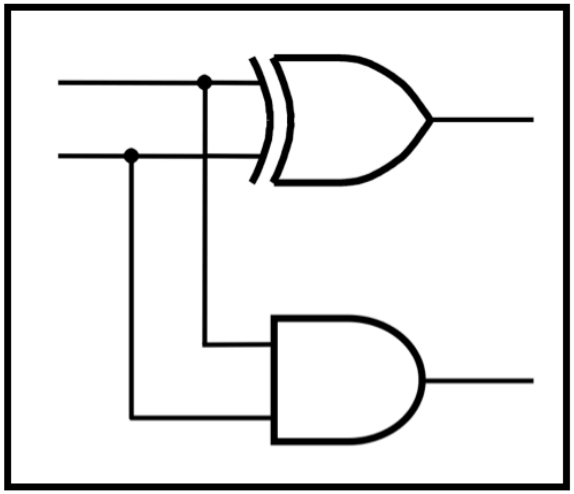


CprE 281: Digital Logic

Instructor: Alexander Stoytchev

http://www.ece.iastate.edu/~alexs/classes/

State Assignment Problem

CprE 281: Digital Logic
Iowa State University, Ames, IA

Administrative Stuff

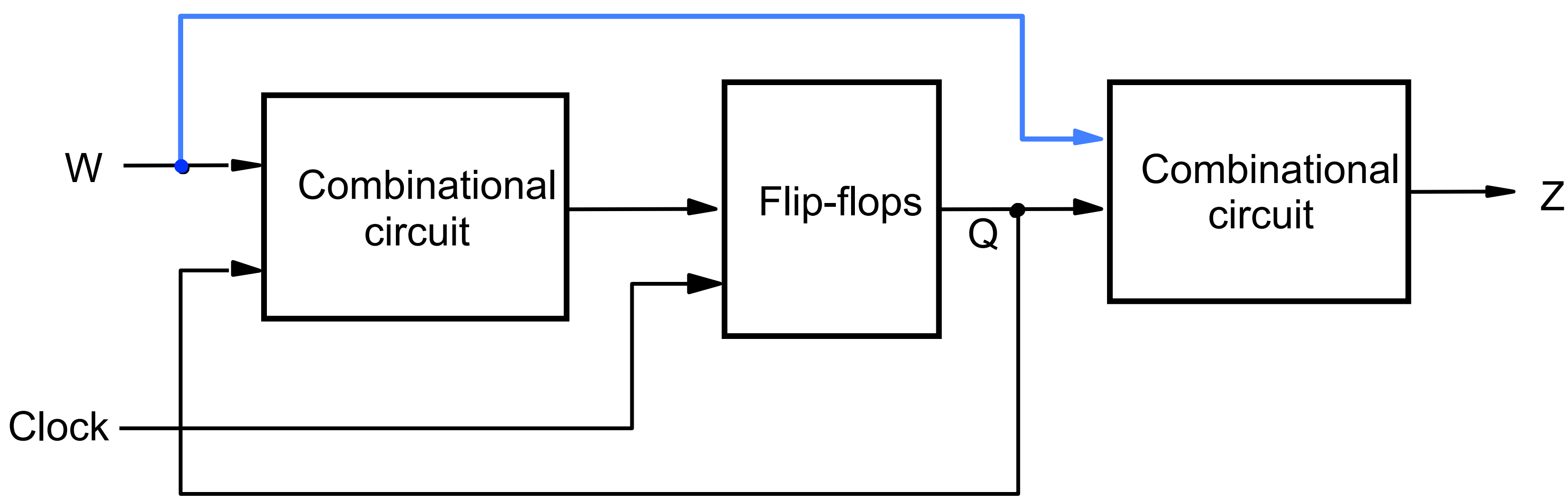
- Homework 9 is due on Monday

Administrative Stuff

- Homework 10 is out
- It is due on Monday Nov 16 @ 4pm

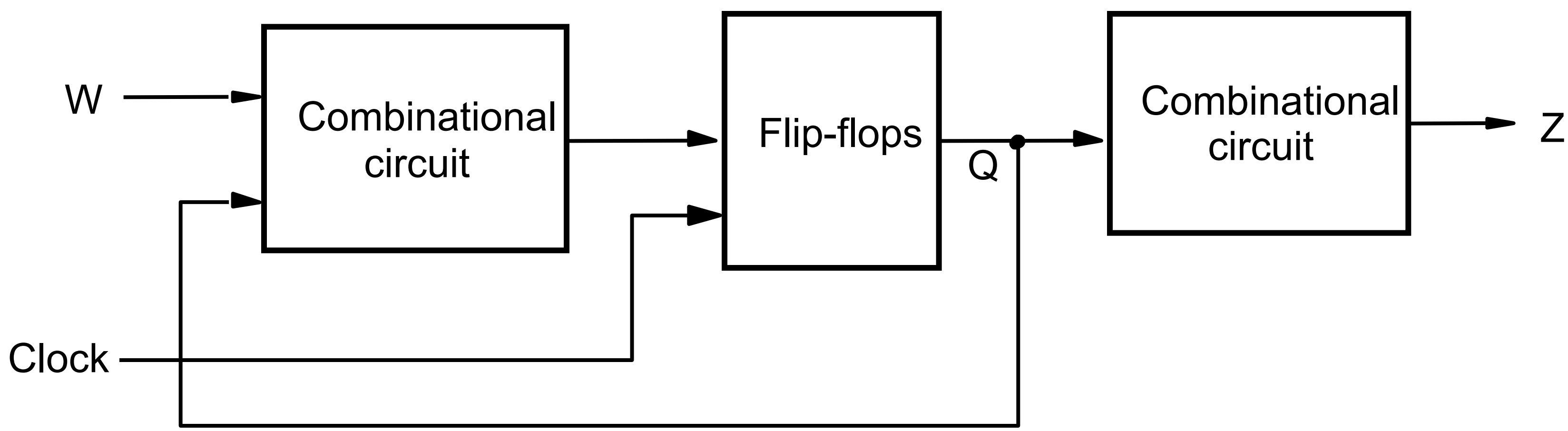
Quick Review

The general form of a synchronous sequential circuit



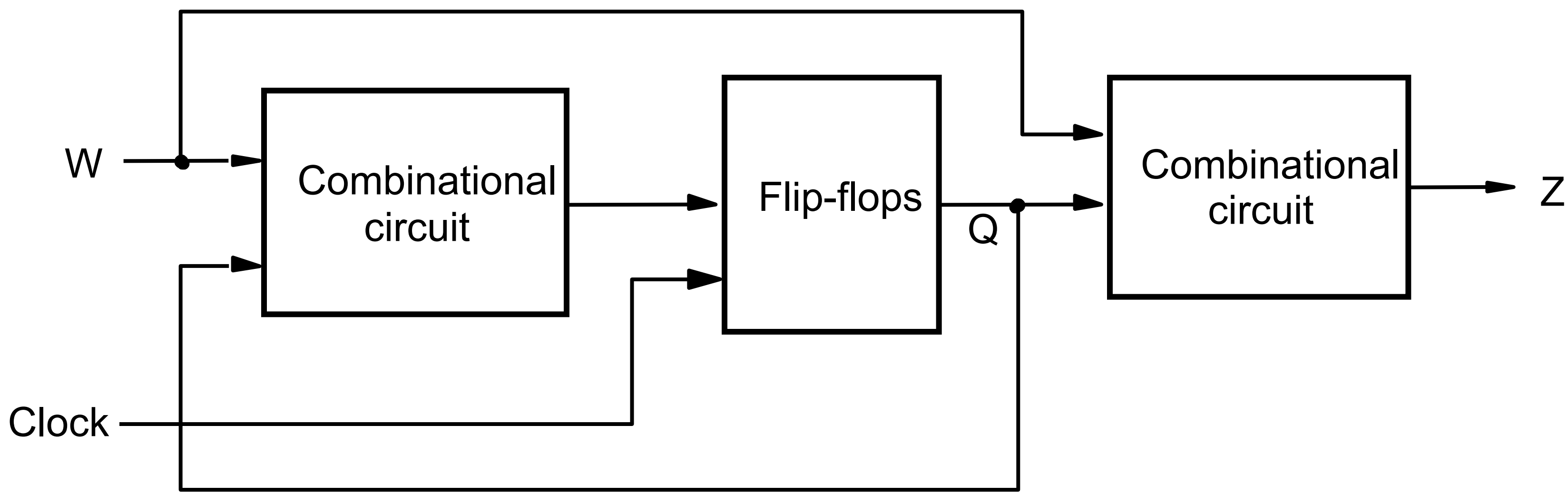
[Figure 6.1 from the textbook]

Moore Type



W
Combinational
circuit
Flip-flops
Q
Combinational
circuit
Z
Clock

Mealy Type


W
Combinational
circuit
Flip-flops
Q
Combinational
circuit
Z
Clock

Moore Machine

- **The machine's current state and current inputs are used to decide which next state to transition into.**

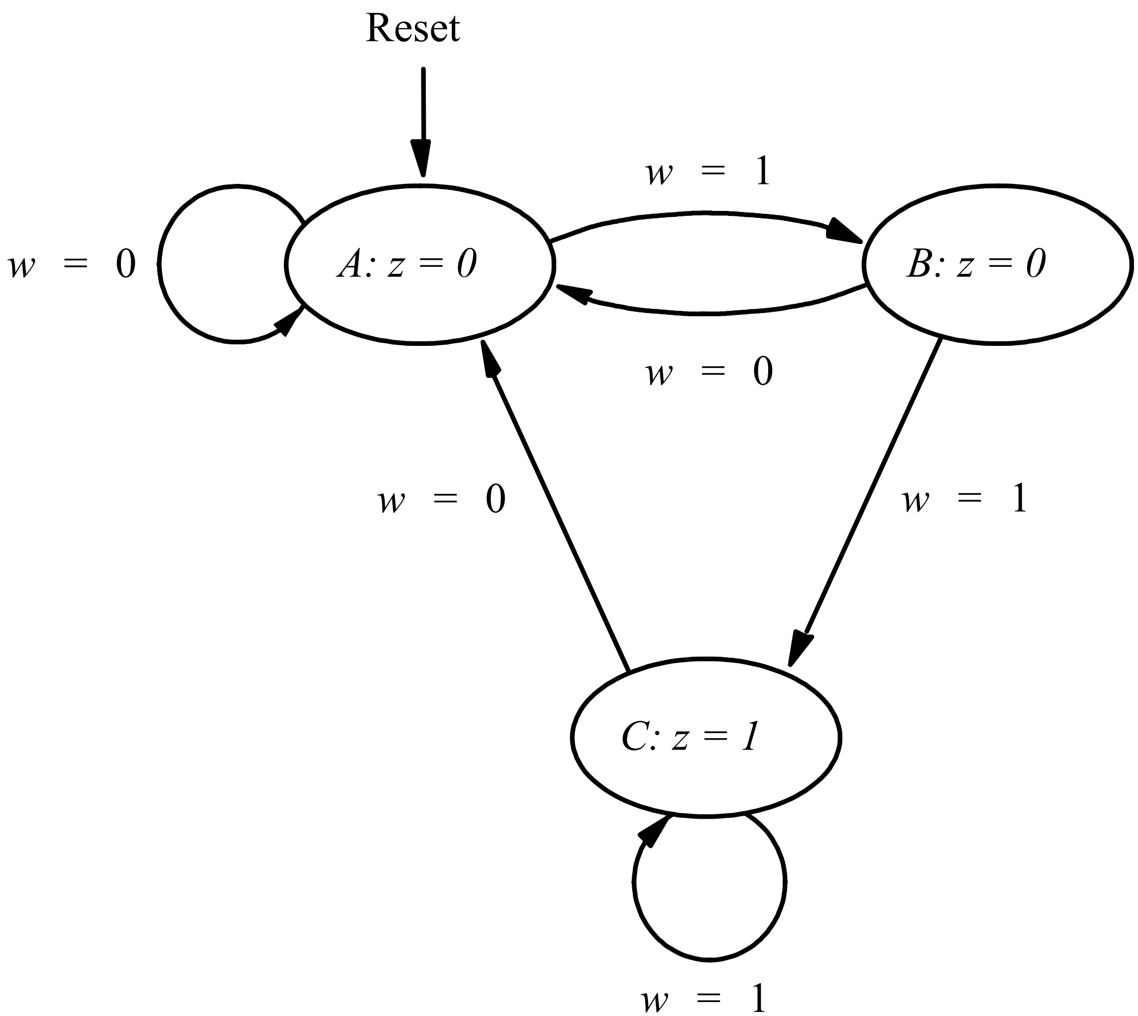
- **The machine's current state decides the current output.**

Mealy Machine

- **The machine's current state and current inputs are used to decide which next state to transition into.**

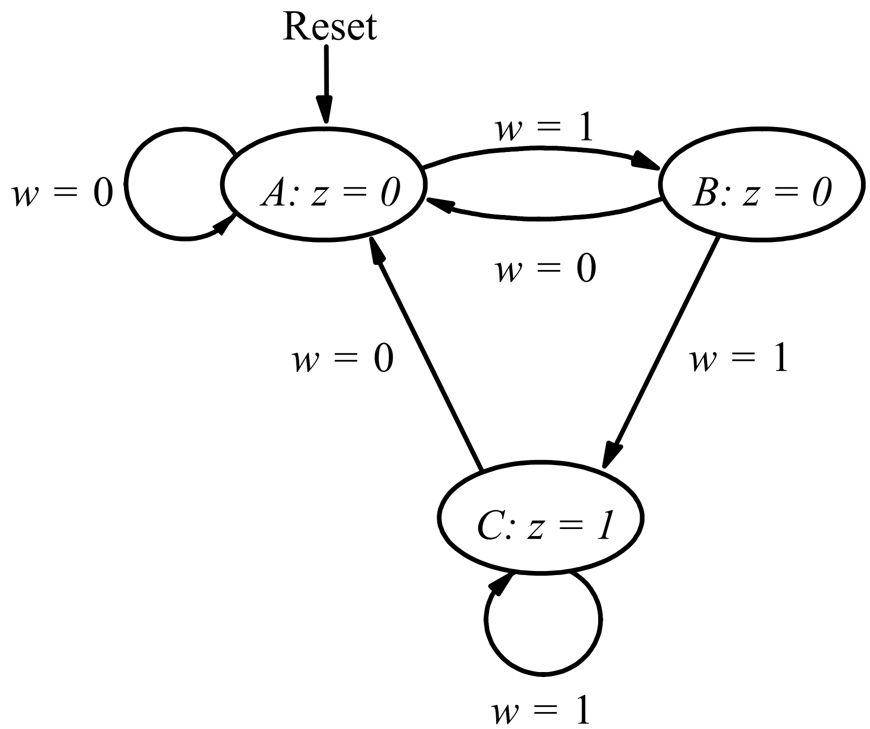
- **The machine's current state and current input values decide the current output.**

Example #1

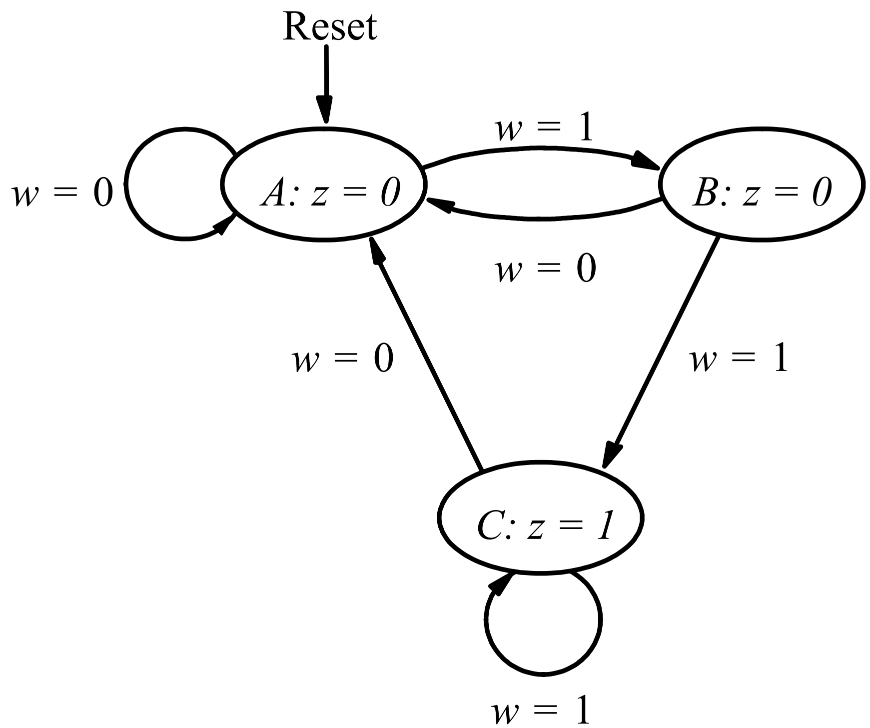


We need to find both the *next state logic* and the *output logic* implied by this machine.

[Figure 6.3 from the textbook]



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



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

[Figure 6.4 from the textbook]

How to represent the States?

One way is to encode each state with a 2-bit binary number

A ~ 00

B ~ 01

C ~ 10

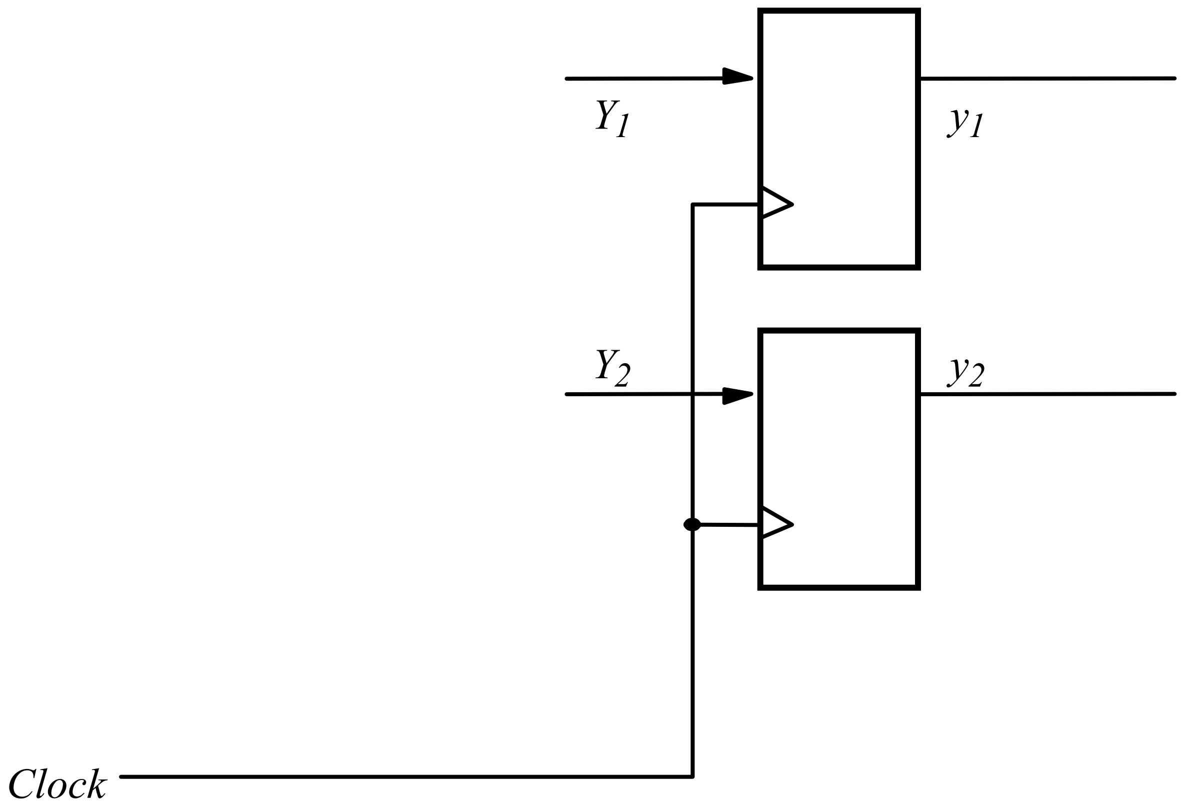
How to represent the states?

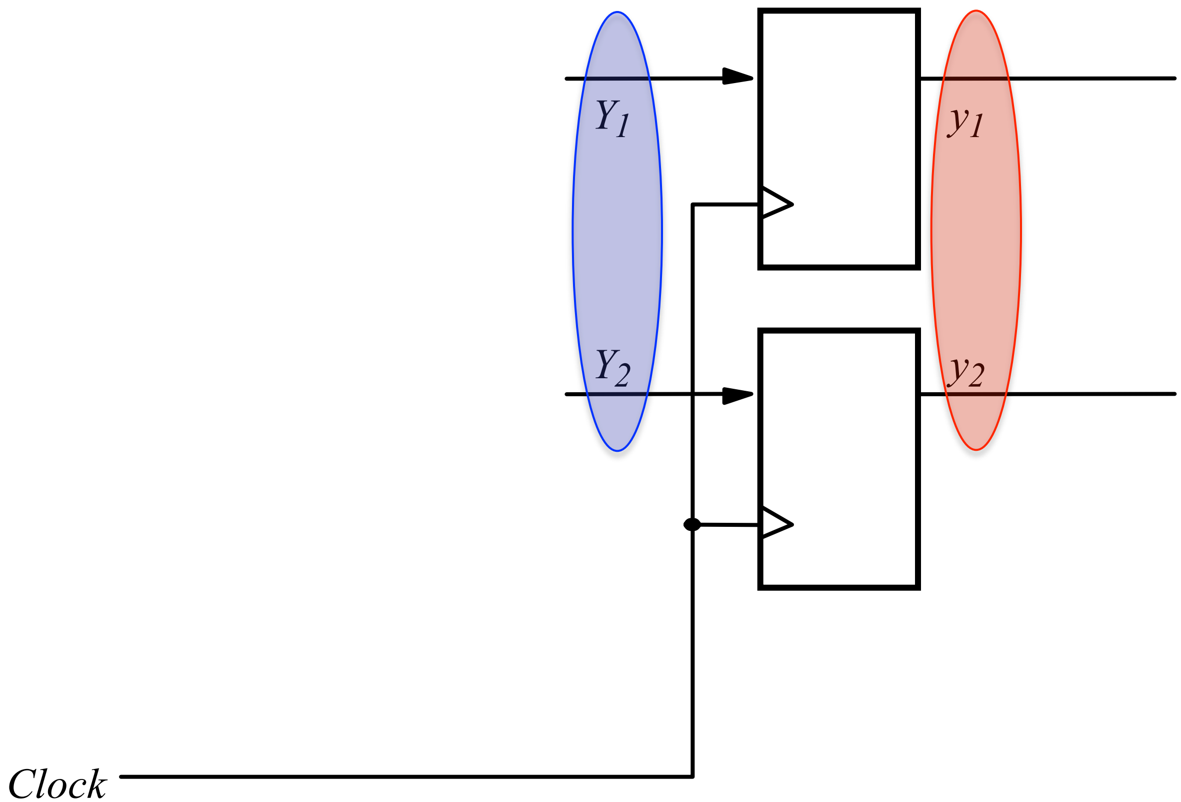
One way is to encode each state with a 2-bit binary number

A ~ 00
B ~ 01
C ~ 10

How many flip-flops do we need?

Let’s use two flip flops
to hold the state machine’s state

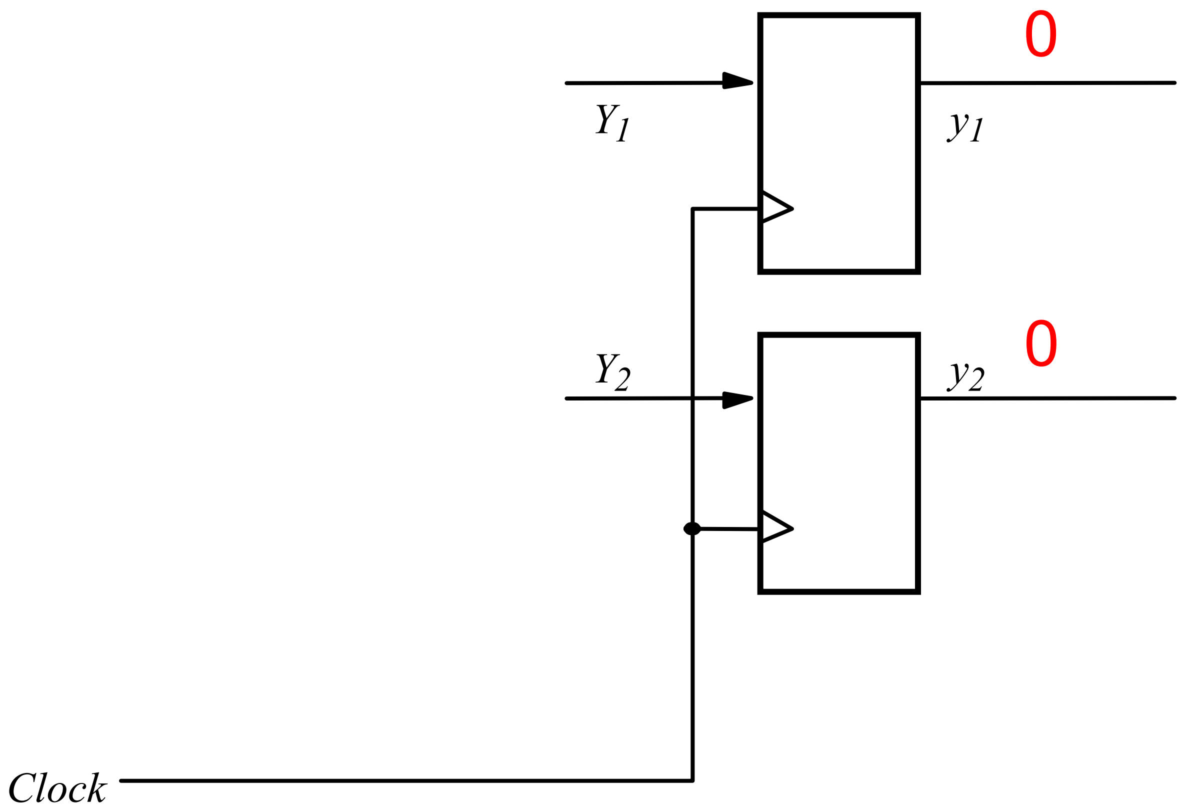

Y_1
y_1
Y_2
y_2
Clock



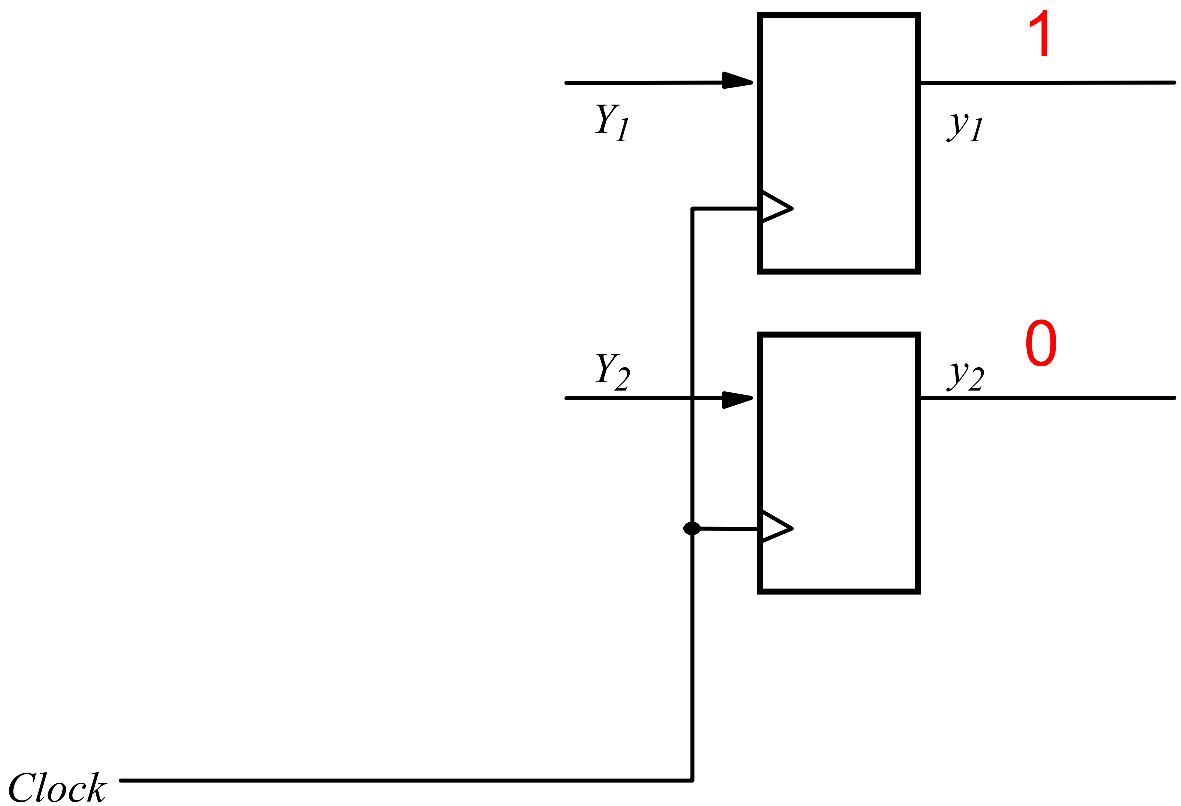
We will call y_1 and y_2 the *present state variables*.

We will call Y_1 and Y_2 the *next state variables*.

