

1

© 2013 Pearson Education, Inc. or its affiliate(s). All rights reserved. Pearson Education, Inc., publishing as Pearson Benjamin Cummings, 101 Philip Drive, Assinippi Park, New York, NY 10984-2135

Outline

- Summary of design procedure for sequential circuits
- Design example – code converter
- Design of iterative circuits
- Design of sequential circuits using ROMs and PLAs

3

Summary of Design Procedure for Sequential Circuits

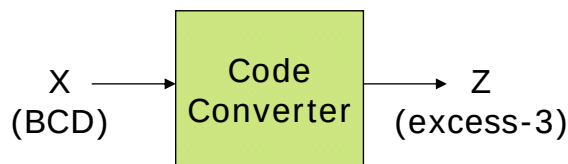
1. Derive state table (and/or state graph)
 - Given the problem statement, determine the required relationship between the input and output sequences and derive a state table
 - For many problems, it is easiest to first construct a state graph
2. Reduce state table
 - State minimization by row matching or using an implication table
3. Perform state assignment
 - $\lceil \log_2 m \rceil$ flip-flops are needed to encode m states
4. Form transition table
 - Substitute the assigned flip-flop states for each state in the reduced state table
5. Derive flip-flop input equations and output functions
 - Plot next-state maps and input maps for each flip-flop to derive the flip-flop input equations
6. Realize flip-flop input equations and output functions using available logic gates
7. Check design
 - Use signal tracing, computer simulation, or lab testing

4

Design Example – Code Converter

□ Convert BCD to excess-3 code

- Add 3 to a binary-coded-decimal digit in the range 0 to 9
- Assume serial input and output with the least significant bit first
- Reset to initial state after receiving every 4 inputs



Can the converter be realized without delaying the output?

$t_0 \quad t_1 \quad t_2 \quad t_3 \quad \dots$
 $X = X^{(0)} X^{(1)} X^{(2)} X^{(3)} \dots$
 $Z = Z^{(0)} Z^{(1)} Z^{(2)} Z^{(3)} \dots$

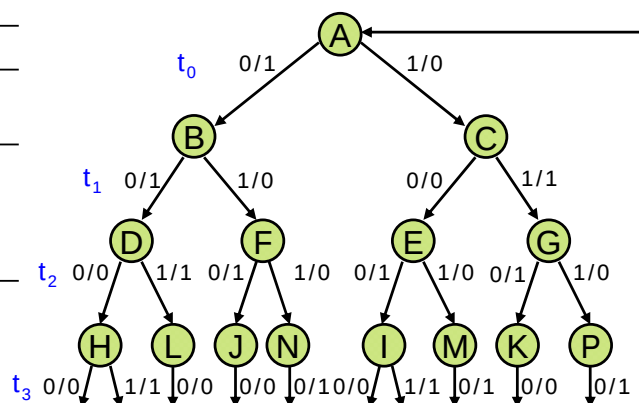
X Input (BCD)				Z Output (excess-3)			
t_3	t_2	t_1	t_0	t_3	t_2	t_1	t_0
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
other codes				-	-	-	-

5

Design Example – Code Converter

□ Step 1: derive state table and state graph

Time	Input Sequence Received (LSB first)	Present State	Next State		Present Output (Z)	
			X=0	1	X=0	1
t_0	reset	A	B	C	1	0
t_1	0	B	D	F	1	0
	1	C	E	G	0	1
t_2	00	D	H	L	0	1
	01	E	I	M	1	0
	10	F	J	N	1	0
	11	G	K	P	1	0
t_3	000	H	A	A	0	1
	001	I	A	A	0	1
	010	J	A	-	0	-
	011	K	A	-	0	-
	100	L	A	-	0	-
	101	M	A	-	1	-
	110	N	A	-	1	-
	111	P	A	-	1	-



