHABLE STATES: DIFF. OUTPUT SEQUENCES

$$z(x(t, t + k - 1), S_v) \neq z(x(t, t + k - 1), S_w)$$

- k-EQUIVALENT STATES: NOT DISTINGUISHABLE FOR SEQUENCES OF LENGTH k
- P_k : PARTITION OF STATES INTO k-EQUIVALENT CLASSES

Reduction of the State Set

Input: $x(t) \in \{a, b, c\}$
Output: $z(t) \in \{0, 1\}$
State: $s(t) \in \{A, B, C, D, E, F\}$
Initial state: $s(0) = A$

Functions: TRANSITION AND OUTPUT

PS	$x = a$	$x = b$	$x = c$
A	$E, 0$	$D, 1$	$B, 0$
B	$F, 0$	$D, 0$	$A, 1$
C	$E, 0$	$B, 1$	$D, 0$
D	$F, 0$	$B, 0$	$C, 1$
E	$C, 0$	$F, 1$	$F, 0$
F	$B, 0$	$C, 0$	$F, 1$
	NS, z		

Reduction of the State Set

Input: $x(t) \in \{a, b, c\}$
 Output: $z(t) \in \{0, 1\}$
 State: $s(t) \in \{A, B, C, D, E, F\}$
 Initial state: $s(0) = A$

Functions: TRANSITION AND OUTPUT

PS	$x = a$	$x = b$	$x = c$
A	$E, 0$	$D, 1$	$B, 0$
B	$F, 0$	$D, 0$	$A, 1$
C	$E, 0$	$B, 1$	$D, 0$
D	$F, 0$	$B, 0$	$C, 1$
E	$C, 0$	$F, 1$	$F, 0$
F	$B, 0$	$C, 0$	$F, 1$
	NS, z		

A and B ARE 1-DISTINGUISHABLE BECAUSE

$$z(b, A) \neq z(b, B)$$

A and C ARE 1-EQUIVALENT BECAUSE

$$z(x(t), A) = z(x(t), C), \quad \text{for all } x(t) \in I$$

Reduction of the State Set

Input: $x(t) \in \{a, b, c\}$
 Output: $z(t) \in \{0, 1\}$
 State: $s(t) \in \{A, B, C, D, E, F\}$
 Initial state: $s(0) = A$

Functions: TRANSITION AND OUTPUT

A and *C* ARE ALSO 2-EQUIVALENT BECAUSE

<i>PS</i>	$x = a$	$x = b$	$x = c$
<i>A</i>	<i>E</i> , 0	<i>D</i> , 1	<i>B</i> , 0
<i>B</i>	<i>F</i> , 0	<i>D</i> , 0	<i>A</i> , 1
<i>C</i>	<i>E</i> , 0	<i>B</i> , 1	<i>D</i> , 0
<i>D</i>	<i>F</i> , 0	<i>B</i> , 0	<i>C</i> , 1
<i>E</i>	<i>C</i> , 0	<i>F</i> , 1	<i>F</i> , 0
<i>F</i>	<i>B</i> , 0	<i>C</i> , 0	<i>F</i> , 1
	<i>NS, z</i>		

$z(aa, A) = z(aa, C) = 00$
 $z(ab, A) = z(ab, C) = 01$
 $z(ac, A) = z(ac, C) = 00$
 $z(ba, A) = z(ba, C) = 10$
 $z(bb, A) = z(bb, C) = 10$
 $z(bc, A) = z(bc, C) = 11$
 $z(ca, A) = z(ca, C) = 00$
 $z(cb, A) = z(cb, C) = 00$
 $z(cc, A) = z(cc, C) = 01$

Clicker Question

Which one is correct?

