ELECTRICAL
Engineering KMITNB

Power | Control | Communication | Computer

ELECTRICAL
Engineering KMITNB

Digital Circuit and Logic Design

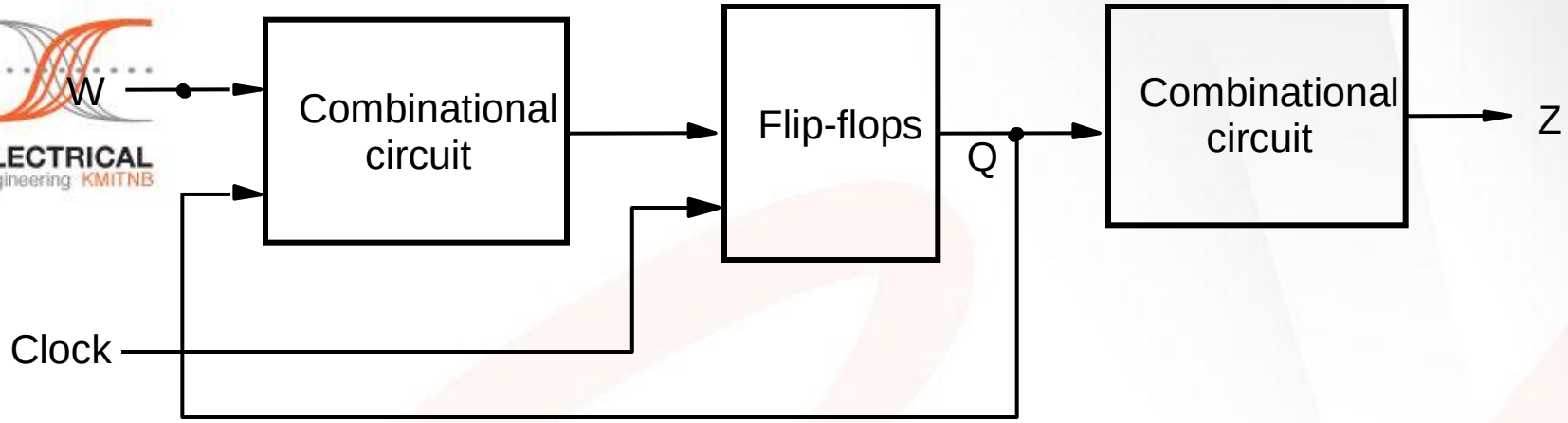
Lecture 8-2 : Synchronous Sequential Circuits

Objectives

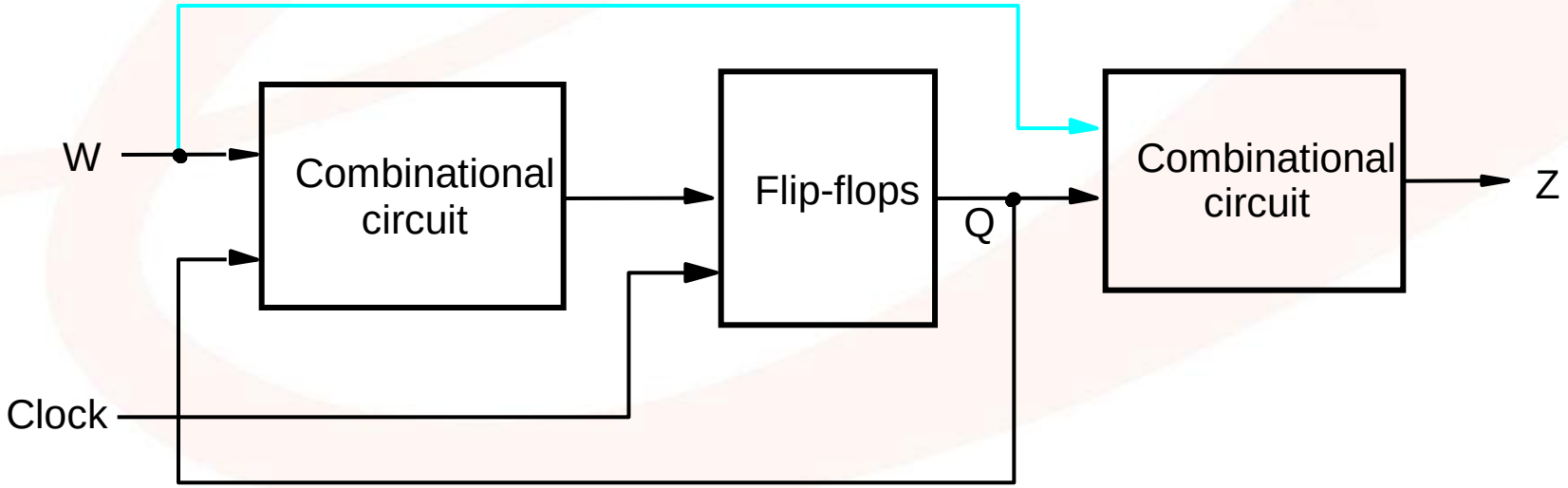
- The concept of states and their implementation with flip-flops
- Sequential behavior of digital circuits
- A procedure for designing synchronous sequential circuits
- The concept of finite state machines

Synchronous Sequential Circuits

- Use a **clock**
- Use combinational logic and one or more flip-flops
- Move from one state to another
- Moore type – outputs depend **only on present state**
- Mealy type – outputs depend on **present state AND inputs**
- Finite State Machines
  - Formal name
  - Functional behavior can be represented by a finite number of states



Moore type Finite State Machine



Mealy type Finite State Machine

ELECTRICAL
Engineering KMITNB

Basic Design Steps

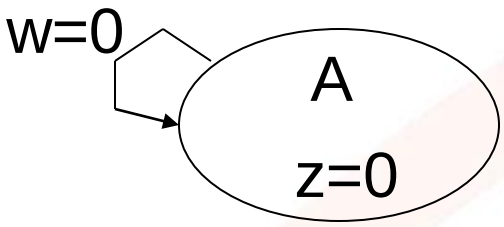
1. Specify Desired Circuit
2. State Diagram
3. State Table
4. Minimize States (Optional) **is not include in this lecture slide.**
5. State Assignment
6. Choice of Flip-flops and Derivation of Next-state and Output Expressions
7. Timing Diagram

1. Specify Desired Circuit

- One input w
- One output z
- All changes occur on positive edge of clock
- Output z = 1 if
  - During two immediately preceding clock cycles the input w was equal to 1
- Output z = 0 otherwise

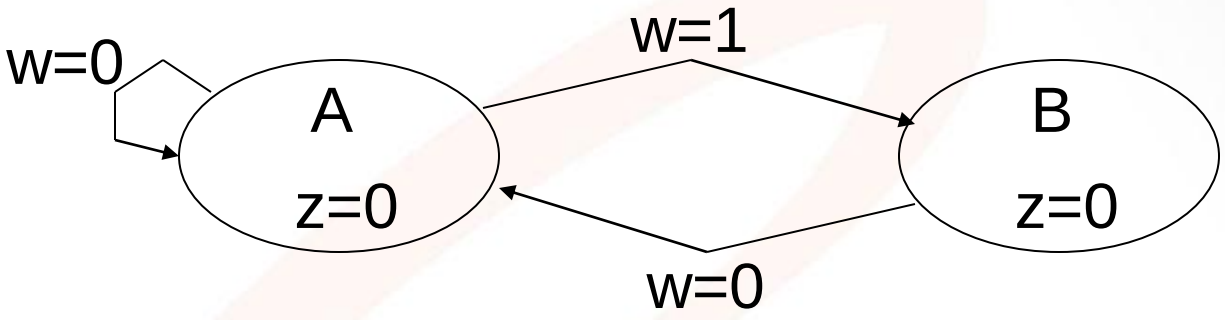
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	0	1	0	0	1	1	0

2. State Diagram



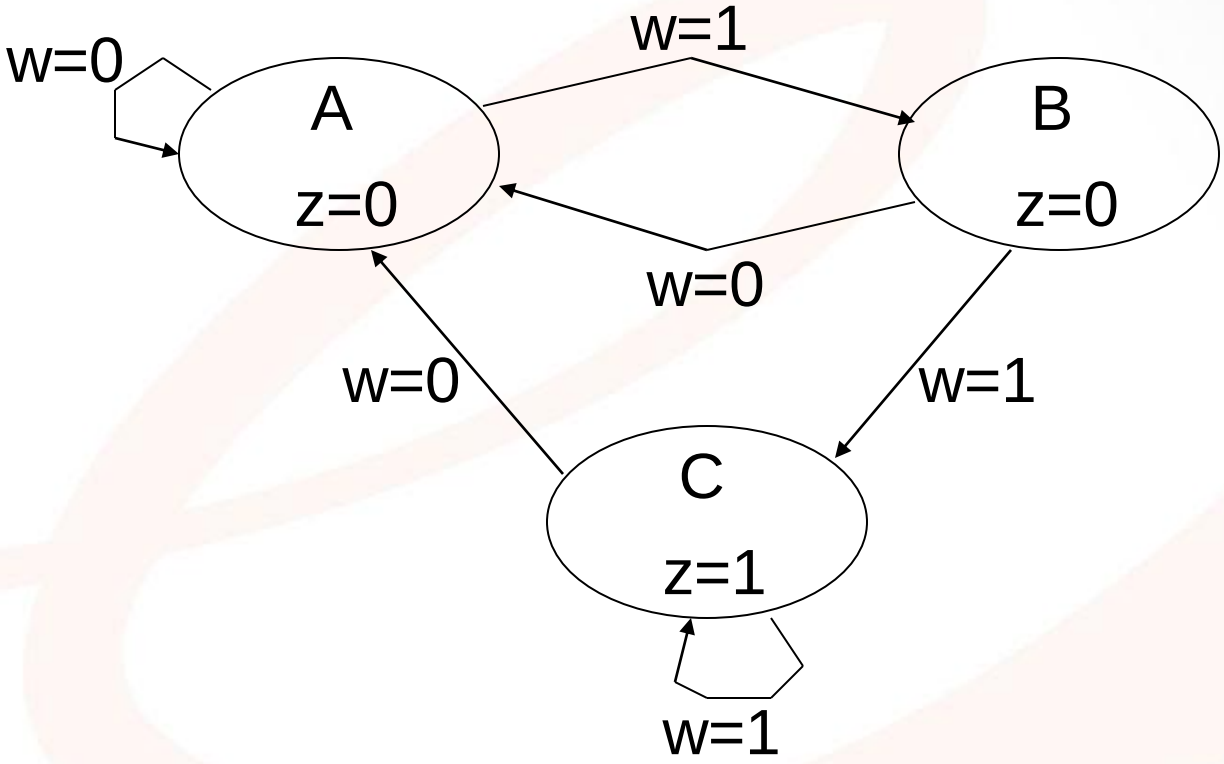
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	0	1	0	0	1	1	0

2. State Diagram (cont.)



Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	0	1	0	0	1	1	0

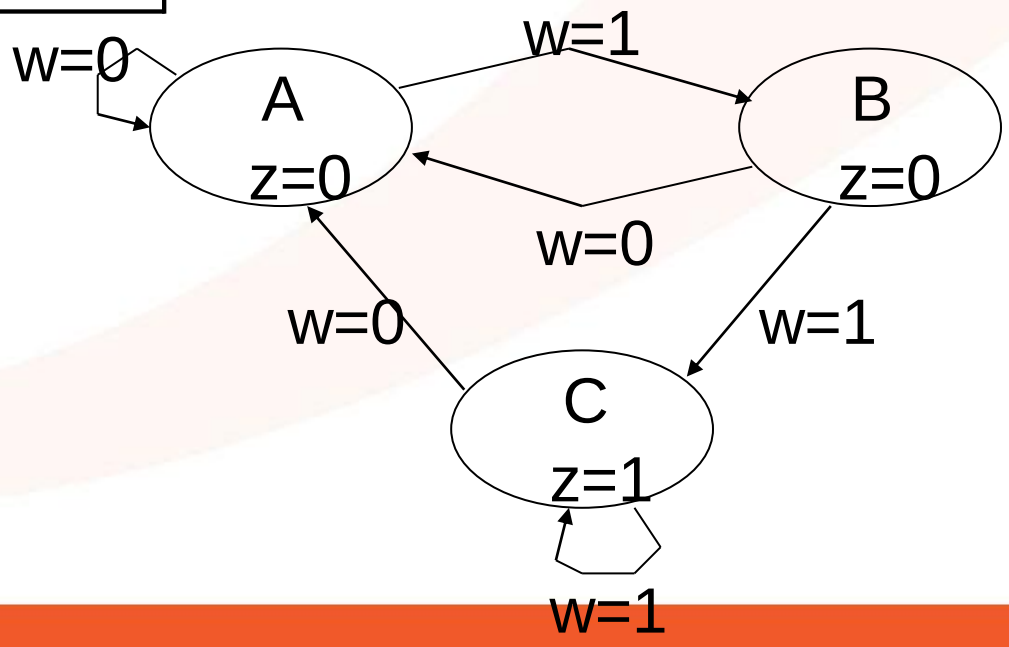
2. State Diagram (cont.)