6

Design Example – Code Converter

□ Step 2: Reduce state table

Time	Input Sequence Received (LSB first)	Present State	Next State		Present Output (Z)	
			X=0	1	X=0	1
t ₀	reset	A	B	C	1	0
t ₁	0	B	D	F	1	0
	1	C	E	G	0	1
t ₂	00	D	H	L	0	1
	01	E	I	M	1	0
	10	F	J	N	1	0
	11	G	K	P	1	0
t ₃	000	H	A	A	0	1
	001	I	A	A	0	1
	010	J	A	-	0	-
	011	K	A	-	0	-
	100	L	A	-	0	-
	101	M	A	-	1	-
	110	N	A	-	1	-
	111	P	A	-	1	-

Time	Present State	Next State		Present Output (Z)	
		X = 0	1	X = 0	1
t ₀	A	B	C	1	0
t ₁	B	D	E	1	0
	C	E	E	0	1
t ₂	D	H	H	0	1
	E	H	M	1	0
t ₃	H	A	A	0	1
	M	A	-	1	-

The matching assumes, under X=1, the next state is A and Z=1

The matching imposes no particular condition on these don't cares

7

Design Example – Code Converter

- In reducing a state table by row matching, sometimes there can be multiple incompatible choices in matching a row

Example

Present State	Next State		Present Output (Z)	
	X=0	1	X=0	1
Q	Q	S	0	1
R	-	-	-	1
S	S	Q	1	-

R can be matched with either Q or S, but not both

8

Design Example – Code Converter

Step 3: Perform state assignment

- Need 3 flip-flops for 7 states

Reduced state table

Time	Present State	Next State		Present Output (Z)	
		X = 0	1	X = 0	1
t_0	A	B	C	1	0
t_1	B	D	E	1	0
	C	E	E	0	1
t_2	D	H	H	0	1
	E	H	M	1	0
t_3	H	A	A	0	1
	M	A	-	1	-

Assignment map

Q_2Q_3		Q_1	
		0	1
00		A	B
01			C
11		H	D
10		M	E

An even better assignment?

9

Design Example – Code Converter

Step 4: Form transition table

Transition table

	$Q_1Q_2Q_3$	$Q_1^+Q_2^+Q_3^+$		Z	
		X=0	X=1	X=0	X=1
A	000	100	101	1	0
B	100	111	110	1	0
C	101	110	110	0	1
D	111	011	011	0	1
E	110	011	010	1	0
H	011	000	000	0	1
M	010	000	xxx	1	x
-	001	xxx	xxx	x	x

Time	Present State	Next State		Present Output (Z)	
		X = 0	1	X = 0	1
t_0	A	B	C	1	0
t_1	B	D	E	1	0
	C	E	E	0	1
t_2	D	H	H	0	1
	E	H	M	1	0
t_3	H	A	A	0	1
	M	A	-	1	-

Q_2Q_3		Q_1	
		0	1
00		A	B
01			C
11		H	D
10		M	E

10

Design Example – Code Converter

Step 5: Derive flip-flop input equations and output functions

X	Q ₁	Q ₂	Q ₃	Z	D ₁	D ₂	D ₃
0	0	0	0	1	1	0	0
0	0	0	1	x	x	x	x
0	0	1	0	1	0	0	0
0	0	1	1	0	0	0	0
0	1	0	0	1	1	1	1
0	1	0	1	0	1	1	0
0	1	1	0	1	0	1	1
0	1	1	1	0	0	1	1
1	0	0	0	0	1	0	1
1	0	0	1	x	x	x	x
1	0	1	0	x	x	x	x
1	0	1	1	1	0	0	0
1	1	0	0	0	1	1	0
1	1	0	1	1	1	1	0
1	1	1	0	0	0	1	0
1	1	1	1	1	0	1	1