PS	$x = a$	$x = b$	$x = c$
A	$E, 0$	$D, 1$	$B, 0$
B	$F, 0$	$D, 0$	$A, 1$
C	$E, 0$	$B, 1$	$D, 0$
D	$F, 0$	$B, 0$	$C, 1$
E	$C, 0$	$F, 1$	$F, 0$
F	$B, 0$	$C, 0$	$F, 1$
	NS, z		

- a) E and F are 1-Distinguishable
- b) E and F are 1-Equivalent
- c) B and F are 1-Equivalent
- d) B and F are 2-Equivalent
- e) Both a, and c are correct

Reduction of the State Set

PS	$x(t) = a$	$x(t) = b$	$x(t) = c$
A	0	1	0
B	0	0	1
C	0	1	0
D	0	0	1
E	0	1	0
F	0	0	1
	NS, z		

- 1-EQUIVALENT IF SAME "row pattern"

$$P_1 = (A, C, E) \quad (B, D, F)$$

Reduction of the State Set

- NUMBER THE CLASSES IN P_1
- TWO STATES ARE IN THE SAME CLASS OF P_2
IF THEIR SUCCESSOR COLUMNS HAVE THE SAME NUMBERS

PS	$x = a$	$x = b$	$x = c$
A	$E, 0$	$D, 1$	$B, 0$
B	$F, 0$	$D, 0$	$A, 1$
C	$E, 0$	$B, 1$	$D, 0$
D	$F, 0$	$B, 0$	$C, 1$
E	$C, 0$	$F, 1$	$F, 0$
F	$B, 0$	$C, 0$	$F, 1$
	NS, z		

	1			2		
P_1	(A, C, E)			(B, D, F)		
a	1	1	1	2	2	2
b	2	2	2	2	2	1
c	2	2	2	1	1	2

BY IDENTIFYING IDENTICAL COLUMNS OF SUCCESSORS, WE GET

$$P_2 = (A, C, E) \quad (B, D) \quad (F)$$

Reduction of the State Set

- SAME PROCESS TO OBTAIN THE NEXT PARTITION:

PS	$x = a$	$x = b$	$x = c$
A	$E, 0$	$D, 1$	$B, 0$
B	$F, 0$	$D, 0$	$A, 1$
C	$E, 0$	$B, 1$	$D, 0$
D	$F, 0$	$B, 0$	$C, 1$
E	$C, 0$	$F, 1$	$F, 0$
F	$B, 0$	$C, 0$	$F, 1$
	NS, z		

	1	2	3
P_2	(A, C, E)	$(B, D),$	(F)
a	1 1 1	3 3	
b	2 2 3	2 2	
c	2 2 3	1 1	

$$P_3 = (A, C) (E) (B, D) (F)$$

- SIMILARLY, WE DETERMINE $P_4 = (A, C) (E) (B, D) (F)$

BECAUSE $P_4 = P_3$ THIS IS ALSO THE EQUIVALENCE PARTITION P

Reduction of the State Set

THE MINIMAL SYSTEM:

PS	$x = a$	$x = b$	$x = c$
A	$E, 0$	$D, 1$	$B, 0$
B	$F, 0$	$D, 0$	$A, 1$
C	$E, 0$	$B, 1$	$D, 0$
D	$F, 0$	$B, 0$	$C, 1$
E	$C, 0$	$F, 1$	$F, 0$
F	$B, 0$	$C, 0$	$F, 1$
	NS, z		

PS	$x = a$	$x = b$	$x = c$
A	$E, 0$	$B, 1$	$B, 0$
B	$F, 0$	$B, 0$	$A, 1$
E	$A, 0$	$F, 1$	$F, 0$
F	$B, 0$	$A, 0$	$F, 1$
	NS, z		

BINARY CODING

PS	a	b	c
A	E, 0	B, 1	B, 0
B	F, 0	B, 0	A, 1
E	A, 0	F, 1	F, 0
F	B, 0	A, 0	F, 1
	NS, z		

What is the state diagram for this system?