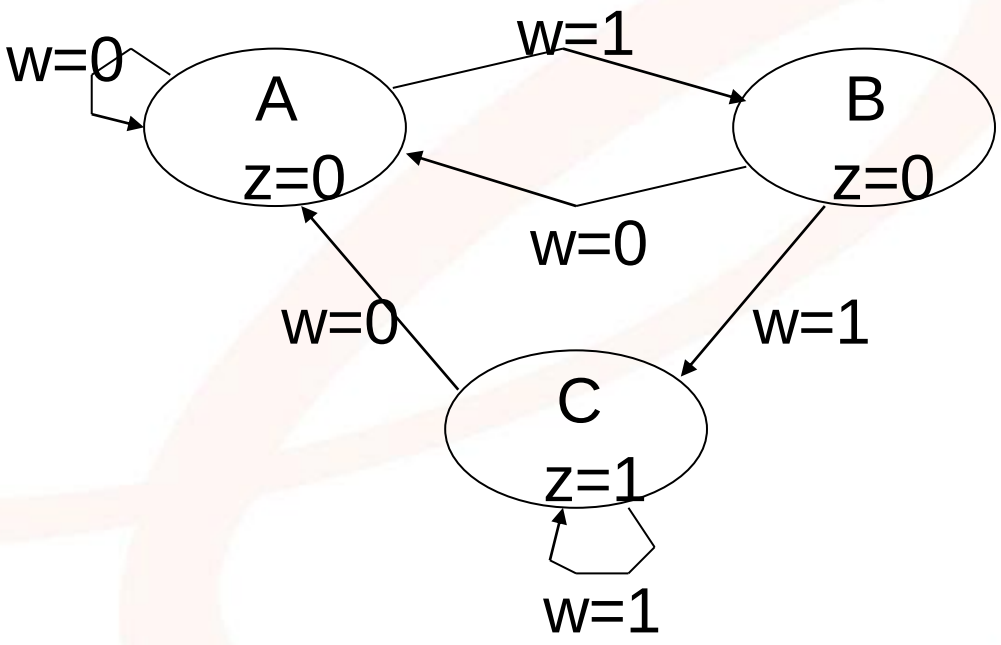
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	0	1	0	0	1	1	0

3. State Table

Present state	Next state		Output z
	$w = 0$	$w = 1$	
A	A	B	0
B	A	C	0
C	A	C	1



5. State Assignment

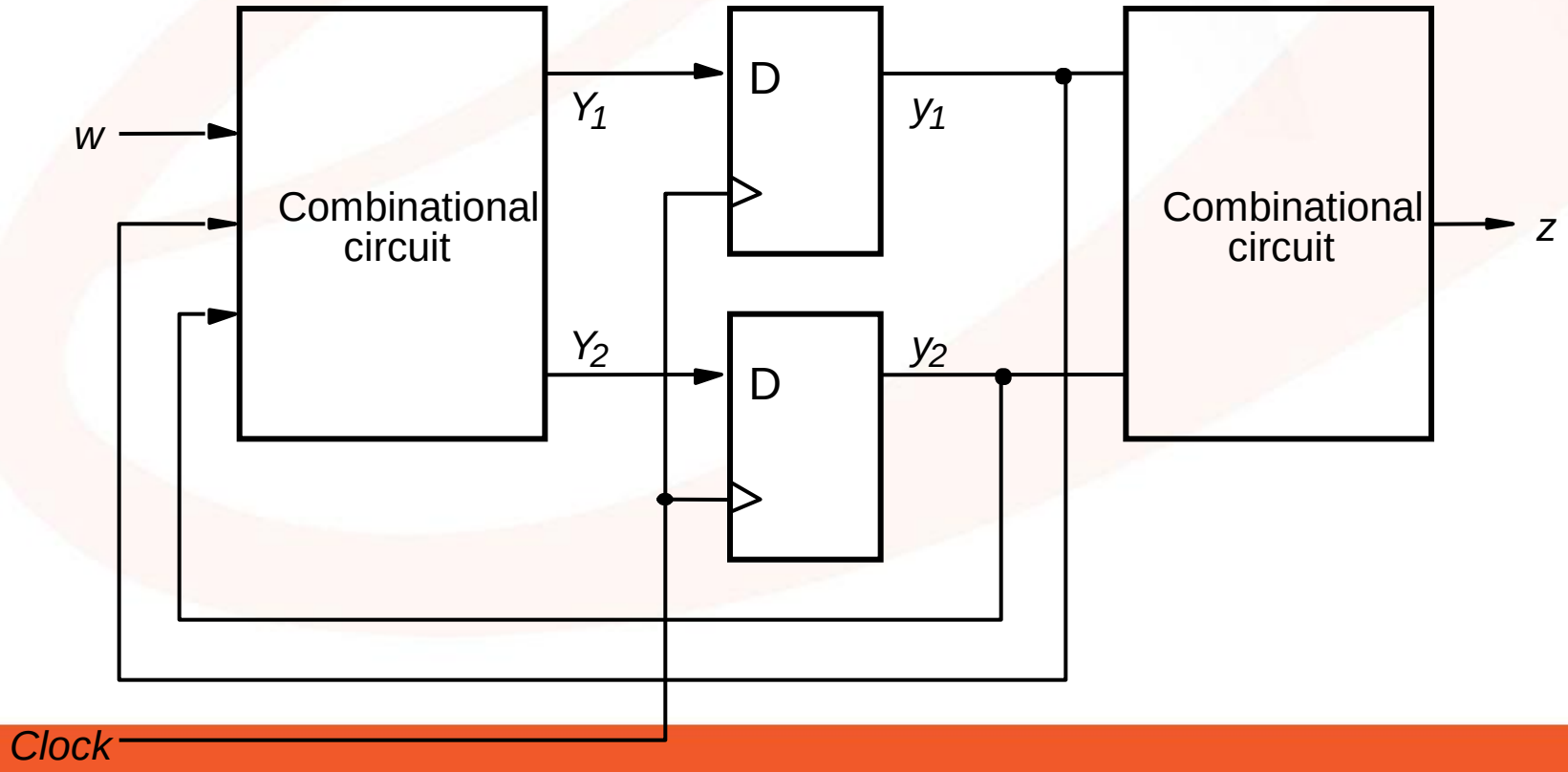


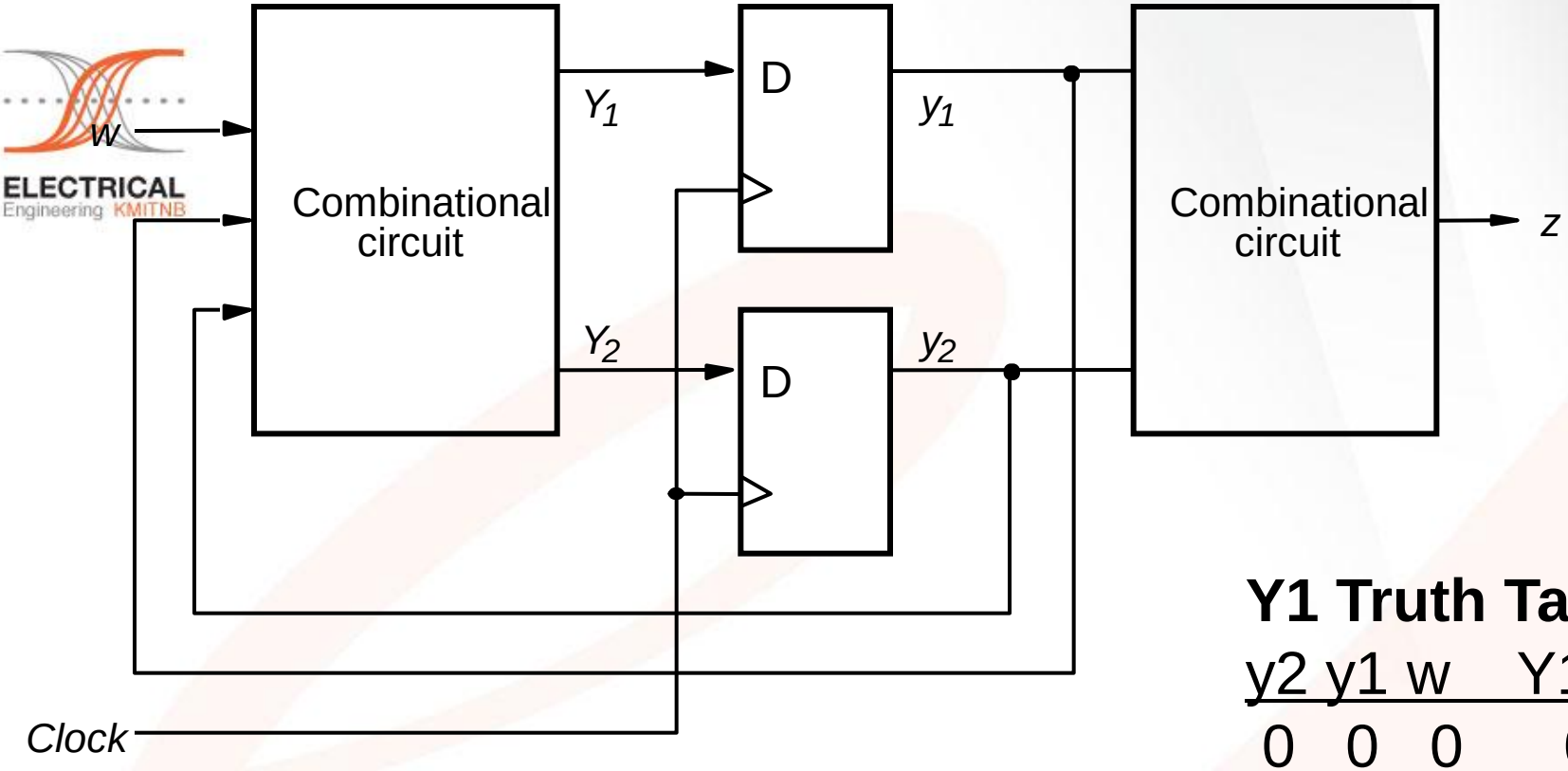
	Present state y_2y_1	Next state $w=0$ Y_2Y_1	Next state $w=1$ Y_2Y_1	Output z
A	00	00	01	0
B	01	00	10	0
C	10	00	10	1
	11	*dd*	*dd*	*d*

- **Moore type** – Output depends only on present state

6. Choice of Flip-flops and Derivation of Next-state and Output Expressions

- D Flip-flops
  - Most straight forward



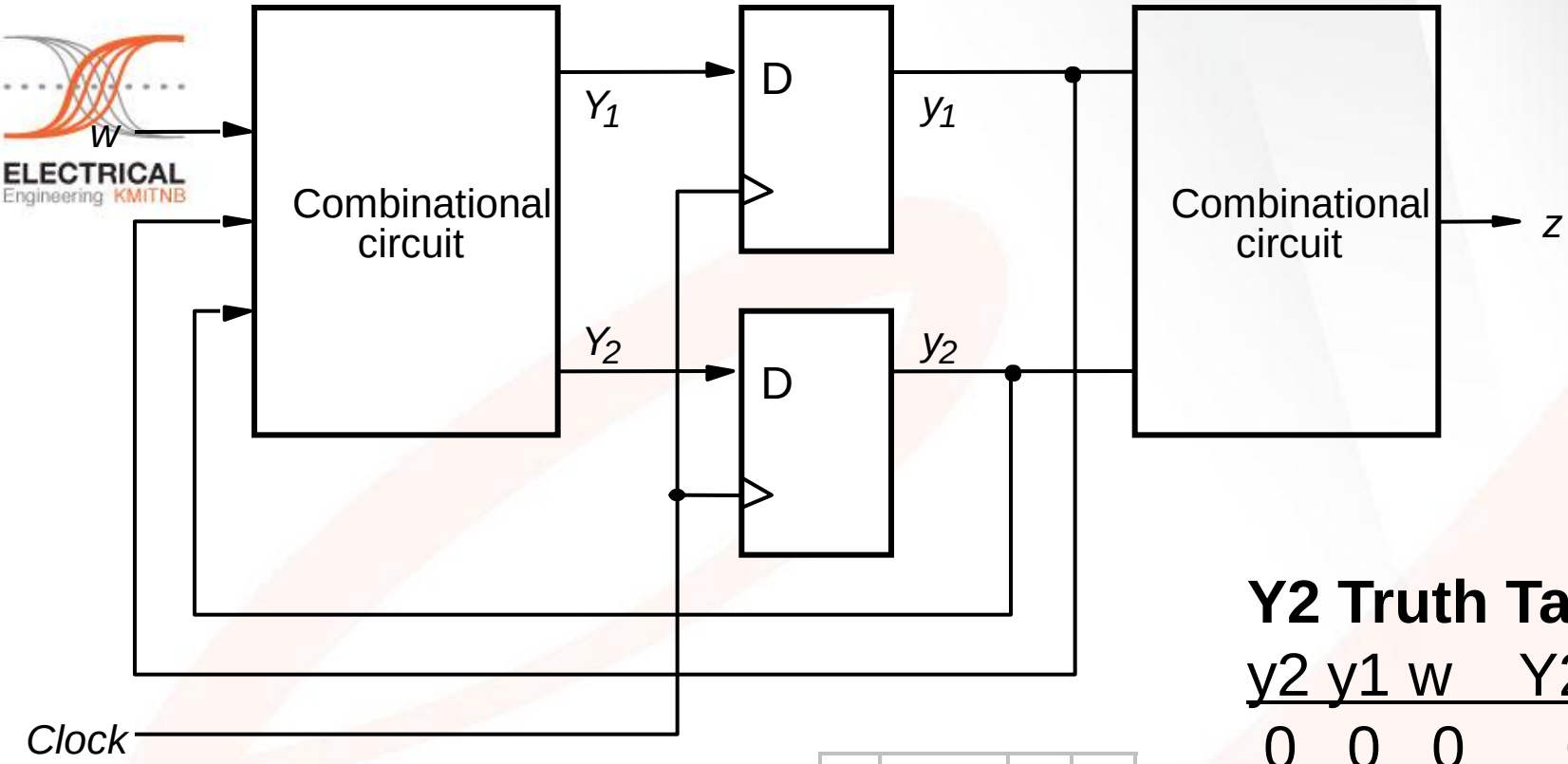


	Present state y_2y_1	Next state $w=0$ Y_2Y_1	Next state $w=1$ Y_2Y_1	Output z
A	00	00	01	0
B	01	00	10	0
C	10	00	10	1
	11	*dd*	*dd*	*d*