XQ ₁	00	01	11	10
Q ₂ Q ₃ 00	1	1	1	1
01	x	1	1	x
11	0	0	0	0
10	0	0	0	x

$$D_1 = Q_1^+ = Q_2'$$

XQ ₁	00	01	11	10
Q ₂ Q ₃ 00	0	1	1	0
01	x	1	1	x
11	0	1	1	0
10	0	1	1	x

$$D_2 = Q_2^+ = Q_1$$

XQ ₁	00	01	11	10
Q ₂ Q ₃ 00	0	1	0	1
01	x	0	0	x
11	0	1	1	0
10	0	1	0	x

$$D_3 = Q_3^+ = Q_1Q_2Q_3 + X'Q_1Q_3' + XQ_1'Q_2'$$

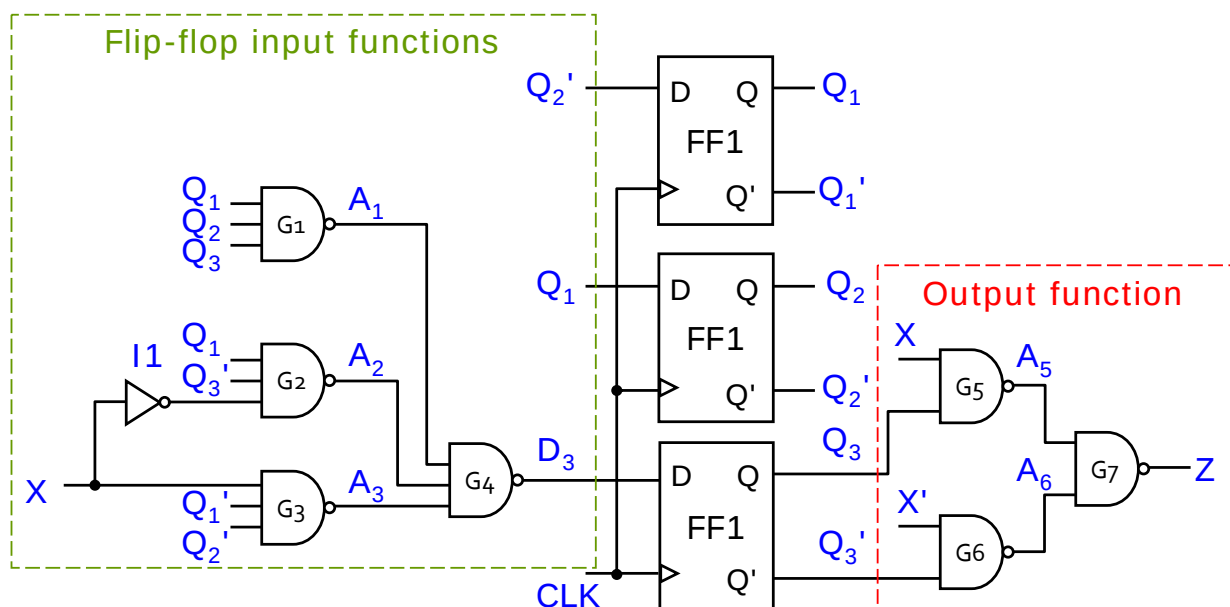
XQ ₁	00	01	11	10
Q ₂ Q ₃ 00	1	1	0	0
01	x	0	1	x
11	0	0	1	1
10	1	1	0	x