BINARY CODING

Input code		Output code		State assignment	
$x(t)$	$x_1(t)x_0(t)$	$z(t)$		$s(t)$	$s_1(t)s_0(t)$
a	00	0	0	A	00
b	01	1	1	B	01
c	10			E	10
				F	11

BINARY CODING

Input code		Output code		State assignment	
$x(t)$	$x_1(t)x_0(t)$	$z(t)$		$s(t)$	$s_1(t)s_0(t)$
a	00	0	0	A	00
b	01	1	1	B	01
c	10			E	10
				F	11

PS	a	b	c
A	E, 0	B, 1	B, 0
B	F, 0	B, 0	A, 1
E	A, 0	F, 1	F, 0
F	B, 0	A, 0	F, 1
	NS, z		

THE RESULTING BINARY SPECIFICATION:

$s_1(t)s_0(t)$	$x_1x_0 = 00$	$x_1x_0 = 01$	$x_1x_0 = 10$
00	10, 0	01, 1	01, 0
01	11, 0	01, 0	00, 1
10	00, 0	11, 1	11, 0
11	01, 0	00, 0	11, 1
	$s_1(t+1)s_0(t+1), z$		

BINARY CODING

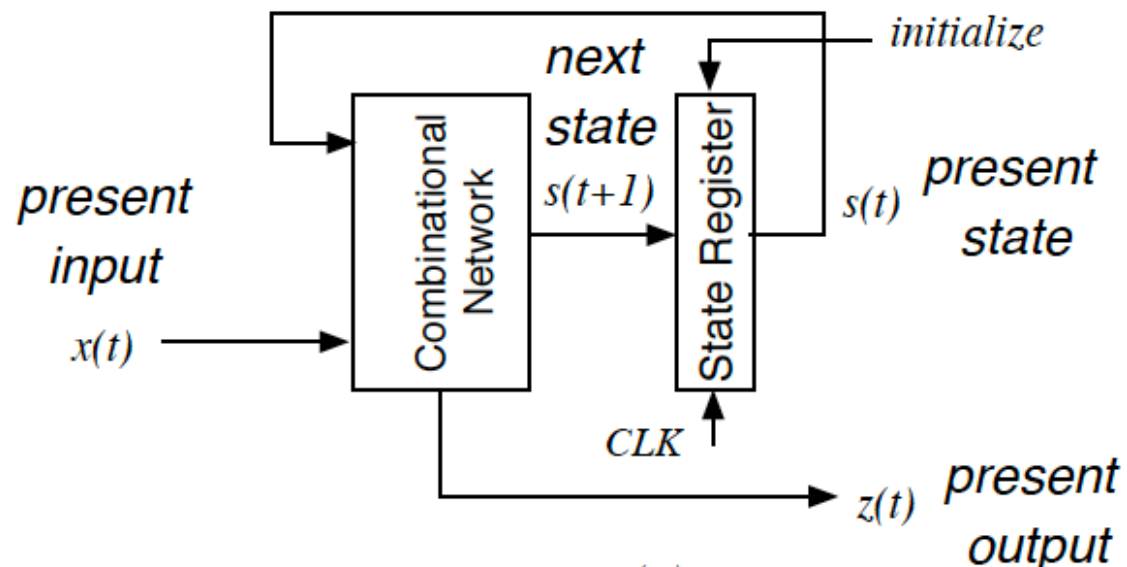
$s_1(t)s_0(t)$	$x_1x_0 = 00$	$x_1x_0 = 01$	$x_1x_0 = 10$
00	10, 0	01, 1	01, 0
01	11, 0	01, 0	00, 1
10	00, 0	11, 1	11, 0
11	01, 0	00, 0	11, 1
	$s_1(t + 1)s_0(t + 1), z$		

What is the state diagram for this system?