$$Y1 = w \cdot \overline{y1} \cdot \overline{y2}$$

Y1 Truth Table

y2	y1	w	Y1
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	d
1	1	1	d



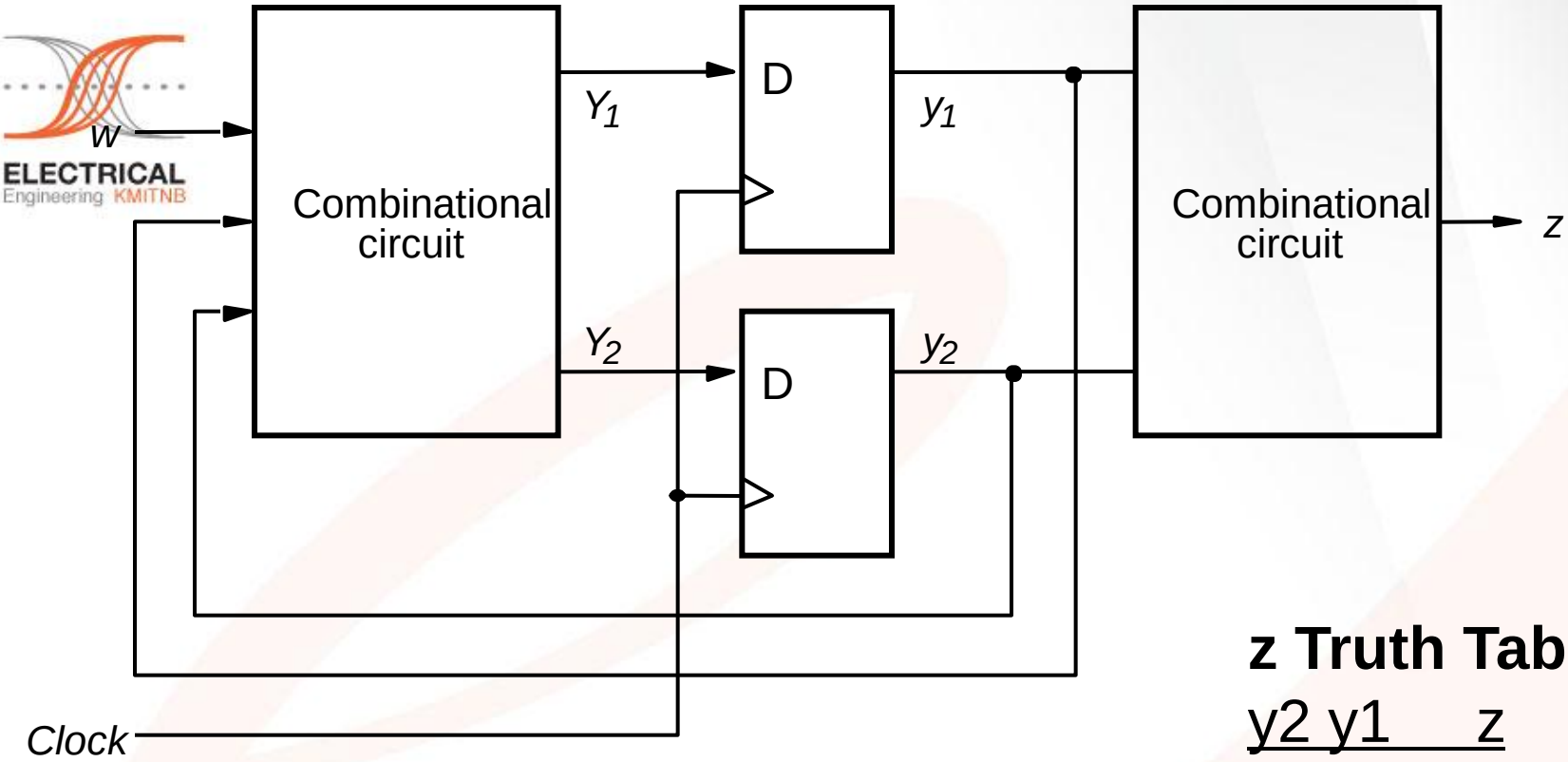
	Present state y_2y_1	Next state $w=0$ Y_2Y_1	Next state $w=1$ Y_2Y_1	Output z
A	00	00	01	0
B	01	00	10	0
C	10	00	10	1
	11	*dd*	*dd*	*d*

	y2 y1			
w	00	01	11	10
0	0	0	d	0
1	0	1	d	1

$$Y2 = w \cdot y1 + w \cdot y2$$
$$= w(y1 + y2)$$

Y2 Truth Table

y2	y1	w	Y2
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	d
1	1	1	d



	Present state y_2y_1	Next state $w=0$ Y_2Y_1	Next state $w=1$ Y_2Y_1	Output z
A	00	00	01	0
B	01	00	10	0
C	10	00	10	1
	11	*dd*	*dd*	*d*

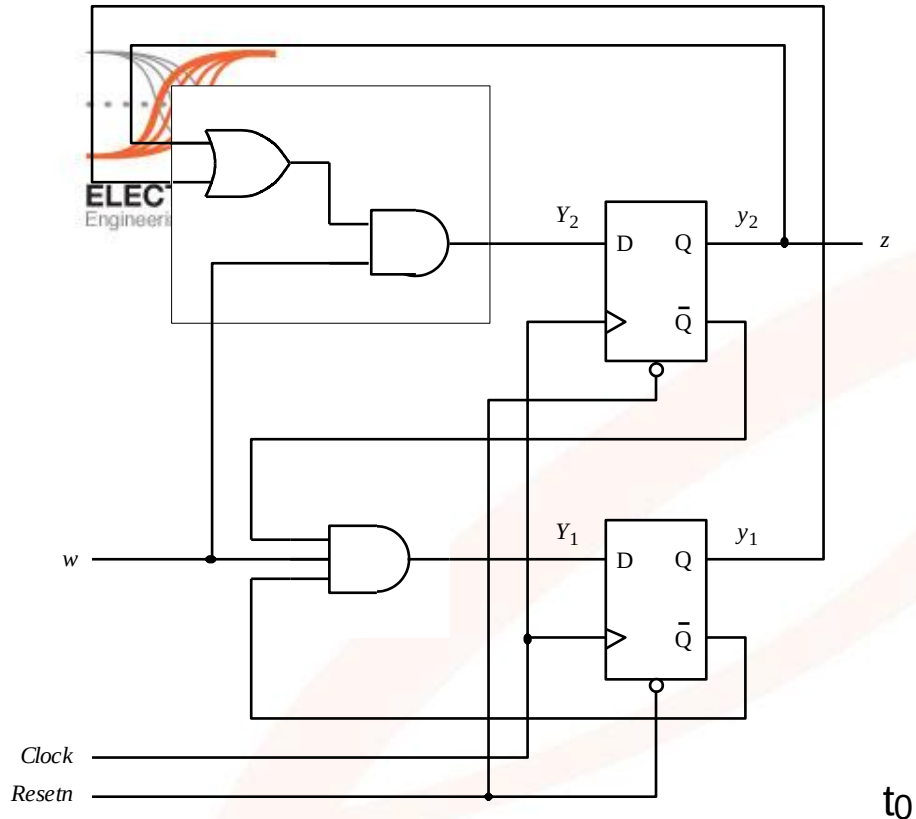
	y1	
y2	0	1
0	0	0
1	1	d

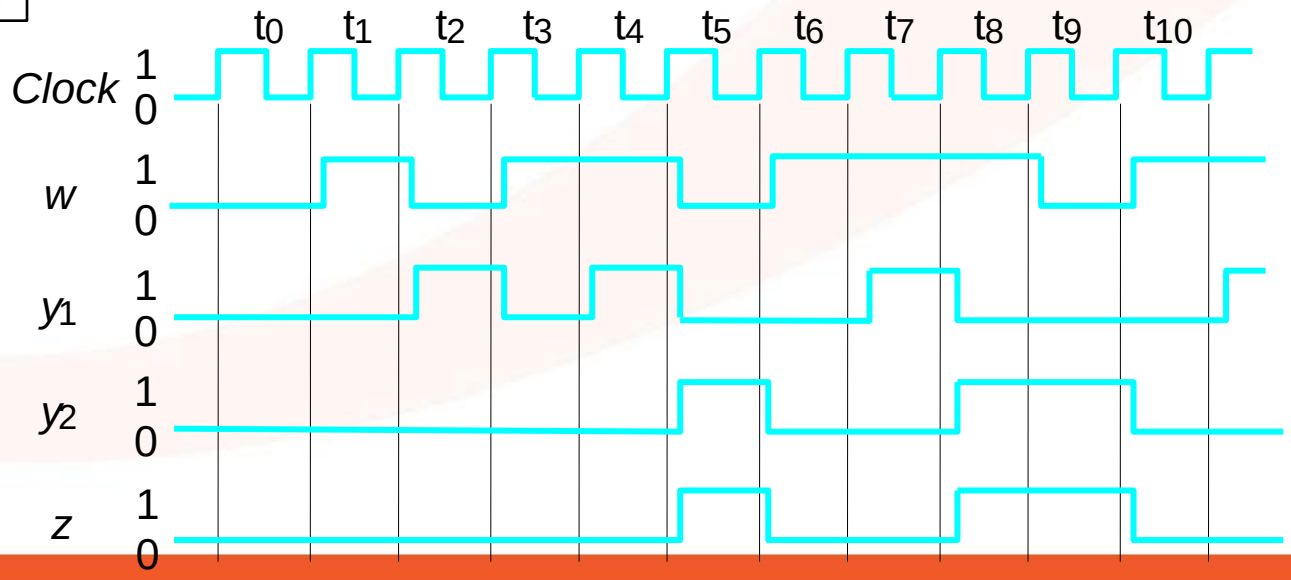
z = y2

z Truth Table

y2	y1	z
0	0	0
0	1	0
1	0	1
1	1	d

7. Timing Diagram

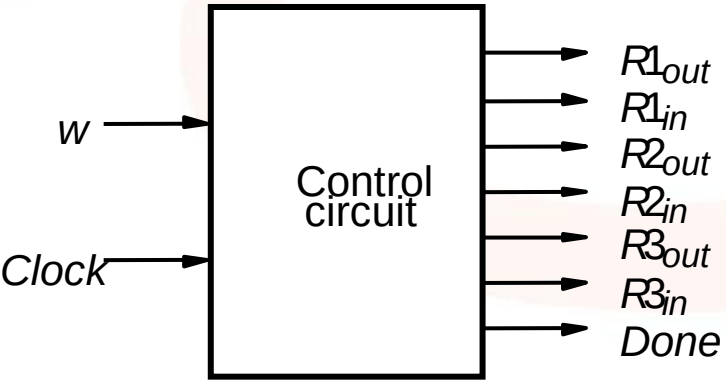


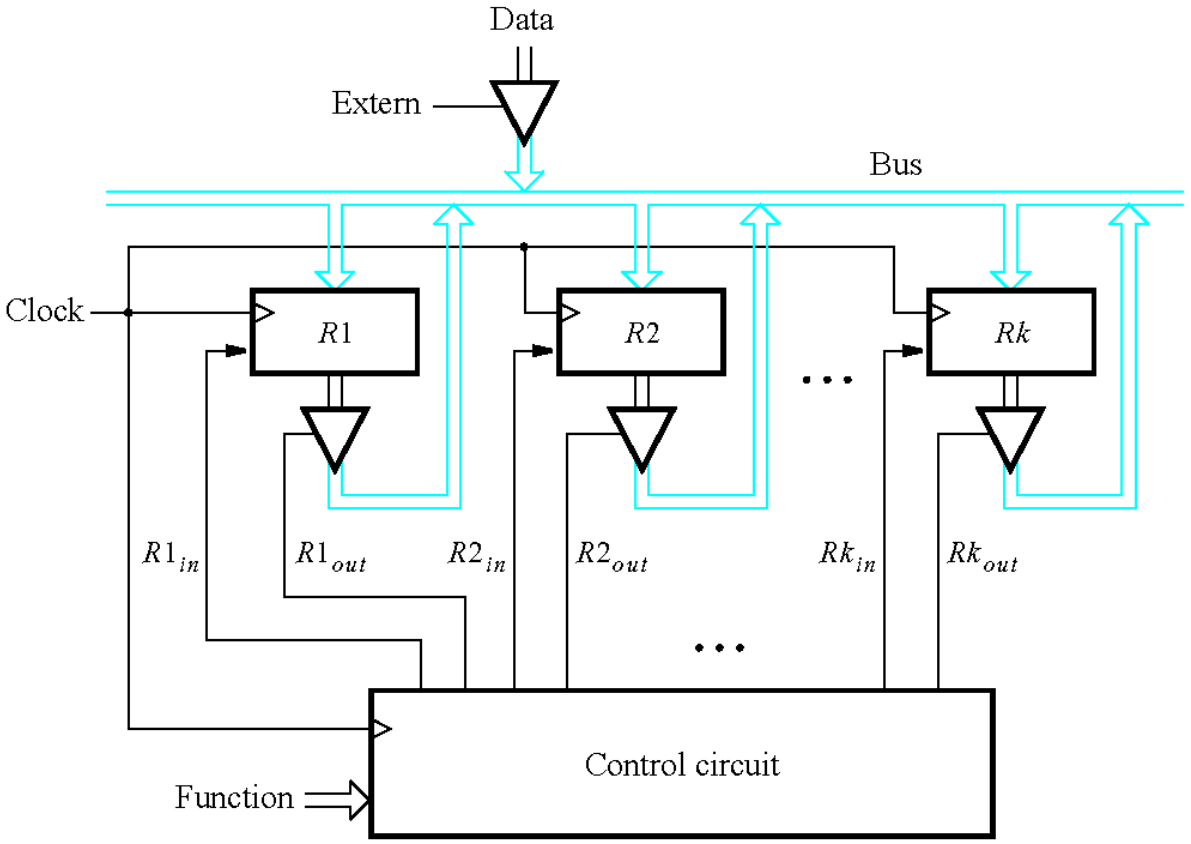


Example

Specify Desired Circuit

- Swap R1 & R2 Using R3 as a temporary register
  - R2=> R3
  - R1 => R2
  - R3 => R1
- Swapping started by a 1 clock cycle pulse on w
- Indicate task complete by setting Done = 1





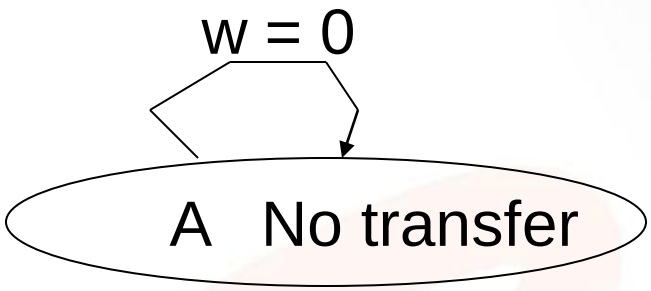
ELECTRICAL
Engineering KMITNB

State Diagram

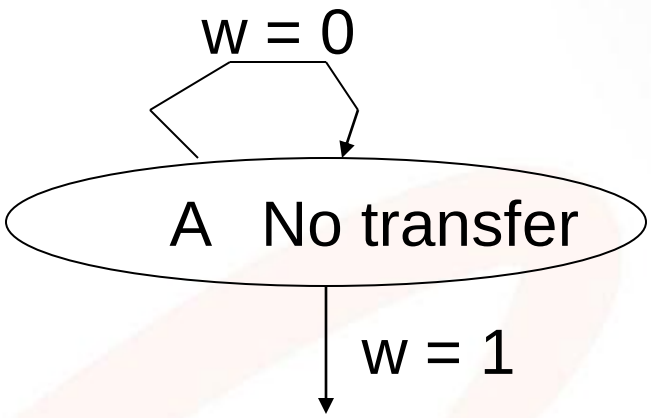
- What are the inputs?
- What are the outputs?
- How many states are required?
- What is the value of the outputs in those states?