$$Z = X'Q_3' + XQ_3$$

11

Design Example – Code Converter

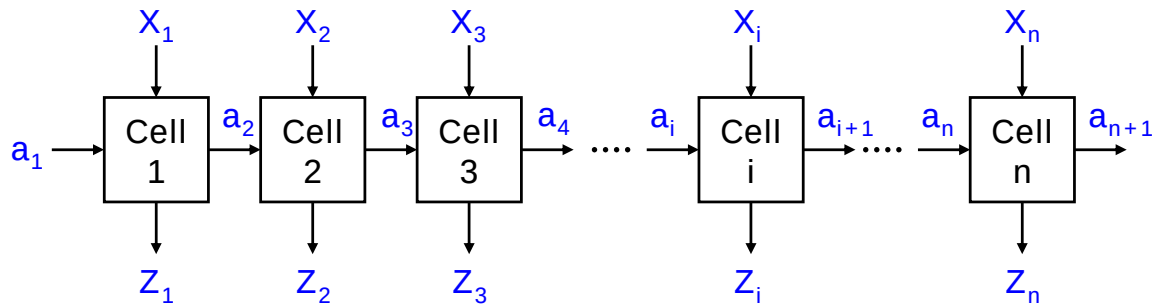
Step 6: Realize circuit



12

Design of Iterative Circuits

- Many of the design procedures used for sequential circuits can be applied to the design of iterative circuits
 - E.g., the design procedure used for the (sequential) serial adder of Unit 13 can be applied to the design of the parallel adder of Unit 4
- **Unilateral iterative circuit** is the simplest form of an iterative circuit
 - A linear array of combinational cells with signals between cells traveling in only one direction



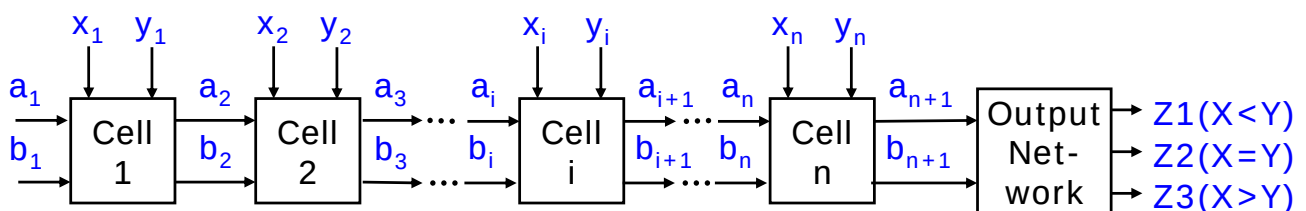
X_i and Z_i are the primary input and output, respectively, of cell i
 a_i and a_{i+1} resemble the present state and next state, respectively, of cell i

13

Design of Iterative Circuits

Design of a Comparator

- Compare two n -bit binary numbers
 $X = x_1x_2...x_n$ and $Y = y_1y_2...y_n$
 and determine if they are equal or which one is larger if they are not equal
 - Assume x_1 and y_1 are the most significant bits (we plan to do the comparison from left to right)



14

Design of Iterative Circuits

Design of a Comparator

Steps 1,2: derive and reduce state table

		S_{i+1}						
	S_i	$x_i y_i = 00$	01	11	10	Z_1	Z_2	Z_3
$X=Y$	S_0	S_0	S_2	S_0	S_1	0	1	0
$X>Y$	S_1	S_1	S_1	S_1	S_1	0	0	1
$X<Y$	S_2	S_2	S_2	S_2	S_2	1	0	0

$\begin{cases} Z_1: X<Y \\ Z_2: X=Y \\ Z_3: X>Y \end{cases}$

Steps 3,4: Perform state assignment and form transition table

		$a_{i+1} b_{i+1}$						
a_i	b_i	$x_i y_i = 00$	01	11	10	Z_1	Z_2	Z_3
0	0	00	10	00	01	0	1	0
0	1	01	01	01	01	0	0	1
1	0	10	10	10	10	1	0	0

state assignment

 $\begin{cases} S_0 = 00 \\ S_1 = 01 \\ S_2 = 10 \end{cases}$

15

Design of Iterative Circuits

Design of a Comparator

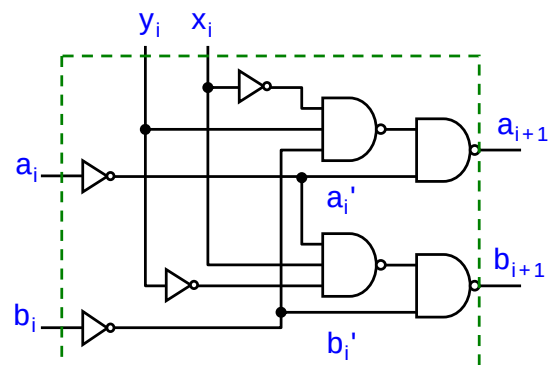
Steps 5,6: derive and realize cell input and output functions