Sequential System Implementation

State-transition function $s(t+1) = G(s(t), x(t))$

Output function $z(t) = H(s(t), x(t))$



MEALY AND MOORE MACHINES

Mealy machine

$$z(t) = H(s(t), x(t))$$

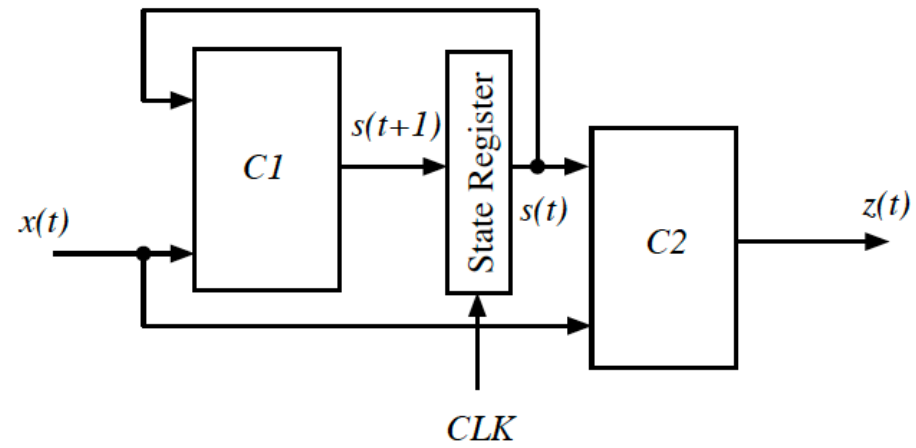
$$s(t + 1) = G(s(t), x(t))$$

Moore machine

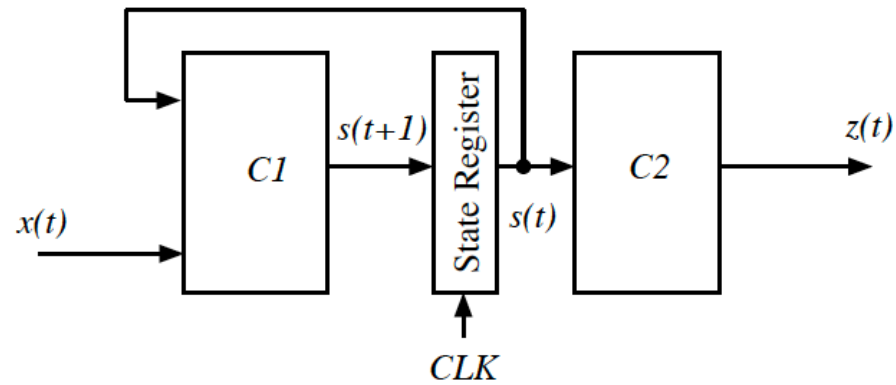
$$z(t) = H(s(t))$$

$$s(t + 1) = G(s(t), x(t))$$

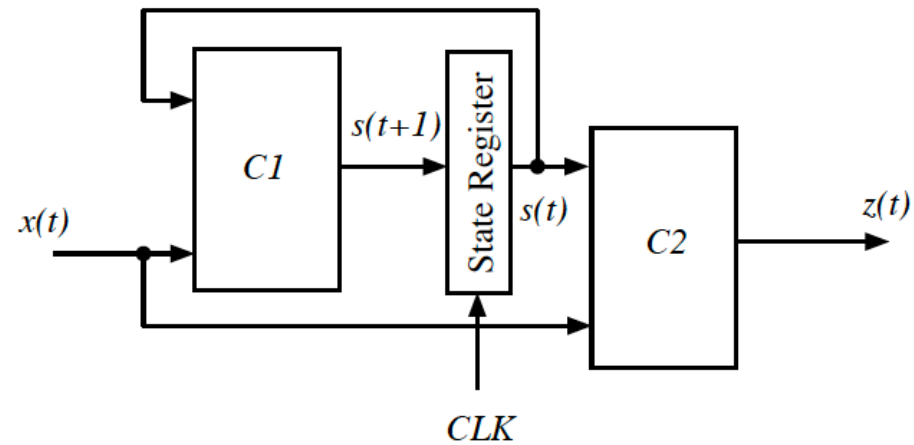
MEALY AND MOORE MACHINES



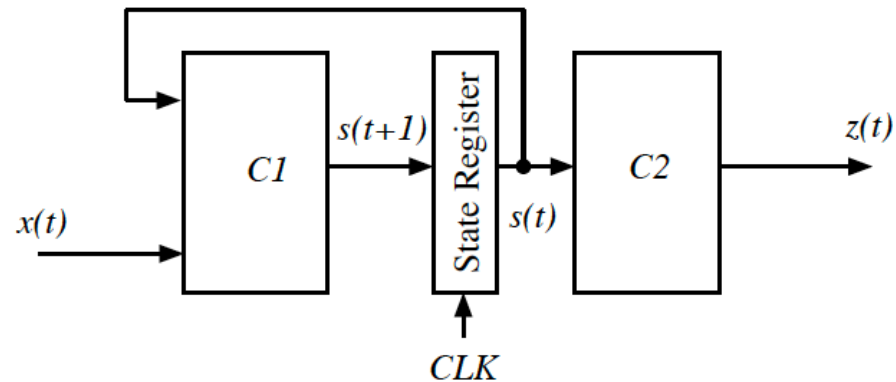
(a)



MEALY AND MOORE MACHINES



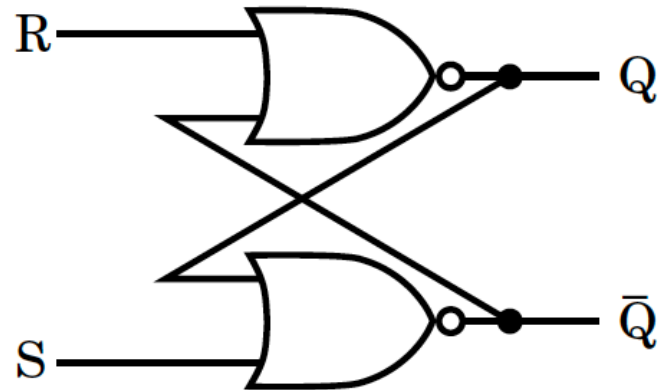
(a)



How to implement the state register?

SR Latch with NOR gates