ELECTRICAL
Engineering KMITNB

A No transfer

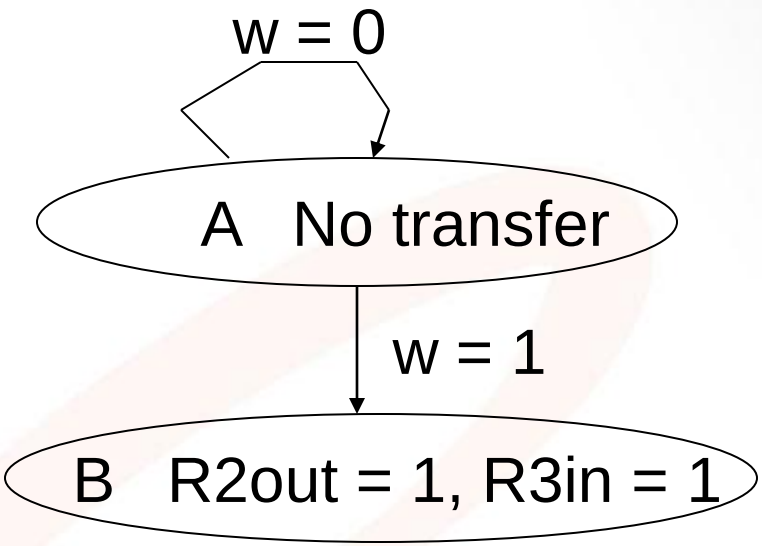

w = 0
A No transfer

R2=> R3
R1 => R2
R3 => R1



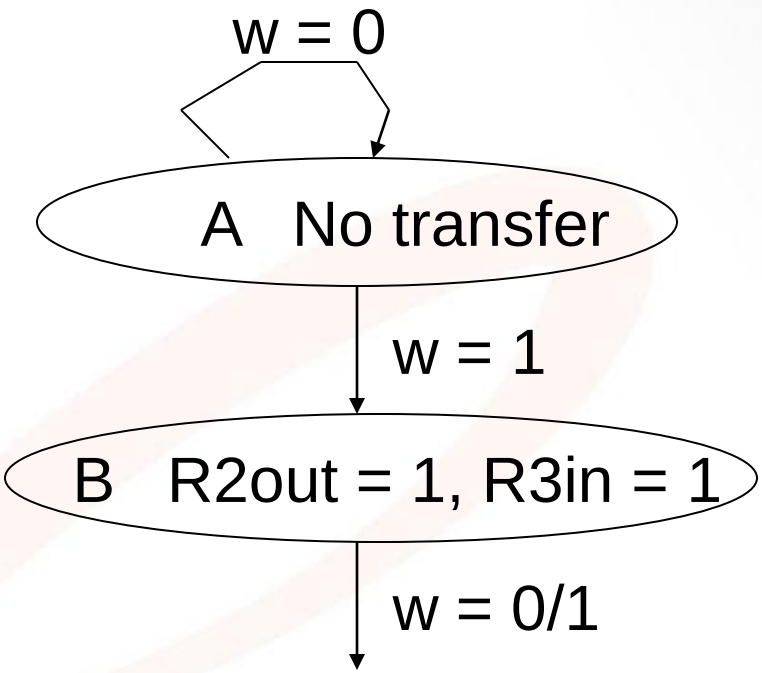
ELECTRICAL
Engineering KMITNB

R2=> R3
R1 => R2
R3 => R1



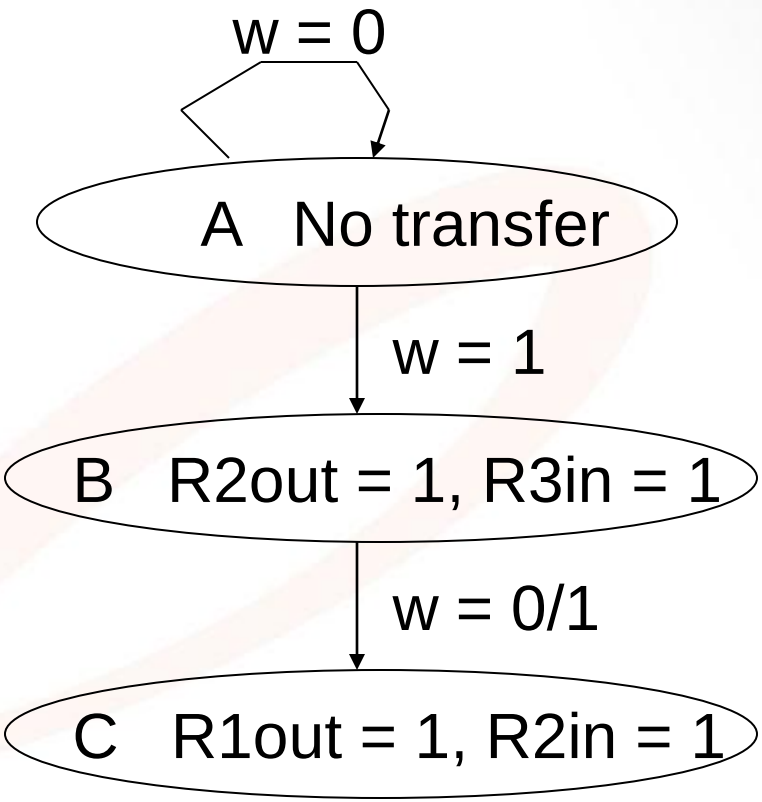
ELECTRICAL
Engineering KMITNB

R2=> R3
R1 => R2
R3 => R1



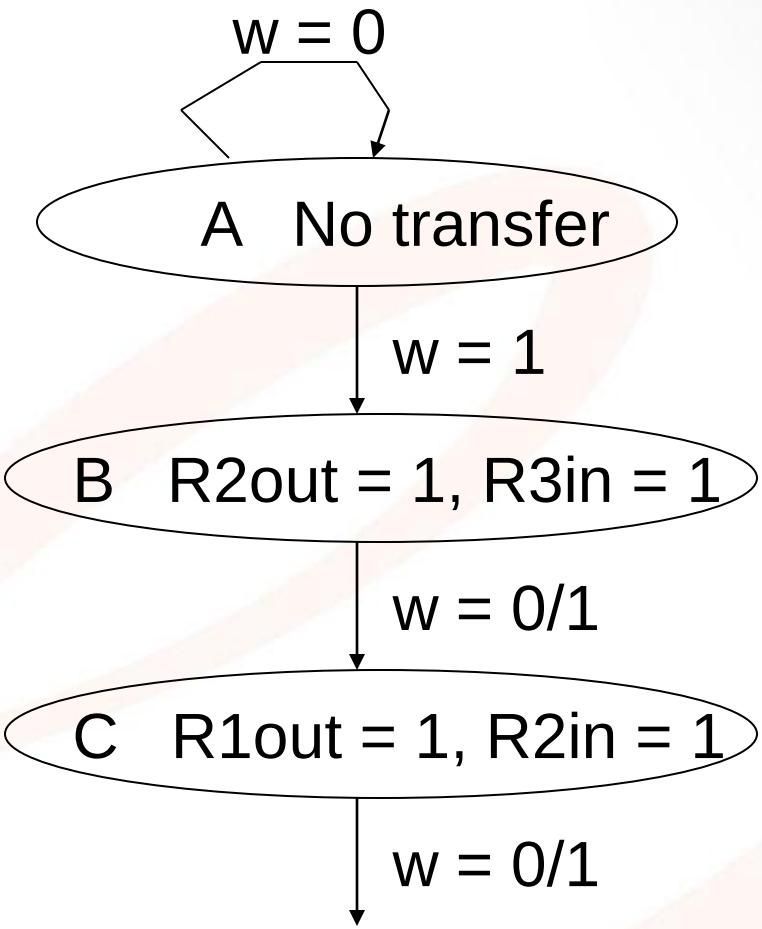
ELECTRICAL
Engineering KMITNB

R2=> R3
R1 => R2
R3 => R1



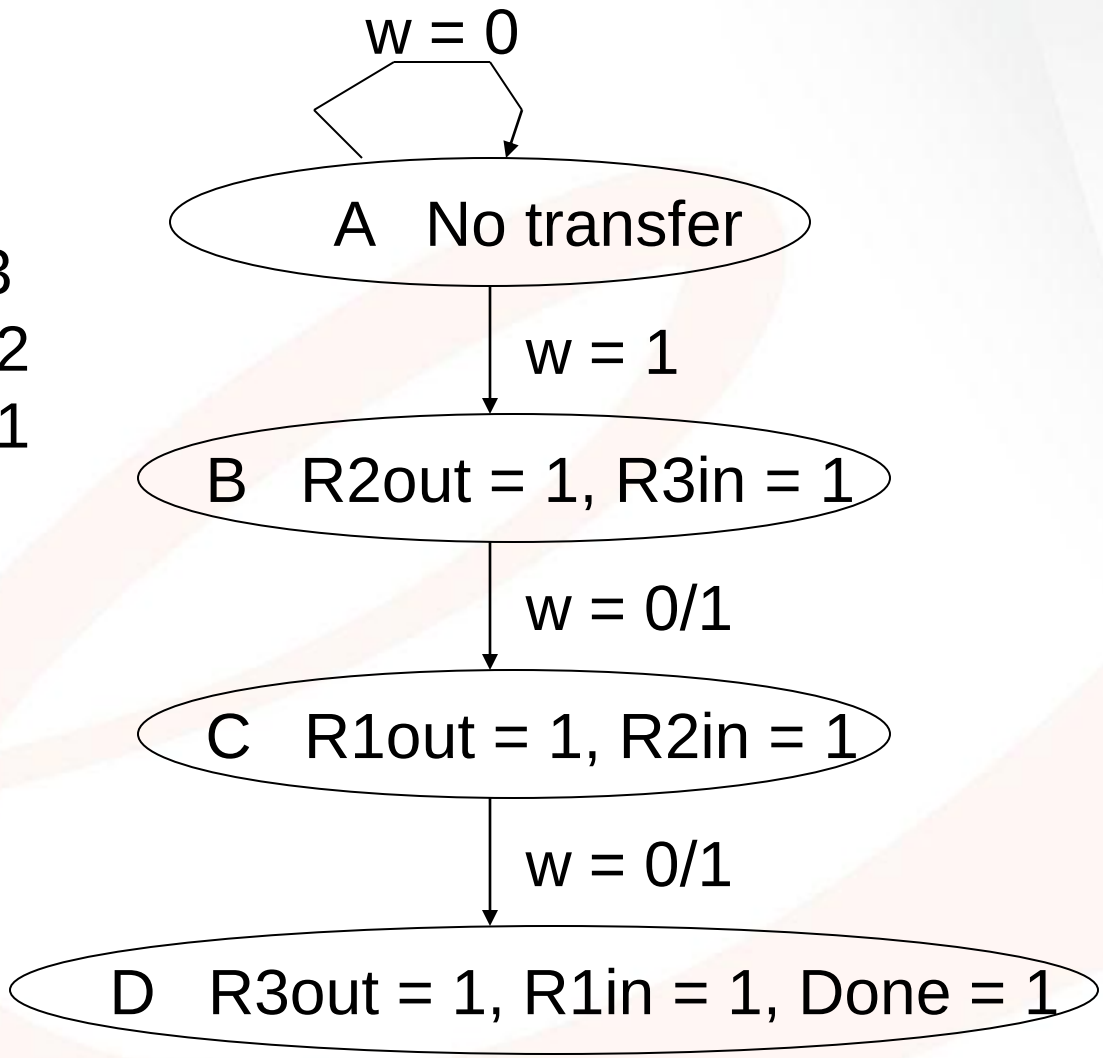
ELECTRICAL
Engineering KMITNB

R2=> R3
R1 => R2
R3 => R1



ELECTRICAL
Engineering KMITNB

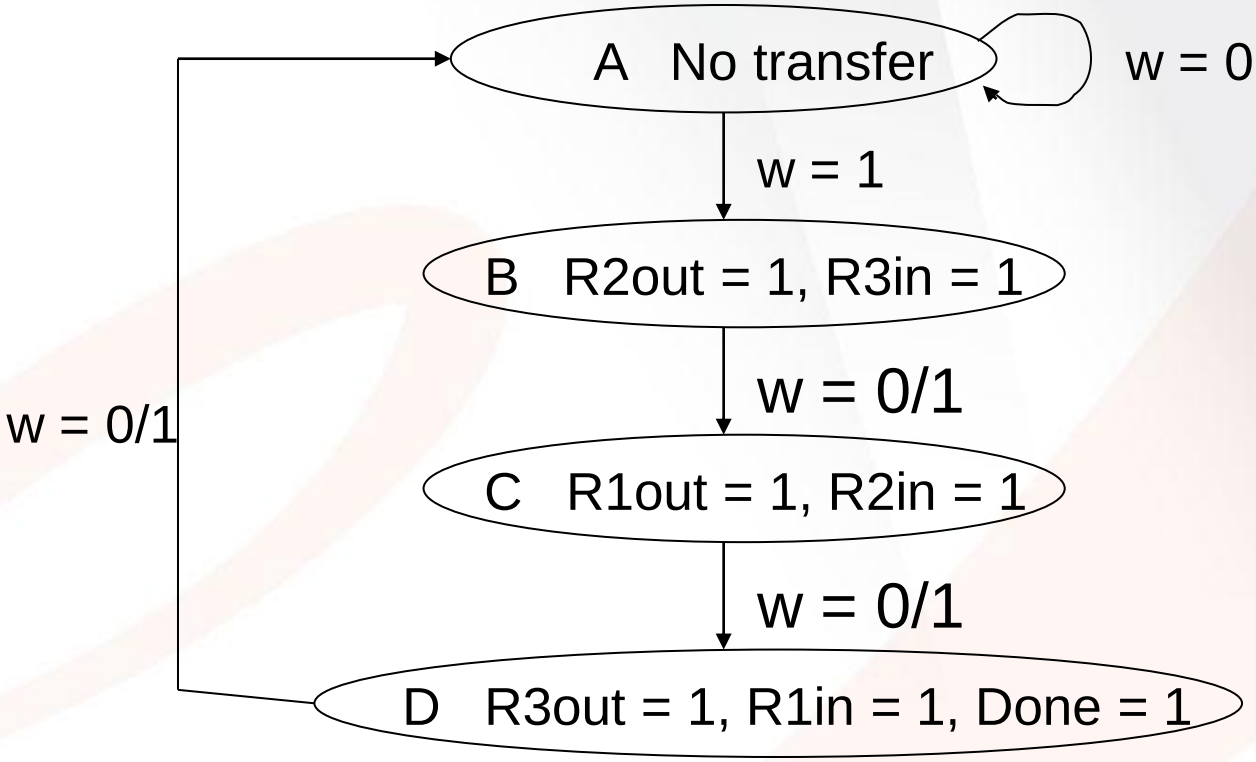
R2=> R3
R1 => R2
R3 => R1



R2=> R3

R1 => R2

R3 => R1



Present state	Next state		Outputs						
	$w = 0$	$w = 1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	$Done$
A	A	B	0	0	0	0	0	0	0
B	C	C	0	0	1	0	0	1	0
C	D	D	1	0	0	1	0	0	0
D	A	A	0	1	0	0	1	0	1