Typical cell for comparator

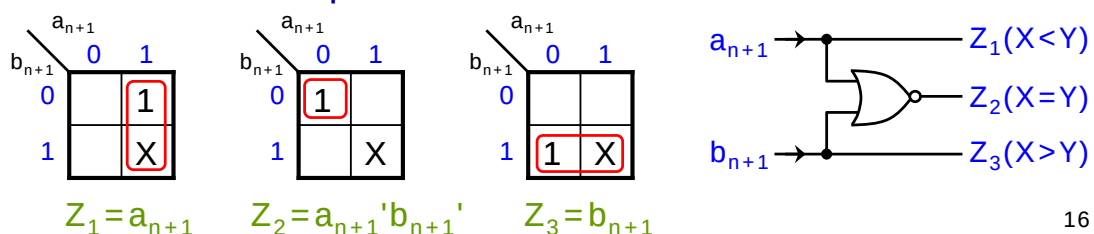
$x_i y_i$	00	01	11	10
$a_i b_i$ 00	0	1	0	0
01	0	0	0	0
11	X	X	X	X
10	1	1	1	1

$x_i y_i$	00	01	11	10
$a_i b_i$ 00	0	0	0	1
01	1	1	1	1
11	X	X	X	X
10	0	0	0	0

$a_{i+1} = a_i + x_i' y_i b_i'$
 $b_{i+1} = b_i + x_i y_i' a_i'$



Output circuit for comparator

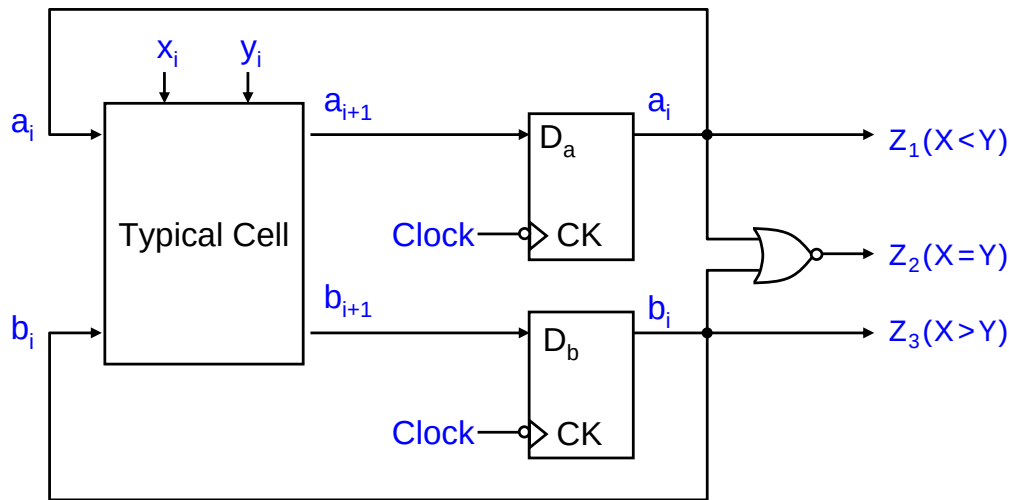


16

Design of Iterative Circuits

Design of a Comparator

□ Sequential comparator



$t_1 \quad t_2 \quad t_n \quad \text{time}$
 $X = x_1 \ x_2 \dots x_n \dots$
 $Y = y_1 \ y_2 \dots y_n \dots$