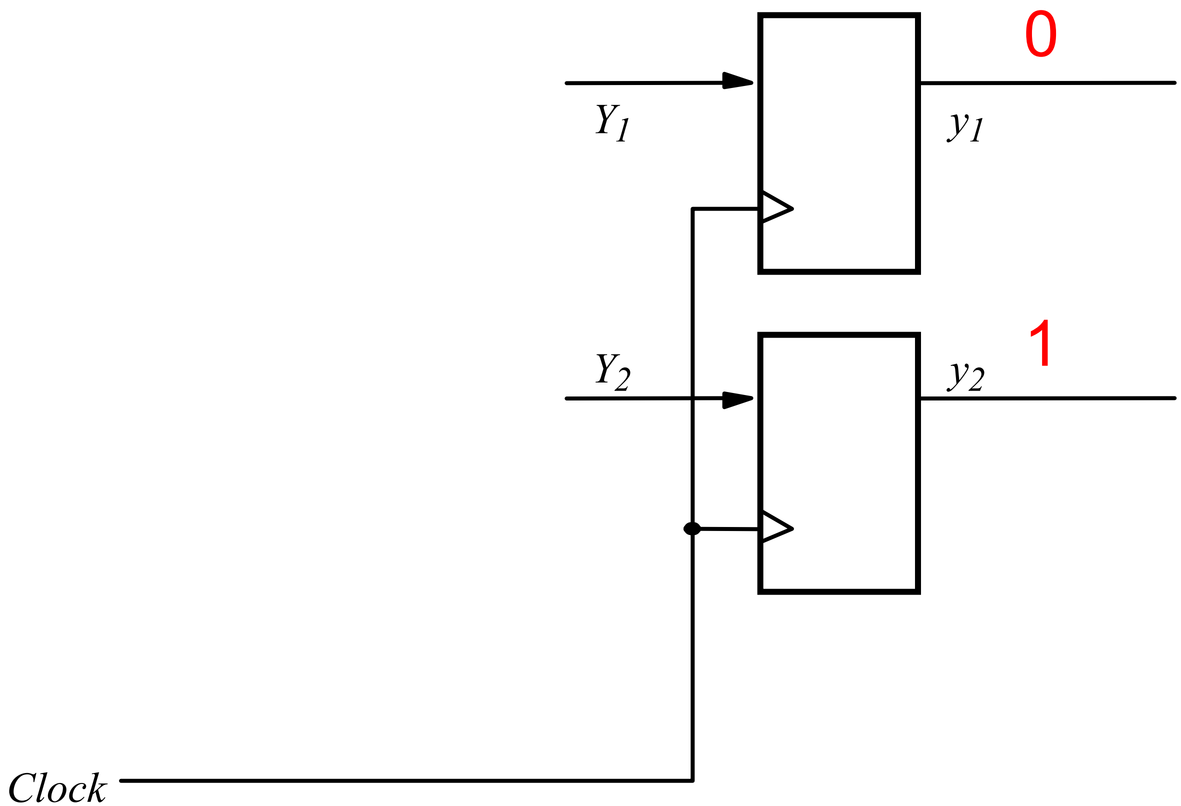
[Figure 6.5 from the textbook]



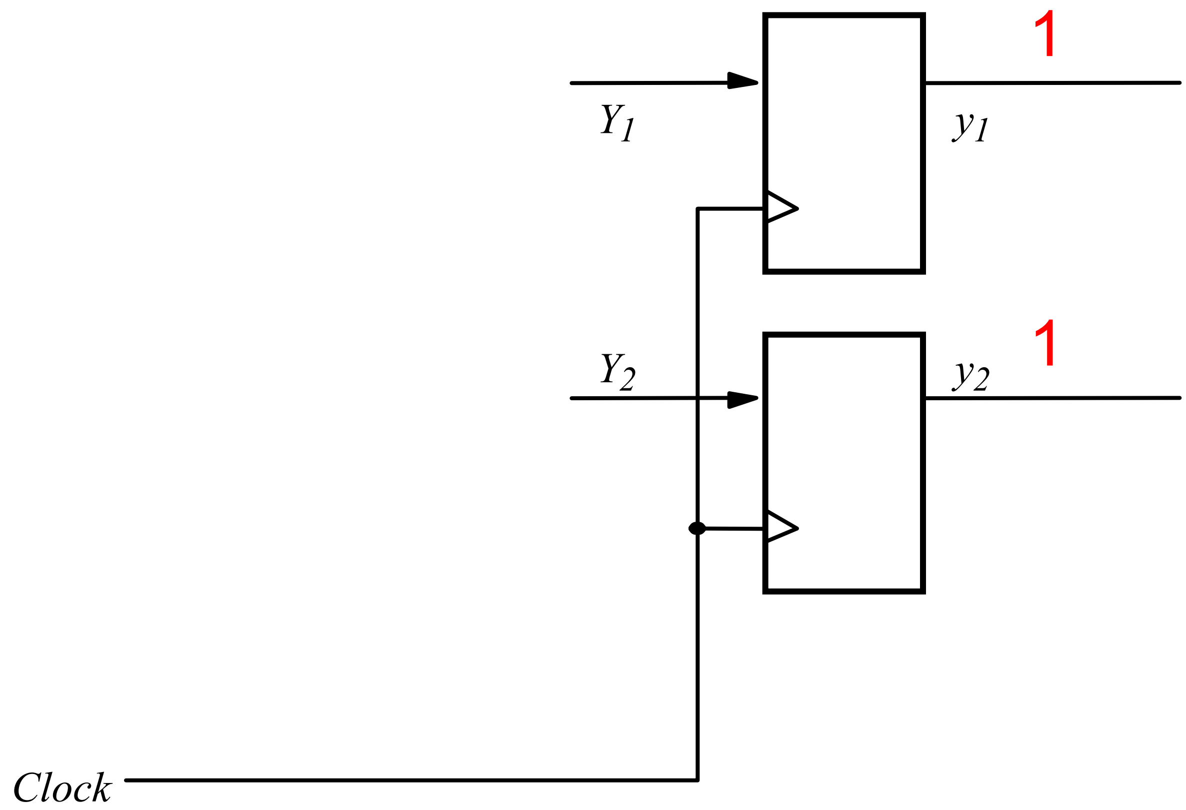
Two zeros on the output JOINTLY represent state A.



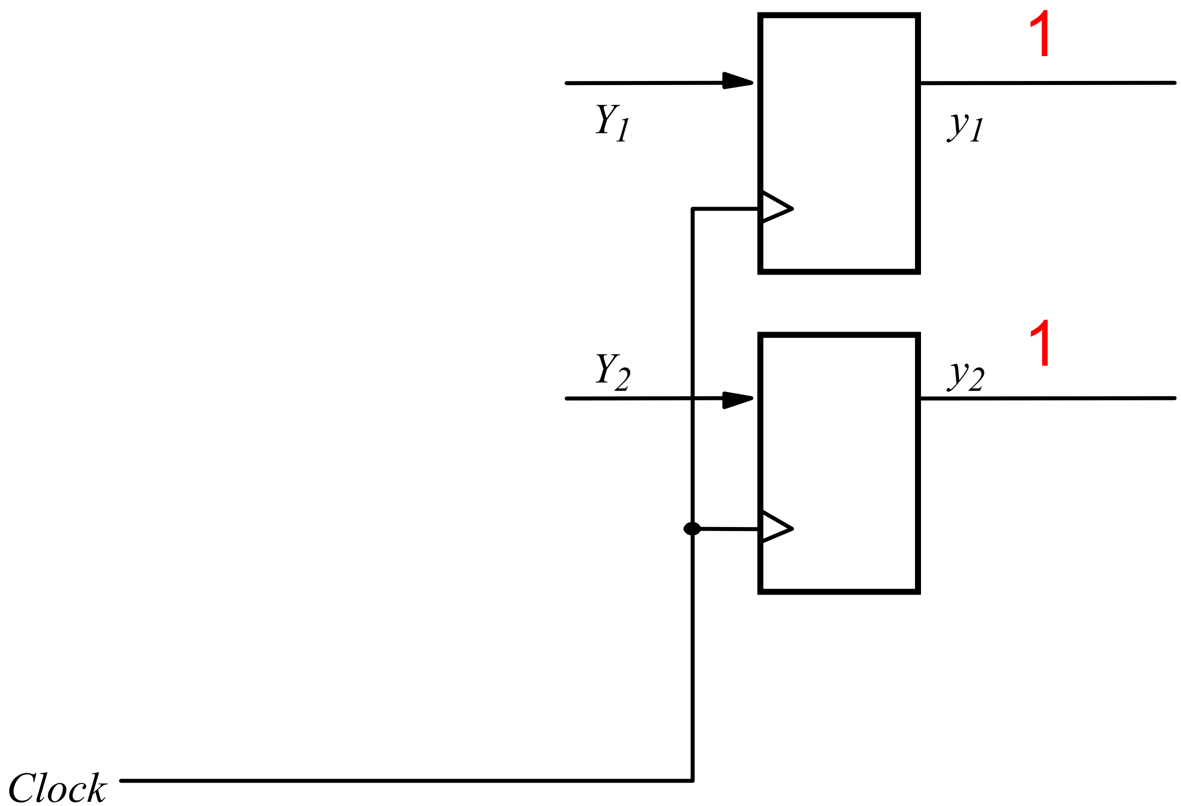
This flip-flop output pattern represents state B.



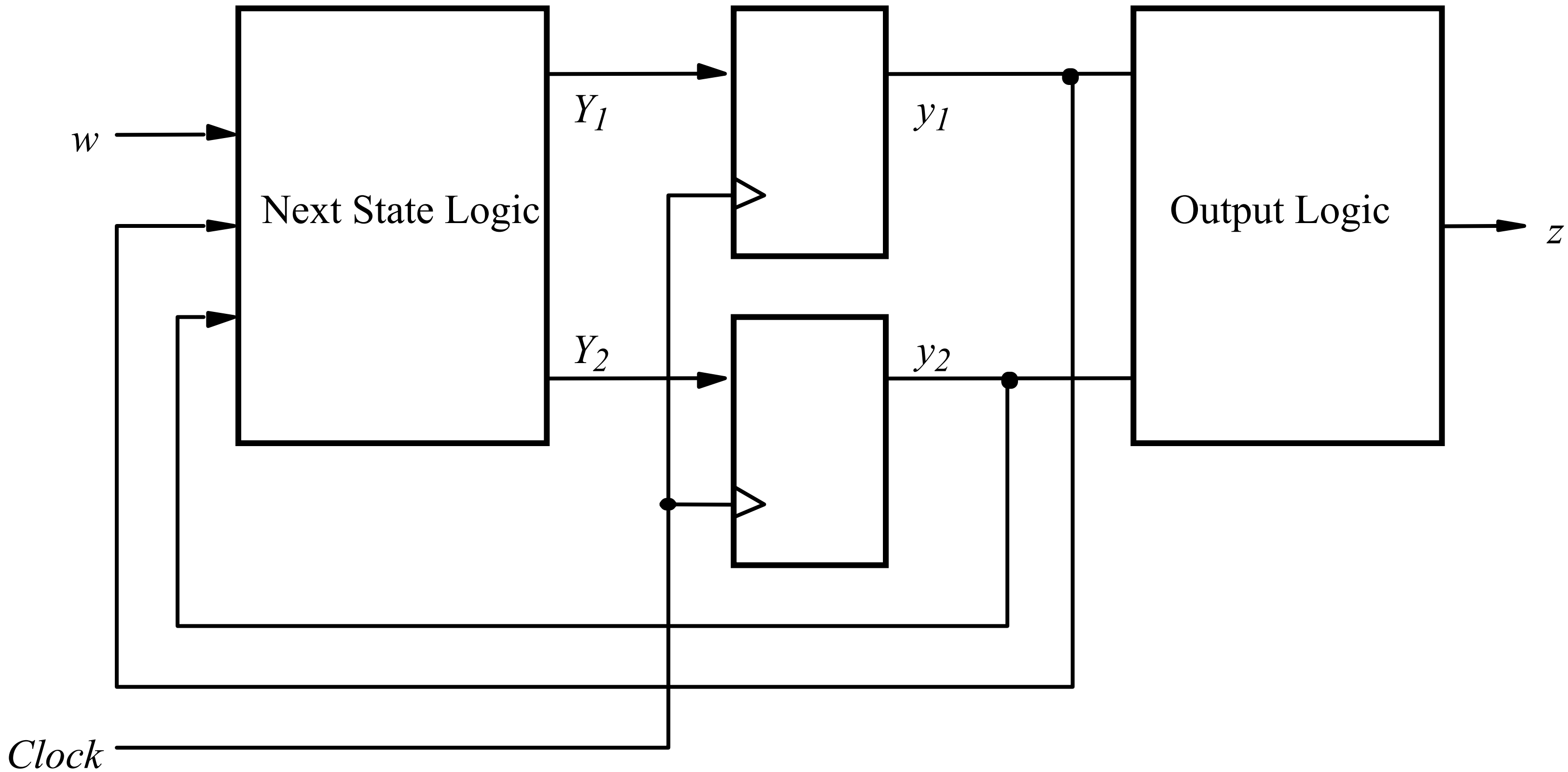
This flip-flop output pattern represents state C.



What does this flip-flop output pattern represent?



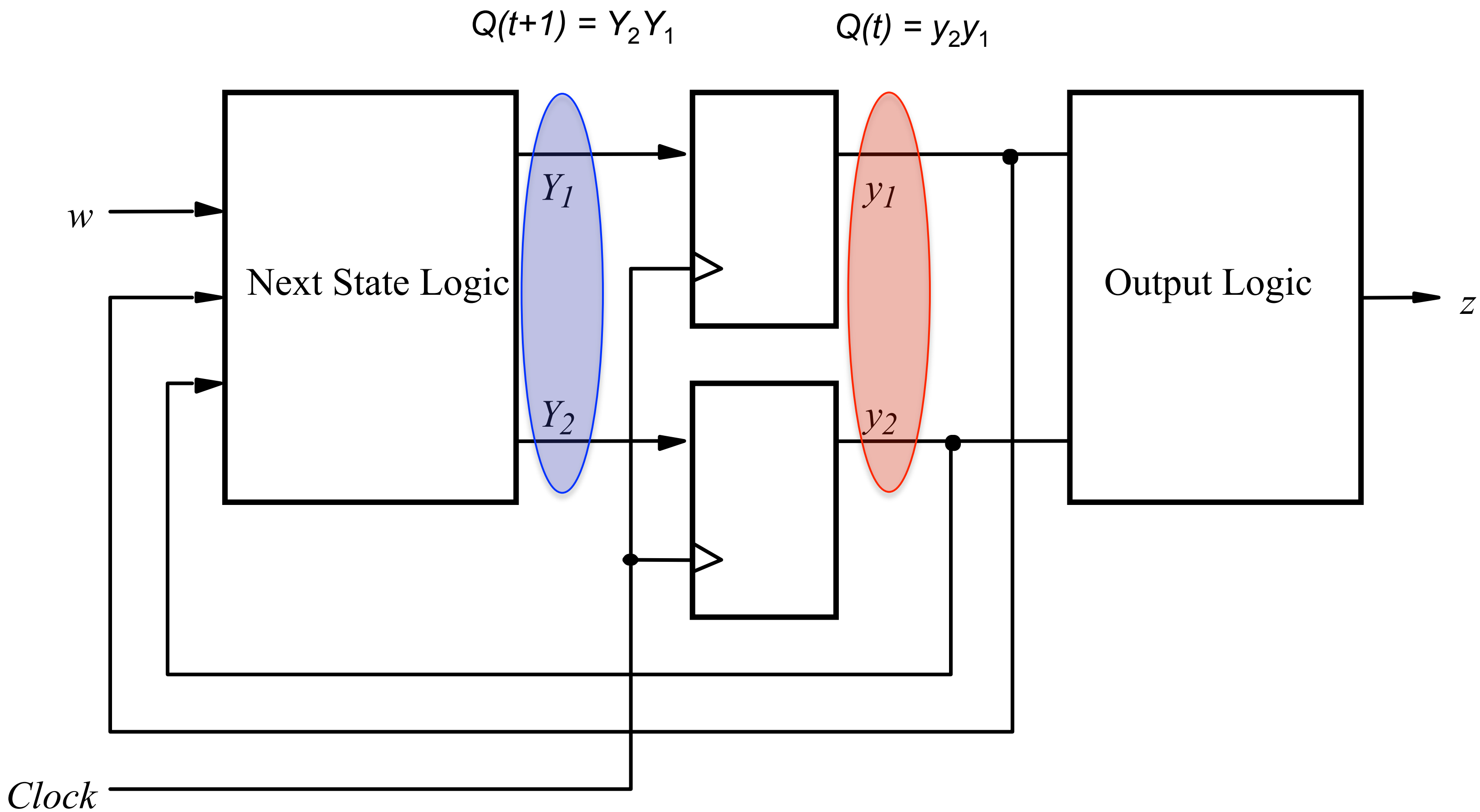
This would be state D, but we don't have one in this example. So this is an impossible state.



We will call y_1 and y_2 the *present state variables*.

We will call Y_1 and Y_2 the *next state variables*.

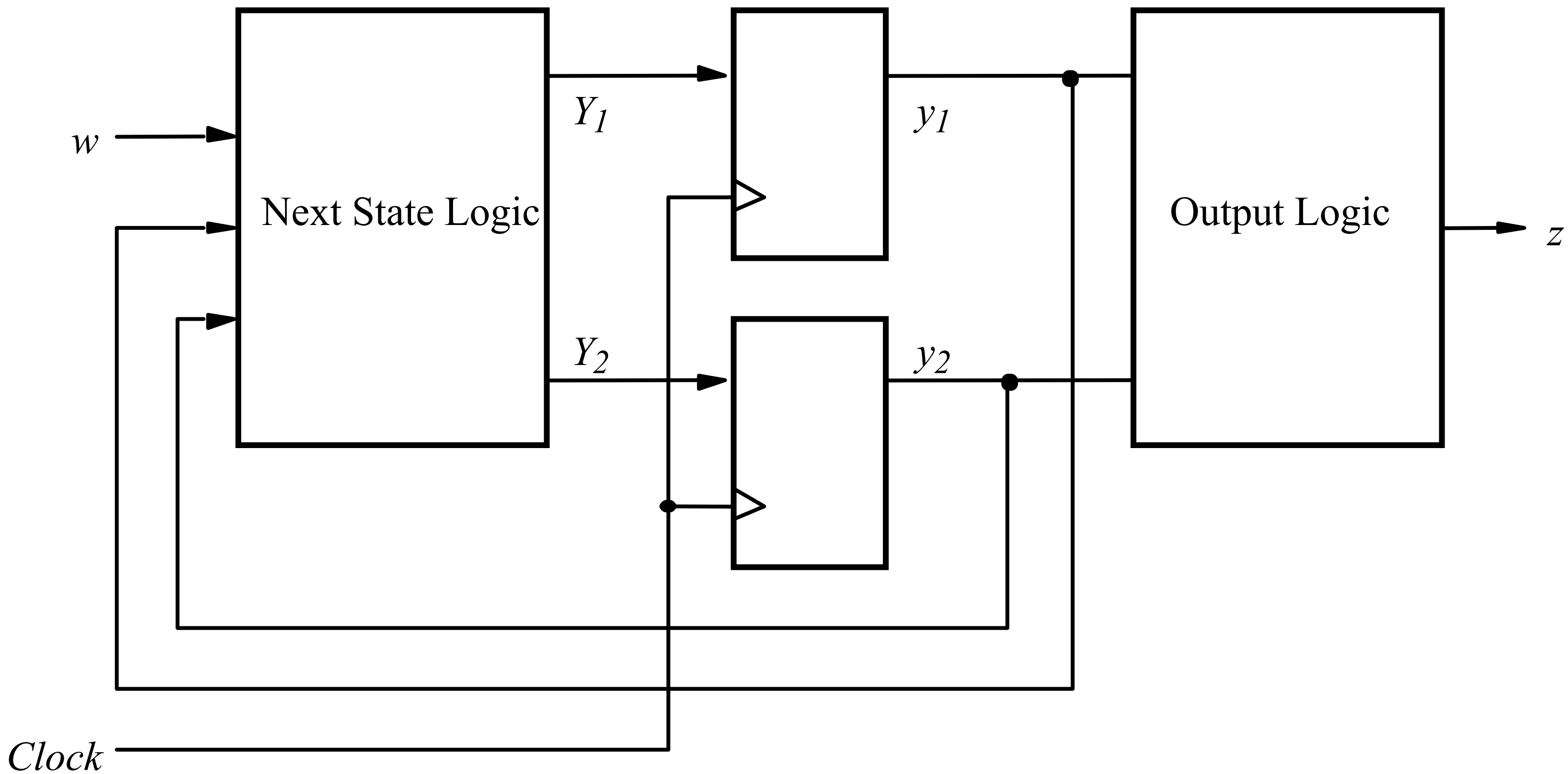
[Figure 6.5 from the textbook]



We will call y_1 and y_2 the *present state variables*.

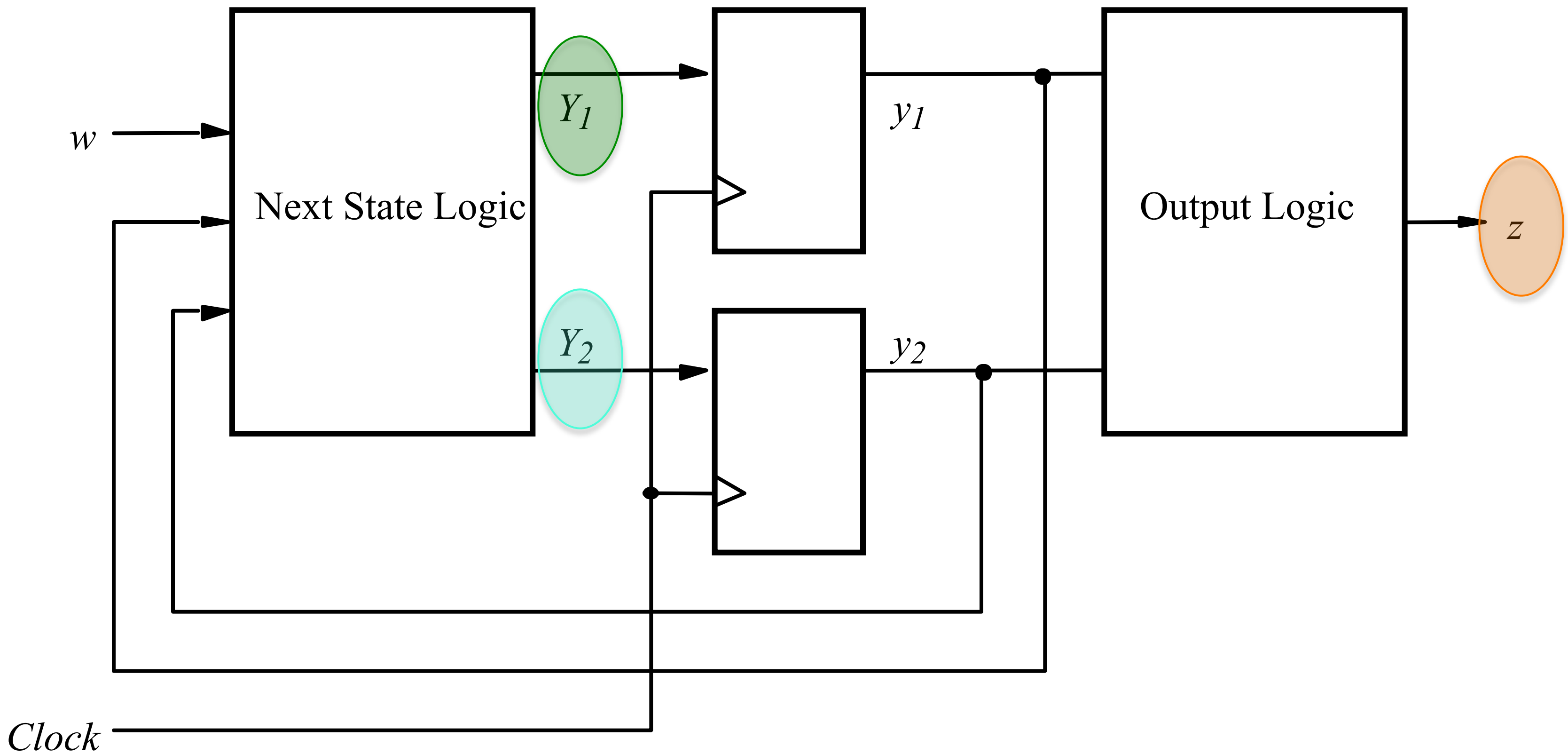
We will call Y_1 and Y_2 the *next state variables*.

[Figure 6.5 from the textbook]



We need to find logic expressions for
$Y_1(w, y_1, y_2)$, $Y_2(w, y_1, y_2)$, and $z(y_1, y_2)$.

[Figure 6.5 from the textbook]



We need to find logic expressions for

$Y_1(w, y_1, y_2)$, $Y_2(w, y_1, y_2)$, and $z(y_1, y_2)$.

[Figure 6.5 from the textbook]

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

Suppose we encoded our states in the same order in which they were labeled:

A ~ 00
B ~ 01
C ~ 10

[Figure 6.4 from the textbook]

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

	Present state	Next state		Output z
		$w = 0$	$w = 1$	
A	00			
B	01			
C	10			
	11			

The finite state machine will never reach a state encoded as 11.

[Figure 6.6 from the textbook]

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

	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*

We arbitrarily chose these as our state encodings. We could have used others.

[Figure 6.6 from the textbook]

$Q(t) = y_2y_1$ and $Q(t+1) = Y_2Y_1$

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

w	y_2	y_1	Y_2	Y_1
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

y_2	y_1	z
0	0	
0	1	
1	0	
1	1	

[Figure 6.6 from the textbook]

Present state $y_2 y_1$	Next state $w = 0$ $Y_2 Y_1$	Next state $w = 1$ $Y_2 Y_1$	Output z
00	00	01	0
01	00	10	0
10	00	10	1
11	*dd*	*dd*	*d*

$Q(t) = y_2 y_1$ and $Q(t+1) = Y_2 Y_1$

w	y_2	y_1	Y_2	Y_1
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

y_2	y_1	z
0	0	0
0	1	0
1	0	1
1	1	d

[Figure 6.6 from the textbook]

Present state	Next state		Output
	$w = 0$	$w = 1$	z
$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	
00	00	01	0
01	00	10	0
10	00	10	1
11	*dd*	*dd*	*d*

[Figure 6.6 from the textbook]

$Q(t) = y_2 y_1$ and $Q(t+1) = Y_2 Y_1$

w	y_2	y_1	Y_2	Y_1
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

y_2	y_1	z
0	0	0
0	1	0
1	0	1
1	1	d

$Q(t) = y_2y_1$ and $Q(t+1) = Y_2Y_1$

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

w	y_2	y_1	Y_2	Y_1
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	d	
1	0	0		
1	0	1		
1	1	0		
1	1	1		

y_2	y_1	z
0	0	0
0	1	0
1	0	1
1	1	d

[Figure 6.6 from the textbook]

Present state	Next state		Output
	$w = 0$	$w = 1$	z
$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	
00	00	01	0
01	00	10	0
10	00	10	1
11	*dd*	*dd*	*d*

[Figure 6.6 from the textbook]

$Q(t) = y_2 y_1$ and $Q(t+1) = Y_2 Y_1$

w	y_2	y_1	Y_2	Y_1
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	d	
1	0	0	0	
1	0	1	1	
1	1	0	1	
1	1	1	d	

y_2	y_1	z
0	0	0
0	1	0
1	0	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
00	00	01	0
01	00	10	0
10	00	10	1
11	*dd*	*dd*	*d*

$Q(t) = y_2y_1$ and $Q(t+1) = Y_2Y_1$

w	y_2	y_1	Y_2	Y_1
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	d	
1	0	0	0	
1	0	1	1	
1	1	0	1	
1	1	1	d	

y_2	y_1	z
0	0	0
0	1	0
1	0	1
1	1	d

[Figure 6.6 from the textbook]

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

[Figure 6.6 from the textbook]

$Q(t) = y_2y_1$ and $Q(t+1) = Y_2Y_1$

w	y_2	y_1	Y_2	Y_1
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	d	d
1	0	0	0	
1	0	1	1	
1	1	0	1	
1	1	1	d	

y_2	y_1	z
0	0	0
0	1	0
1	0	1
1	1	d

Present state	Next state		Output
	$w = 0$	$w = 1$	
$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	z
00	00	01	0
01	00	10	0
10	00	10	1
11	*dd*	*dd*	*d*

[Figure 6.6 from the textbook]

$Q(t) = y_2 y_1$ and $Q(t+1) = Y_2 Y_1$

w	y_2	y_1	Y_2	Y_1
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	d	d
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	d	d

y_2	y_1	z
0	0	0
0	1	0
1	0	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
00	00	01	0
01	00	10	0
10	00	10	1
11	*dd*	*dd*	*d*

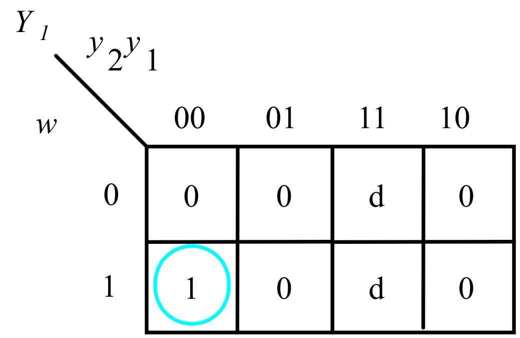
$Q(t) = y_2y_1$ and $Q(t+1) = Y_2Y_1$

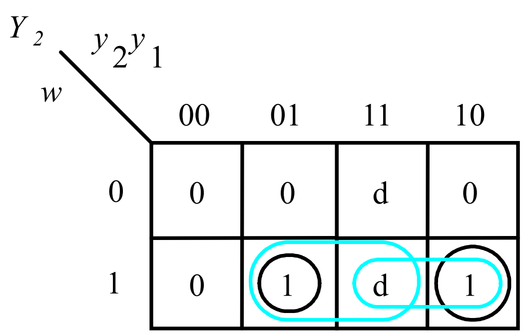
w	y_2	y_1	Y_2	Y_1
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	d	d
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	d	d

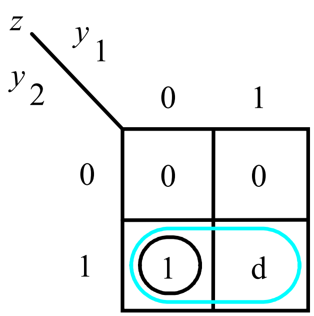
y_2	y_1	z
0	0	0
0	1	0
1	0	1
1	1	d

[Figure 6.6 from the textbook]

Note that the textbook draws these K-Maps differently from all previous K-maps
(the least significant bits index the columns, instead of the most significant bits).