- SR Latch with NOR gates



Functional Description of SR Latch

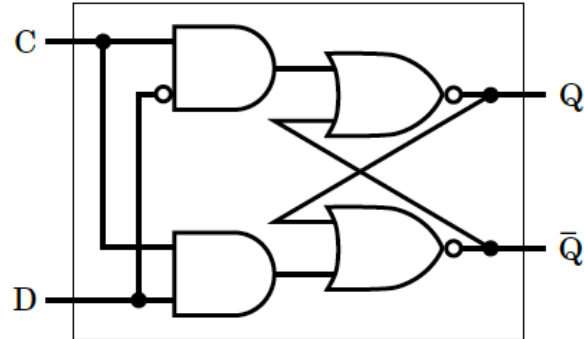
S	R	Q	\bar{Q}	
0	0	Q	\bar{Q}	Latch state (no change)
0	1	0	1	Reset state
1	0	1	0	Set state
1	1	?	?	Undefined

Functional Description of SR Latch

S	R	Q	\overline{Q}	
0	0	Q	\overline{Q}	Latch state (no change)
0	1	0	1	Reset state
1	0	1	0	Set state
1	1	?	?	Undefined

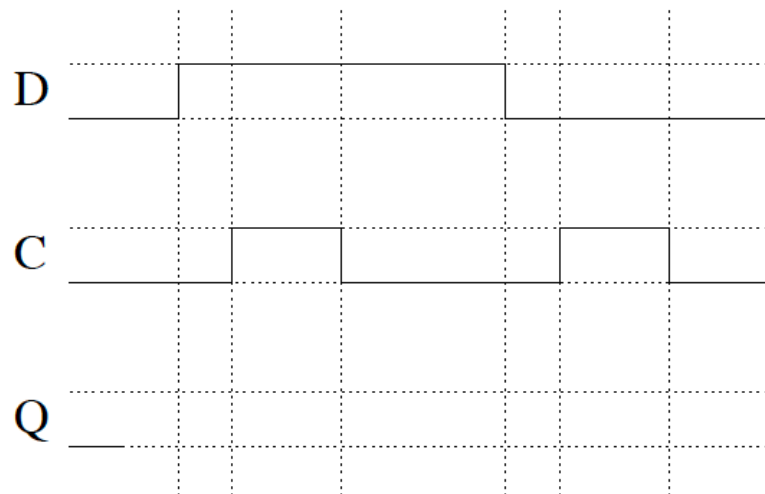
- Advantages:
 - Can “remember” value
 - Natural “reset” and “set” signals
(SR=01 is “reset” to 0, SR=10 is “set” to 1)
- Disadvantages:
 - SR=11 input has to be avoided
 - No notion of a clock or change at discrete points in time yet

