CSE271 Section 2
Introduction to Digital Systems

Lecture 30

Instructor: Lin Li
http://www.cse.psu.edu/~lili/cse271

Derivation of State Table/Diagram

- Recall the first three steps of sequential system design:
  - Determine what needs to be stored in memory -- what are the possible states?
  - Code the inputs and outputs in binary.
  - Derive a state table or state diagram.
- Now, we start with verbal description of sequential system and develop state table/diagram.

Example 1

- A system with one input *x* and one output *z*, such that *z*=1 iff *x* has been 1 for at least three consecutive clock times.

  Solution 1: store the last three inputs. Knowing them we could determine the output.

$q_1q_2q_3$	$q_1^*q_2^*q_3^*$ x = 0	x = 1	z
0 0 0	0 0 0	0 0 1	0
0 0 1	0 1 0	0 1 1	0
0 1 0	1 0 0	1 0 1	0
0 1 1	1 1 0	1 1 1	0
1 0 0	0 0 0	0 0 1	0
1 0 1	0 1 0	0 1 1	0
1 1 0	1 0 0	1 0 1	0
1 1 1	1 1 0	1 1 1	1

We store the oldest input in q_1, and most recent one in q_3.

When updating memory, we discard the oldest input stored.

8 states are required and 3 flip flops are needed.

$q_3^*=x$ $q_2^*=q_3$ $q_1^*=q_2$

Example 1

- A system with one input *x* and one output *z*, such that *z*=1 iff *x* has been 1 for at least three consecutive clock times.

  Solution 2: store the number of consecutive 1's in memory.

  A: none -- the last input was 0.
  B: one
  C: two
  D: three or more

q	q^* x = 0	x = 1	z
A	A	B	0
B	A	C	0
C	A	D	0
D	A	D	1

4 states are required and 2 flip flops are needed.

Example 2

- A system with one input *x* and one output *z*, such that *z*=1 iff *x* is currently 1 and was also 1 at the previous two clock times.

Timing trace for CE6: It's a Moore model.

x	0	1	1	0	1	1	1	0	0	1	0	1	1	1	1	1	0	0	
z	?	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	0	0	0	0

Timing trace for CE7: It's a Mealey model.

x	0	1	1	0	1	1	1	0	0	1	0	1	1	1	1	1	0	0	
z	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1	0	0	0	0

Example 2

- A system with one input *x* and one output *z*, such that *z*=1 iff *x* is currently 1 and was also 1 at the previous two clock times.

Solution 1: store the last two inputs.

	$q_1^*q_2^*$		z	
q_1q_2	x=0	x=1	x=0	x=1
0 0	0 0	0 1	0	0
0 1	1 0	1 1	0	0
1 0	0 0	0 1	0	0
1 1	1 0	1 1	0	1

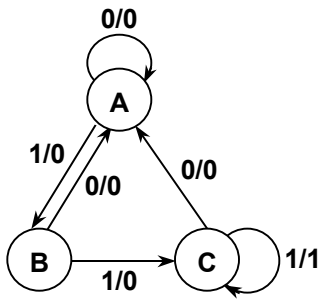
Example 2

- A system with one input *x* and one output *z*, such that *z*=1 iff *x* is currently 1 and was also 1 at the previous two clock times.

  Solution 2: store the number of consecutive 1's in memory.

  A: none -- the last input was 0.
  B: one
  C: two

q	q*		z	
	x=0	x=1	x=0	x=1
A	A	B	0	0
B	A	C	0	0
C	A	C	0	1



7

Example 3

- Design both a Moore and a Mealy system with one input *x* and one output *z*, such that *z*=1 iff *x* has been 1 for *exactly* three consecutive clock times.

  Sample timing trace:

x	0	1	1	1	1	1	0	1	1	0	1	1	1	0	1		
z-Mealy	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
z-Moore	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

  We cannot tell whether there should be a 1 output when the third consecutive 1 input occurs.
  We have to wait the next input arrives.

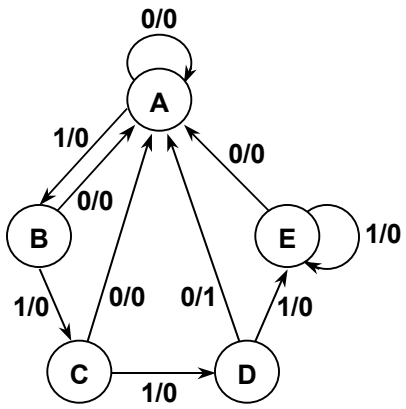
8

Example 3

- Design both a Moore and a Mealy system with one input *x* and one output *z*, such that *z*=1 iff *x* has been 1 for *exactly* three consecutive clock times.

Solution for Mealy system:

A: no 1 in a row
B: one 1 in a row
C: two 1's in a row
D: three 1's in a row
E: more than three 1's in a row



9

Example 3

- Design both a Moore and a Mealy system with one input *x* and one output *z*, such that *z*=1 iff *x* has been 1 for *exactly* three consecutive clock times.

Solution for Moore system:

A: no 1 in a row
B: one 1 in a row
C: two 1's in a row
D: three 1's in a row
E: more than three 1's in a row
F: three 1's plus a 0

	q*		
q	x = 0	x = 1	z
A	A	B	0
B	A	C	0
C	A	D	0
D	F	E	0
E	A	E	0
F	A	B	1

10

Example 4

- Design a Mealy system where the inputs are considered in blocks of three. The output is 1 iff the input is 1 for all three inputs in a block.

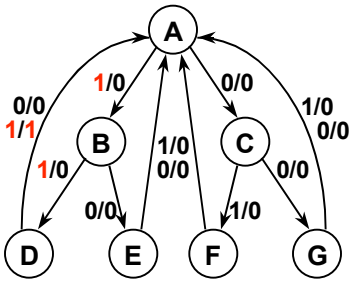
  Sample timing trace:

x	0	1	1		1	1	1		1	0	1		1	1	0		1	1	1
z	0	0	0		0	0	1		0	0	0		0	0	0		0	0	1

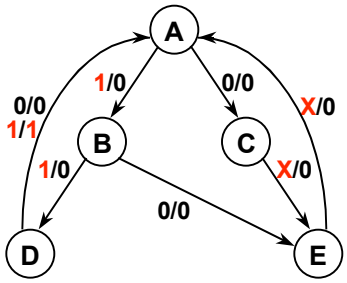
Example 4

- Design a Mealy system where the inputs are considered in blocks of three. The output is 1 iff the input is 1 for all three inputs in a block.

Solution 1:



Solution 2:



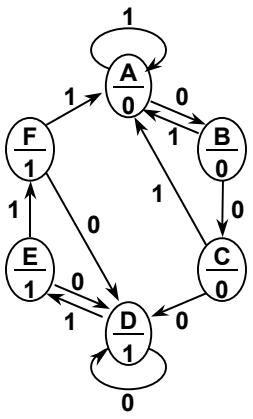
Example 5

- Design a Moore system whose output is 1 iff three consecutive 0 inputs occurred more recently than three consecutive 1 inputs.

Sample timing trace:

x	1	1	1	0	0	1	0	1	1	1	0
z	?	?	?	0	0	0	0	0	0	0	0

x	0	1	0	0	0	0	1	1	1	1	0
z	0	0	0	0	0	1	1	1	1	0	0



13