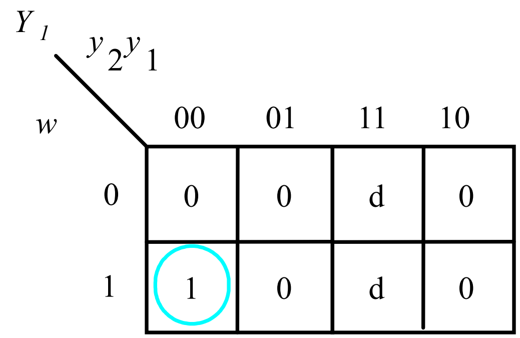


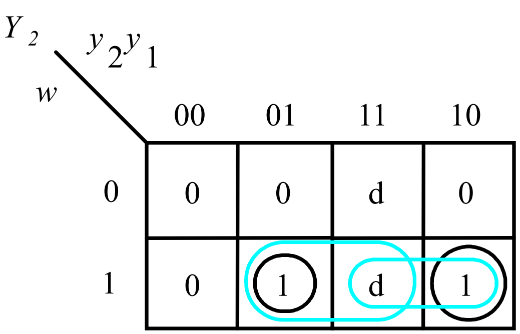
$Q(t) = y_2y_1$ and $Q(t+1) = Y_2Y_1$

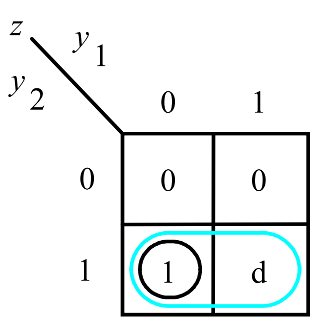
w	y_2	y_1	Y_2	Y_1
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	d	d
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	d	d

y_2	y_1	z
0	0	0
0	1	0
1	0	1
1	1	d

Don’t care conditions simplify the combinatorial logic

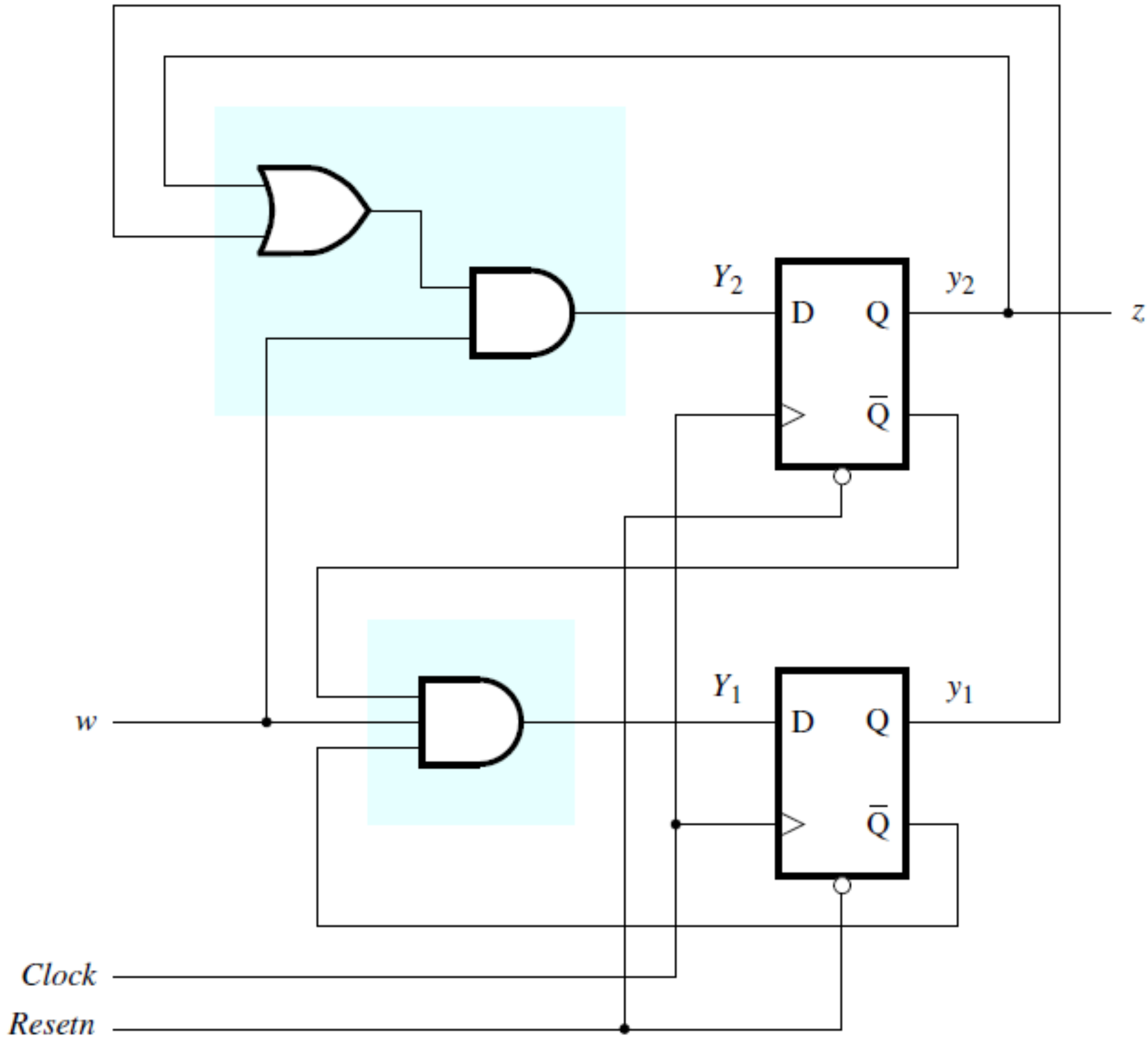




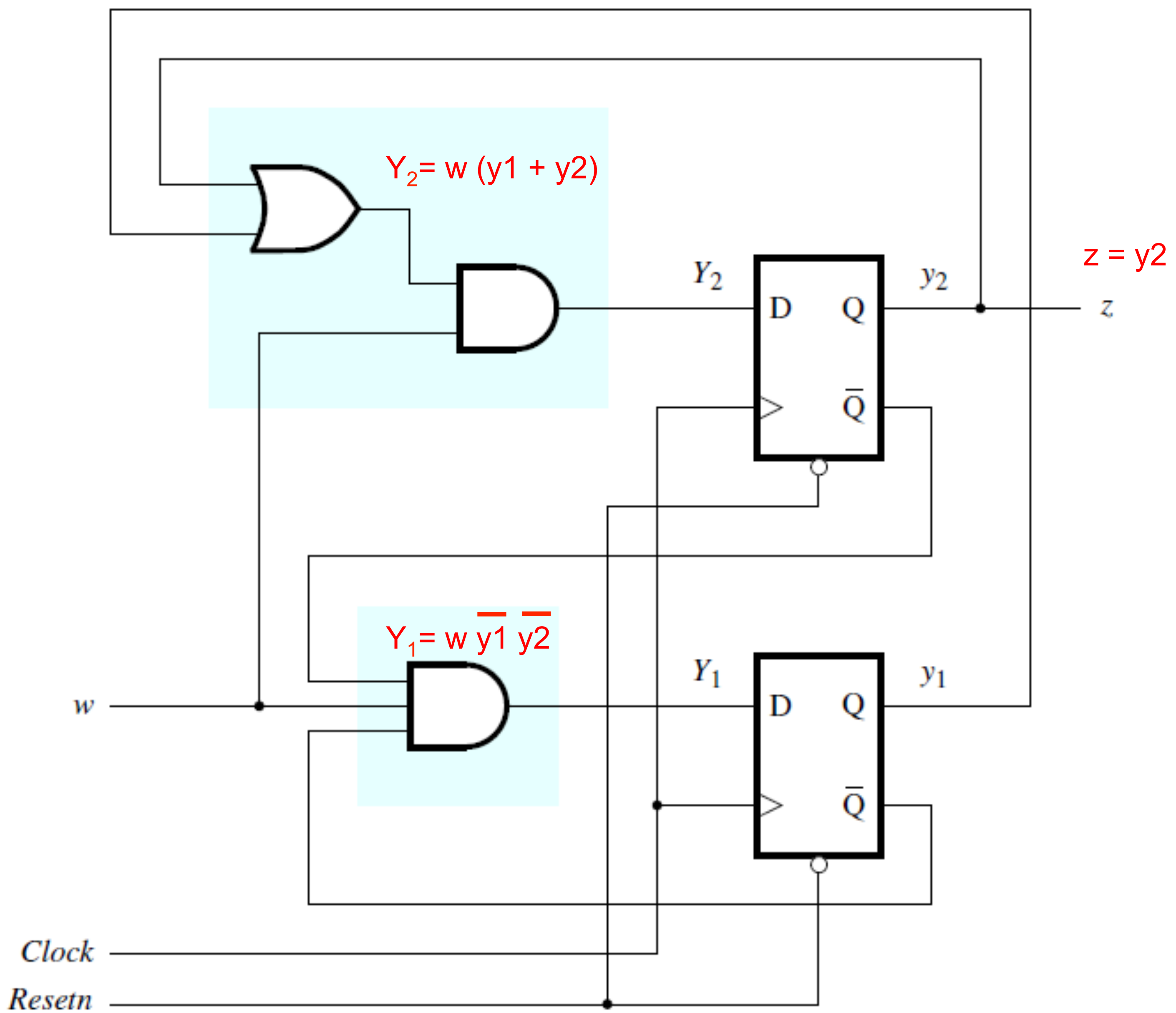


Ignoring don't cares	Using don't cares
$Y_1 = w\bar{y}_1\bar{y}_2$	$Y_1 = w\bar{y}_1\bar{y}_2$
$Y_2 = wy_1\bar{y}_2 + \bar{w}y_1y_2$	$Y_2 = wy_1 + wy_2$ $= w(y_1 + y_2)$
$z = \bar{y}_1y_2$	$z = y_2$

[Figure 6.7 from the textbook]

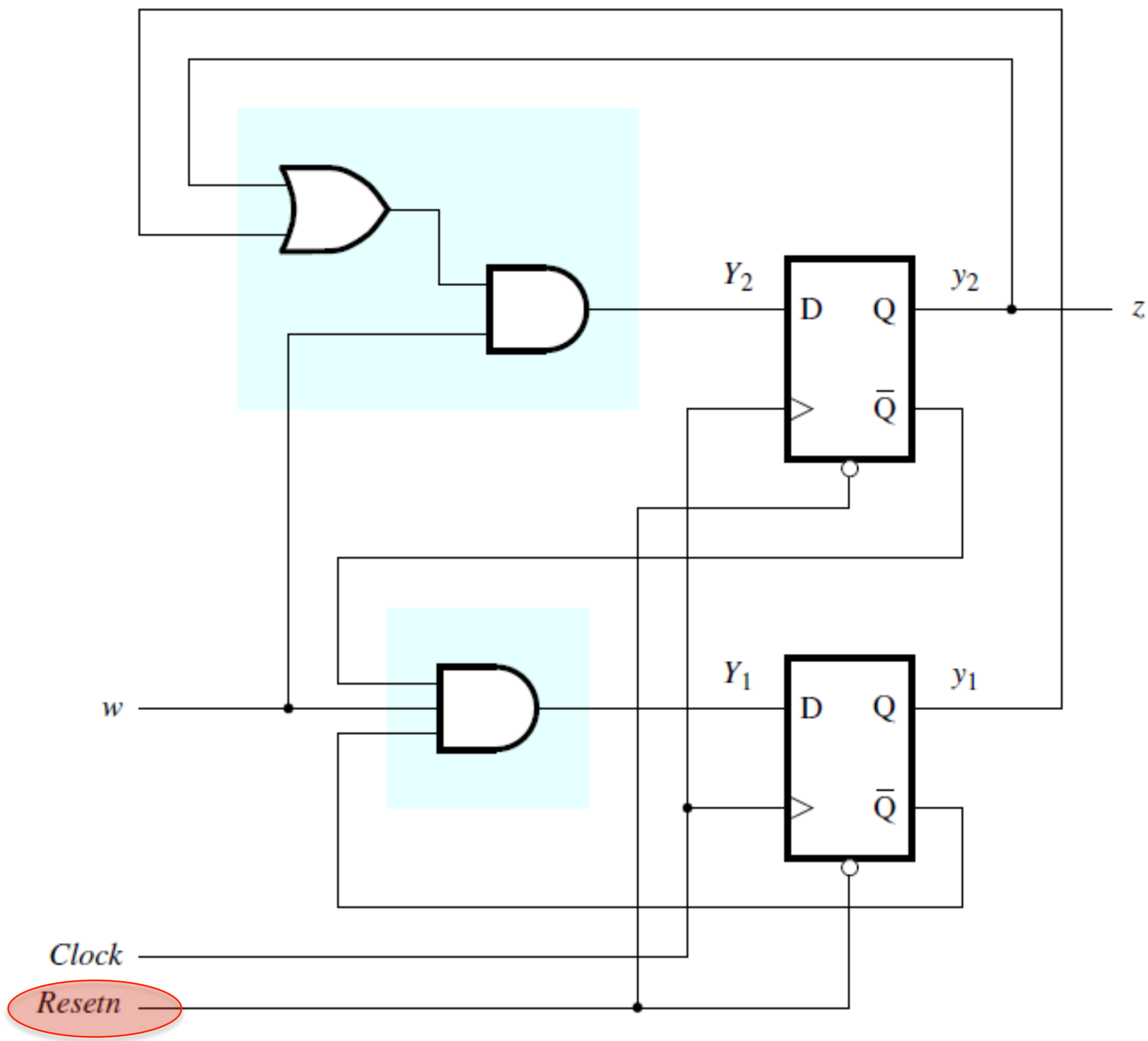


[Figure 6.8 from the textbook]



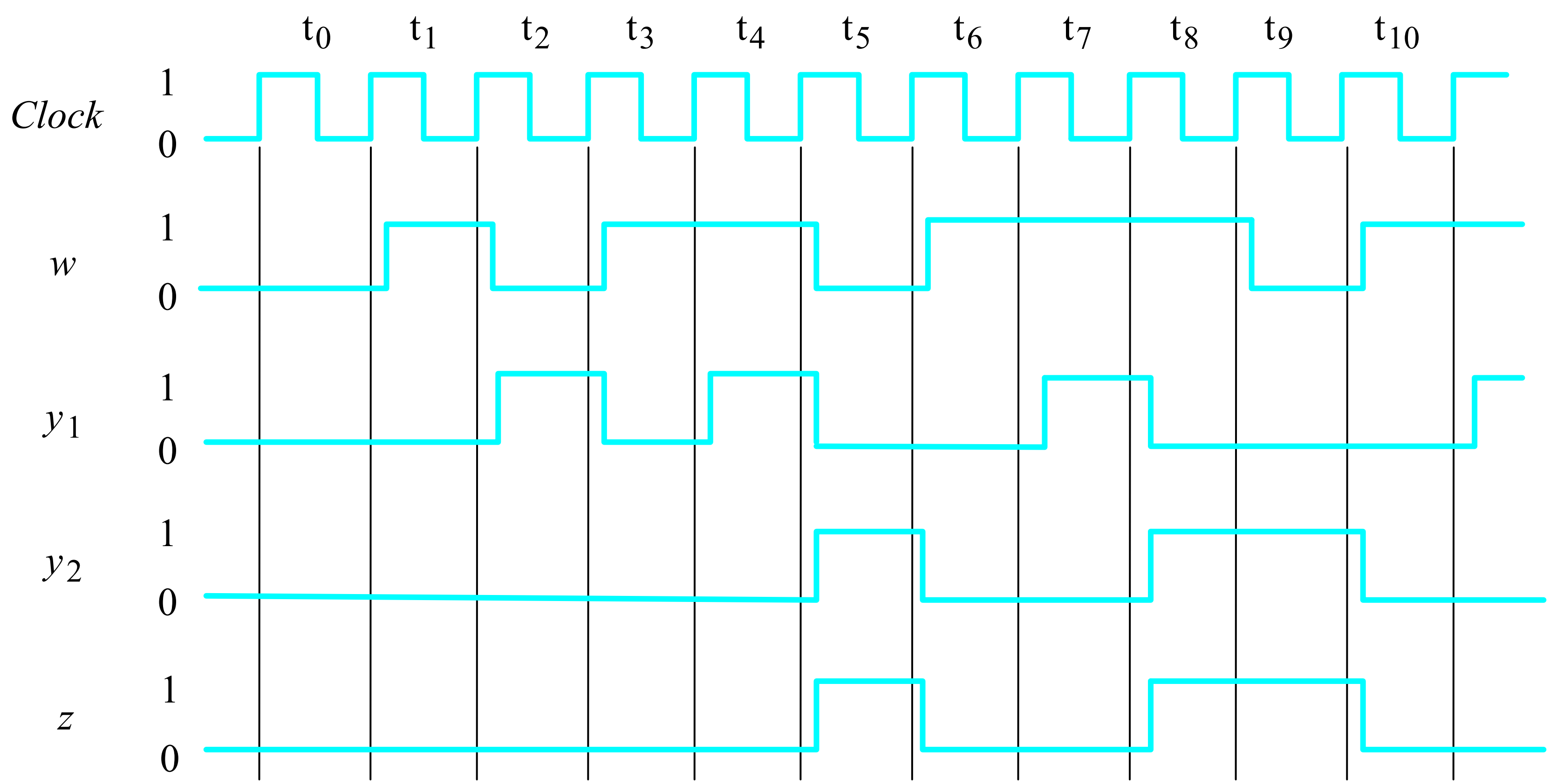
[Figure 6.8 from the textbook]

Lastly, we add a reset signal, which forces the machine back to its start state, which is state *00* in this case.



[Figure 6.8 from the textbook]

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



[Figure 6.9 from the textbook]

Summary: Designing a Moore Machine

- Obtain the circuit specification.
- Derive a state diagram.
- Derive the state table.
- Decide on a state encoding.
- Encode the state table.
- Derive the output logic and next-state logic.
- Add a reset signal.

An Alternative State Encoding For Example #1

A Better State Encoding

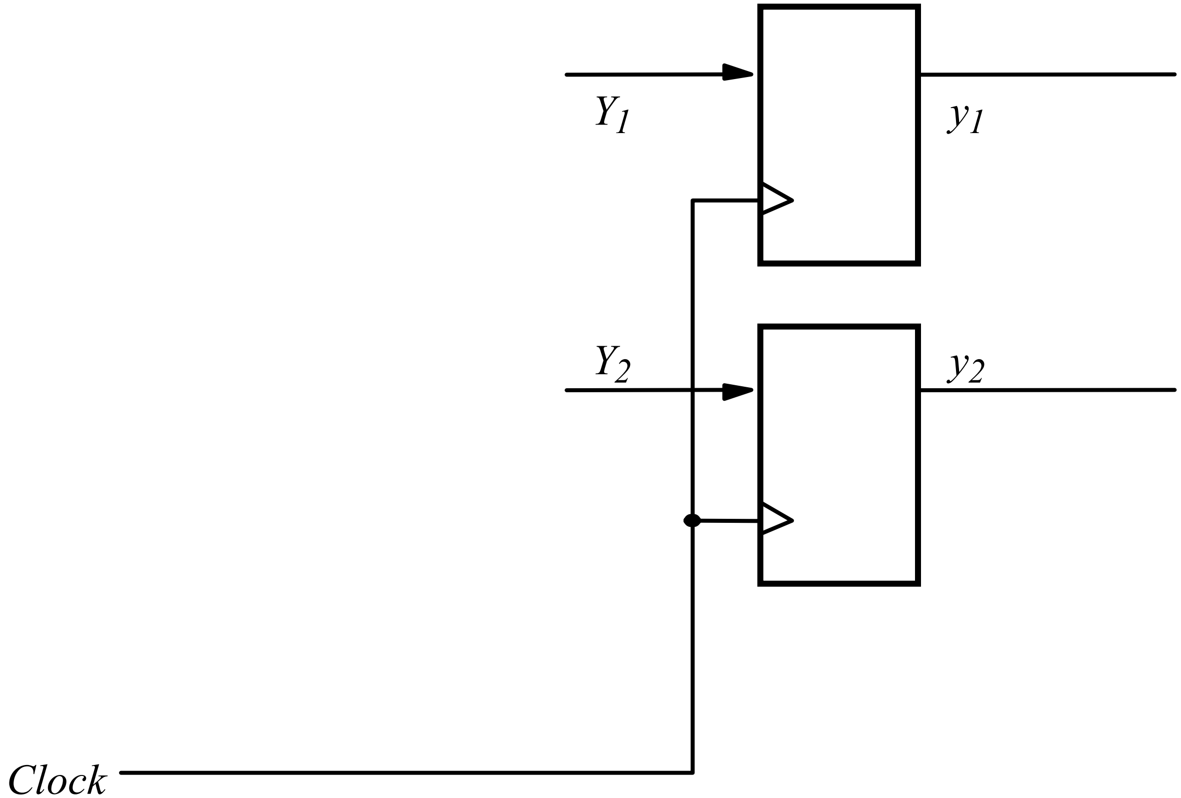
Present state	Next state		Output
	$w = 0$	$w = 1$	z
A	A	B	0
B	A	C	0
C	A	C	1

Suppose we encoded our states another way:

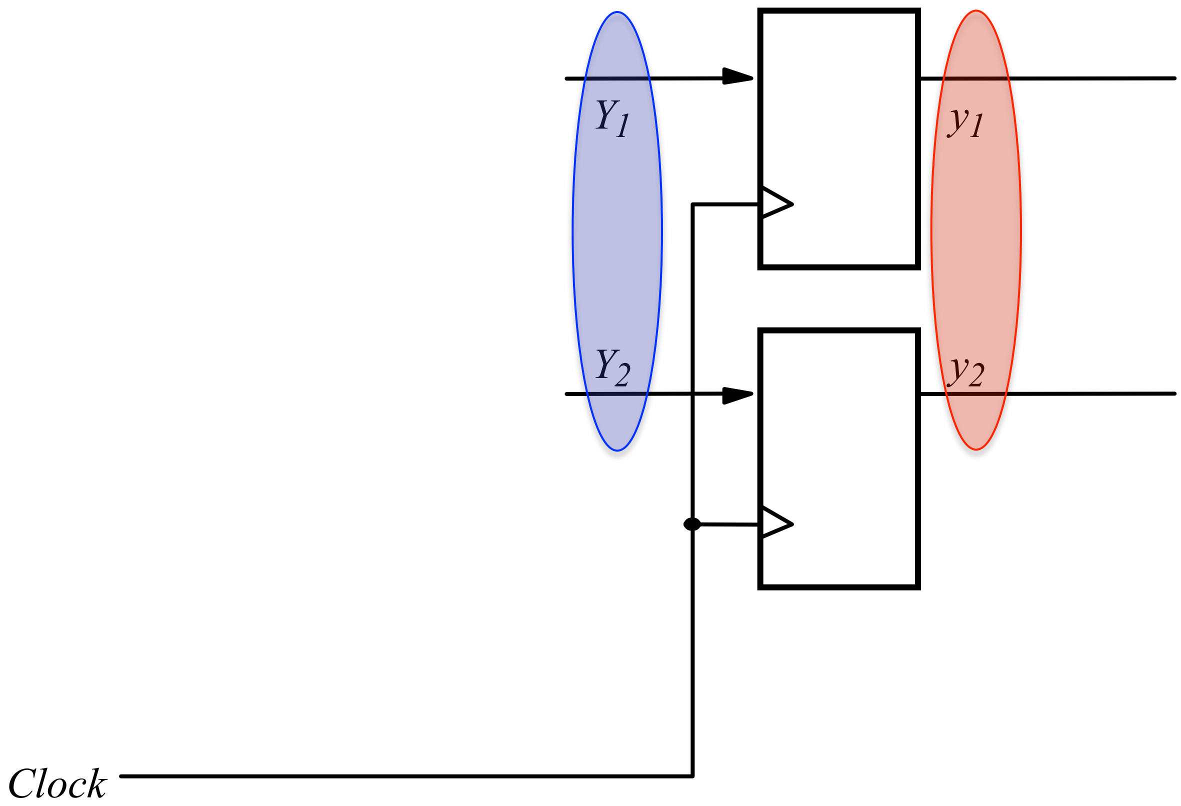
A ~ 00

B ~ 01

C ~ 11



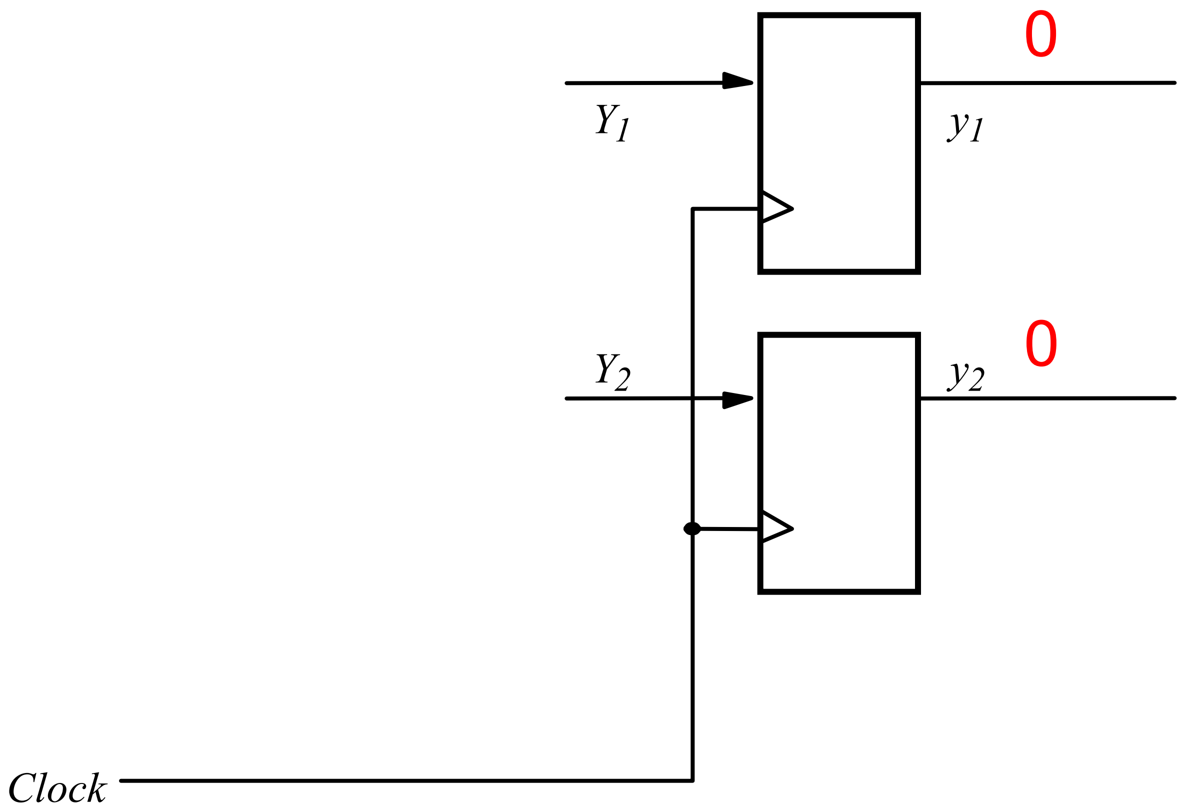
Y_1
y_1
Y_2
y_2
Clock



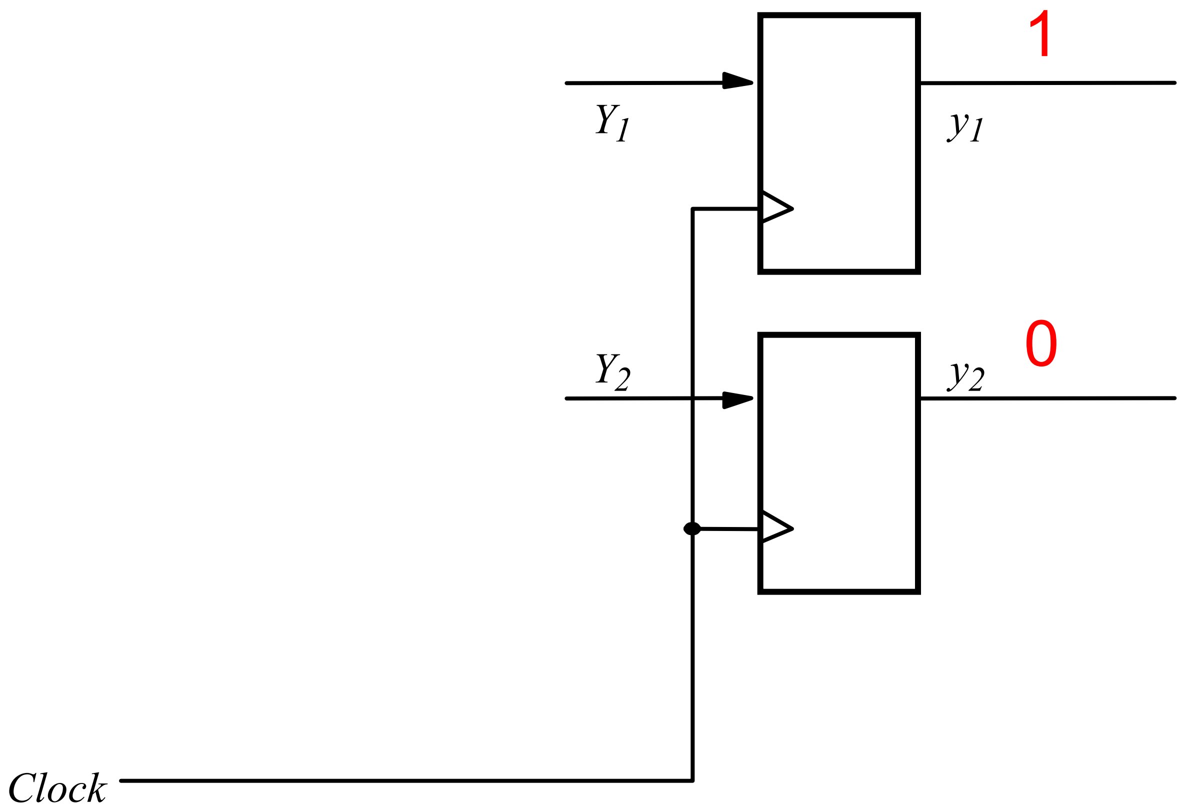
We will call y_1 and y_2 the *present state variables*.