The D Latch



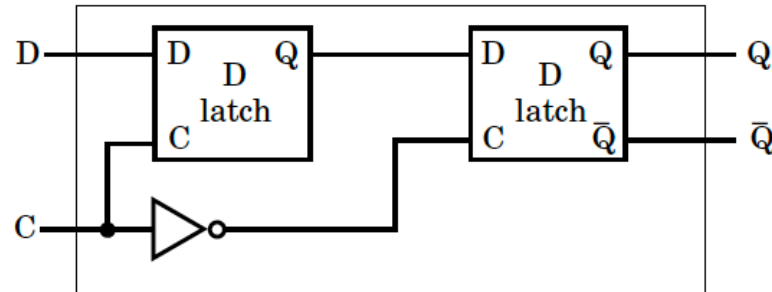
C	D	Next state of Q
0	X	No change
1	0	$Q = 0$ (Reset)
1	1	$Q = 1$ (Set)

Graphical example:

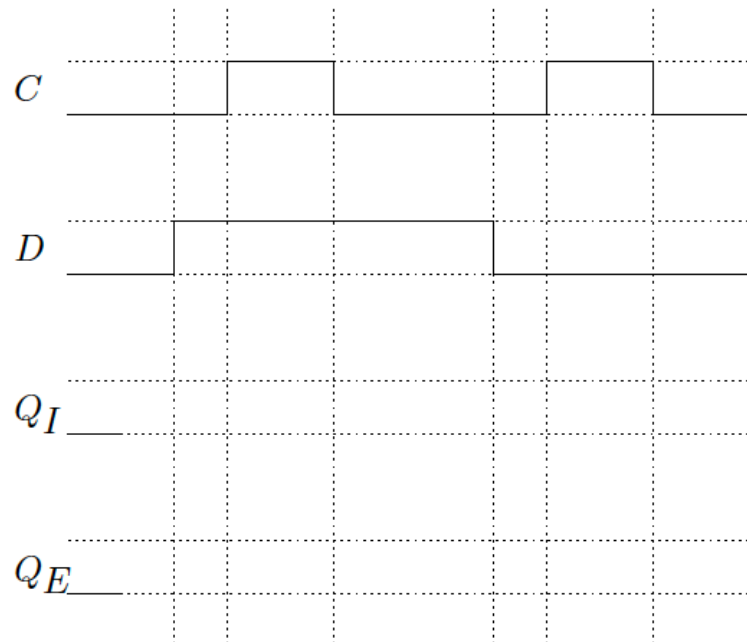


The D Flip-Flop