State Table

Present state	Next state		Outputs						
	$w = 0$	$w = 1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	$Done$
A	A	B	0	0	0	0	0	0	0
B	C	C	0	0	1	0	0	1	0
C	D	D	1	0	0	1	0	0	0
D	A	A	0	1	0	0	1	0	1

	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	y_2y_1	Y_2Y_1	Y_2Y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	$Done$
A	00	00	0 1	0	0	0	0	0	0	0
B	01	10	1 0	0	0	1	0	0	1	0
C	10	11	1 1	1	0	0	1	0	0	0
D	11	00	0 0	0	1	0	0	1	0	1

	Present state	Next state		Outputs						
		$w=0$	$w=1$							
	y_2y_1	Y_2Y_1	Y_2Y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	$Done$
A	00	00	01	0	0	0	0	0	0	0
B	01	10	10	0	0	1	0	0	1	0
C	10	11	11	1	0	0	1	0	0	0
D	11	00	00	0	1	0	0	1	0	1

Y1 Truth Table

y2 y1 w Y1

	y2 y1			
w	00	01	11	10
0				
1				

Y1=

Y2 Truth Table

y2 y1 w Y2

	y2 y1			
w	00	01	11	10
0				
1				

Y2=

	Present state	Next state		Outputs						
		$w=0$	$w=1$							
	y_2y_1	Y_2Y_1	Y_2Y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	$Done$
A	00	00	0 1	0	0	0	0	0	0	0
B	01	10	1 0	0	0	1	0	0	1	0
C	10	11	1 1	1	0	0	1	0	0	0
D	11	00	0 0	0	1	0	0	1	0	1

Y1 Truth Table

y2	y1	w	Y1
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

	y2 y1			
w	00	01	11	10
0	0	0	0	1
1	1	0	0	1

$Y1 = y2 \cdot \overline{y1} + w \cdot \overline{y1}$

Y2 Truth Table

y2	y1	w	Y2
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

	y2 y1			
w	00	01	11	10
0	0	1	0	1
1	0	1	0	1

$Y2 = \overline{y2} \cdot y1 + y2 \cdot \overline{y1}$

	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	y_2y_1	Y_2Y_1	Y_2Y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	$Done$
A	00	00	0 1	0	0	0	0	0	0	0
B	01	10	1 0	0	0	1	0	0	1	0
C	10	11	1 1	1	0	0	1	0	0	0
D	11	00	0 0	0	1	0	0	1	0	1

R1out =

R2in =

R1in =

R3out =

Done =

R2out =

R3in =

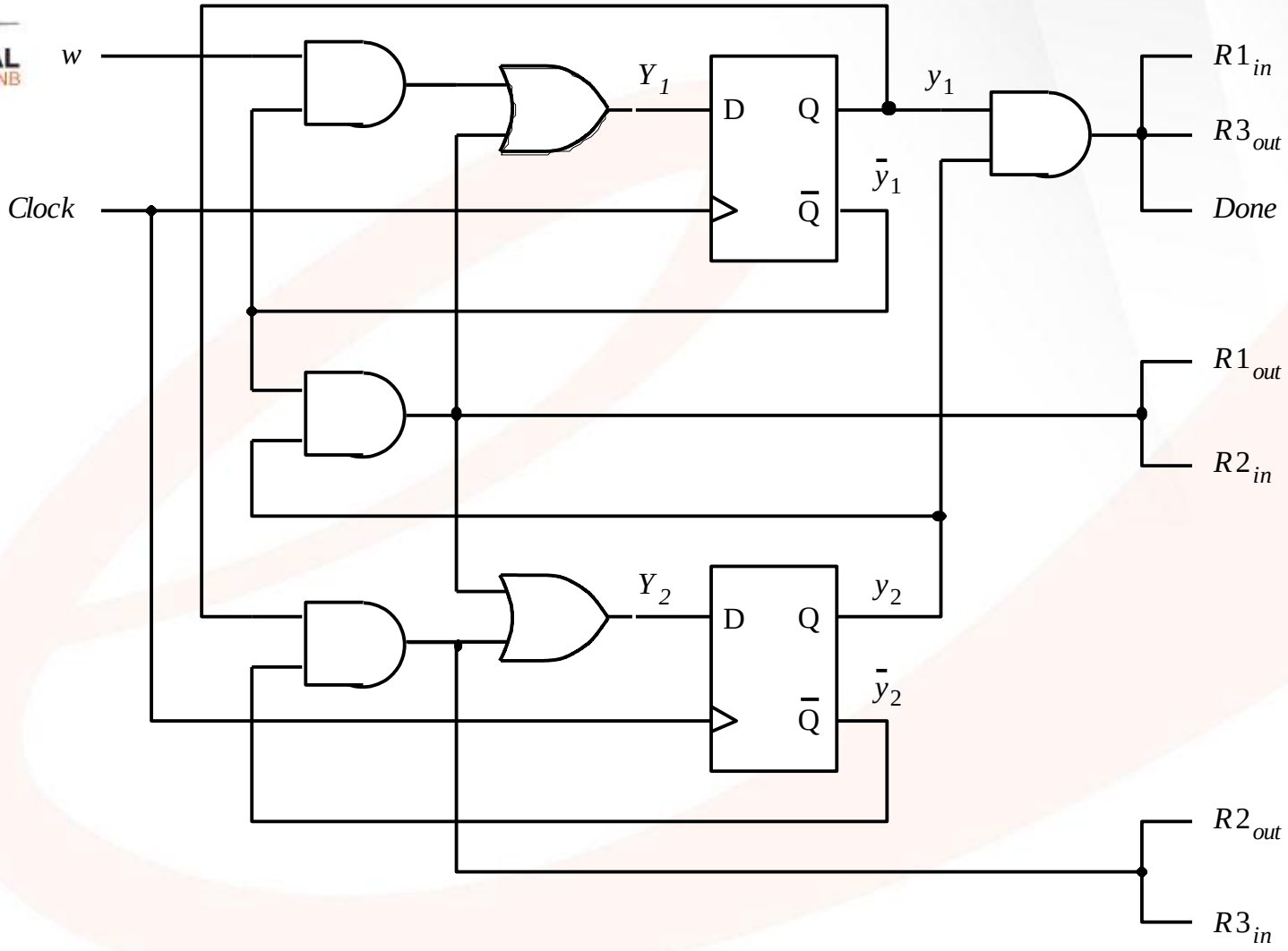
ELECTRICAL
Engineering KMITNB

	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	y_2y_1	Y_2Y_1	Y_2Y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	$Done$
A	00	00	0 1	0	0	0	0	0	0	0
B	01	10	1 0	0	0	1	0	0	1	0
C	10	11	1 1	1	0	0	1	0	0	0
D	11	00	0 0	0	1	0	0	1	0	1

R1out = R2in = $y2 \cdot \overline{y1}$

R1in = R3out = Done = y2·y1

R2out = R3in = $\overline{y2} \cdot y1$


w
Clock
Y_1
D
Q
$\bar{Q}$
y_1
$\bar{y}_1$
Y_2
D
Q
$\bar{Q}$
y_2
$\bar{y}_2$
$R1_{in}$
$R3_{out}$
Done
$R1_{out}$
$R2_{in}$
$R2_{out}$
$R3_{in}$

ELECTRICAL
Engineering KMITNB

FSM Model Variants

- Moore-type
  - Outputs depend only on Present State
- Mealy-type
  - Outputs depend on Present State AND Inputs

ELECTRICAL
Engineering KMITNB

Mealy State Model

Specify Desired Circuit

- One input w
- One output z
- All changes occur on positive edge of clock
- Output z = 1 if
  - During immediately preceding clock cycle the input w was equal to 1 AND is currently equal to 1
- Output z = 0 otherwise

Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0

ELECTRICAL
Engineering KMITNB

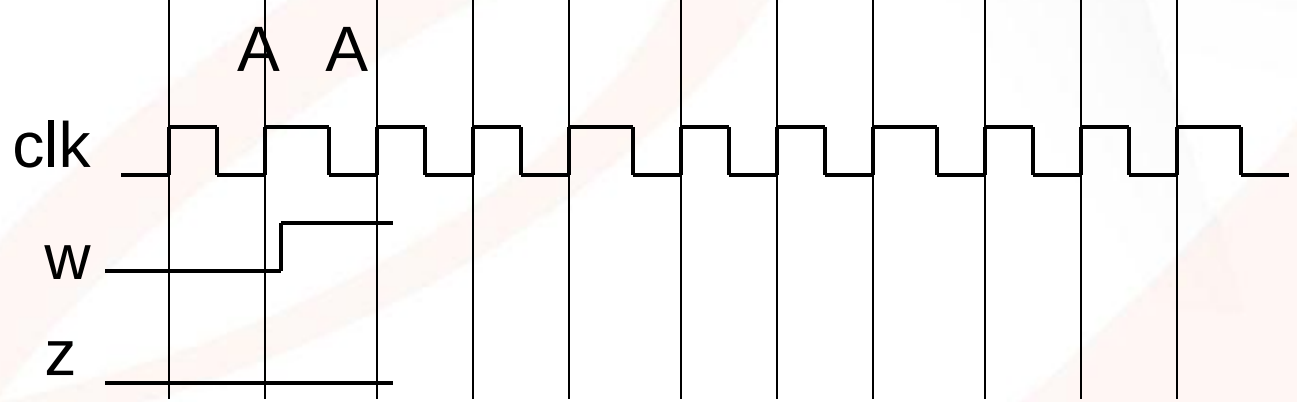
New Machine (Mealy)

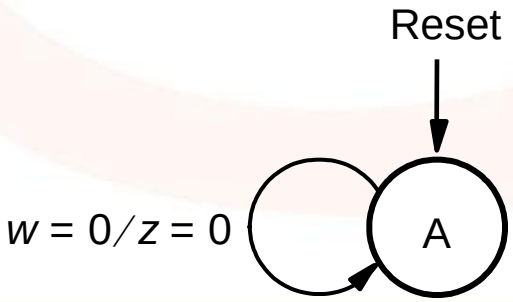
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0

Old Machine (Moore)

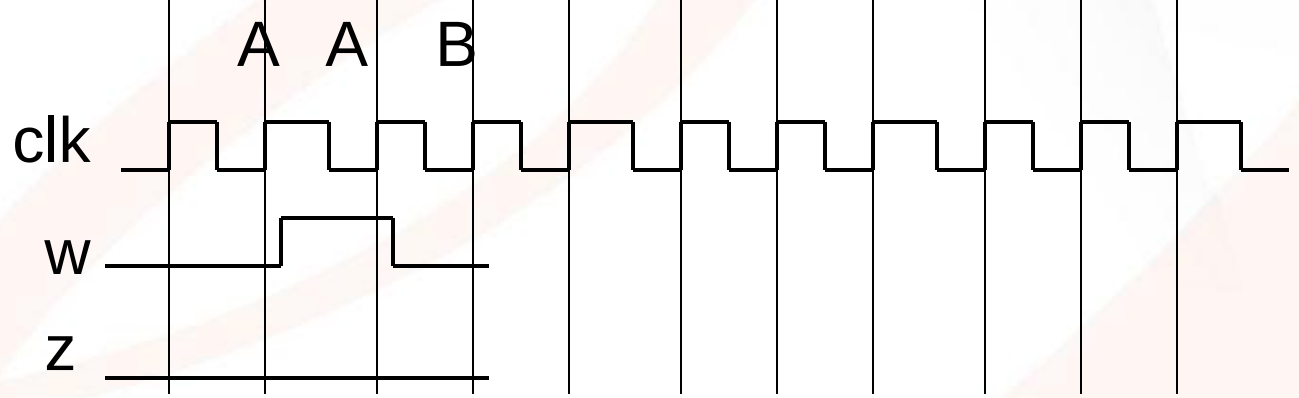
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	0	1	0	0	1	1	0

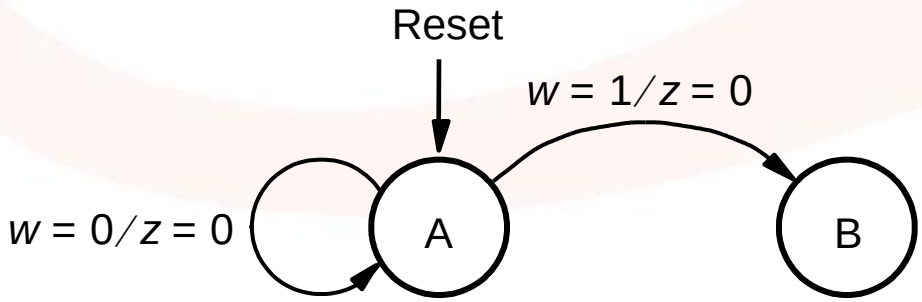
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0