We will call Y_1 and Y_2 the *next state variables*.

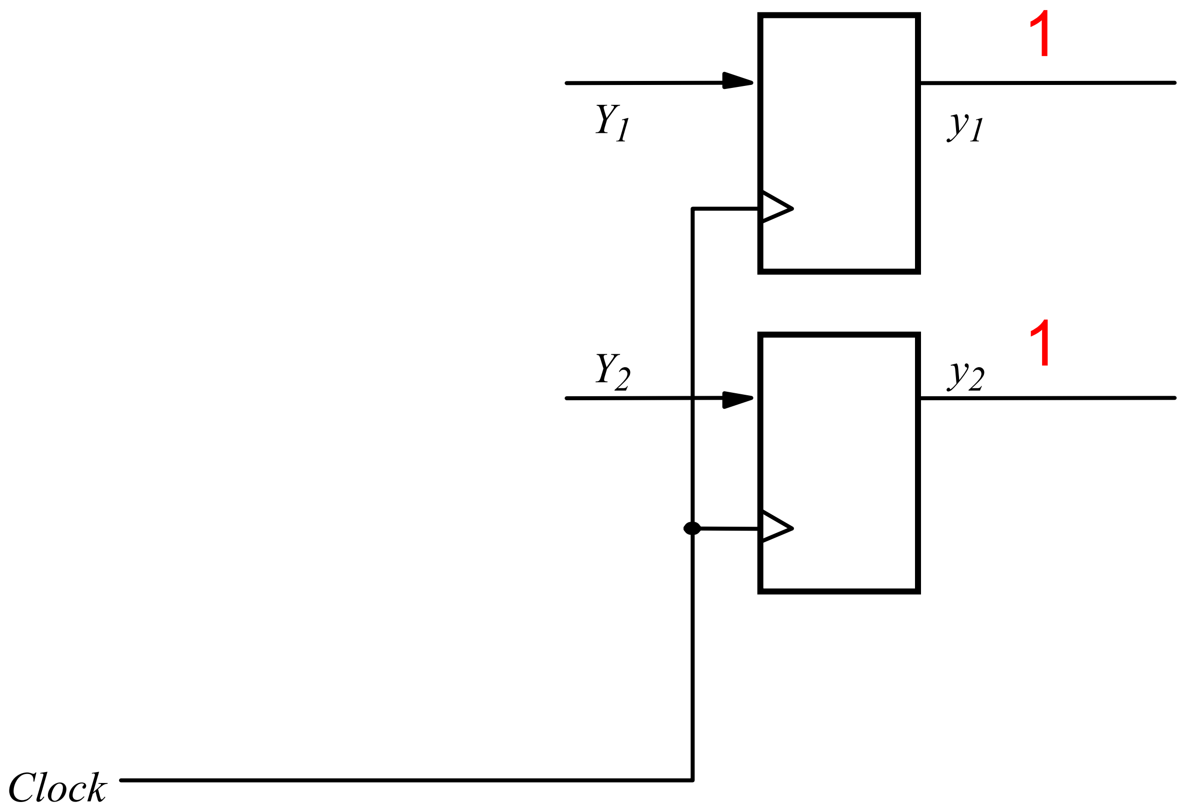
[Figure 6.5 from the textbook]



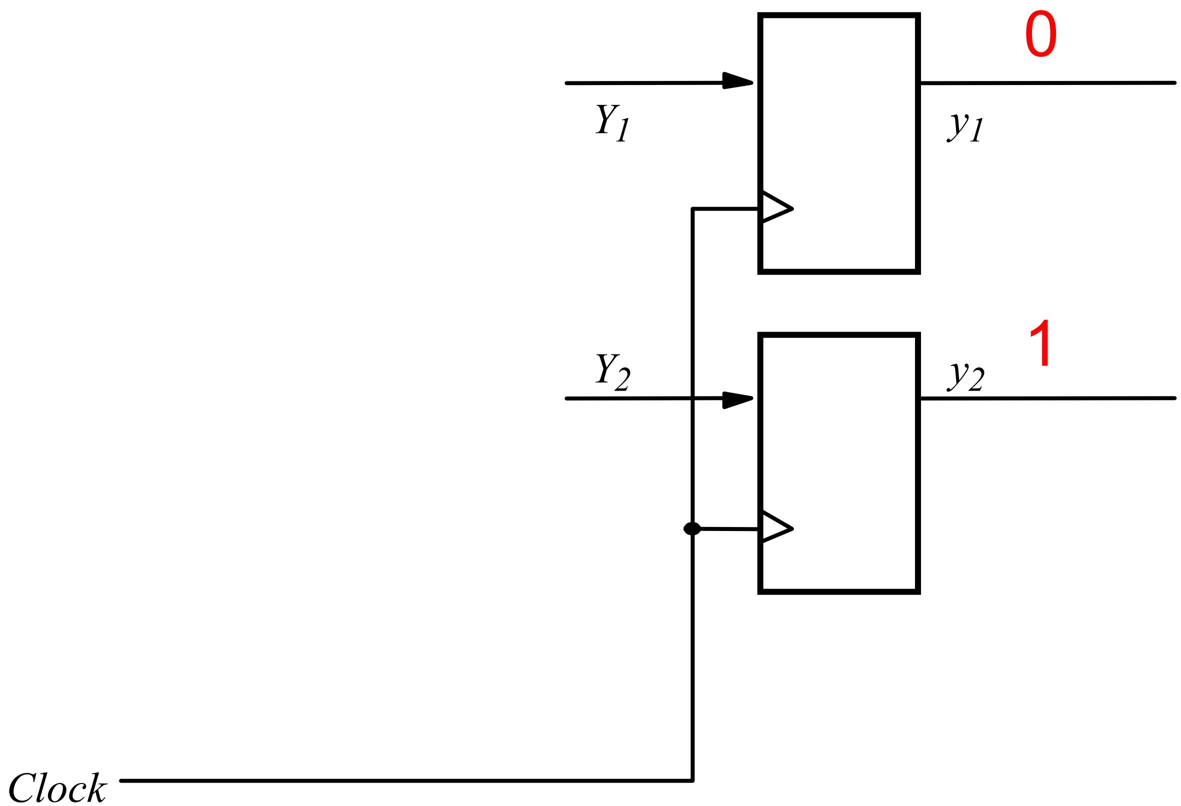
Two zeros on the output JOINTLY represent state A.



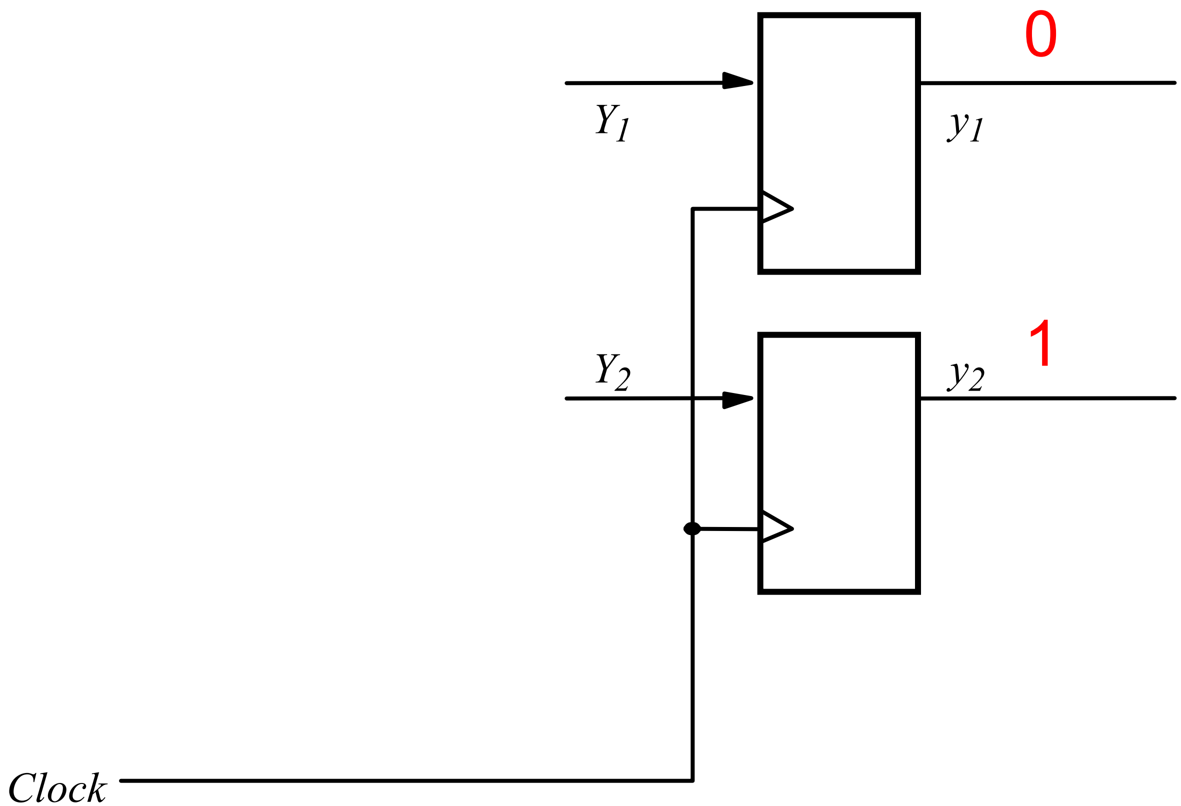
This flip-flop output pattern represents state B.



This flip-flop output pattern represents state C.



What does this flip-flop output pattern represent?



This would be state D, but we don't have one in this example. So this is an impossible state.

A Better State Encoding

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

Suppose we encoded our states another way:

A ~ 00

B ~ 01

C ~ 11

A Better State Encoding

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

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

A ~ 00

B ~ 01

C ~ 11

A Better State Encoding

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

	Present state $y_2 y_1$	Next state $w = 0$ $Y_2 Y_1$	Next state $w = 1$ $Y_2 Y_1$	Output z
A	00	00	01	0
B	01	00	11	0
C	11	00	11	1
	10	*dd*	*dd*	*d*

Let's Derive the Logic Expressions

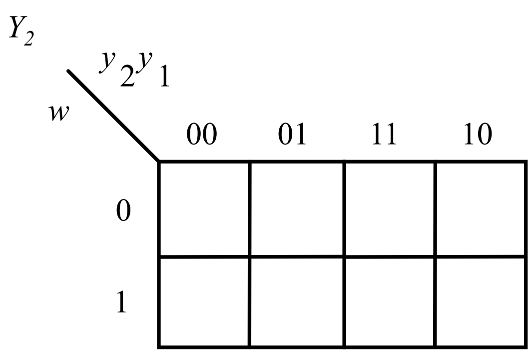
	Present state	Next state		Output
		$w = 0$	$w = 1$	
	$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	z
A	00	00	01	0
B	01	00	11	0
C	11	00	11	1
	10	*dd*	*dd*	*d*

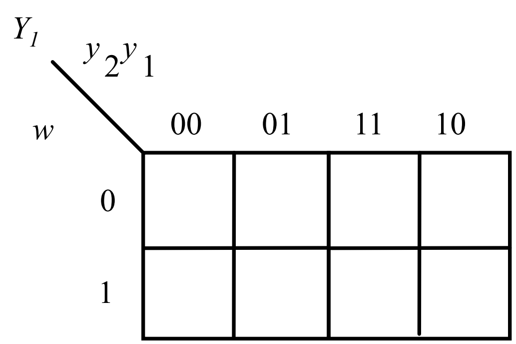
[Figure 6.16 from the textbook]

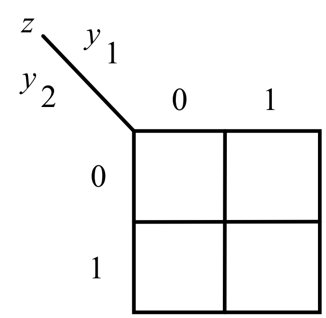
Let's Derive the Logic Expressions

Warning:
This table does not enumerate y_2y_1, in the standard way, so be careful when filling out the K-Map.

	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	11	0
C	11	00	11	1
	10	*dd*	*dd*	*d*



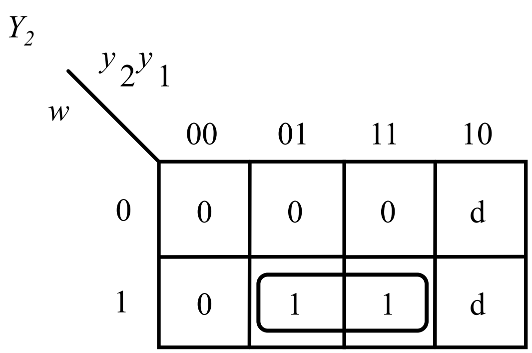




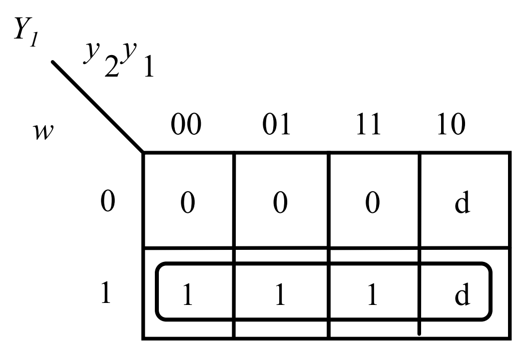
Let's Derive the Logic Expressions

Warning:
This table does not enumerate y_2y_1, in the standard way, so be careful when filling out the K-Map.

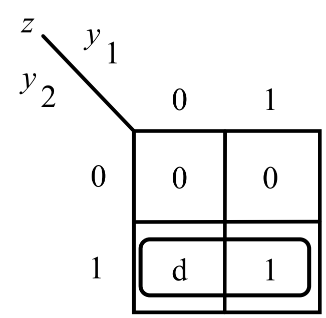
	Present state	Next state		Output
		$w = 0$	$w = 1$	
	y_2y_1	Y_2Y_1	Y_2Y_1	z
A	00	00	01	0
B	01	00	11	0
C	11	00	11	1
	10	*dd*	*dd*	*d*



$$Y_2(w, y_2, y_1) = wy_1$$

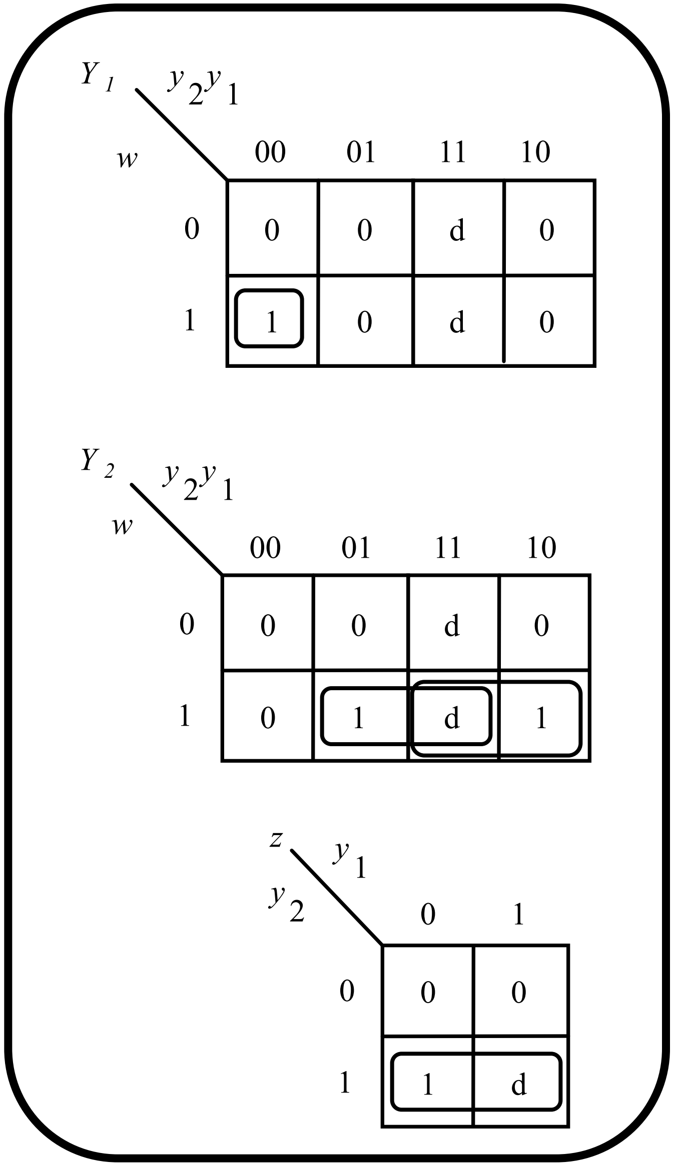


$$Y_1(w, y_2, y_1) = w$$

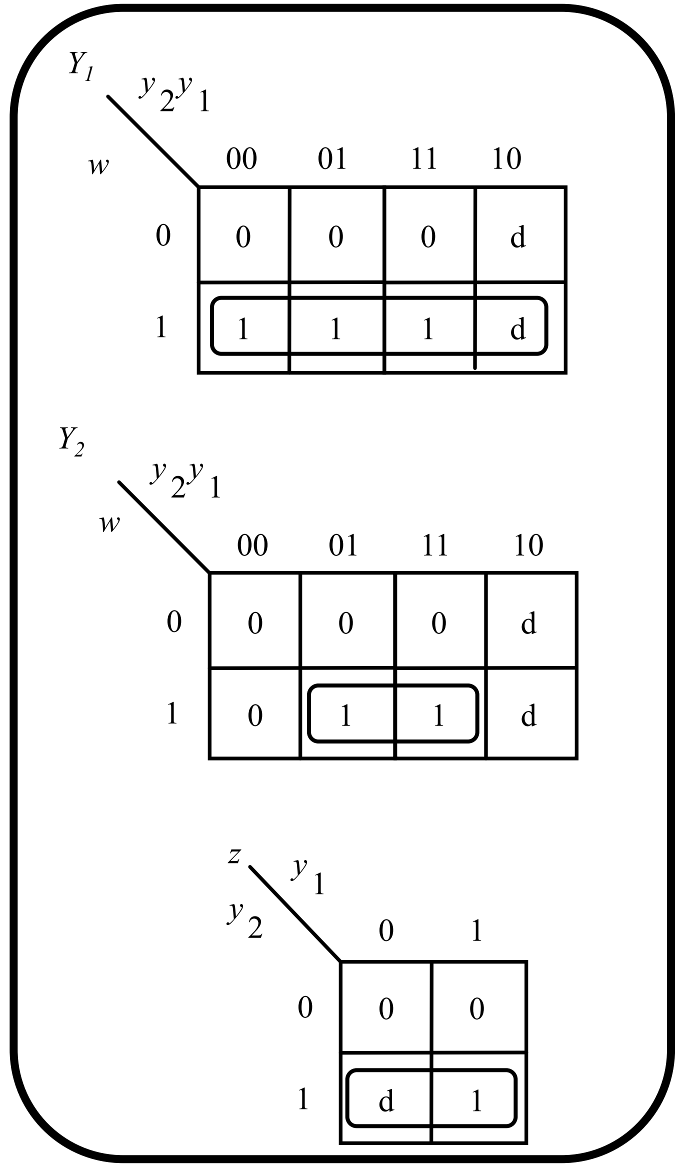


$$z(y_2, y_1) = y_2$$

Original State Encodings



New State Encodings

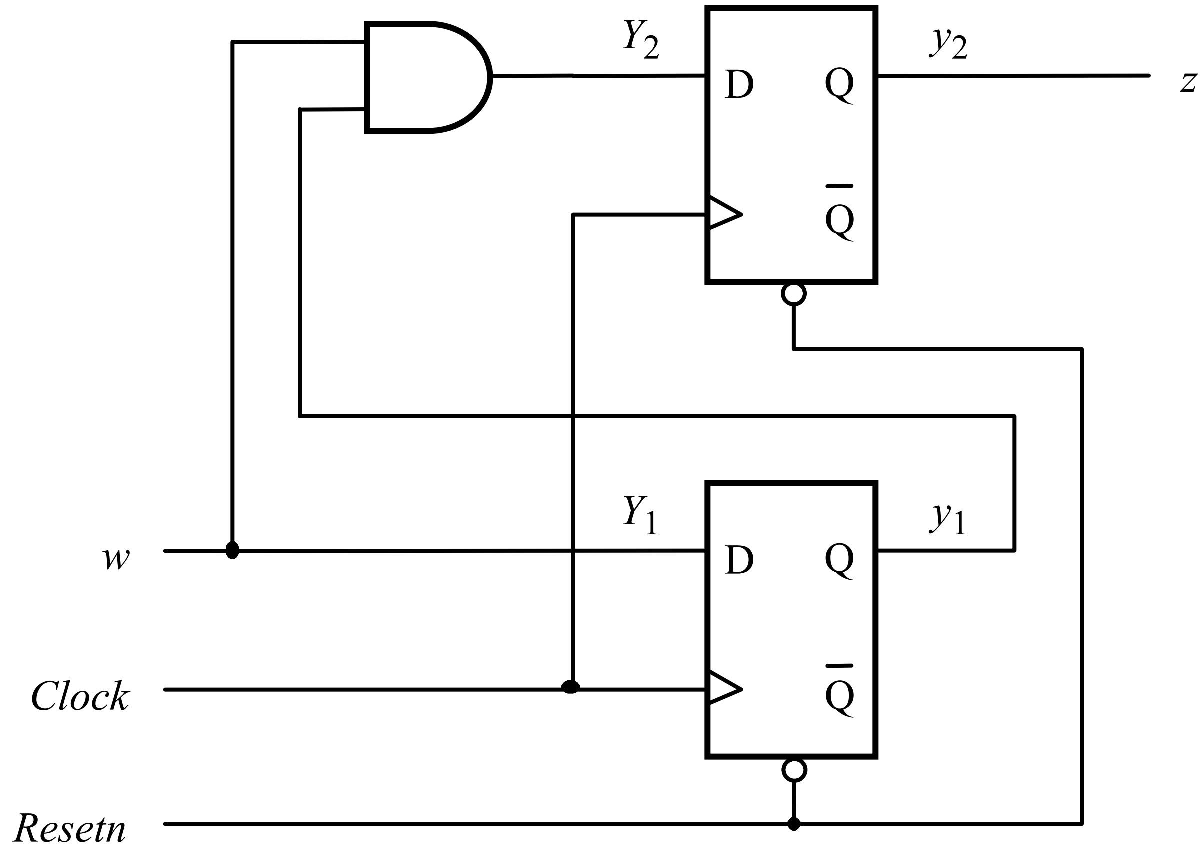


The New and Improved Circuit Diagram

$$Y_1(w, y_2, y_1) = w$$

$$Y_2(w, y_2, y_1) = wy_1$$

$$z(y_2, y_1) = y_2$$



[Figure 6.17 from the textbook]

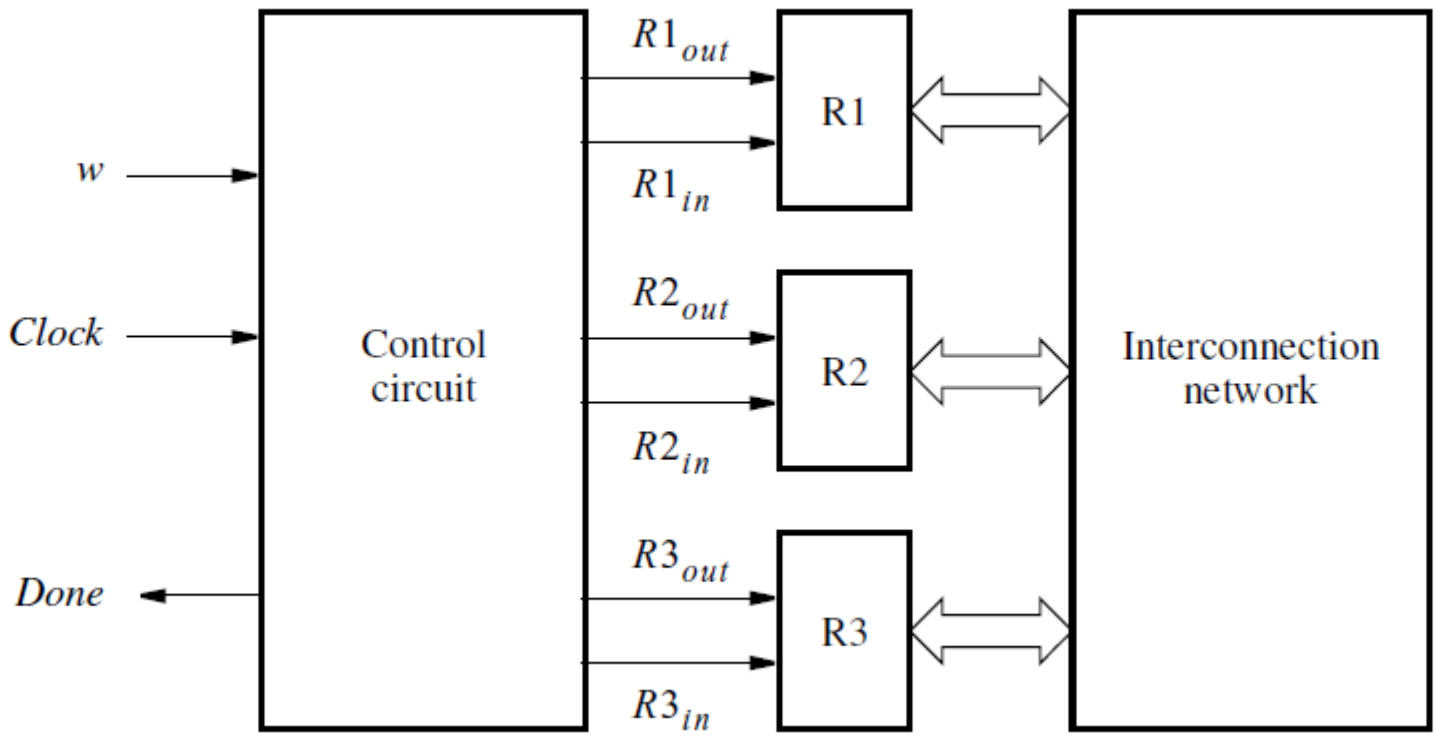
Main Idea

Different state assignments of the same Moore machine generally lead to different circuits.

Some may be better than others.