17

Design of Sequential Circuits Using ROMs and PLAs

□ For a Mealy sequential circuit, the combinational part can be realized using a ROM (PLA)

- For the circuit with m inputs, n outputs, and k state variables, we need k D flip-flops and a ROM with $m+k$ inputs (2^{m+k} words) and $n+k$ outputs

□ Similarly, for a Moore sequential circuit, the next-state and output combinational subcircuits can be realized using two ROMs (PLAs) or, alternatively, a single ROM (PLA)

18

Design of Sequential Circuits Using ROMs

Example

- Realize the prior BCD to excess-3 code converter using a ROM and D-FFs

State table

Present state	Next State		Present Output (Z)	
	X = 0	1	X = 0	1
A	B	C	1	0
B	D	E	1	0
C	E	E	0	1
D	H	H	0	1
E	H	M	1	0
H	A	A	0	1
M	A	-	1	-

Transition table

	$Q_1Q_2Q_3$	$Q_1^+Q_2^+Q_3^+$		Z	
		X=0	X=1	X=0	X=1
A	0 0 0	001	010	1	0
B	0 0 1	011	100	1	0
C	0 1 0	100	100	0	1
D	0 1 1	101	101	0	1
E	1 0 0	101	110	1	0
H	1 0 1	000	000	0	1
M	1 1 0	000	-	1	-

19

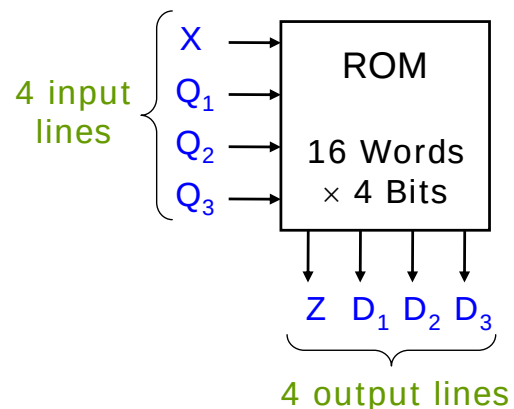
Design of Sequential Circuits Using ROMs

Example (cont'd)

- Realize the prior BCD to excess-3 code converter using a ROM and D-FFs

Truth table

X	Q_1	Q_2	Q_3	Z	D_1	D_2	D_3
0	0	0	0	1	1	0	0
0	0	0	1	x	x	x	x
0	0	1	0	1	0	0	0
0	0	1	1	0	0	0	0
0	1	0	0	1	1	1	1
0	1	0	1	0	1	1	0
0	1	1	0	1	0	1	1
0	1	1	1	0	0	1	1
1	0	0	0	0	1	0	1
1	0	0	1	x	x	x	x
1	0	1	0	x	x	x	x
1	0	1	1	1	0	0	0
1	1	0	0	0	1	1	0
1	1	0	1	1	1	1	0
1	1	1	0	0	0	1	0
1	1	1	1	1	0	1	1