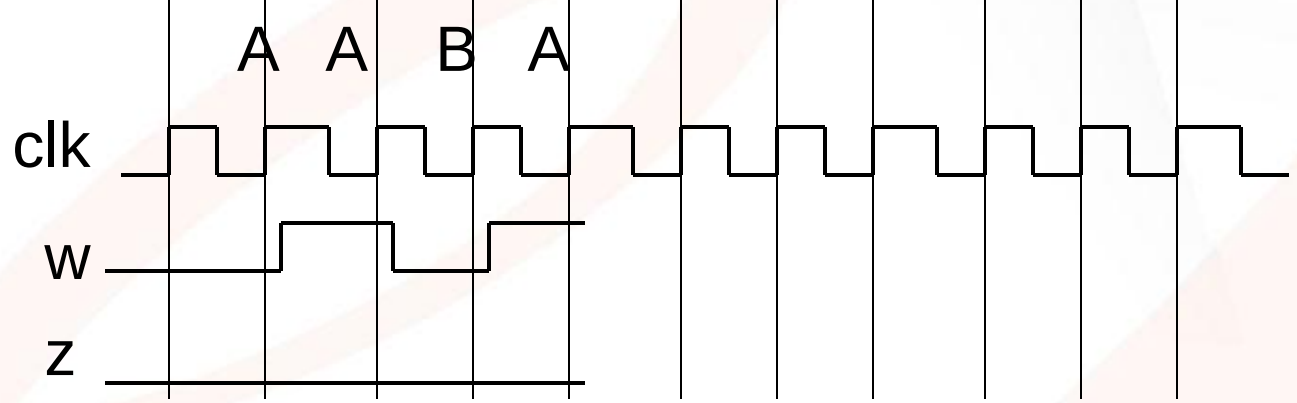


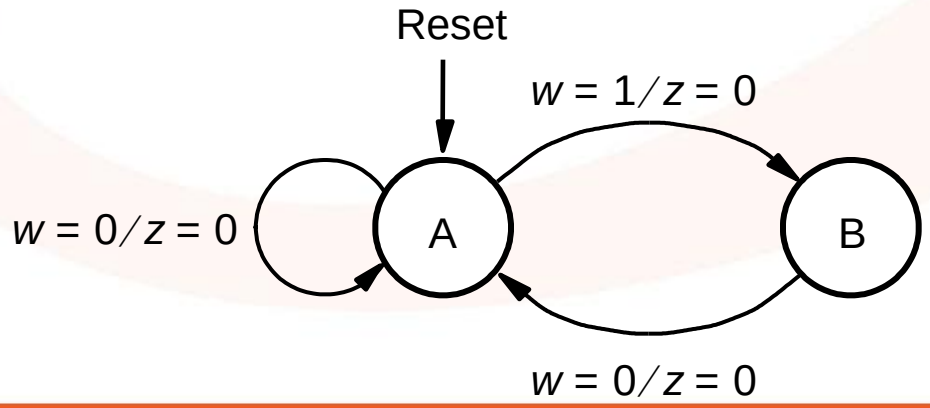
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0



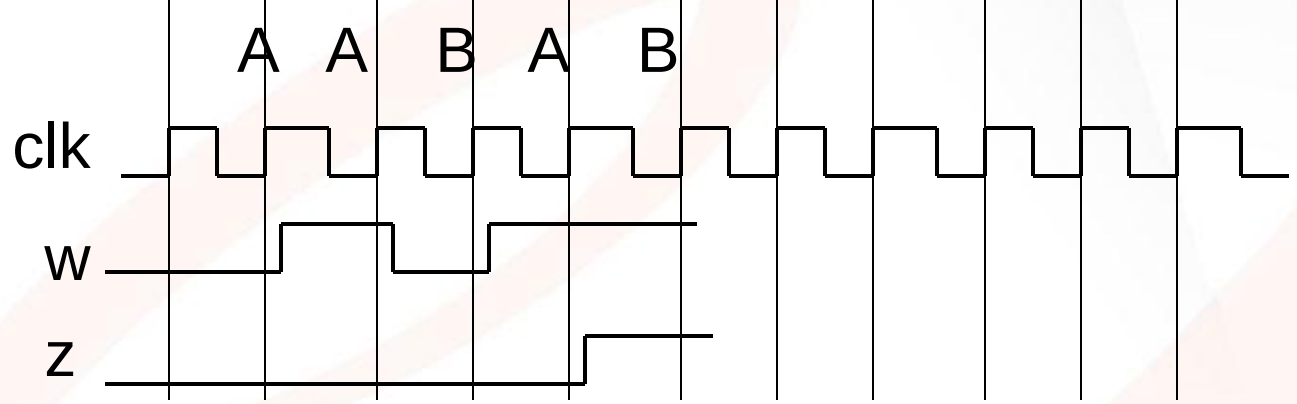


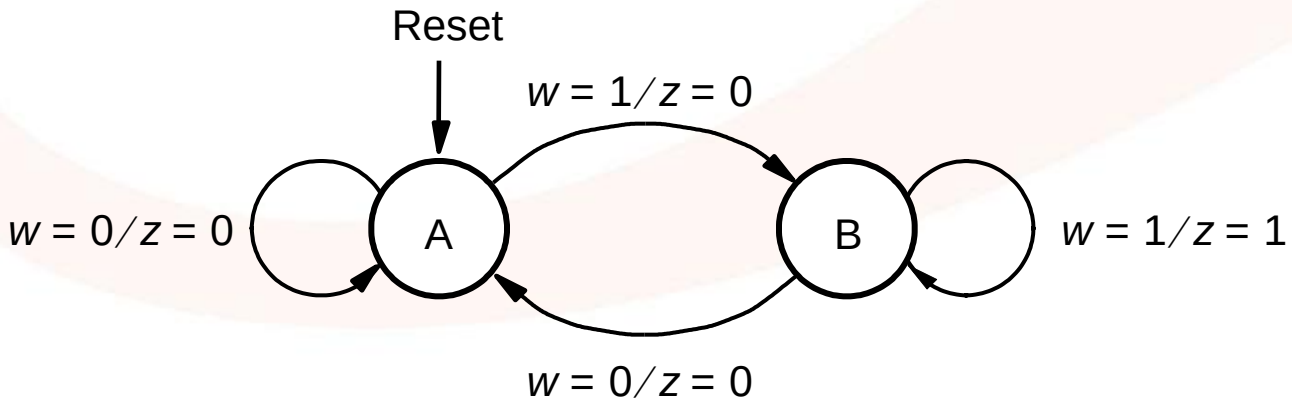
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0



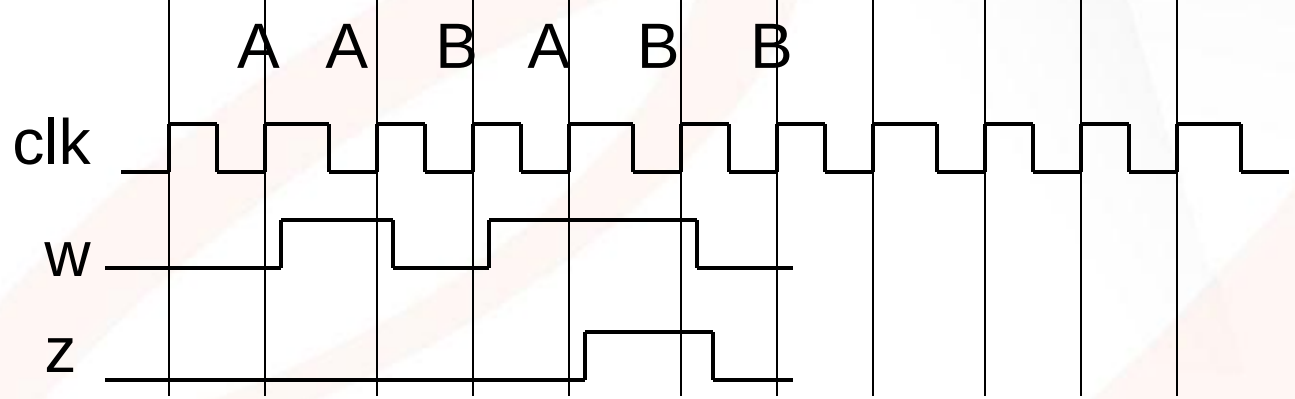


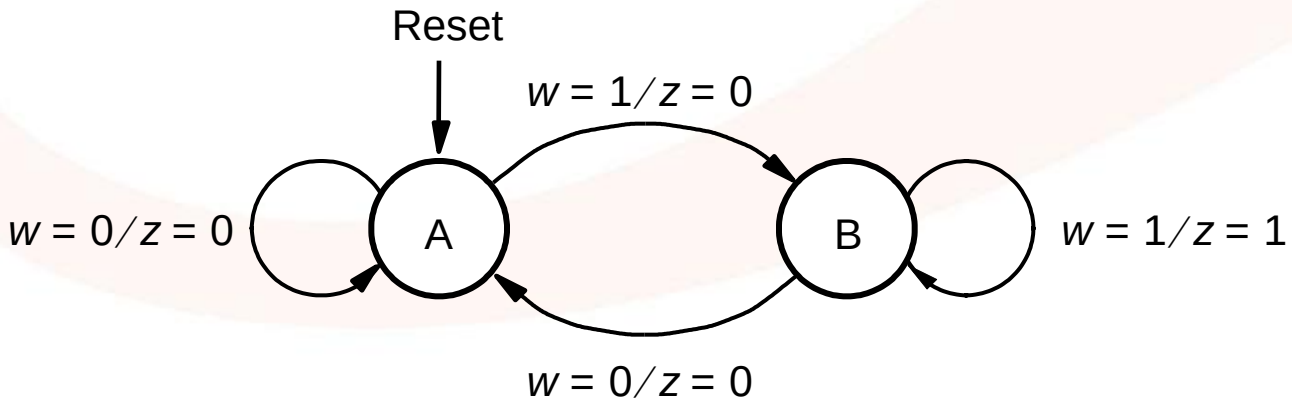
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0



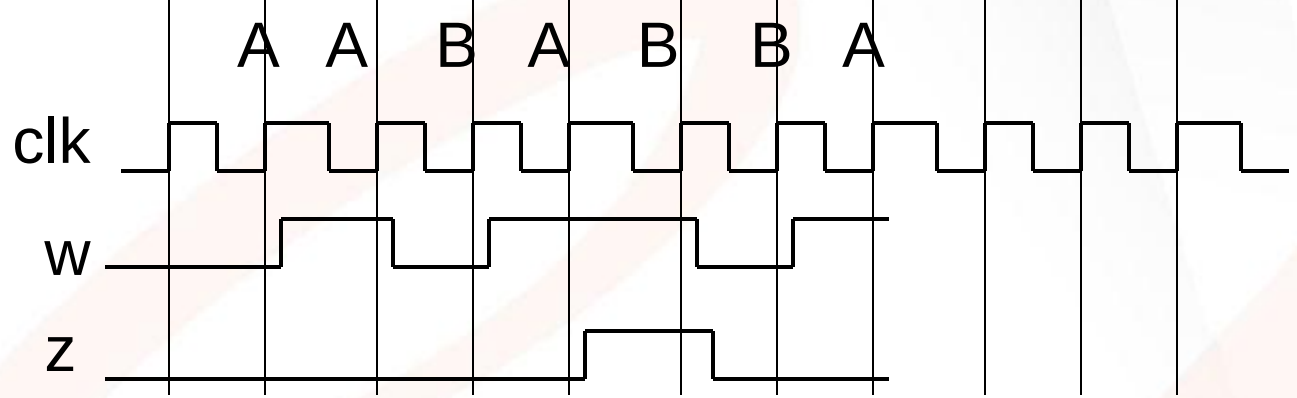


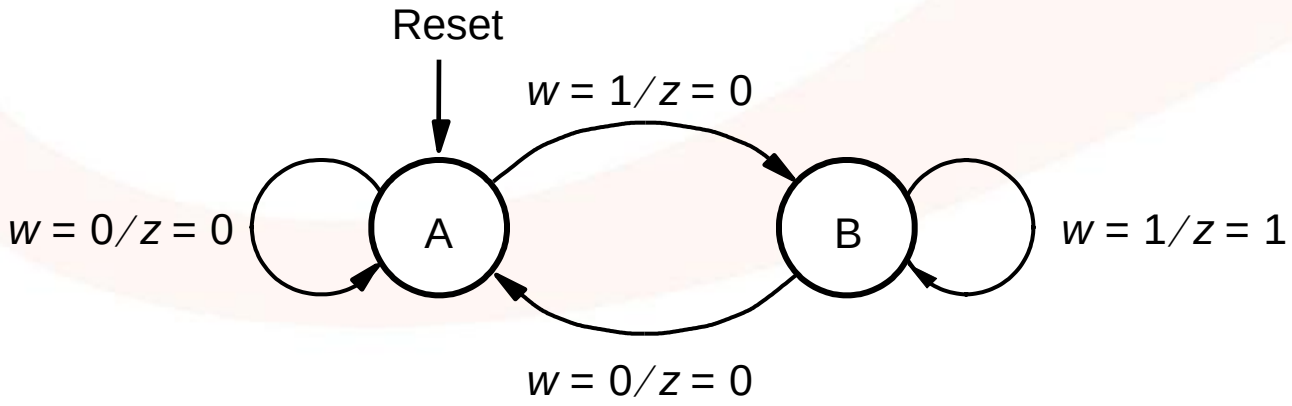
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0