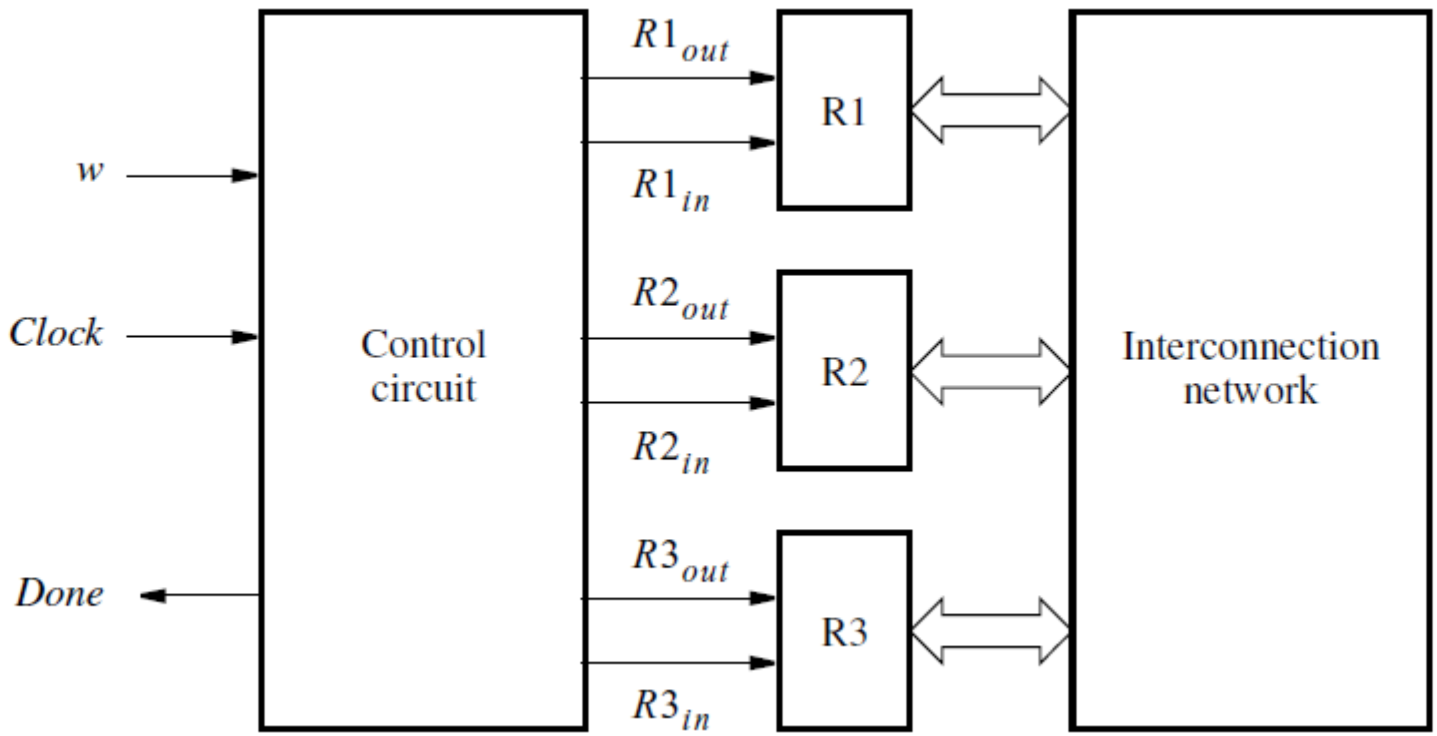
Example #2

Register Swap Controller



[Figure 6.10 from the textbook]

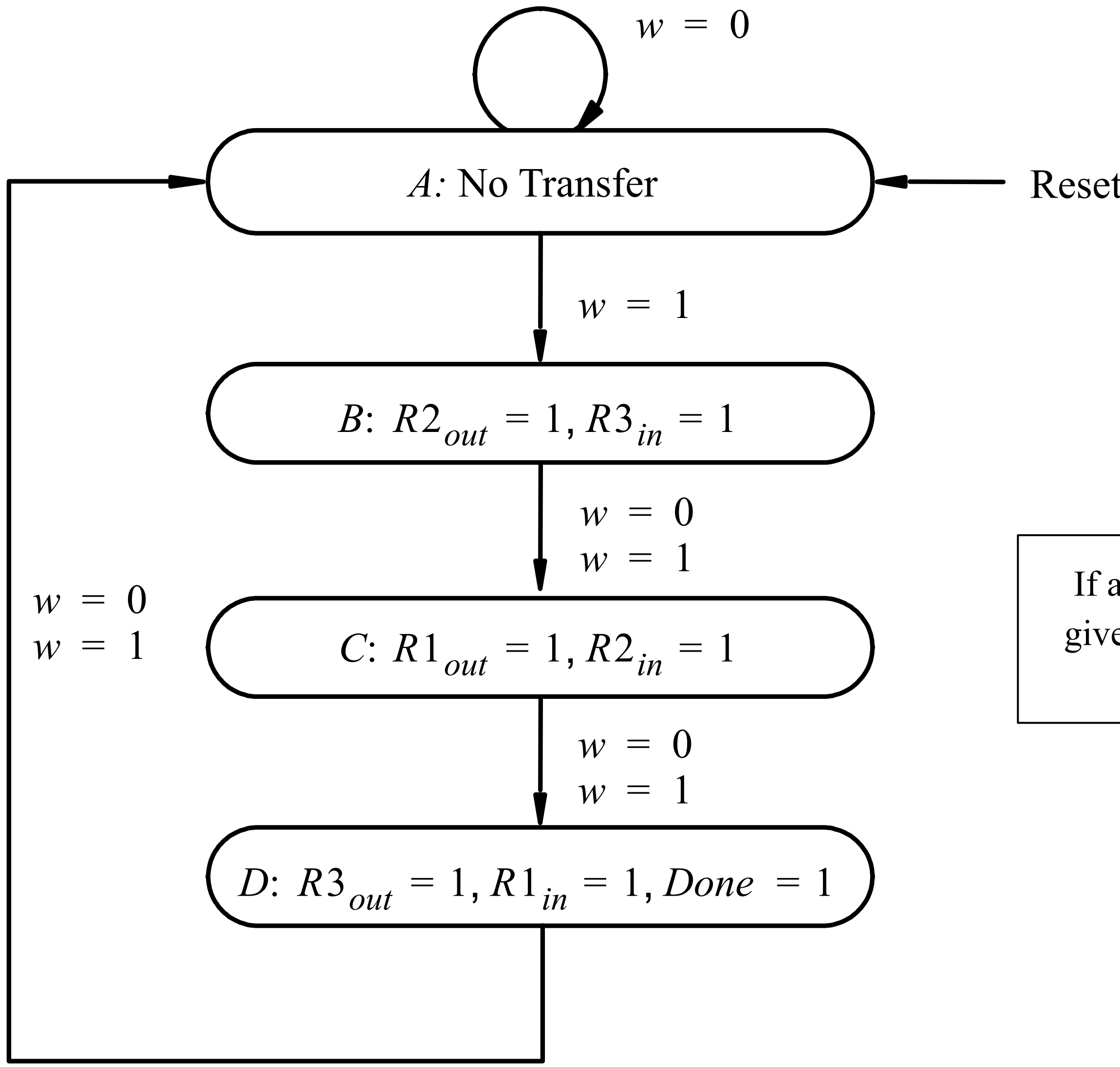
Register Swap Controller



Design a Moore machine control circuit for swapping the contents of registers R1 and R2 by using R3 as a temporary.

[Figure 6.10 from the textbook]

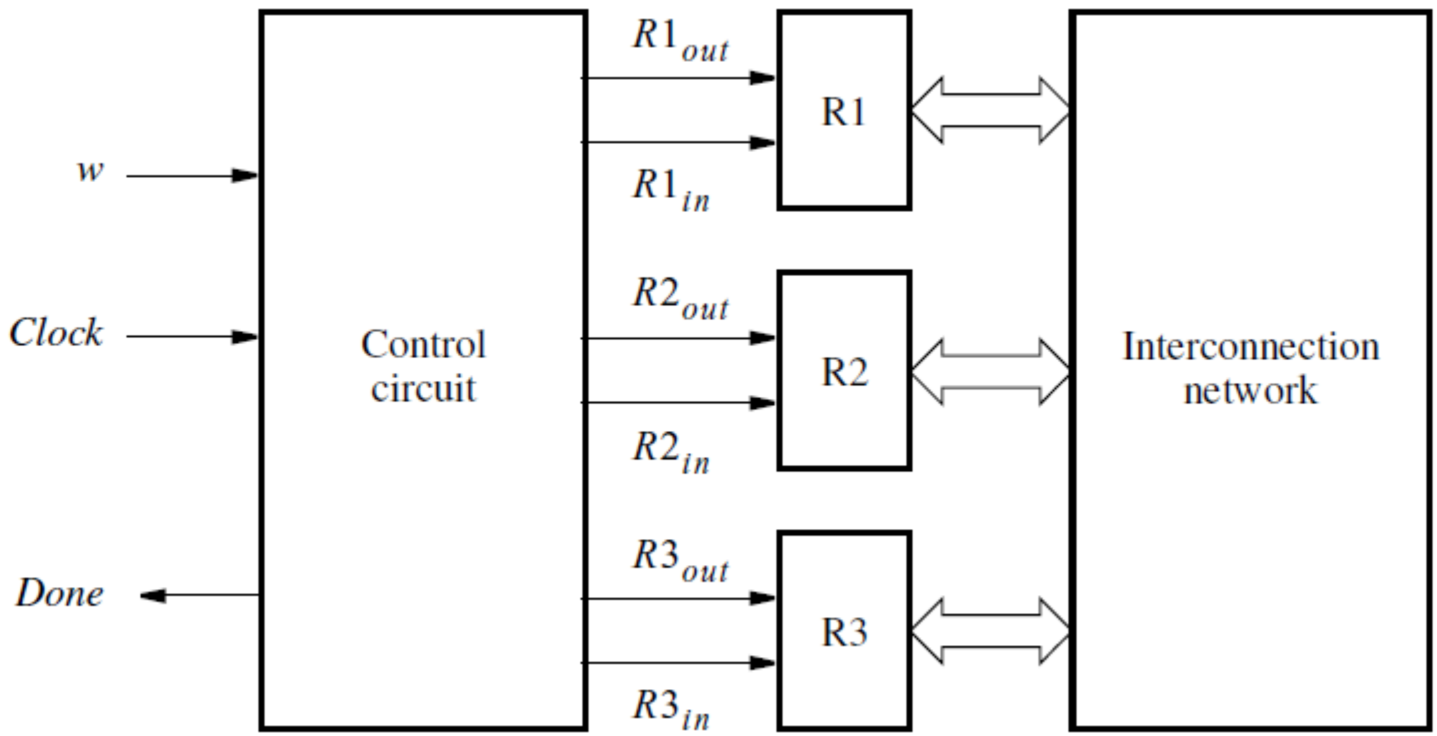
State Diagram



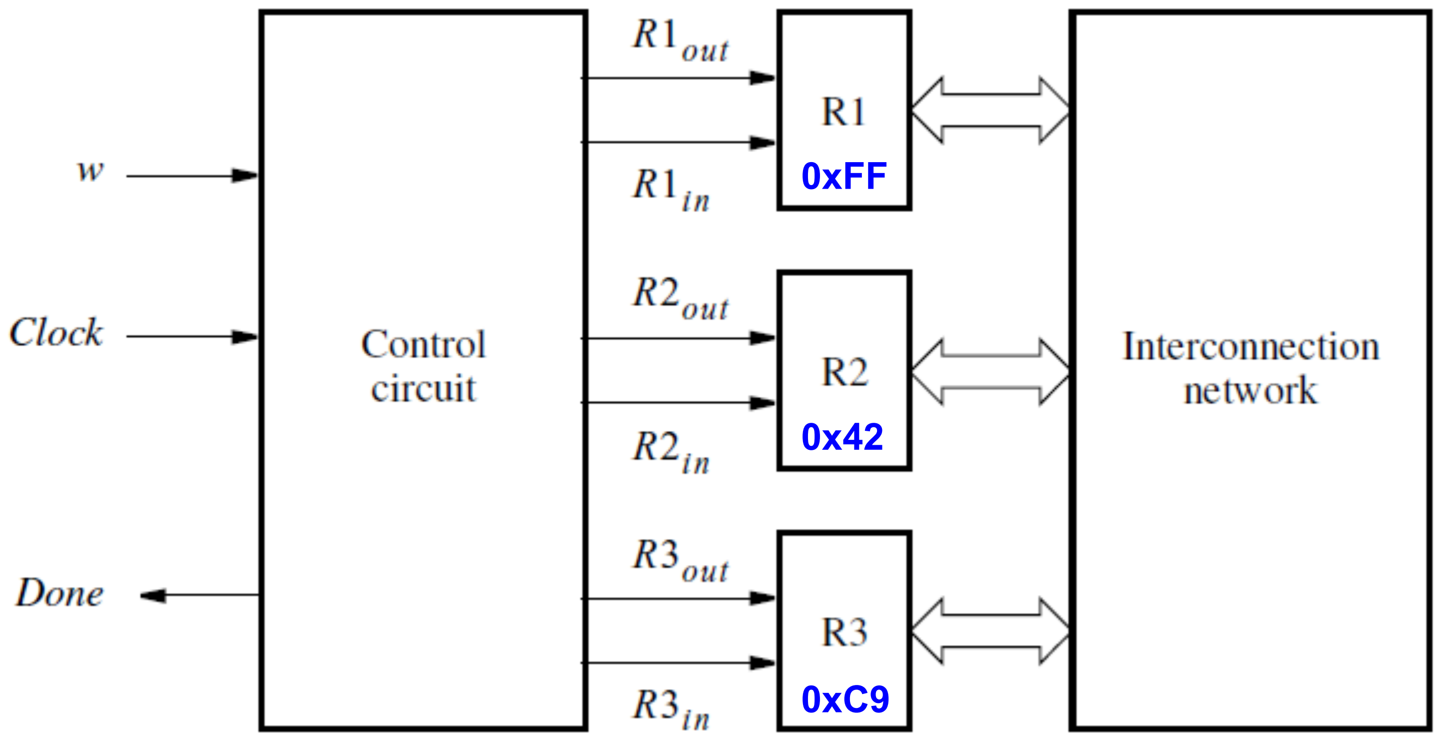
If an output is not given, then assume that it is 0.

[Figure 6.11 from the textbook]

Animated Register Swap

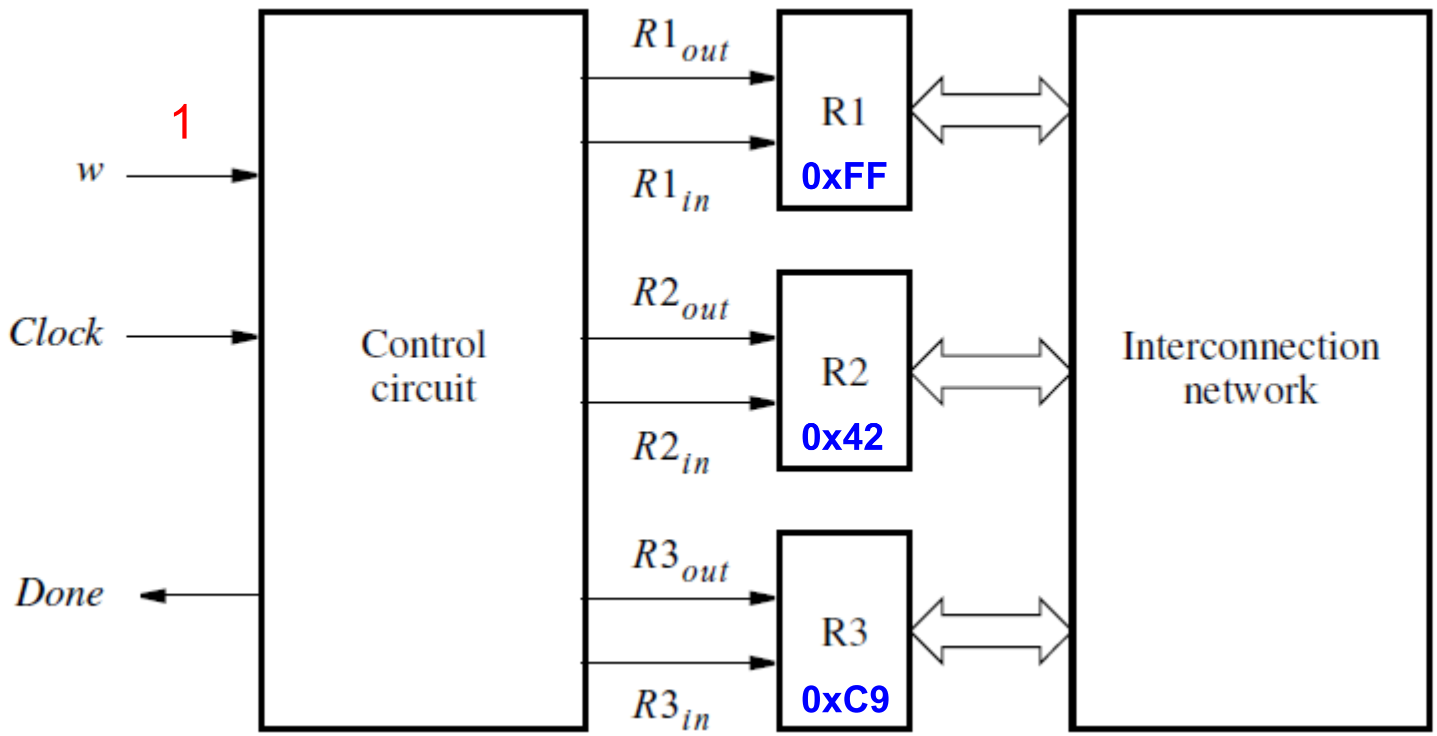


Animated Register Swap



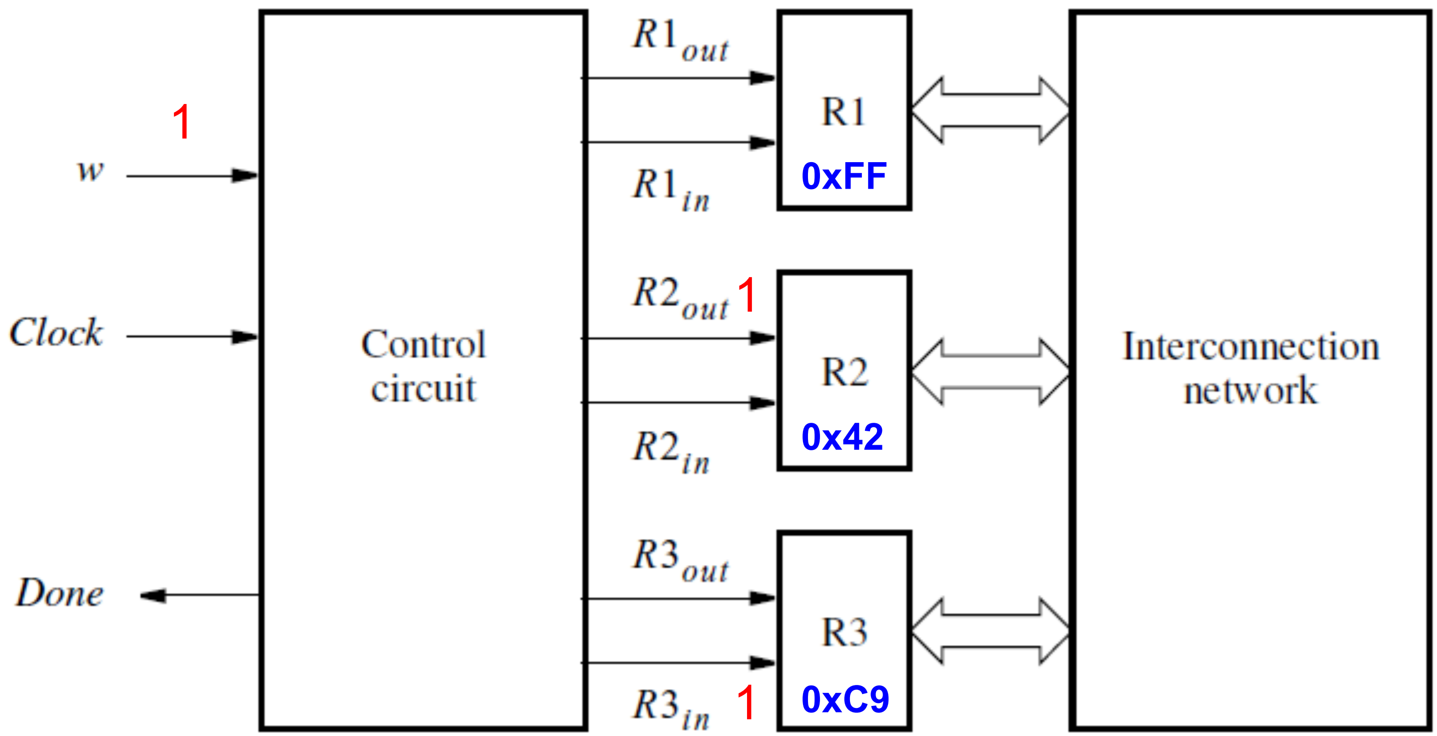
These are the original values of the 8-bit registers

Animated Register Swap

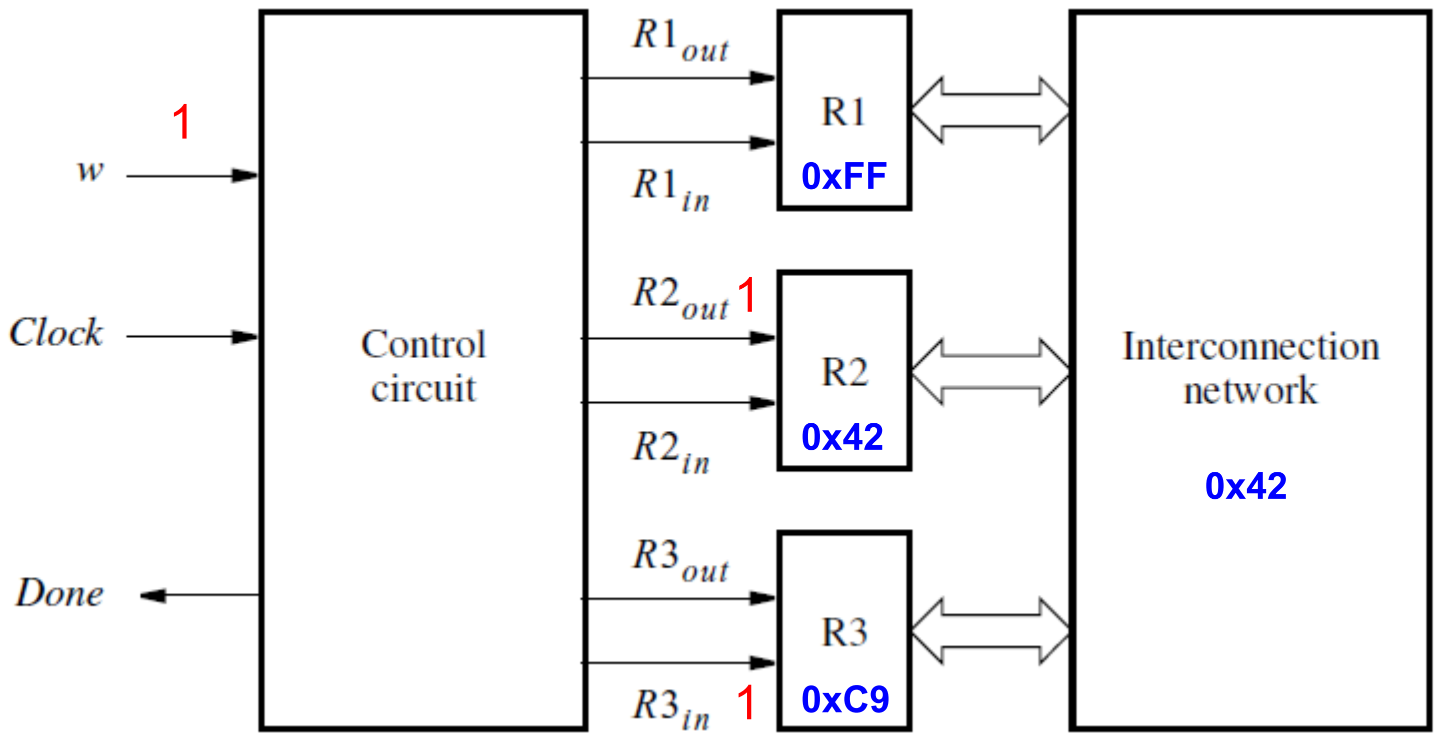


For clarity, only inputs that are equal to 1 will be shown.