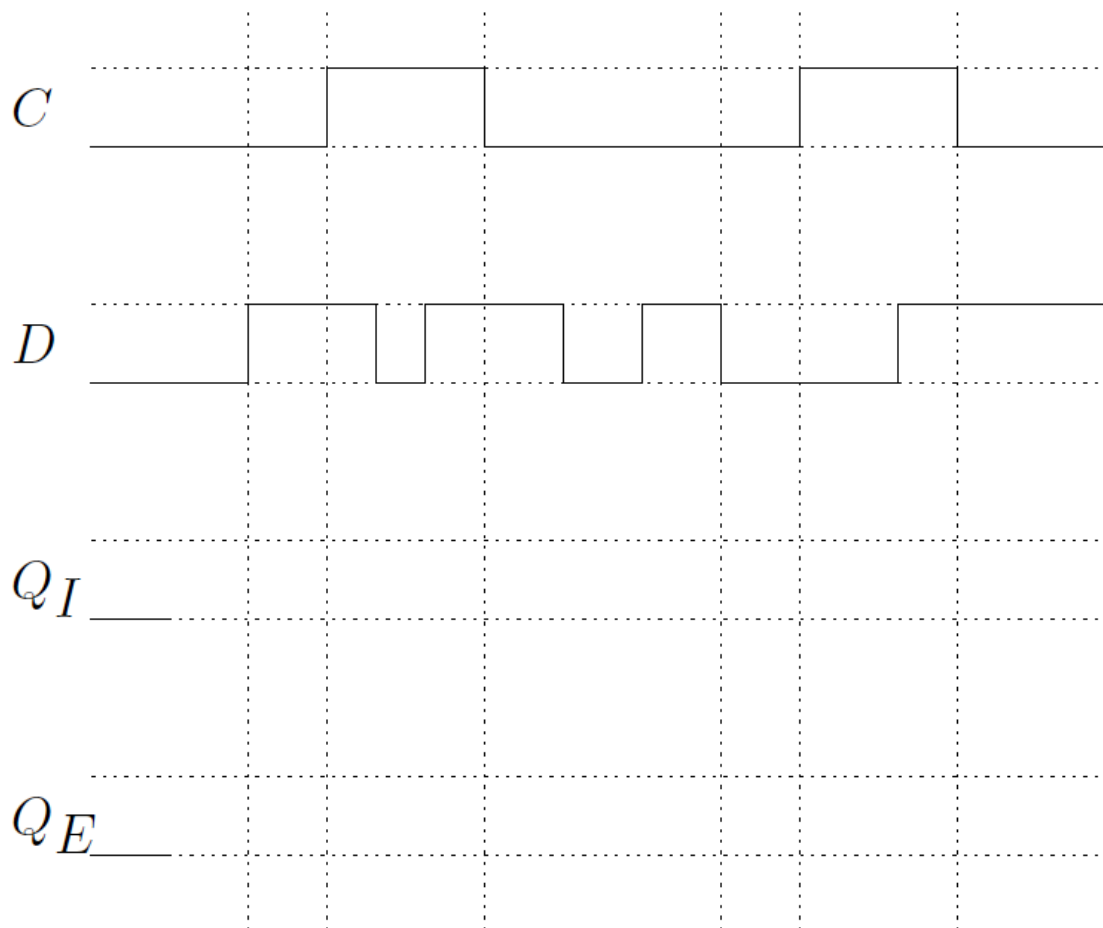
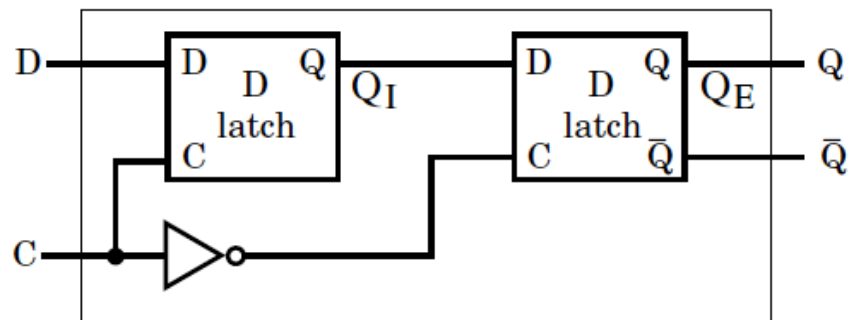
- We want state to be affected only at discrete points in time; a master-slave design achieves this.



- Graphical example:



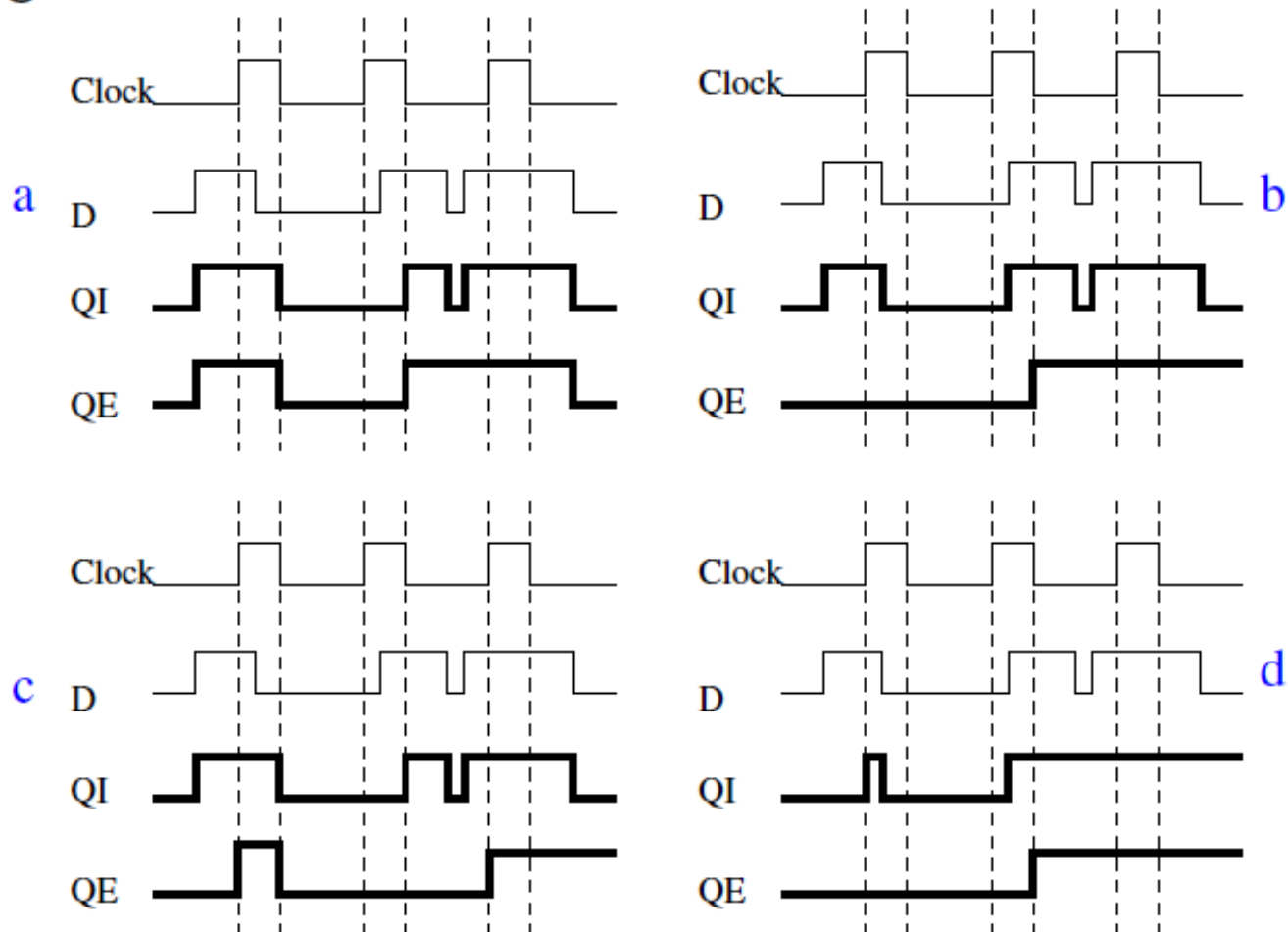
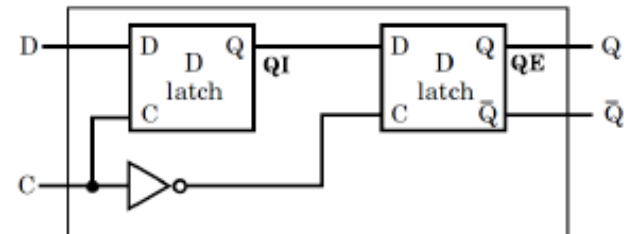
D Flip-Flop



Clicker Question

Flipflops

Which of the following is a trace of QI and QE of a D-flipflop for the given D and Clock traces?



Clicker Question

Question on previous midterm:

How many bits can you store in one flipflop? Circle one.

1 2 4 8 16

A) 1

B) 2

C) 4

D) 8

E) 16