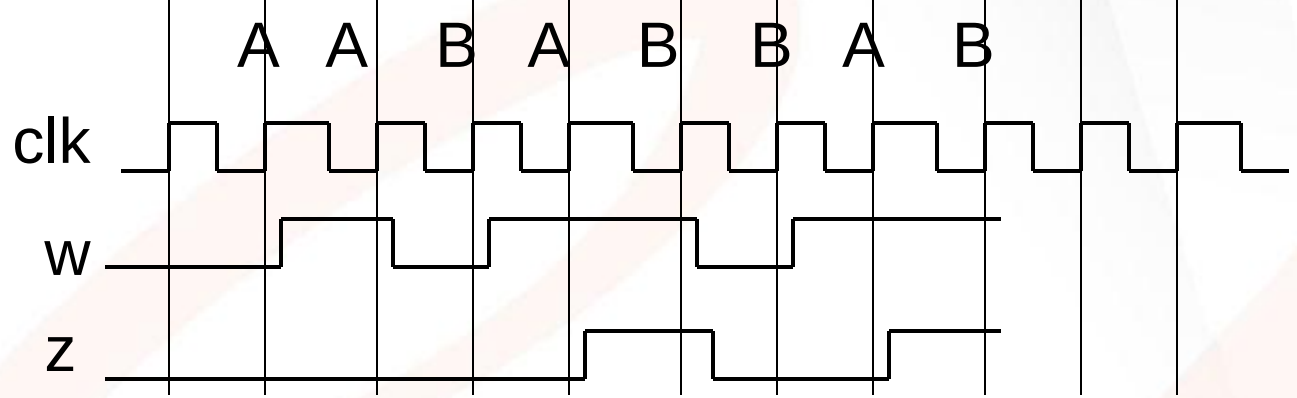


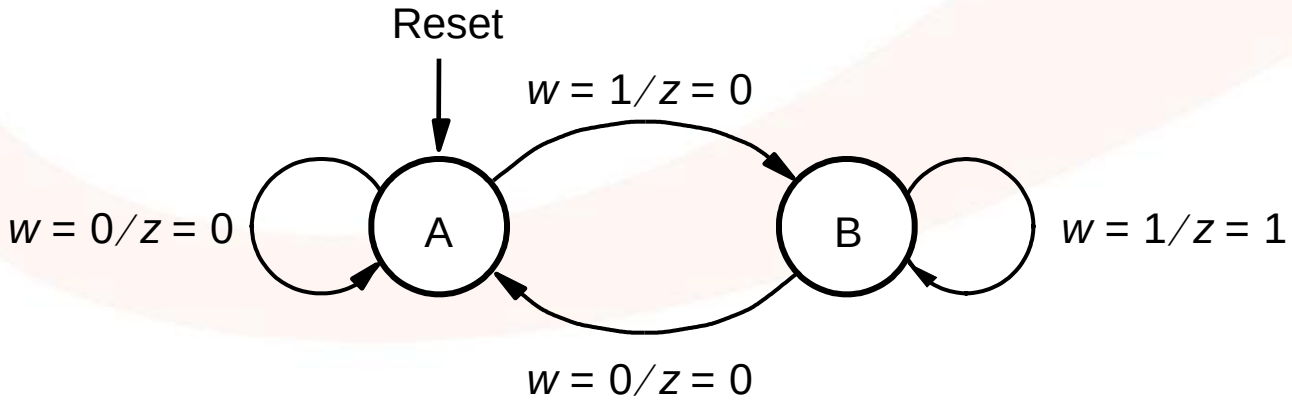
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0



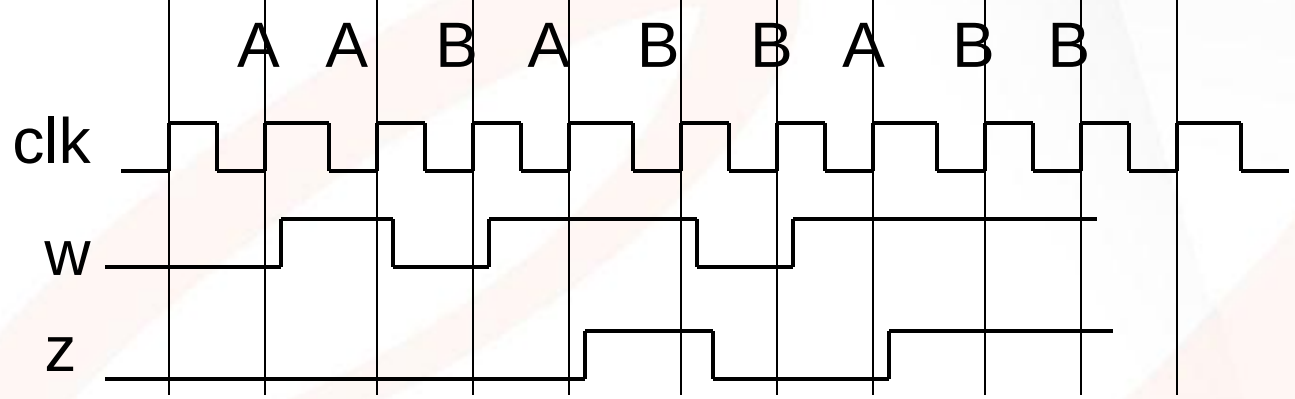


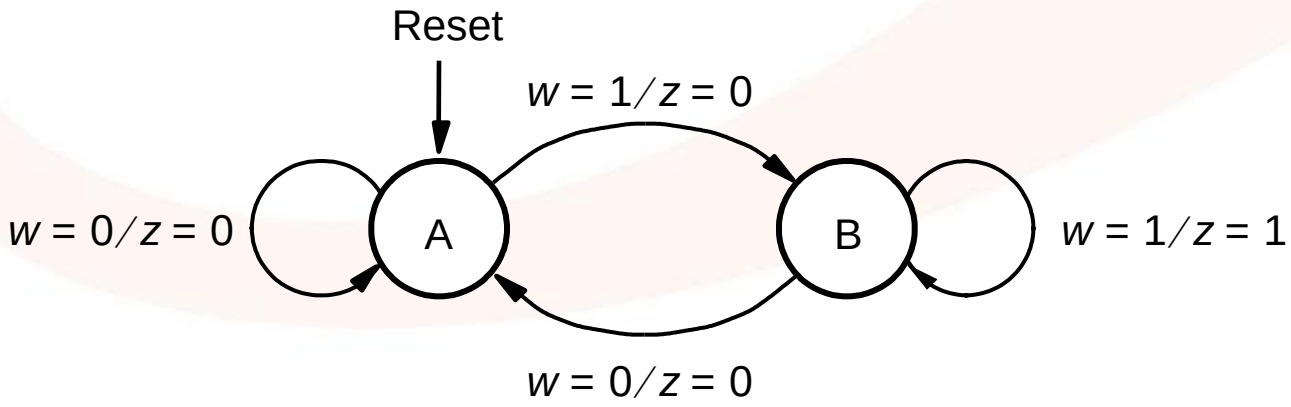
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0



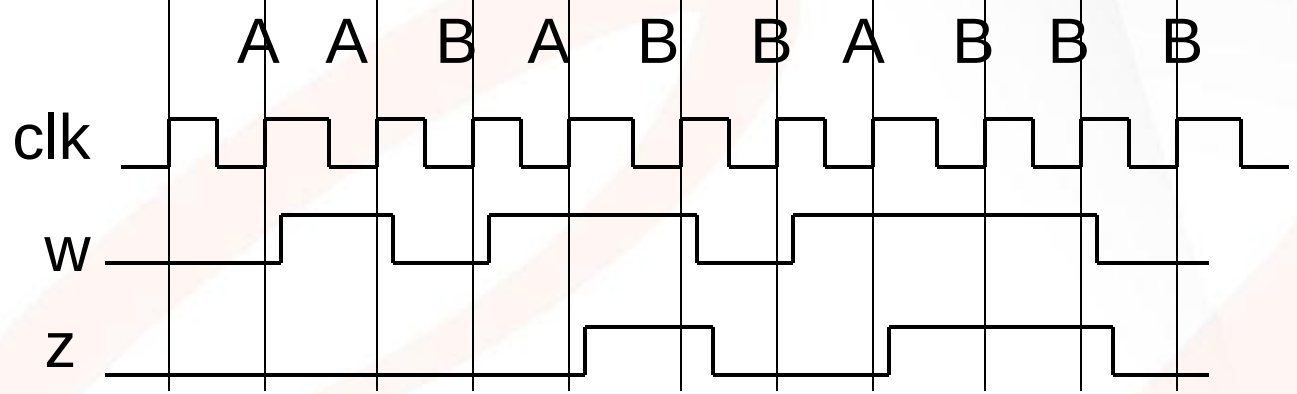


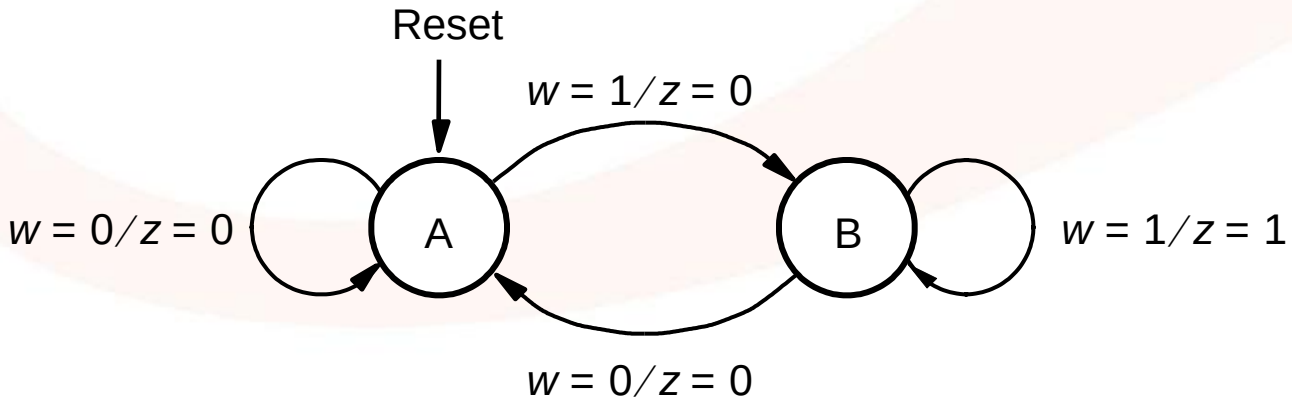
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0



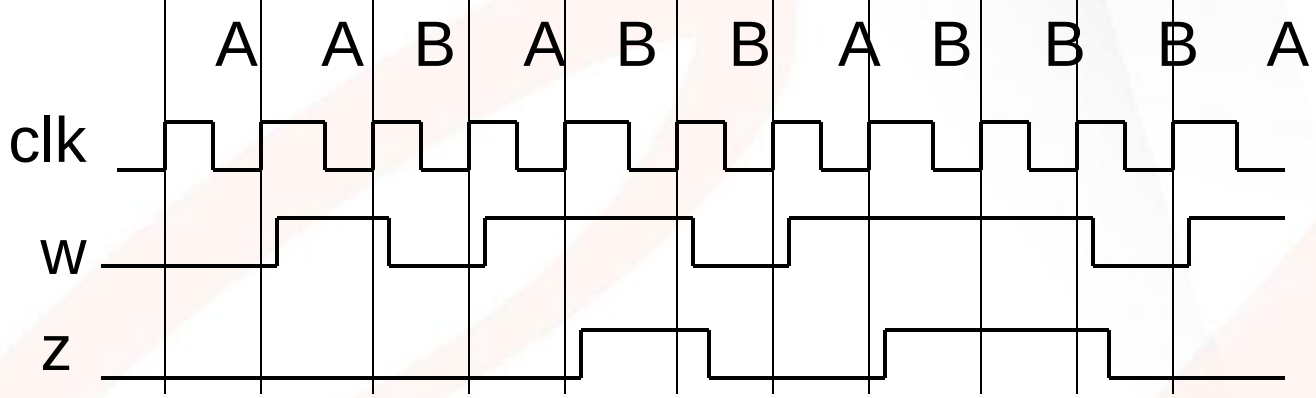


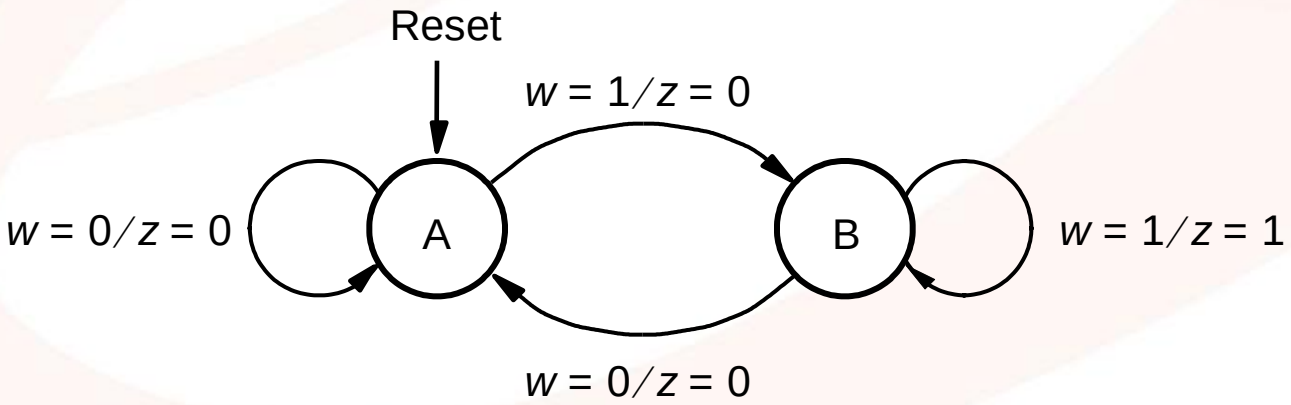
Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0





Clockcycle:	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
w:	0	1	0	1	1	0	1	1	1	0	1
z:	0	0	0	0	1	0	0	1	1	0	0





During present clock cycle output value on arc from Present State node
Note difference from Moore model in where output is shown output changes asynchronously

State Table & State Assignment

Present state	Next state		Output z	
	$w = 0$	$w = 1$	$w = 0$	$w = 1$
A	A	B	0	0
B	A	B	0	1

	Present state	Next state		Output	
		$w = 0$	$w = 1$	$w = 0$	$w = 1$
	y	Y	Y	z	z
A	0	0	1	0	0
B	1	0	1	0	1

Derivation of Next-state and Output Expressions

	Present state	Next state		Output	
		$w = 0$	$w = 1$	$w = 0$	$w = 1$
	y	Y	Y	z	z
A	0	0	1	0	0
B	1	0	1	0	1