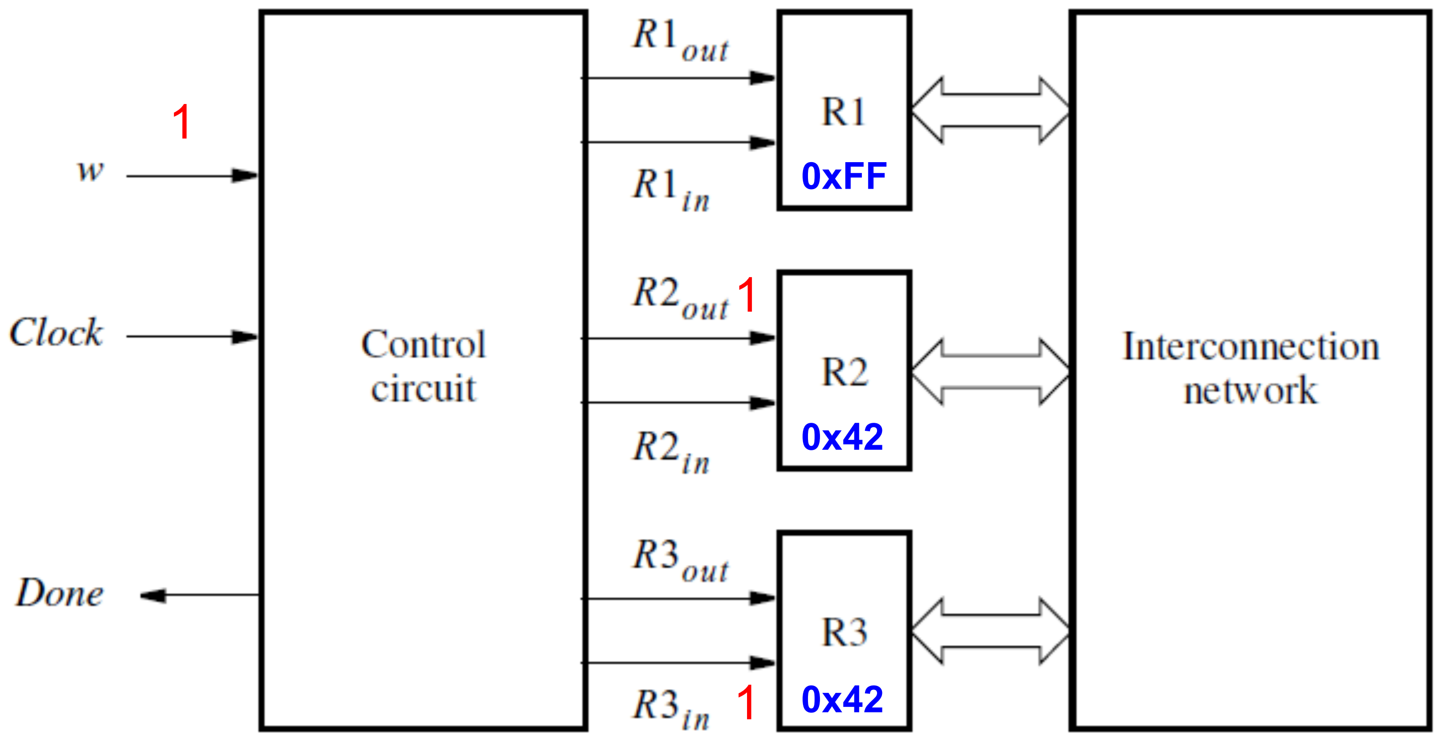
Animated Register Swap



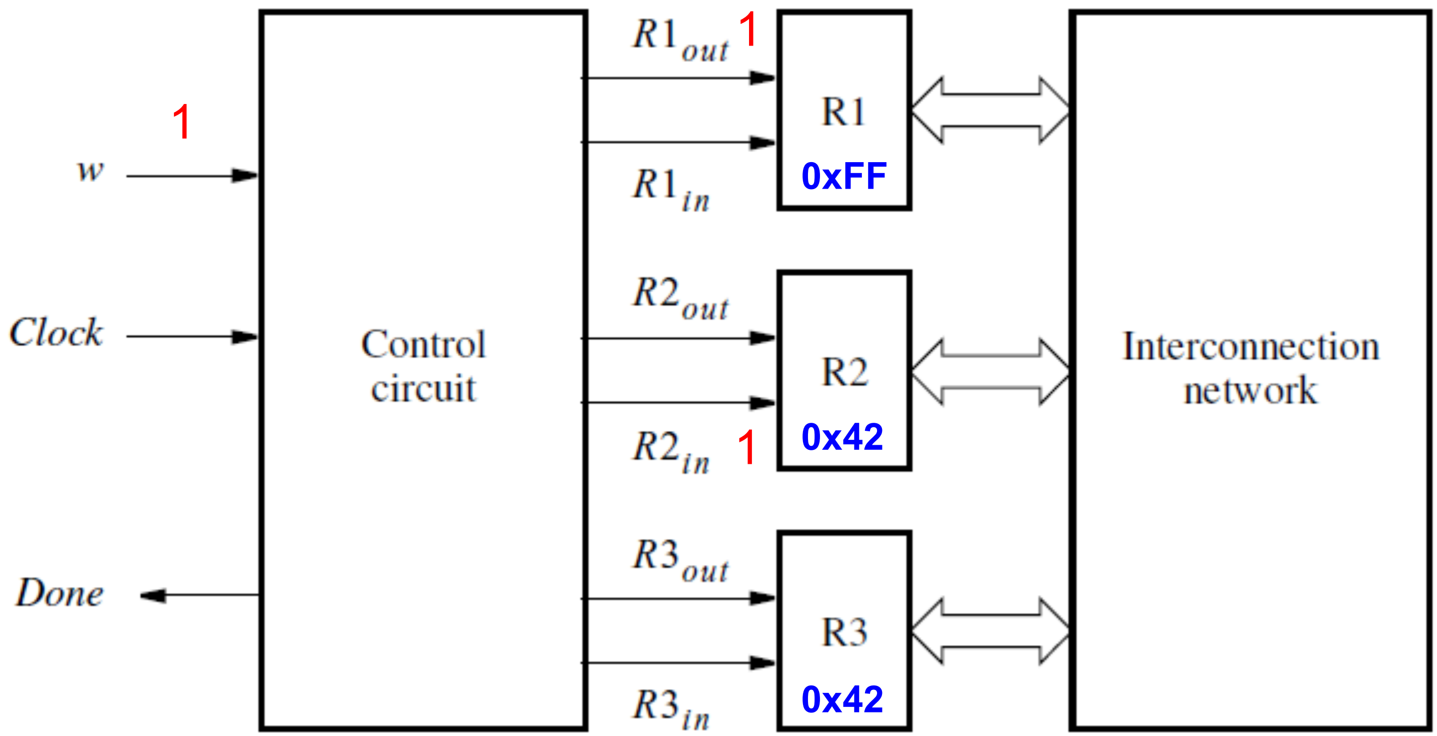
Animated Register Swap



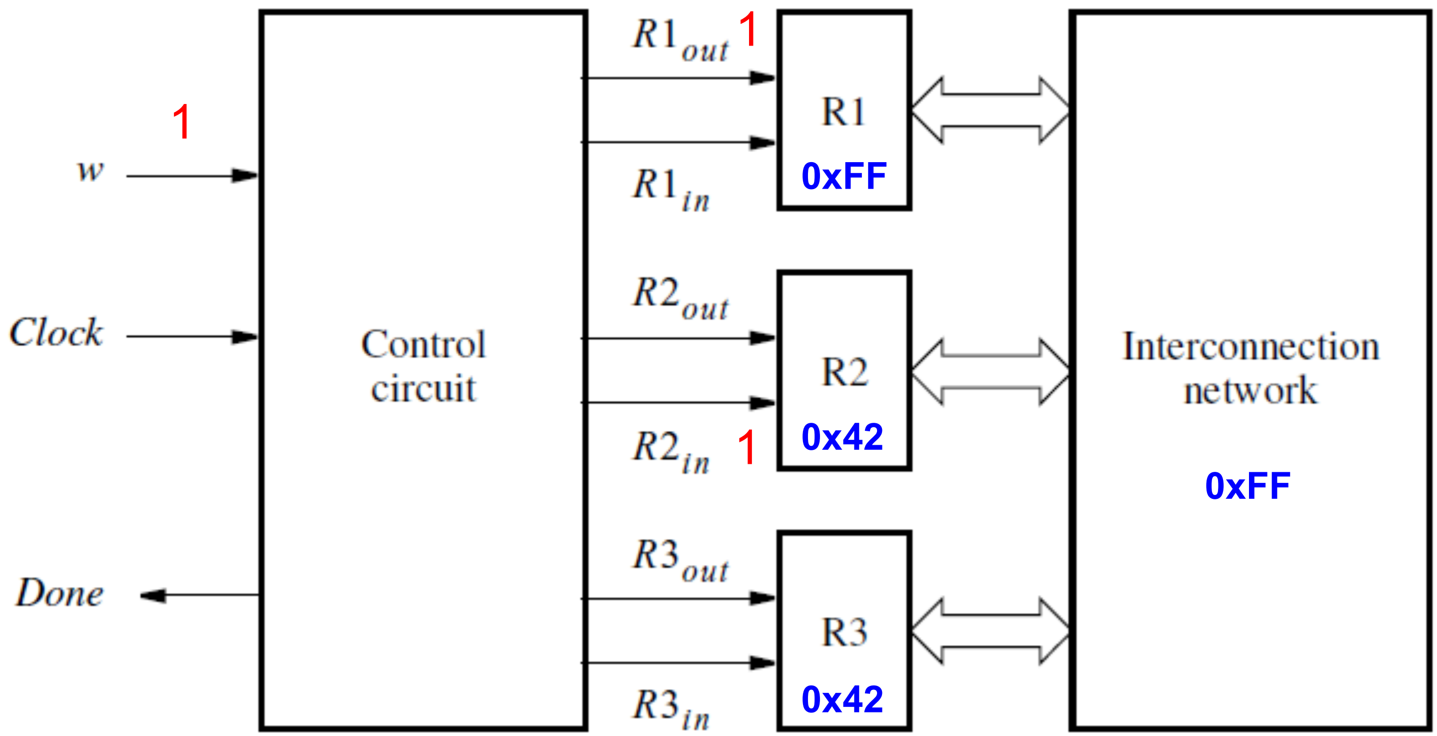
Animated Register Swap



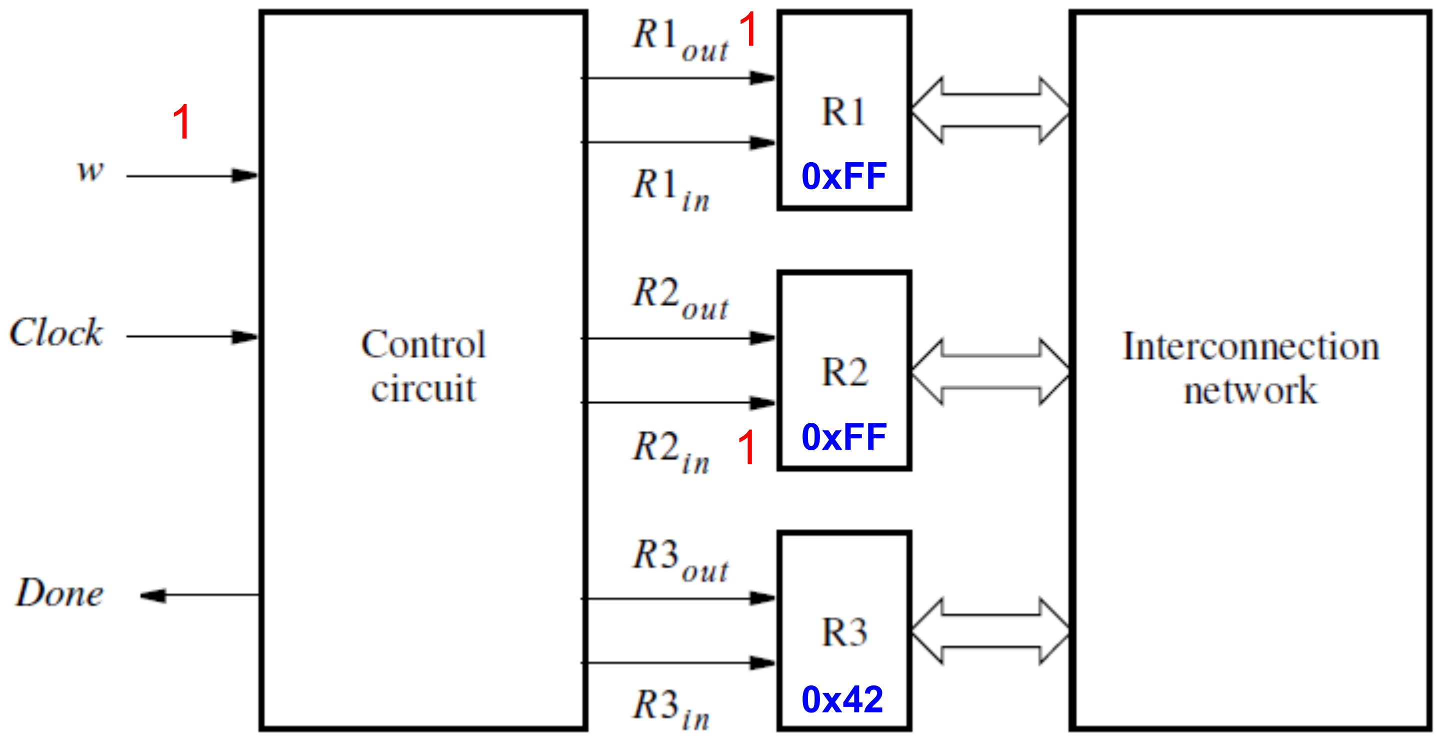
Animated Register Swap



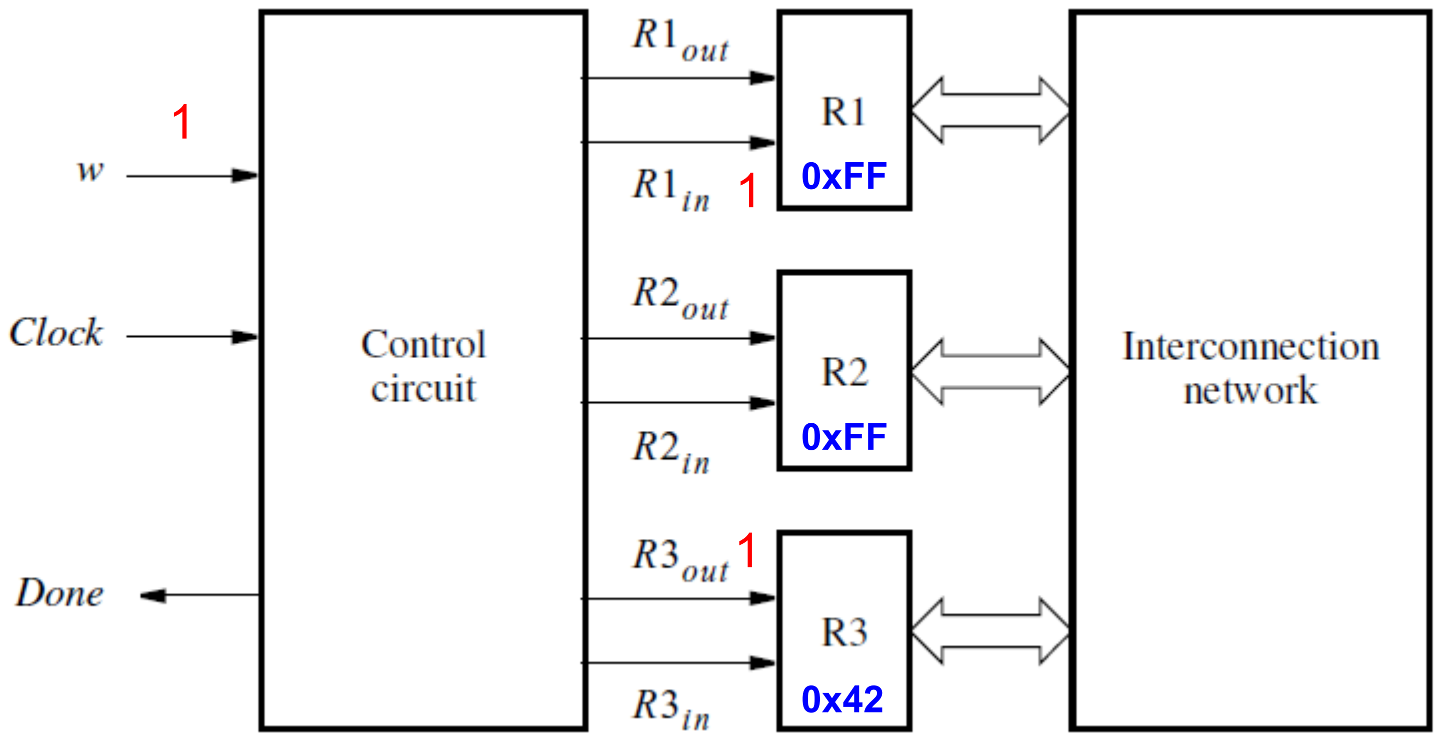
Animated Register Swap



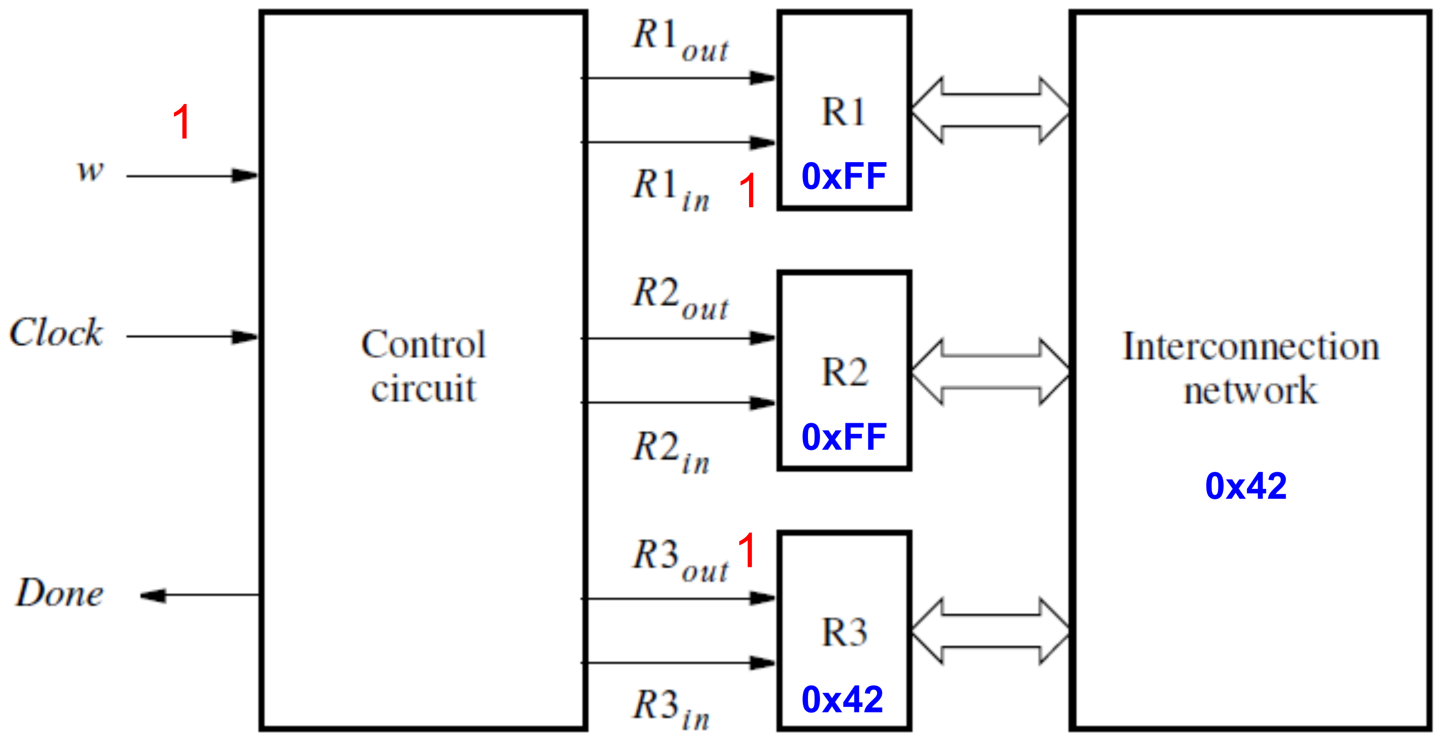
Animated Register Swap



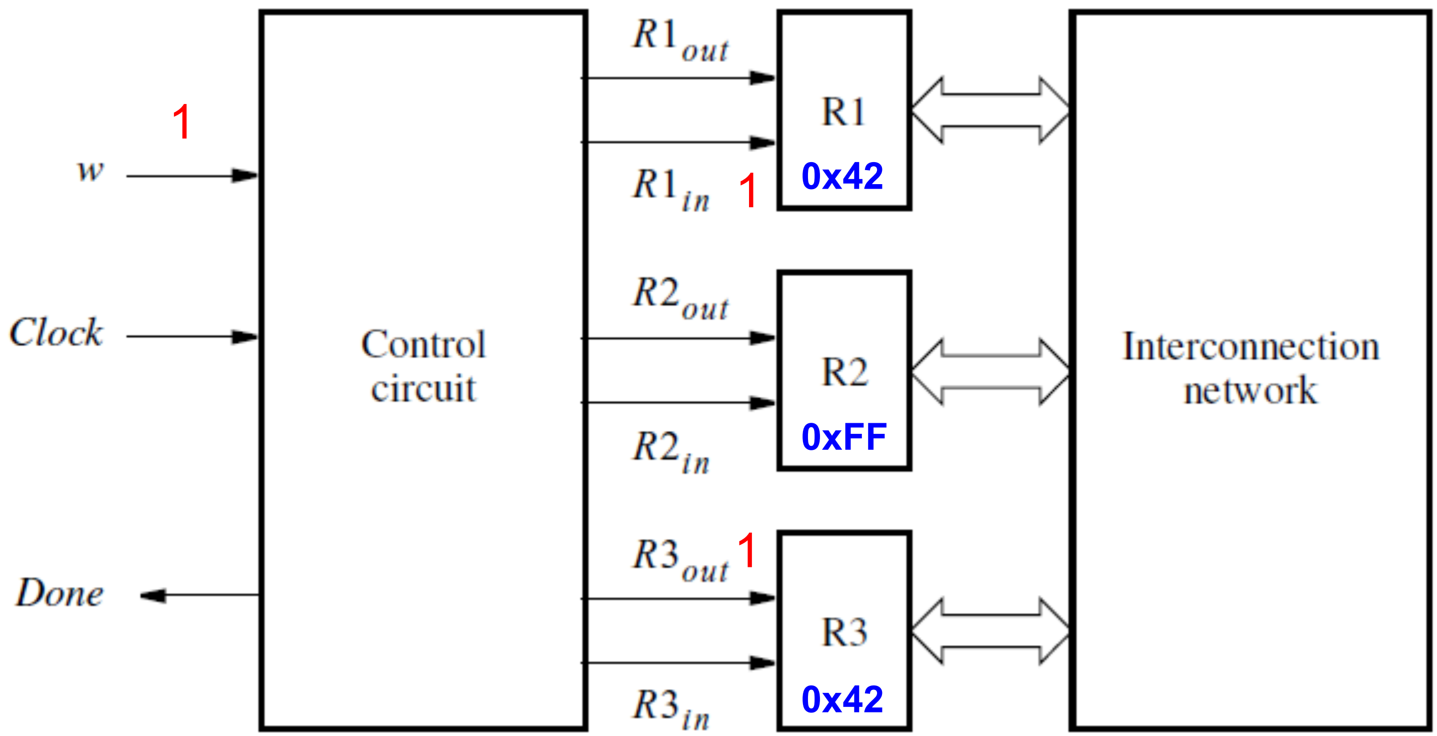
Animated Register Swap



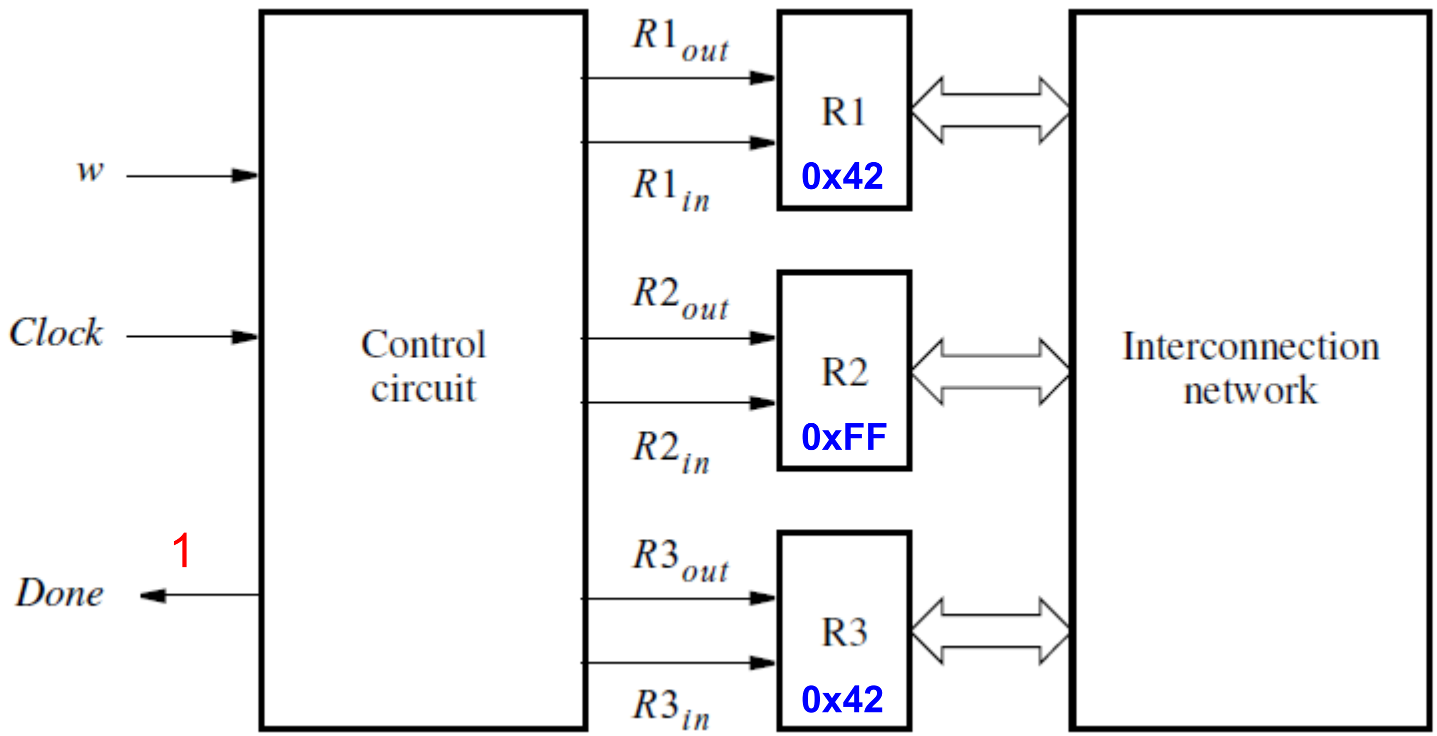
Animated Register Swap



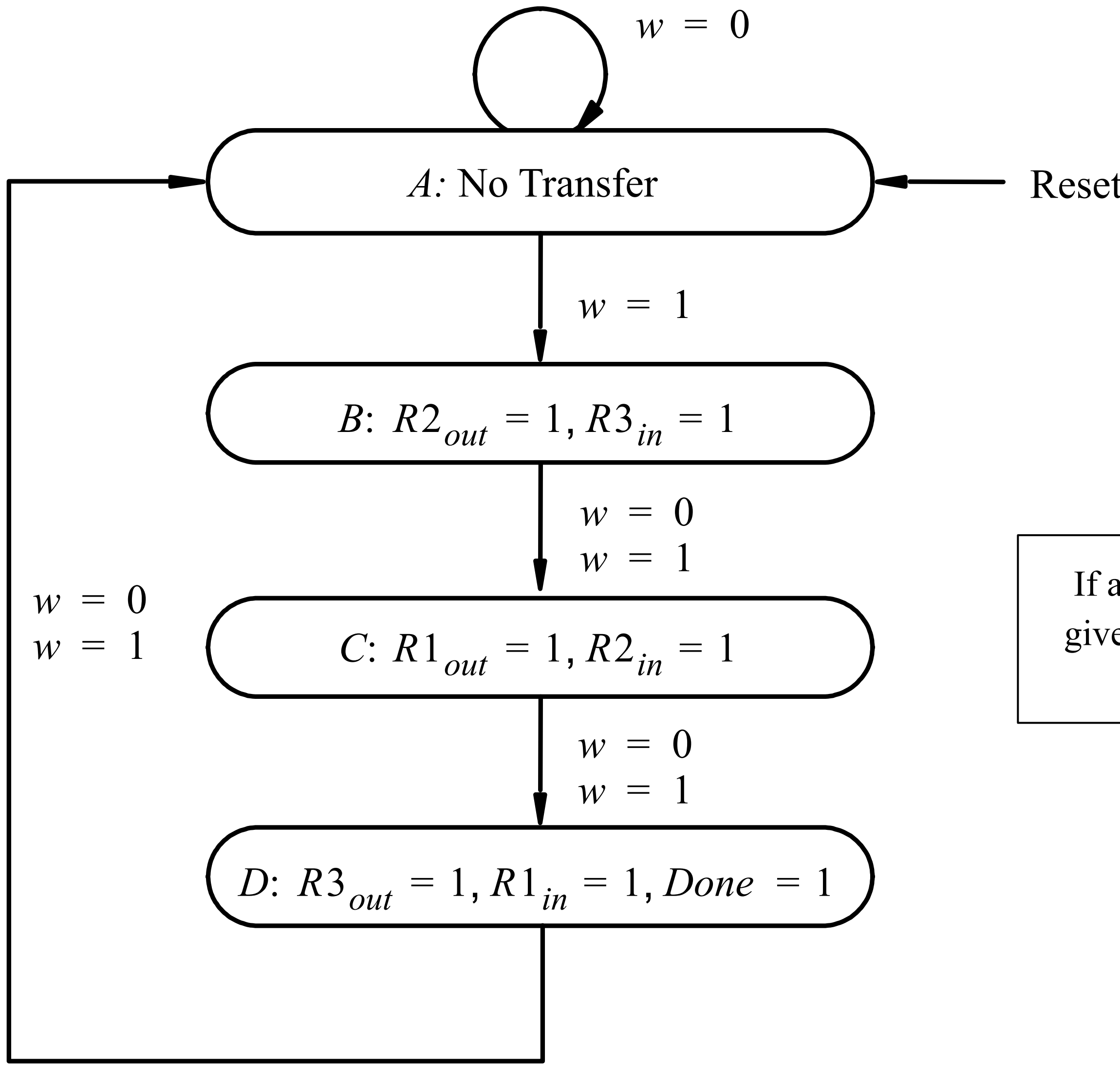
Animated Register Swap



Animated Register Swap



State Diagram



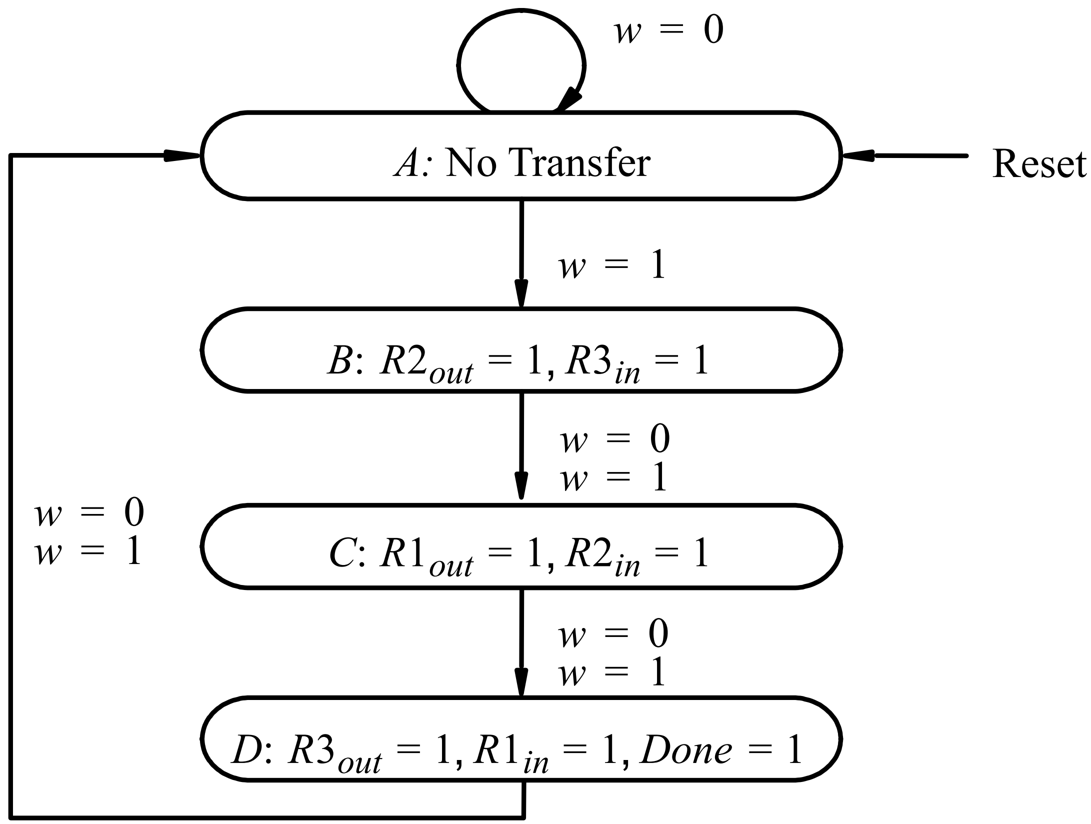
If an output is not given, then assume that it is 0.

[Figure 6.11 from the textbook]

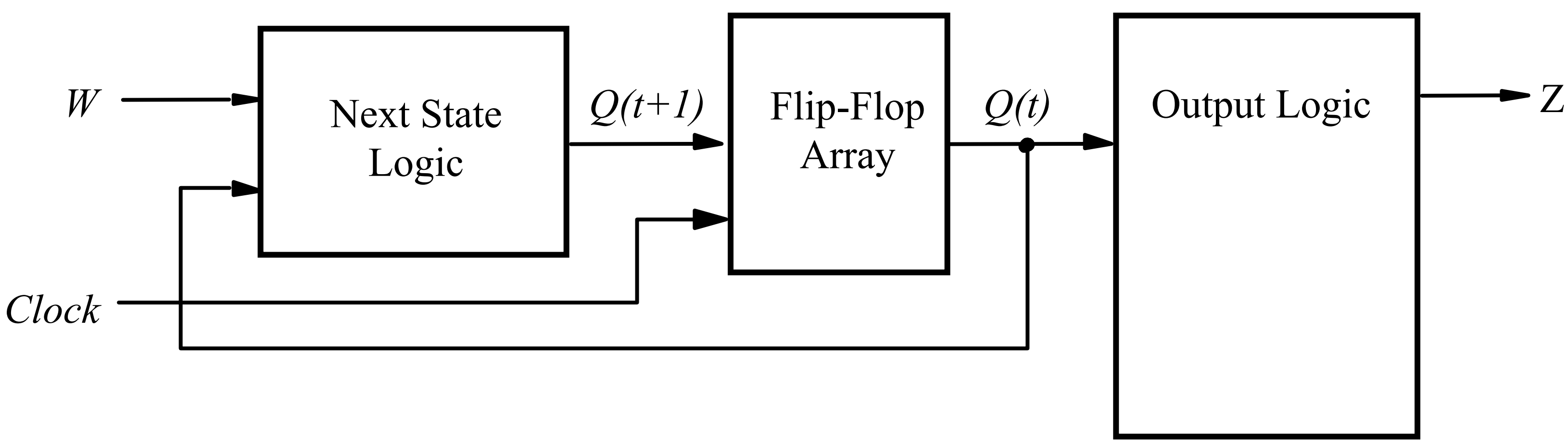
Some Questions

- How many flip-flops are we going to use?