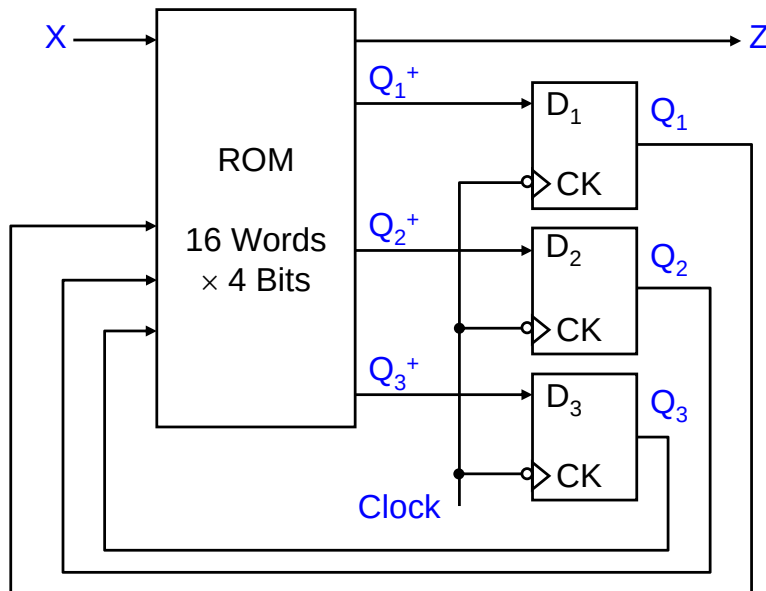


20

Design of Sequential Circuits Using ROMs

Example (cont'd)

- Realize the prior BCD to excess-3 code converter using a ROM and D-FFs



21

Design of Sequential Circuits Using PLAs

Example

- Realize the prior BCD to excess-3 code converter using a PLA and D-FFs

Assignment map

$Q_2 Q_3$ \ Q_1	0	1
00	A	B
01		C
11	H	D
10	M	E

Transition table

	$Q_1 Q_2 Q_3$	$Q_1^+ Q_2^+ Q_3^+$		Z	
		X=0	X=1	X=0	X=1
A	000	100	101	1	0
B	100	111	110	1	0
C	101	110	110	0	1
D	111	011	011	0	1
E	110	011	010	1	0
H	011	000	000	0	1
M	010	000	xxx	1	x

22

Design of Sequential Circuits Using PLAs

Example (cont'd)

- Realize the prior BCD to excess-3 code converter using a PLA and D-FFs

		XQ ₁			
		00	01	11	10
Q ₂ Q ₃	00	1	1	1	1
	01	X	1	1	X
	11	0	0	0	0
	10	0	0	0	X

$$D_1 = Q_1^+ = Q_2'$$

		XQ ₁			
		00	01	11	10
Q ₂ Q ₃	00	0	1	1	0
	01	X	1	1	X
	11	0	1	1	0
	10	0	1	1	X

$$D_2 = Q_2^+ = Q_1$$

		XQ ₁			
		00	01	11	10
Q ₂ Q ₃	00	0	1	0	1
	01	X	0	0	X
	11	0	1	1	0
	10	0	1	0	X

$$D_3 = Q_3^+ = Q_1Q_2Q_3 + X'Q_1Q_3' + XQ_1'Q_2'$$

		XQ ₁			
		00	01	11	10
Q ₂ Q ₃	00	1	1	0	0
	01	X	0	1	X
	11	0	0	1	1
	10	1	1	0	X

$$Z = X'Q_3' + XQ_3$$

23

Design of Sequential Circuits Using PLAs

Example (cont'd)

- Realize the prior BCD to excess-3 code converter using a PLA and D-FFs

PLA table

	X	Q ₁	Q ₂	Q ₃	Z	D ₁	D ₂	D ₃
Q ₂ '	-	-	0	-	0	1	0	0
Q ₁	-	1	-	-	0	0	1	0
Q ₁ Q ₂ Q ₃	-	1	1	1	0	0	0	1
X'Q ₁ Q ₃ '	0	1	-	0	0	0	0	1
XQ ₁ 'Q ₂ '	1	0	0	-	0	0	0	1
X'Q ₃ '	0	-	-	0	1	0	0	0
XQ ₃	1	-	-	1	1	0	0	0

$$D_1 = Q_1^+ = Q_2'$$

$$D_2 = Q_2^+ = Q_1$$

$$D_3 = Q_3^+ = Q_1Q_2Q_3 + X'Q_1Q_3' + XQ_1'Q_2'$$

$$Z = X'Q_3' + XQ_3$$

AND plane

OR plane

24

Design of Sequential Circuits Using PALs

Example

- A segment of a sequential PAL realizing the next-state equation $Q^+ = D = A'BQ' + AB'Q$