Y Truth Table

y	w	Y
0	0	0
0	1	1
1	0	0
1	1	1

	y	
w	0	1
0	0	0
1	1	1

Y = w

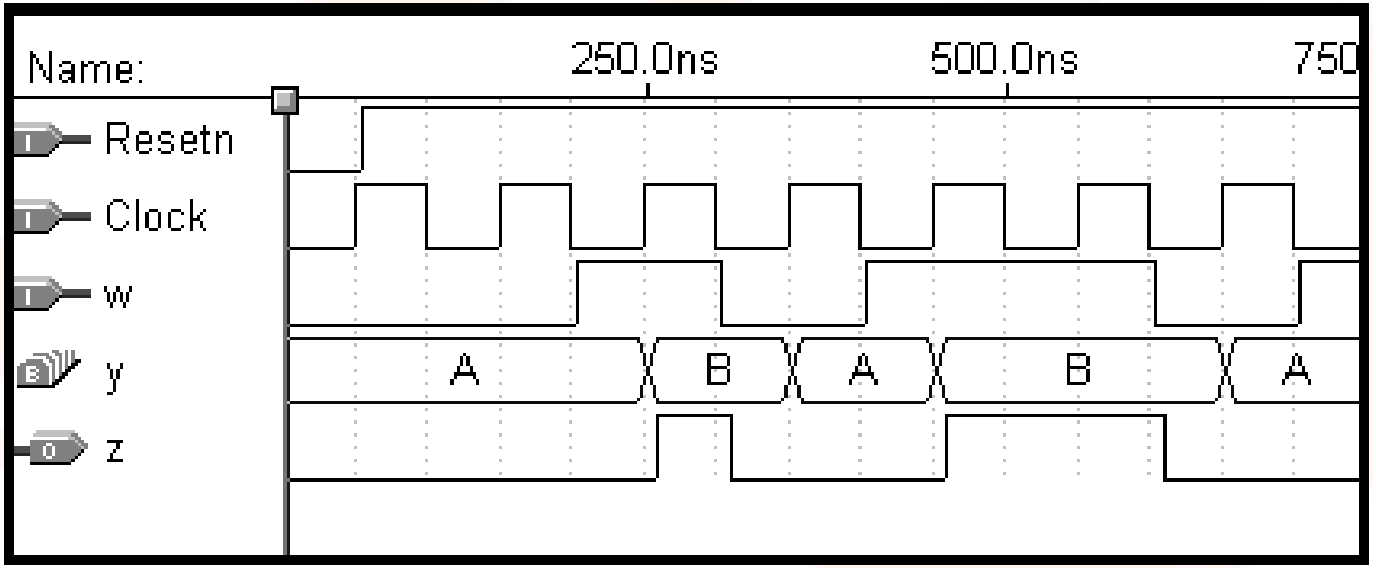
z Truth Table

y	w	z
0	0	0
0	1	0
1	0	0
1	1	1

	y	
w	0	1
0	0	0
1	0	1

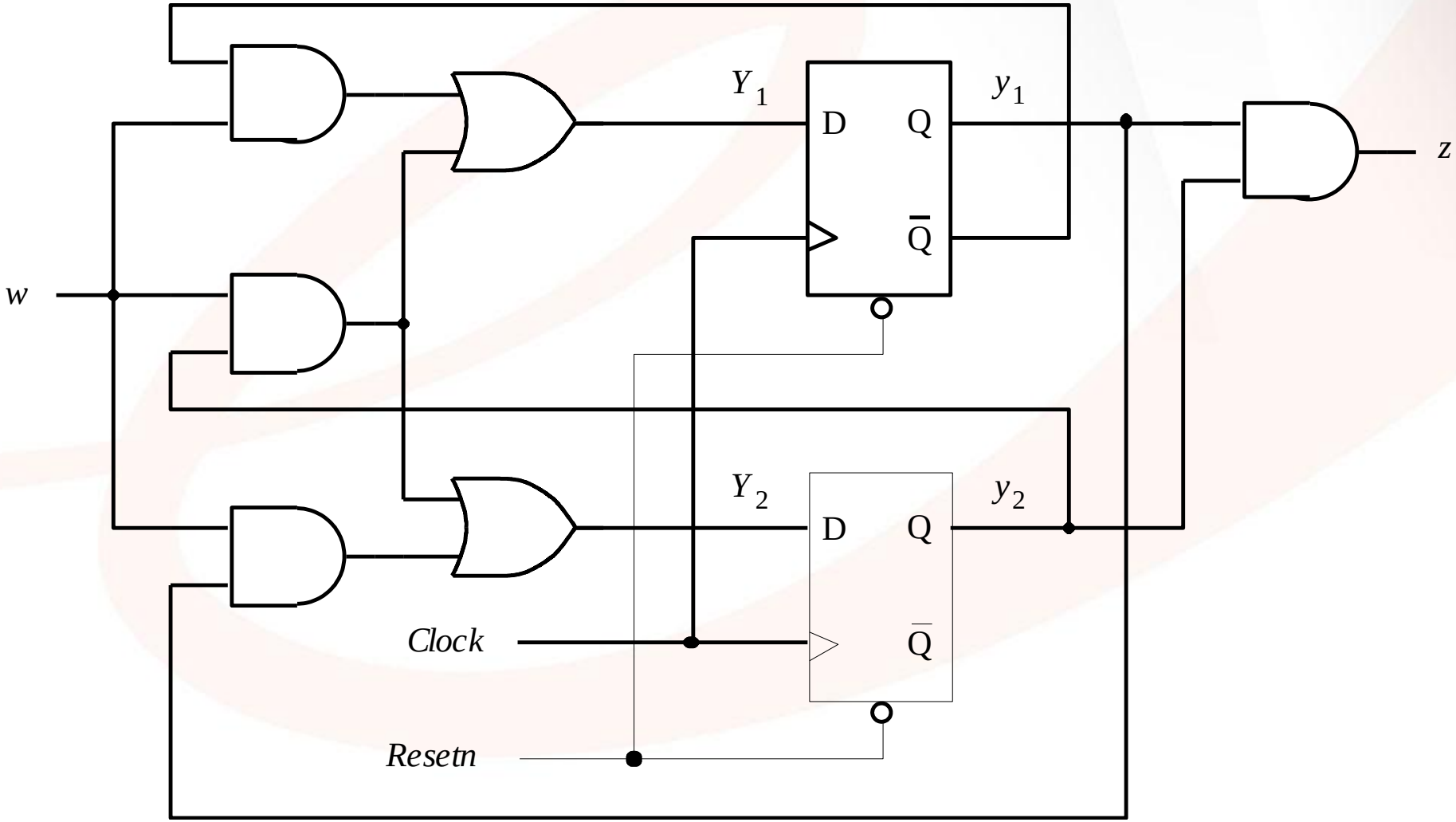
z = y·w

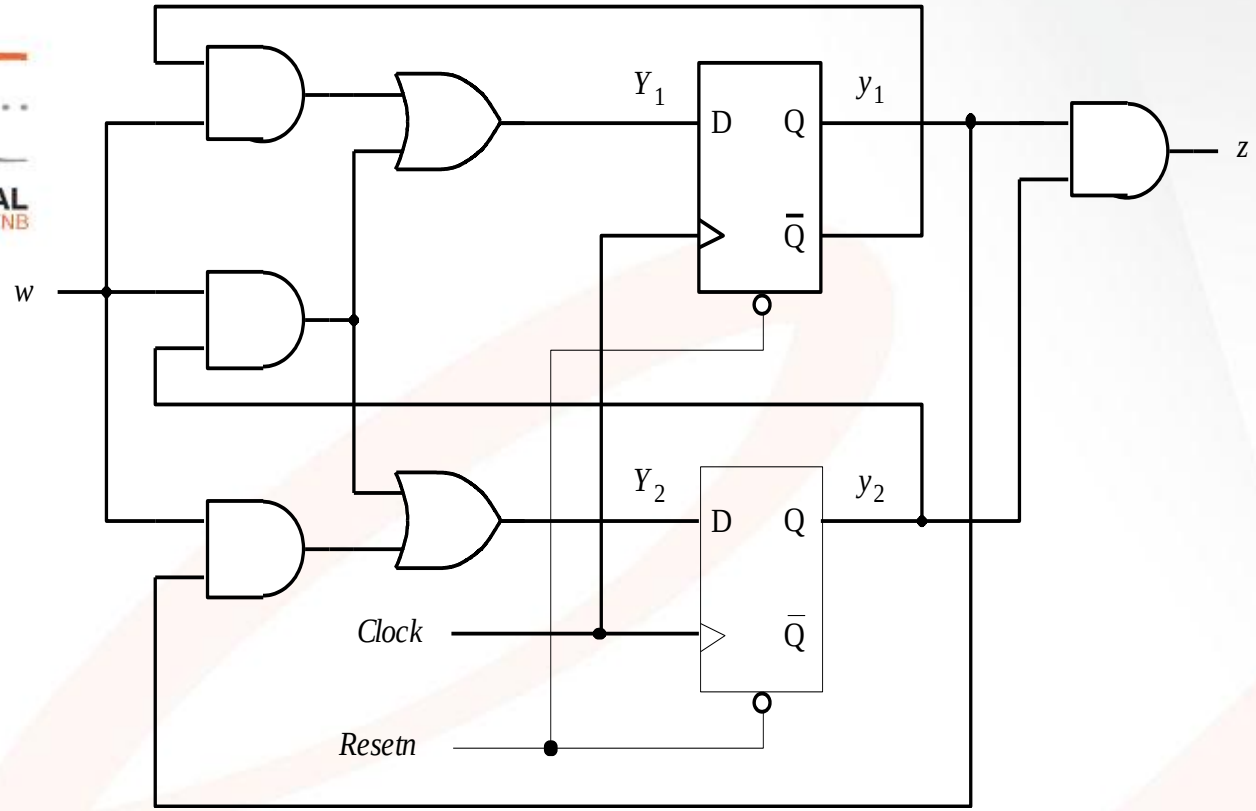
Asynchronous Behavior of Mealy Model



- If z is connected to another synchronous clock no problem
- If z is connected to another circuit not controlled by the same clock a potential problem exists

Analysis of Synchronous State Machines

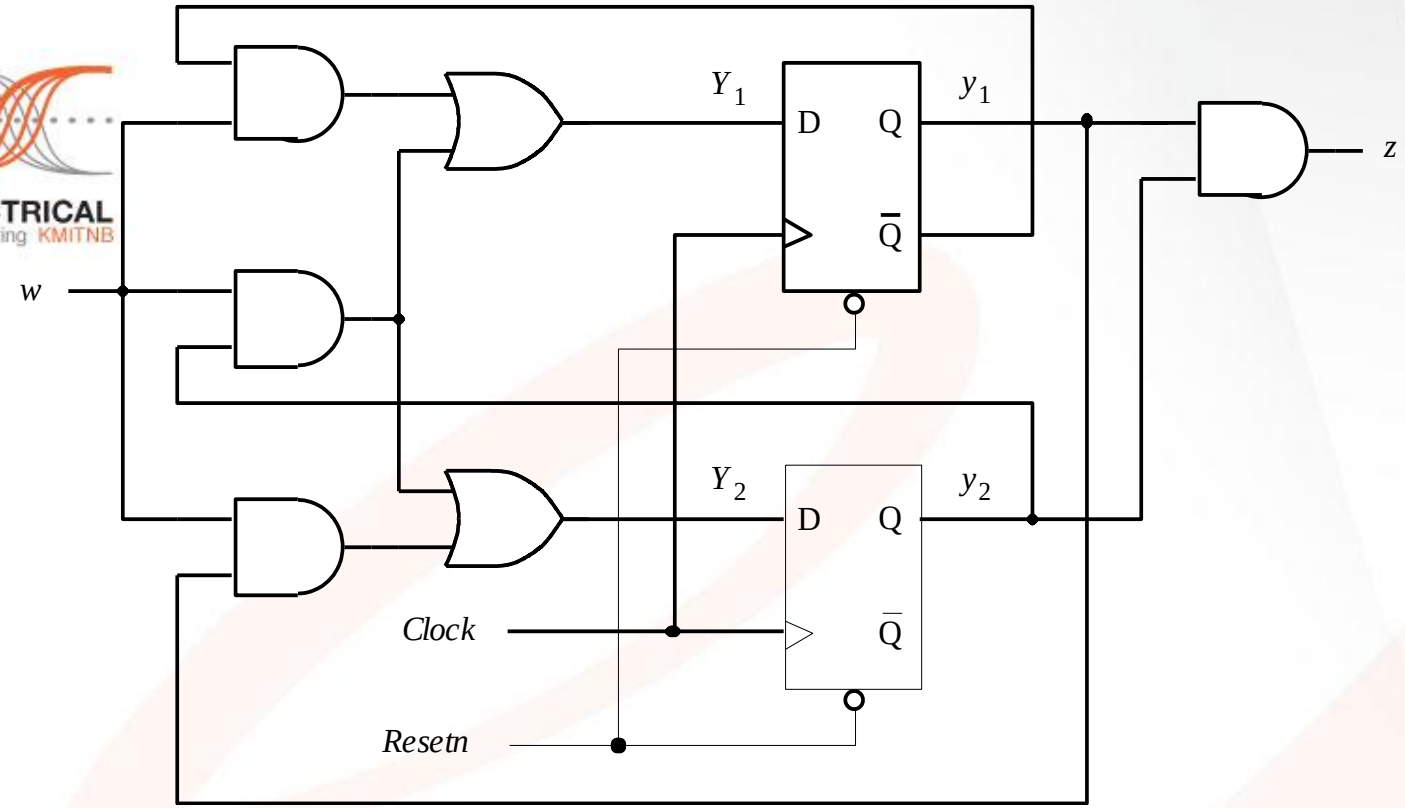




How many states?

Create Present State/Next State/Output table

Present state	Next State	Output z



Present state y_2y_1	Next State		Output z
	w = 0	w = 1	
	Y_2Y_1	Y_2Y_1	
0 0	0 0	01	0
0 1	0 0	10	0
1 0	0 0	11	0
1 1	0 0	11	1

Exercise#1: Counter Using Sequential State Machine

- Counting Sequence is 0,1,2,…,6,7,0,1,…
- Input w
  - If w = 0 present count remains the same
  - If w = 1 count is incremented
- Create State Diagram
- Create State Table
- Create State Assignments

Exercise#2: FSM with Two Inputs

- Inputs w1 and w2
- Outputs z
- If w1 = w2 during any four consecutive clock cycles z = 1
- Otherwise z = 0

w1	0	1	1	0	1	1	1	0	0	0	1	1	0
w2	1	1	1	0	1	0	1	0	0	0	1	1	1
z	0	0	0	0	1	0	0	0	0	0	1	1	0

ELECTRICAL
Engineering KMITNB

Homework

- 8.3 & 8.5 (Brown)
  - Create
    - State Diagram
    - State Table
    - State Assignment Table