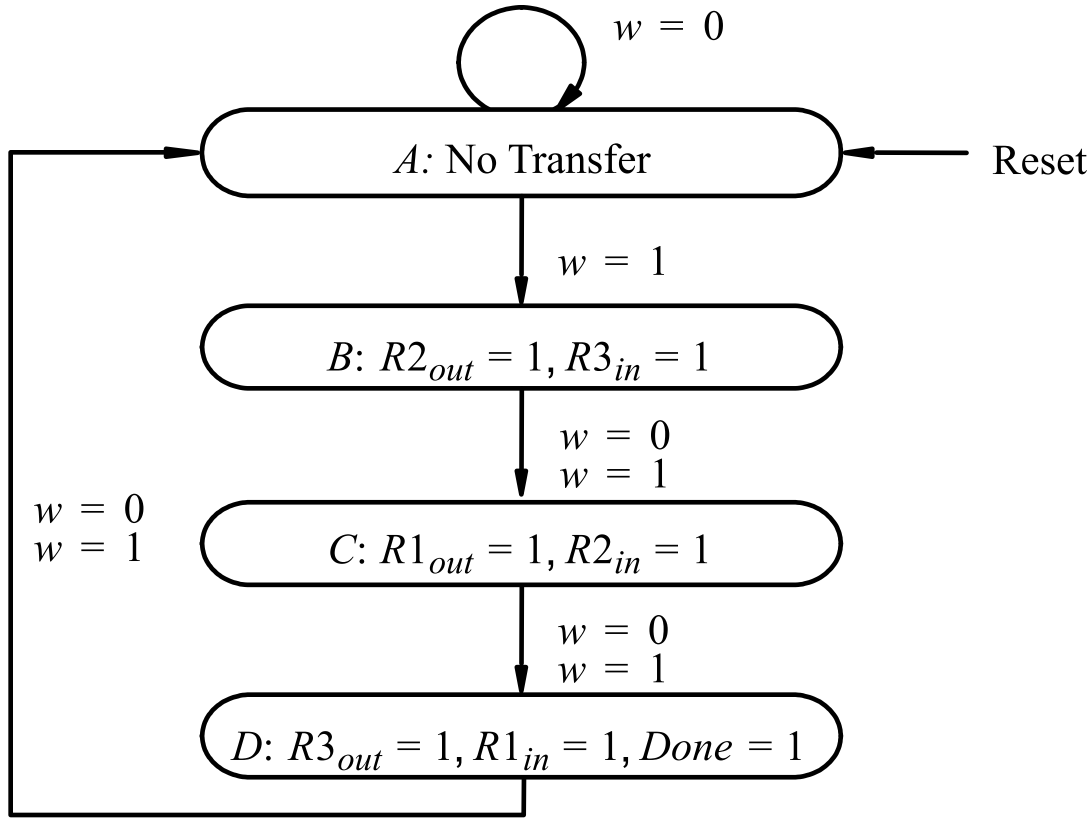
- How many logic expressions do we need to find?

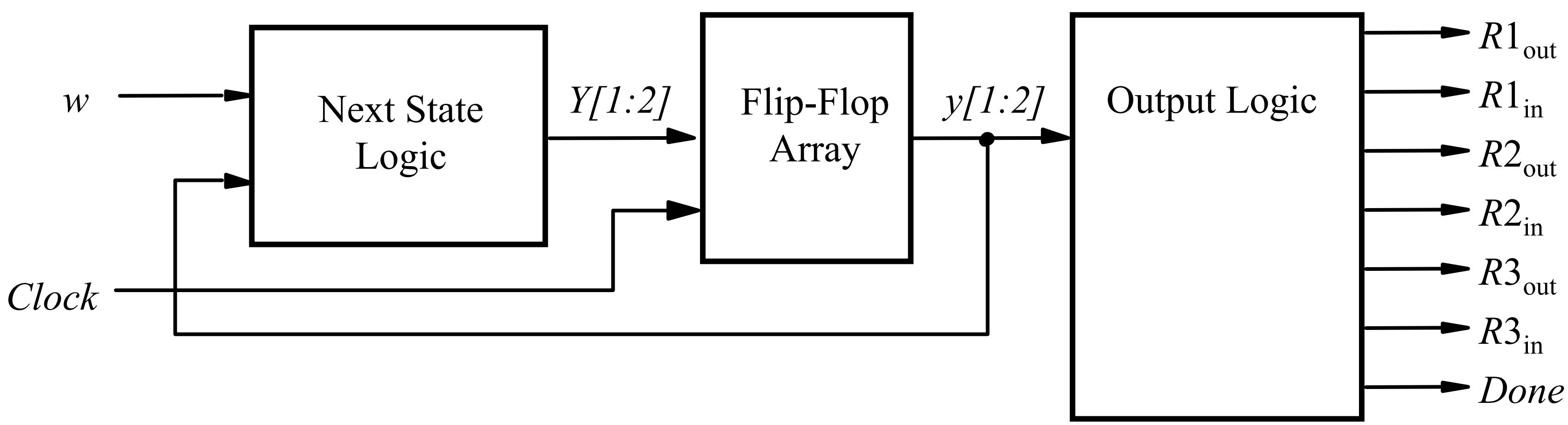


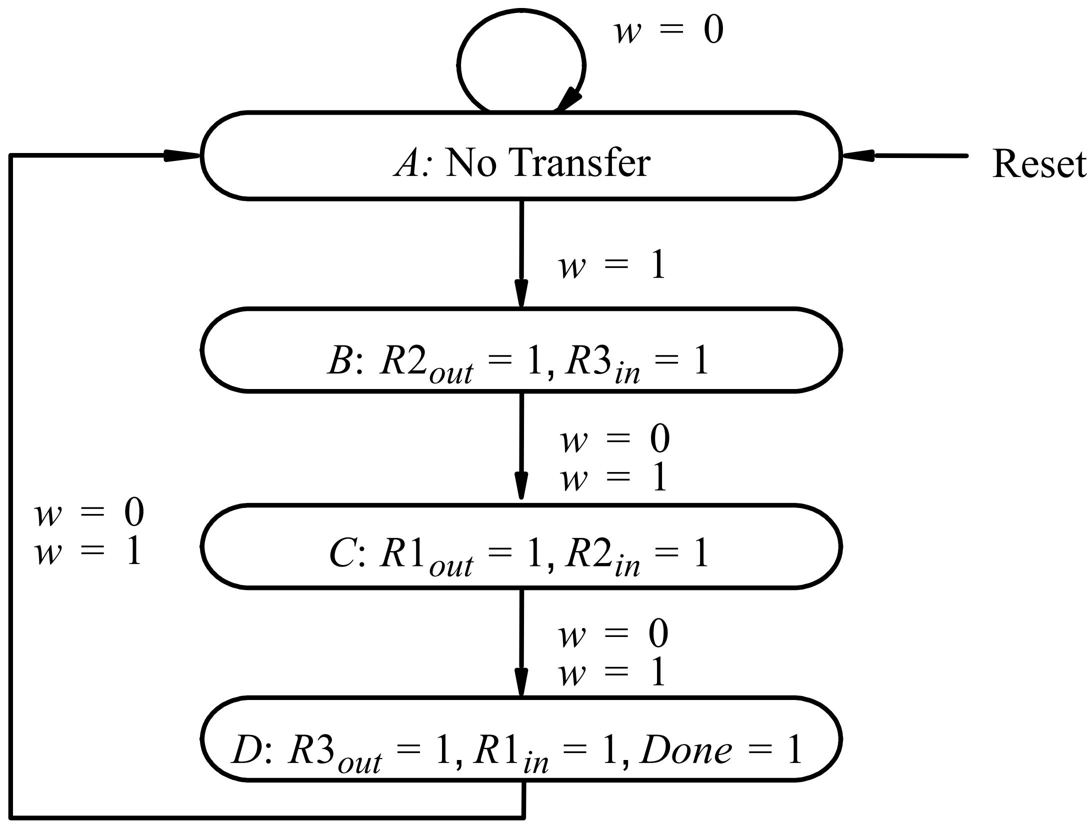
w = 0
A: No Transfer
Reset
w = 1
B: R2out = 1, R3in = 1
w = 0
w = 1
C: R1out = 1, R2in = 1
w = 0
w = 1
w = 0
w = 1
D: R3out = 1, R1in = 1, Done = 1



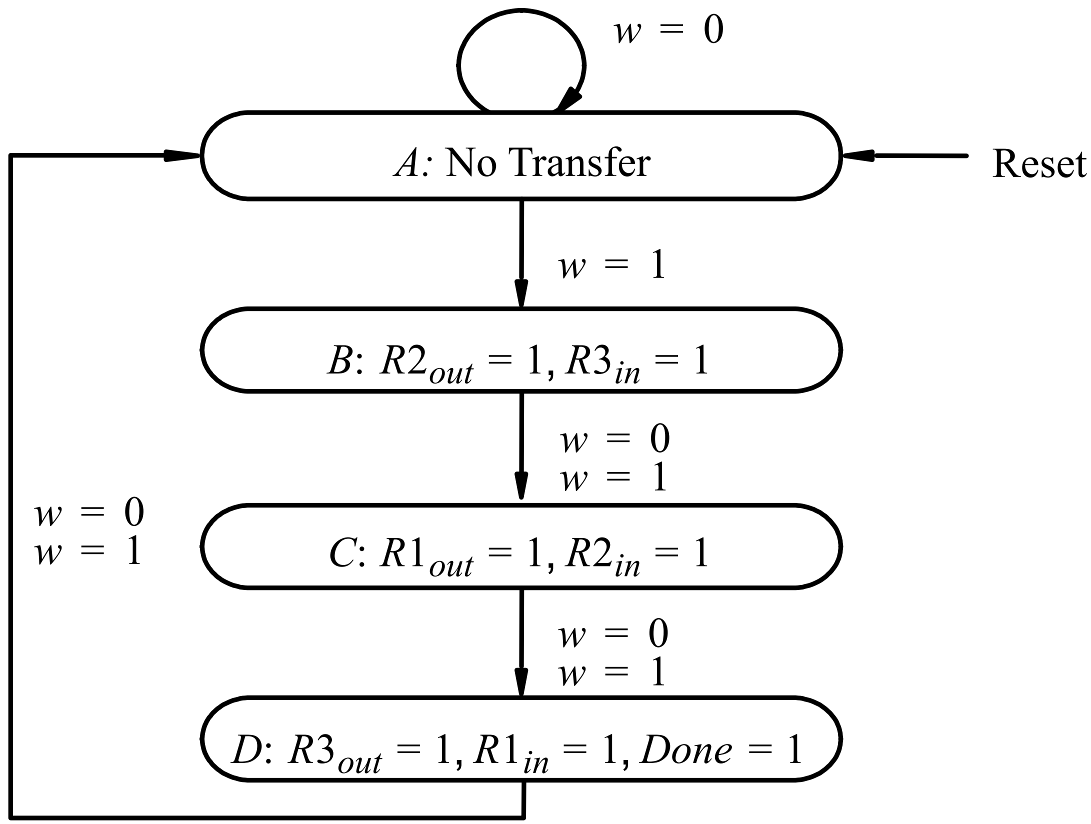
W
Next State Logic
Q(t+1)
Flip-Flop Array
Q(t)
Output Logic
Z
Clock


w = 0
A: No Transfer
Reset
w = 1
B: R2out = 1, R3in = 1
w = 0
w = 1
C: R1out = 1, R2in = 1
w = 0
w = 1
w = 0
w = 1
D: R3out = 1, R1in = 1, Done = 1


w
Clock
Next State Logic
Y[1:2]
Flip-Flop Array
y[1:2]
Output Logic
R1out
R1in
R2out
R2in
R3out
R3in
Done



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



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

As we saw before, we can expect that some state encodings will be better than others.

We will consider three encoding schemes.

Encoding #1:
A=00, B=01, C=10, D=11

(Uses Two Flip-Flops)

State Table

Present state	Next state $w = 0$	Next state $w = 1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*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-Assigned Table

	Present state y_2y_1	Next state $w = 0$ Y_2Y_1	Next state $w = 1$ Y_2Y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A										
B										
C										
D										

[Figure 6.12 & 6.13 from the textbook]

State Table

Present state	Next state $w=0$	Next state $w=1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*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 Assigned Table

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

[Figure 6.12 & 6.13 from the textbook]

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

State Assigned Table

	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							
B	01	10	1 0							
C	10	11	1 1							
D	11	00	0 0							

[Figure 6.12 & 6.13 from the textbook]

State Table

Present state	Next state $w=0$	Next state $w=1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*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 Assigned Table

	Present state y_2y_1	Next state $w=0$ Y_2Y_1	Next state $w=1$ 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

[Figure 6.12 & 6.13 from the textbook]

	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*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

y_2	y_1	w	Y_2	Y_1
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

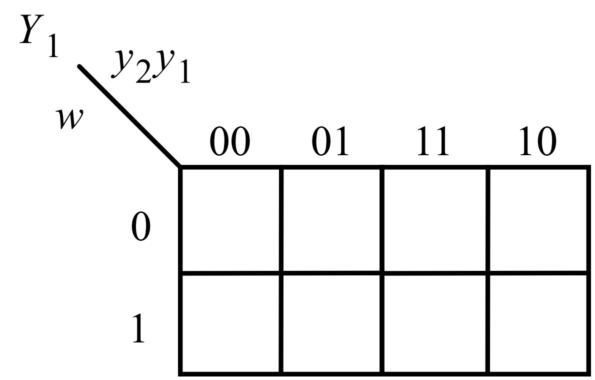
Let's derive the next-state expressions

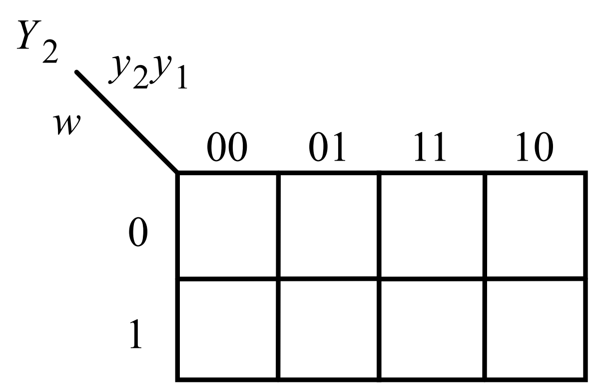
	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	y_2y_1	Y_2Y_1	Y_2Y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1	0	0	0	0	0	0	0
B	01	10	1 0	0	0	1	0	0	1	0
C	10	11	1 1	1	0	0	1	0	0	0
D	11	00	0 0	0	1	0	0	1	0	1

y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	0

	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

y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	0





	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

y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	0

Y_1

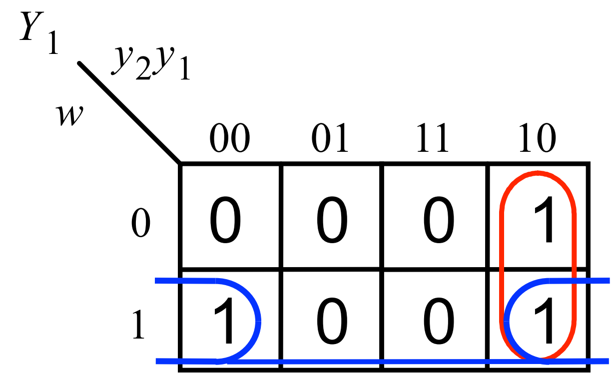
w \ y_2y_1	00	01	11	10
0	0	0	0	1
1	1	0	0	1

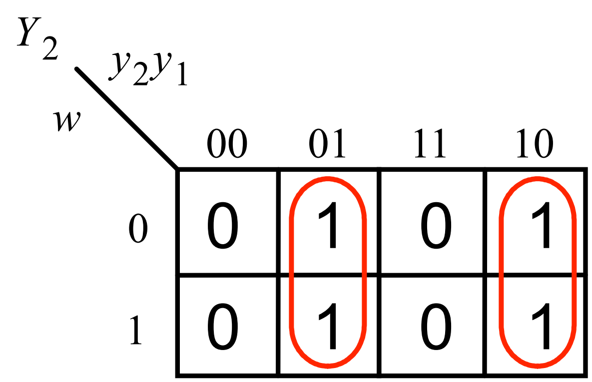
Y_2

w \ y_2y_1	00	01	11	10
0	0	1	0	1
1	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

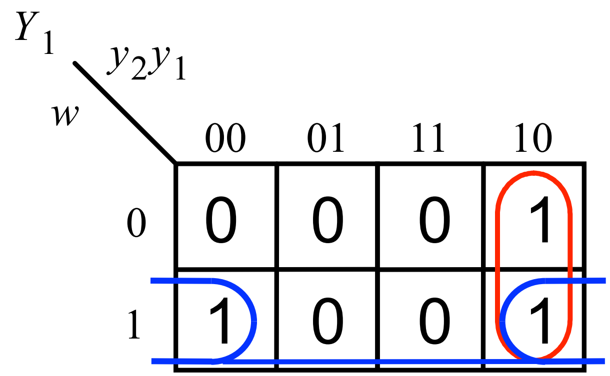
y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	0



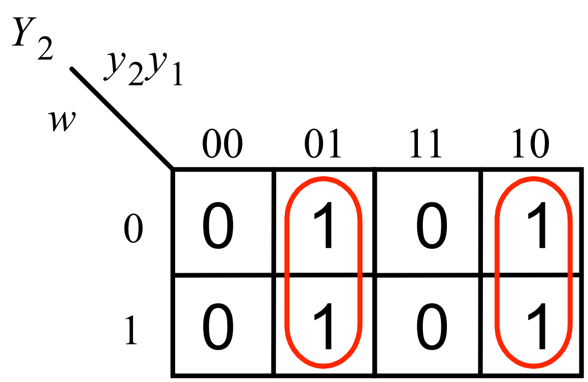


	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	y_2y_1	Y_2Y_1	Y_2Y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1	0	0	0	0	0	0	0
B	01	10	1 0	0	0	1	0	0	1	0
C	10	11	1 1	1	0	0	1	0	0	0
D	11	00	0 0	0	1	0	0	1	0	1

y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	0
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	0



$$Y_1 = w\bar{y}_1 + \bar{y}_1 y_2$$



$$Y_2 = y_1\bar{y}_2 + \bar{y}_1 y_2$$

	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

y_2	y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$
0	0			
0	1			
1	0			
1	1			

Let's derive the output expressions

	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

y_2	y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$
0	0			
0	1			
1	0			
1	1			

Let's derive the output expressions

We need to derive only these 3 unique ones

	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

y_2	y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$
0	0	0	0	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

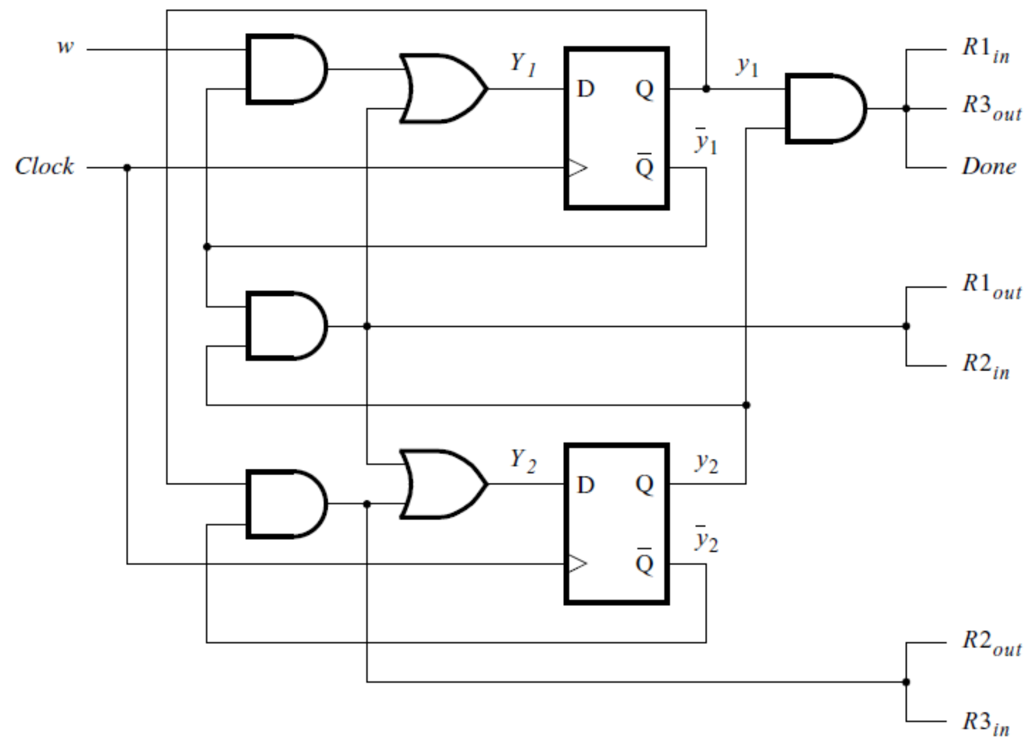
	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1	0	0	0	0	0	0	0
B	01	10	1 0	0	0	1	0	0	1	0
C	10	11	1 1	1	0	0	1	0	0	0
D	11	00	0 0	0	1	0	0	1	0	1

y_2	y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$
0	0	0	0	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

$$R1_{out} = R2_{in} = \overline{y_1}\, y_2$$

$$R1_{in} = R3_{out} = Done = y_1\, y_2$$

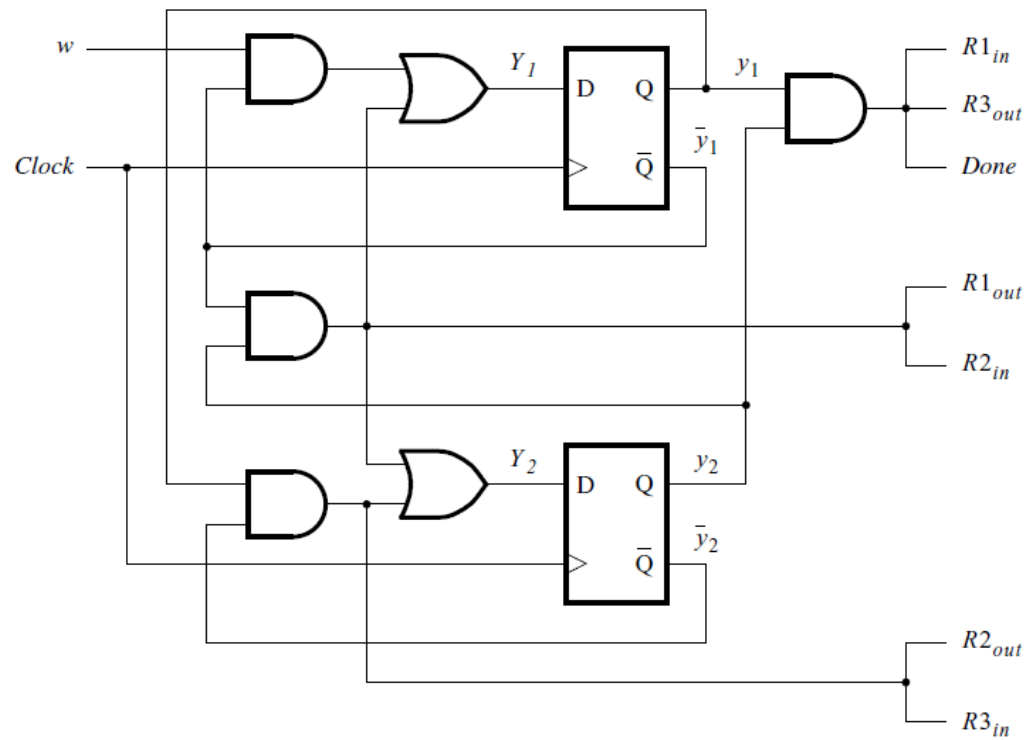
$$R2_{out} = R3_{in} = y_1\, \overline{y_2}$$



	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

$$Y_1 = w\bar{y}_1 + \bar{y}_1 y_2$$

$$Y_2 = y_1\bar{y}_2 + \bar{y}_1 y_2$$



$y_1 y_2$

$\overline{y_1} y_2$

$y_1 \overline{y_2}$

	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

Encoding #2:
A=00, B=01, C=11, D=10

(Also Uses Two Flip-Flops)

State Table (same as before)

Present state	Next state $w = 0$	Next state $w = 1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*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-Assigned Table

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

[Figure 6.12 & 6.18 from the textbook]

State Table (same as before)

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-Assigned Table

	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00									
B	01									
C	11									
D	10									

[Figure 6.12 & 6.18 from the textbook]

State Table (same as before)

Present state	Next state $w = 0$	Next state $w = 1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*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-Assigned Table

	Present state $y_2 y_1$	Next state $w = 0$ $Y_2 Y_1$	Next state $w = 1$ $Y_2 Y_1$	Outputs $R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1							
B	01	11	1 1							
C	11	10	1 0							
D	10	00	0 0							

[Figure 6.12 & 6.18 from the textbook]

State Table (same as before)

Present state	Next state $w = 0$	Next state $w = 1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*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-Assigned Table

	Present state y_2y_1	Next state $w = 0$ Y_2Y_1	Next state $w = 1$ 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	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

[Figure 6.12 & 6.18 from the textbook]

	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	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

y_2	y_1	w	Y_2	Y_1
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

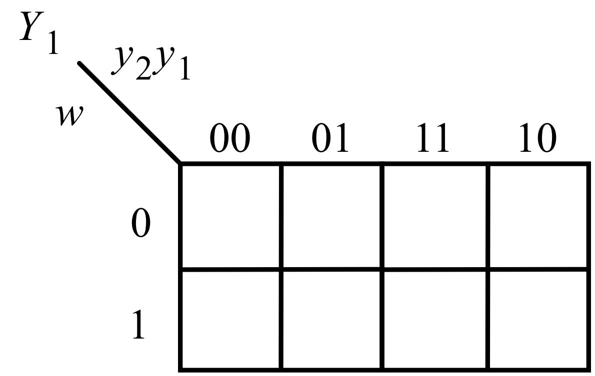
Let's derive the next-state expressions

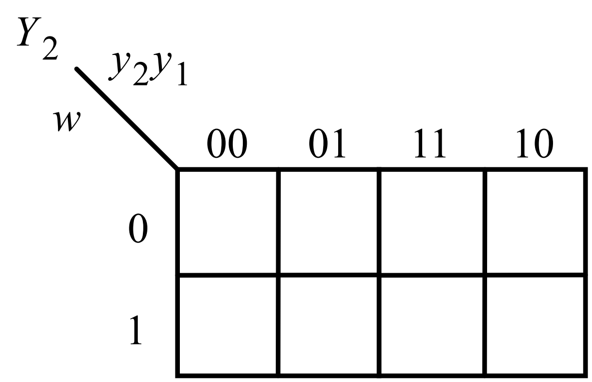
	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1	0	0	0	0	0	0	0
B	01	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	0
1	1	1	1	0

	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	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	0
1	1	1	1	0





	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	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	0
1	1	1	1	0

Y_1

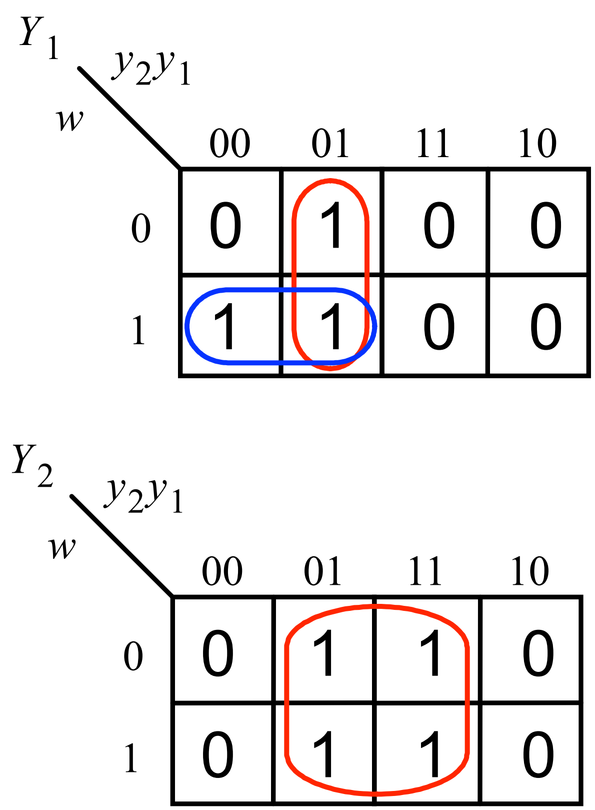
w \ y_2y_1	00	01	11	10
0	0	1	0	0
1	1	1	0	0

Y_2

w \ y_2y_1	00	01	11	10
0	0	1	1	0
1	0	1	1	0

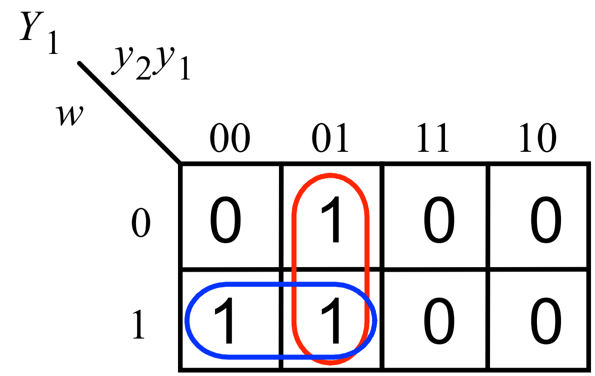
	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1	0	0	0	0	0	0	0
B	01	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	0
1	1	1	1	0

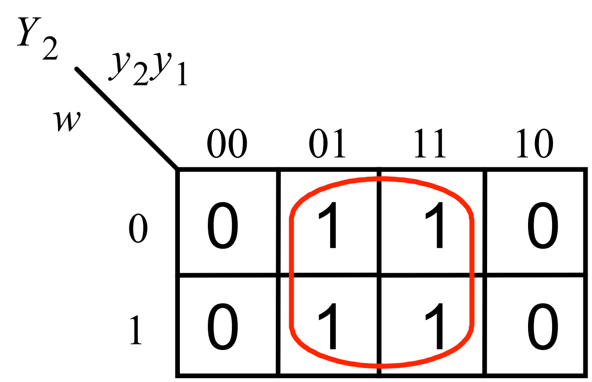


	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	y_2y_1	Y_2Y_1	Y_2Y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1	0	0	0	0	0	0	0
B	01	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

y_2	y_1	w	Y_2	Y_1
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	0
1	1	1	1	0



$$Y_1 = w\bar{y}_2 + y_1\bar{y}_2$$



$$Y_2 = y_1$$

	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1	0	0	0	0	0	0	0
B	01	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

y_2	y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$
0	0			
0	1			
1	0			
1	1			

Let's derive the output expressions

	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1	0	0	0	0	0	0	0
B	01	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

y_2	y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$
0	0			
0	1			
1	0			
1	1			

Let's derive the output expressions

Once again, we only need to derive these three unique ones.

	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	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

	y_2	y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$
A	0	0	0		
B	0	1	0		
D	1	0	0		
C	1	1	1		

Note that C and D are swapped in the truth table due to the new state encoding that was chosen.

	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	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

	y_2	y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$
A	0	0	0	0	0
B	0	1	0	0	1
D	1	0	0	1	0
C	1	1	1	0	0

	Present state	Next state		Outputs						
		$w = 0$	$w = 1$							
	$y_2 y_1$	$Y_2 Y_1$	$Y_2 Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	00	00	0 1	0	0	0	0	0	0	0
B	01	11	1 1	0	0	1	0	0	1	0
C	11	10	1 0	1	0	0	1	0	0	0
D	10	00	0 0	0	1	0	0	1	0	1

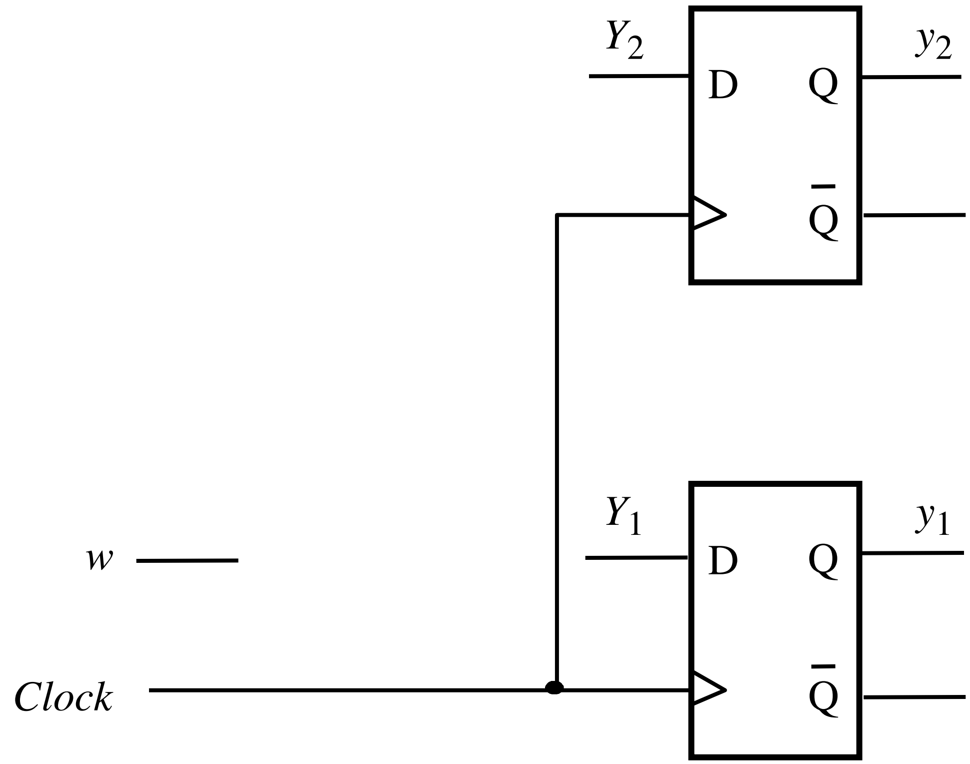
	y_2	y_1	$R1_{out}$	$R1_{in}$	$R2_{out}$
A	0	0	0	0	0
B	0	1	0	0	1
D	1	0	0	1	0
C	1	1	1	0	0

$R1_{out} = R2_{in} = y_1 \, y_2$

$R1_{in} = R3_{out} = Done = \overline{y_1} \, y_2$

$R2_{out} = R3_{in} = y_1 \, \overline{y_2}$

Let's Complete the Circuit Diagram



$$Y_1 = w\overline{y}_2 + y_1\overline{y}_2$$

$$Y_2 = y_1$$

$R1_{out} = R2_{in} = y_1\, y_2$

$R1_{in} = R3_{out} = Done = \overline{y_1}\, y_2$

$R2_{out} = R3_{in} = y_1\, \overline{y_2}$

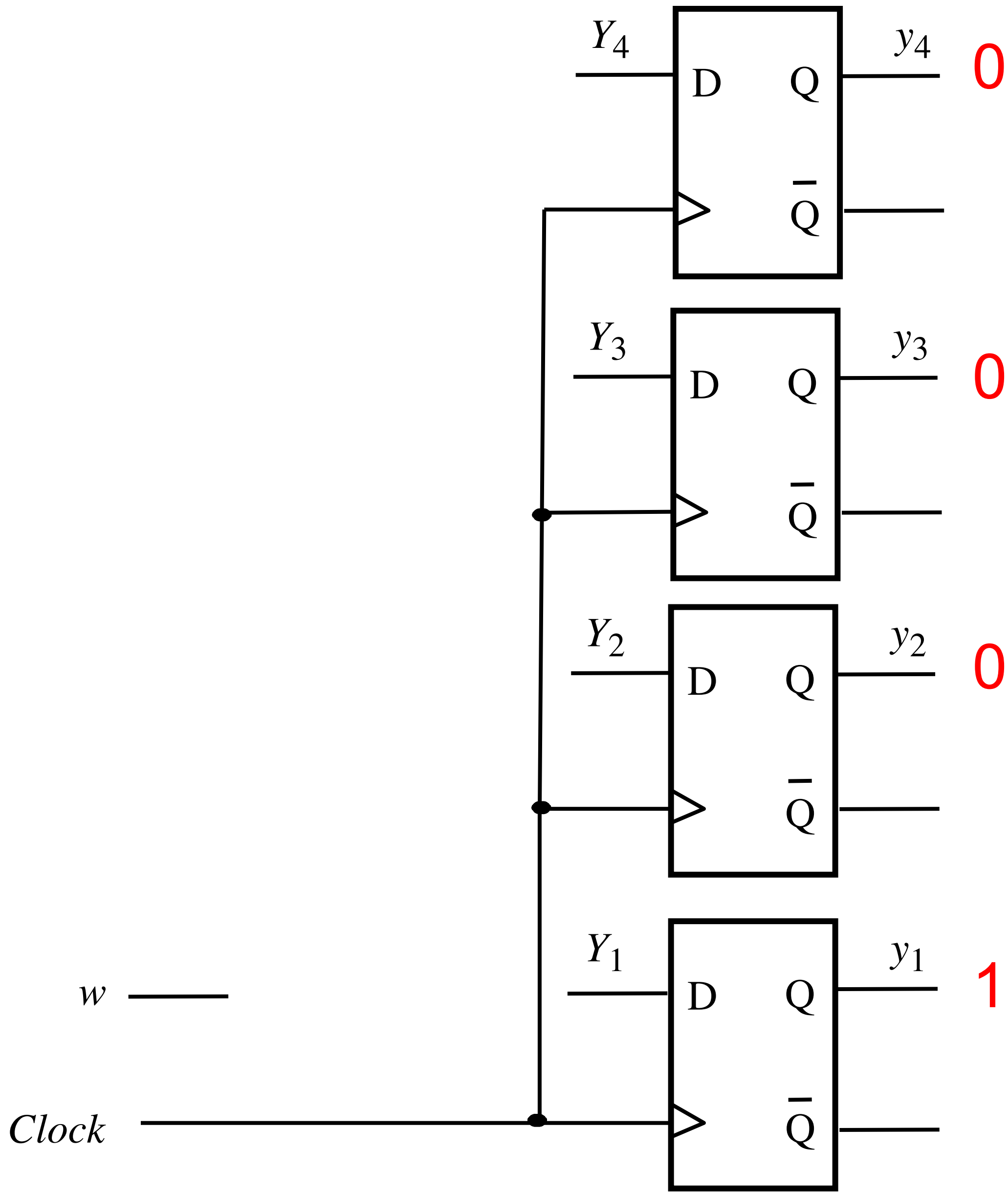
Encoding #3:
A=0001, B=0010, C=0100, D=1000

(One-Hot Encoding – Uses Four Flip-Flops)

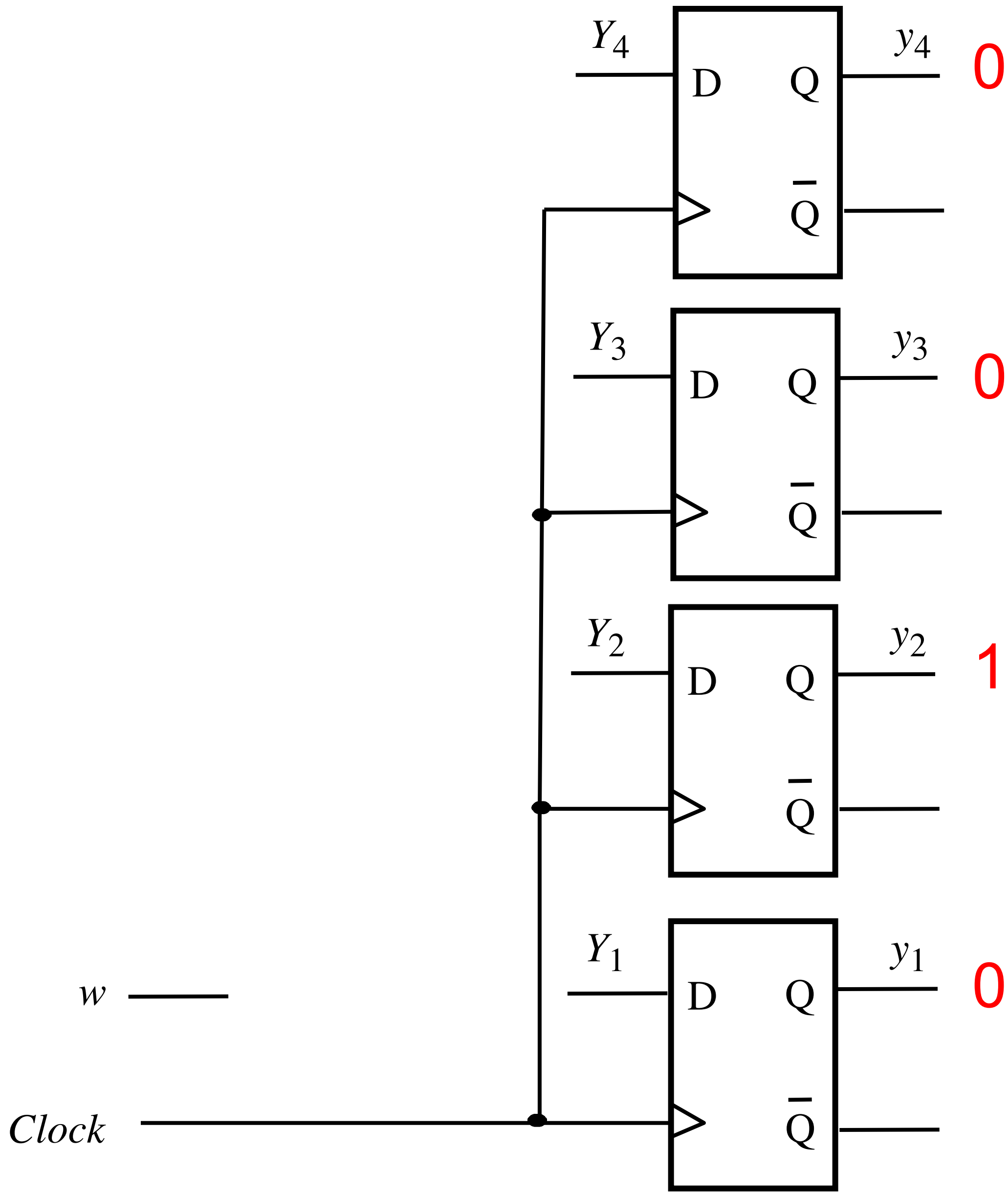
One-Hot State Encoding

- So far, we have been encoding states in a way that minimizes the number of flip-flops.
- But sometimes we can decrease the complexity of our logic if we encode states more sparsely.

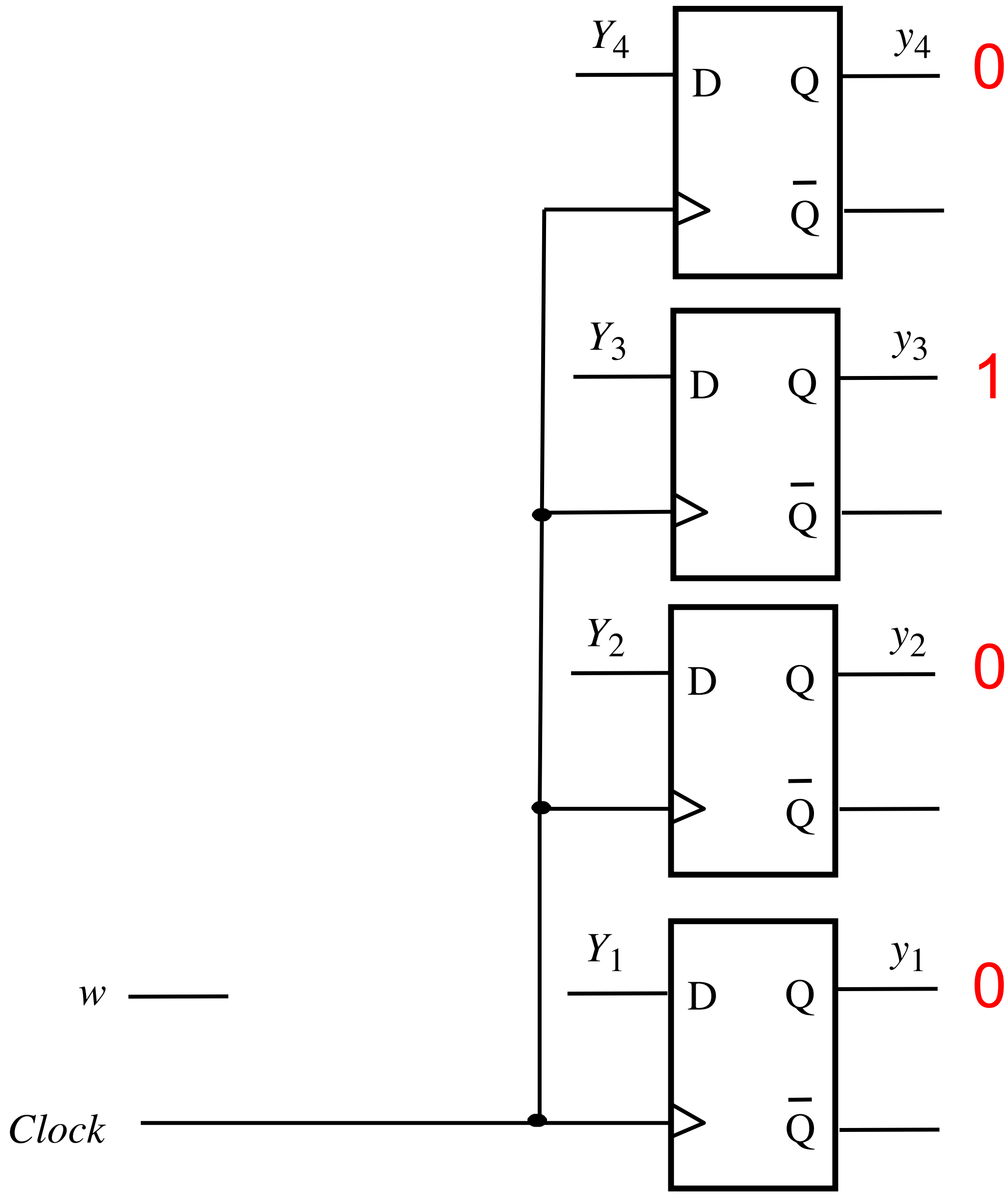
Encoding for State A



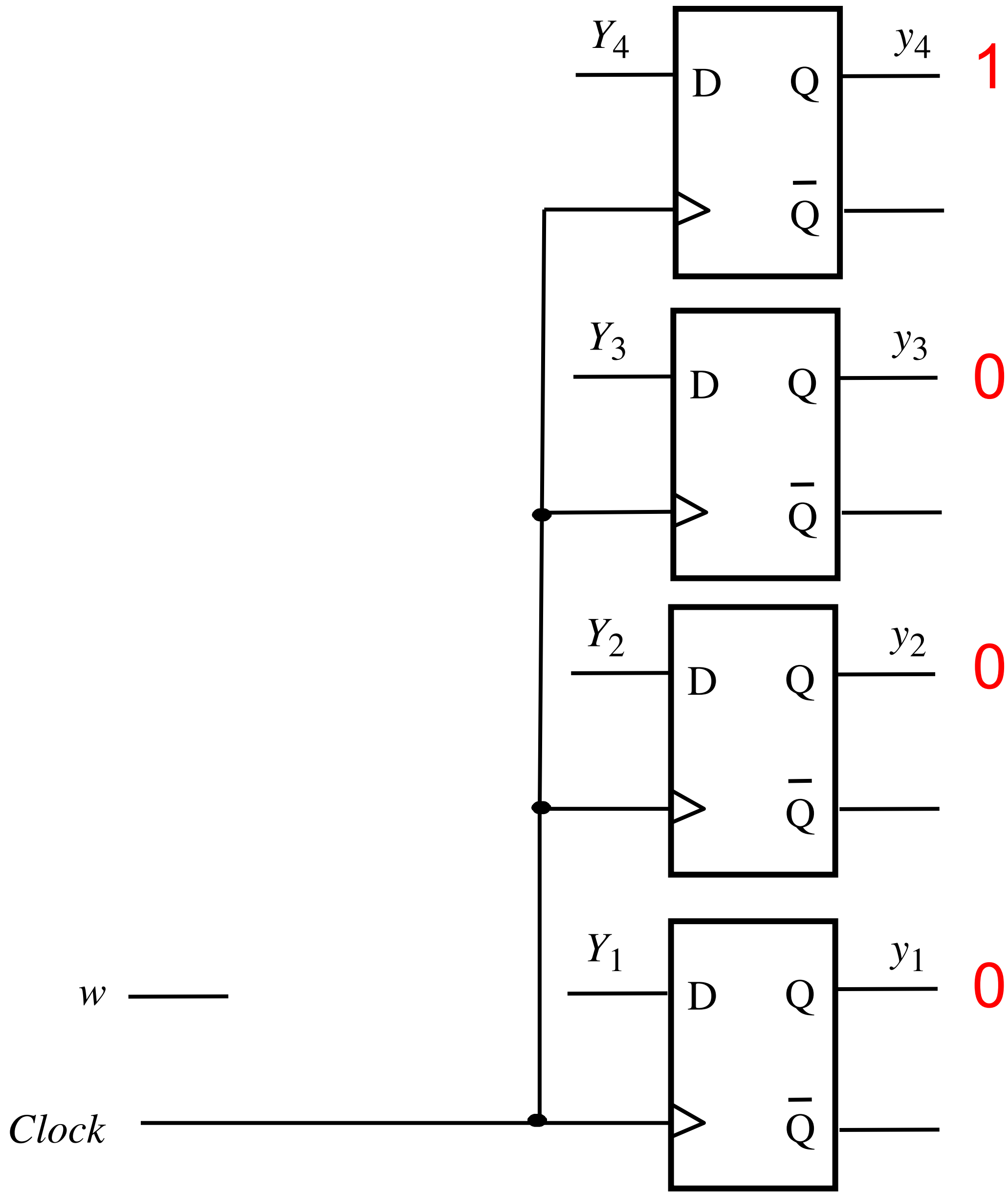
Encoding for State B



Encoding for State C



Encoding for State D



Register Swap Controller

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

Register Swap Controller

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

Let's use four flip-flops and the following one-hot state encoding scheme:

A = 0001

B = 0010

C = 0100

D = 1000

State Table (same as before)

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-Assigned Table

	Present State	Next State		Outputs						
		$w = 0$	$w = 1$							
	$y_4y_3y_2y_1$	$Y_4Y_3Y_2Y_1$	$Y_4Y_3Y_2Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A										
B										
C										
D										

[Figure 6.12 & 6.21 from the textbook]

State Table (same as before)

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-Assigned Table

	Present State	Next State		Outputs						
		$w = 0$	$w = 1$							
	$y_4y_3y_2y_1$	$Y_4Y_3Y_2Y_1$	$Y_4Y_3Y_2Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	0 001									
B	0 010									
C	0 100									
D	1 000									

[Figure 6.12 & 6.21 from the textbook]

State Table (same as before)

Present state	Next state $w = 0$	Next state $w = 1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*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-Assigned Table

	Present State $y_4y_3y_2y_1$	Next State $w = 0$ $Y_4Y_3Y_2Y_1$	Next State $w = 1$ $Y_4Y_3Y_2Y_1$	Outputs $R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	0 001	0001	0010							
B	0 010	0100	0100							
C	0 100	1000	1000							
D	1 000	0001	0001							

[Figure 6.12 & 6.21 from the textbook]

State Table (same as before)

Present state	Next state $w = 0$	Next state $w = 1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*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-Assigned Table

	Present State $y_4y_3y_2y_1$	Next State $w = 0$ $Y_4Y_3Y_2Y_1$	Next State $w = 1$ $Y_4Y_3Y_2Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	0 001	0001	0010	0	0	0	0	0	0	0
B	0 010	0100	0100	0	0	1	0	0	1	0
C	0 100	1000	1000	1	0	0	1	0	0	0
D	1 000	0001	0001	0	1	0	0	1	0	1

[Figure 6.12 & 6.21 from the textbook]

Let's Derive the Next-State Expressions

	Present State	Next State $w = 0$	Next State $w = 1$	Outputs						
	$y_4y_3y_2y_1$	$Y_4Y_3Y_2Y_1$	$Y_4Y_3Y_2Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	0 001	0001	0010	0	0	0	0	0	0	0
B	0 010	0100	0100	0	0	1	0	0	1	0
C	0 100	1000	1000	1	0	0	1	0	0	0
D	1 000	0001	0001	0	1	0	0	1	0	1

Let's Derive the Next-State Expressions

$$Y_1(w, y_4, y_3, y_2, y_1)$$
$$Y_2(w, y_4, y_3, y_2, y_1)$$
$$Y_3(w, y_4, y_3, y_2, y_1)$$
$$Y_4(w, y_4, y_3, y_2, y_1)$$

We need to do four 5-variable K-maps!

	Present State	Next State		Outputs						
		$w = 0$	$w = 1$							
	$y_4y_3y_2y_1$	$Y_4Y_3Y_2Y_1$	$Y_4Y_3Y_2Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	0 001	0001	0010	0	0	0	0	0	0	0
B	0 010	0100	0100	0	0	1	0	0	1	0
C	0 100	1000	1000	1	0	0	1	0	0	0
D	1 000	0001	0001	0	1	0	0	1	0	1

Let's Derive the Next-State Expressions

$$Y_1(w, y_4, y_3, y_2, y_1) = \overline{w}y_1 + y_4$$
$$Y_2(w, y_4, y_3, y_2, y_1) = wy_1$$
$$Y_3(w, y_4, y_3, y_2, y_1) = y_2$$
$$Y_4(w, y_4, y_3, y_2, y_1) = y_3$$

Or we can be smarter than that ☺

	Present State	Next State		Outputs						
		$w = 0$	$w = 1$							
	$y_4y_3y_2y_1$	$Y_4Y_3Y_2Y_1$	$Y_4Y_3Y_2Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	0 001	0001	0010	0	0	0	0	0	0	0
B	0 010	0100	0100	0	0	1	0	0	1	0
C	0 100	1000	1000	1	0	0	1	0	0	0
D	1 000	0001	0001	0	1	0	0	1	0	1

Let's Derive the Output Expressions

	Present State	Next State		Outputs						
		$w = 0$	$w = 1$							
	$y_4y_3y_2y_1$	$Y_4Y_3Y_2Y_1$	$Y_4Y_3Y_2Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	0 001	0001	0010	0	0	0	0	0	0	0
B	0 010	0100	0100	0	0	1	0	0	1	0
C	0 100	1000	1000	1	0	0	1	0	0	0
D	1 000	0001	0001	0	1	0	0	1	0	1

Let's Derive the Output Expressions

$R1_{out}(y_4, y_3, y_2, y_1)$
$R1_{in}(y_4, y_3, y_2, y_1)$
$R2_{out}(y_4, y_3, y_2, y_1)$
$R2_{in}(y_4, y_3, y_2, y_1)$
$R3_{out}(y_4, y_3, y_2, y_1)$
$R3_{in}(y_4, y_3, y_2, y_1)$
$Done(y_4, y_3, y_2, y_1)$

We need to do seven 4-variable K-maps!

	Present State	Next State		Outputs						
		$w = 0$	$w = 1$							
	$y_4y_3y_2y_1$	$Y_4Y_3Y_2Y_1$	$Y_4Y_3Y_2Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	0 001	0001	0010	0	0	0	0	0	0	0
B	0 010	0100	0100	0	0	1	0	0	1	0
C	0 100	1000	1000	1	0	0	1	0	0	0
D	1 000	0001	0001	0	1	0	0	1	0	1

Let's Derive the Output Expressions

$R1_{out}(y_4, y_3, y_2, y_1) = y_3$
$R1_{in}(y_4, y_3, y_2, y_1) = y_4$
$R2_{out}(y_4, y_3, y_2, y_1) = y_2$
$R2_{in}(y_4, y_3, y_2, y_1) = y_3$
$R3_{out}(y_4, y_3, y_2, y_1) = y_4$
$R3_{in}(y_4, y_3, y_2, y_1) = y_2$
$Done(y_4, y_3, y_2, y_1) = y_4$

Or we can be smarter than that by exploiting the one-hot property

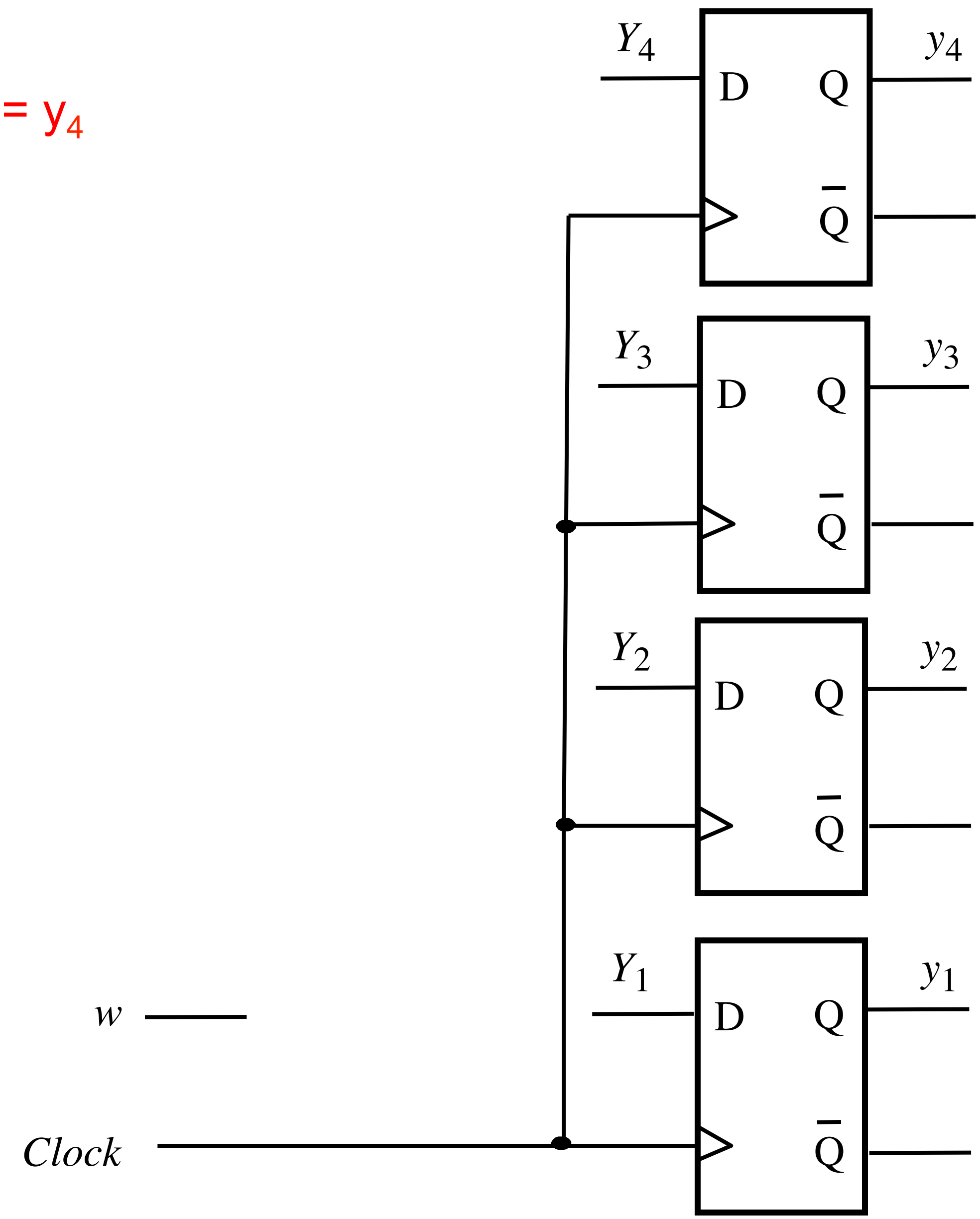
	Present State	Next State		Outputs						
		$w = 0$	$w = 1$							
	$y_4y_3y_2y_1$	$Y_4Y_3Y_2Y_1$	$Y_4Y_3Y_2Y_1$	$R1_{out}$	$R1_{in}$	$R2_{out}$	$R2_{in}$	$R3_{out}$	$R3_{in}$	*Done*
A	0 001	0001	0010	0	0	0	0	0	0	0
B	0 010	0100	0100	0	0	1	0	0	1	0
C	0 100	1000	1000	1	0	0	1	0	0	0
D	1 000	0001	0001	0	1	0	0	1	0	1

Let's Complete the Circuit Diagram

$R1_{out} = R2_{in} = y_3$

$R1_{in} = R3_{out} = Done = y_4$

$R2_{out} = R3_{in} = y_2$



$Y_1 = \overline{w}\, y_1 + y_4$

$Y_2 = w\, y_1$

$Y_3 = y_2$

$Y_4 = y_3$

Questions?

THE END