 Copyright © 1999 The Regents of the University of California

0 Admin
1 Recap
2 State minimization
3 FSM partitioning – Hierarchical

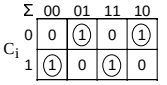
 Copyright © 1999 The Regents of the University of California

1 RECAP

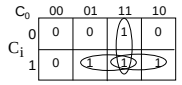
K-map: $AB + A\overline{B} = A$ Property? – adjacency

Example

A	B	C_i	Σ	C_0
0	0	0	0	0
0	0	1	1	0
	1	0	1	0
	1	1	0	1
1	0	0	1	0
	0	1	0	1
	1	0	0	1
	1	1	1	1



$\Sigma = A \oplus B \oplus C_i$



$C_0 = A \cdot B + C_i \cdot (A \oplus B)$
$+ C_i \cdot \overline{A}B + C_i \cdot A\overline{B}$

State assignment – done by machine
Minimization/partitioning more important

 Copyright © 1999 The Regents of the University of California

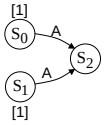
2 STATE MINIMIZATION

Goal: To remove redundant states.

Why?:
- Reduce number of FF
- Logic simplification
- Improve testability
- Reduce cost

Two states are equivalent iff:
Rule 1. ∀ input combinations, outputs are the same
Rule 2. ∀ input combinations, go to equivalent next state

Trivial example:



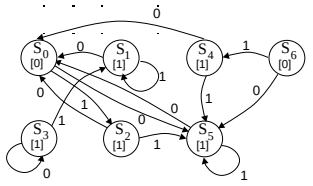
S_0 and S_1 are equivalent

 Copyright © 1999 The Regents of the University of California

2 STATE MINIMIZATION (cont.)

2.1 Implication Chart Method: pairwise comparison of all states for equivalence

Example (Ex. 9.1 in text):



 Copyright © 1999 The Regents of the University of California

2 STATE MINIMIZATION (cont.)

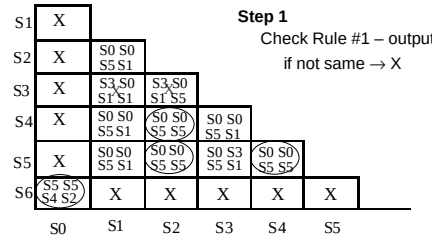
2.2 State table (Moore Machine)

PS	IN	NS	OUT
S0	0	S5	0
	1	S2	
S1	0	S0	1
	1	S1	
S2	0	S0	1
	1	S5	
S3	0	S3	1
	1	S1	
S4	0	S0	1
	1	S5	
S5	0	S0	1
	1	S5	
S6	0	S5	0
	1	S4	

 Copyright © 1999 The Regents of the University of California

2 STATE MINIMIZATION (cont.)

2.3 Implication table – Pass 1



Step 1
Check Rule #1 – outputs same
if not same → X

Step 2
List next states

 Copyright © 1999 The Regents of the University of California

2 STATE MINIMIZATION (cont.)

2.3 Implication table – Pass 1 (cont.)

Step 3

X out next states that have unequal outputs (already X'd)

$S_2 \stackrel{?}{=} S_1$

S_2	S_1
S_0	S_0
S_5	S_1

perhaps =

$S_3 \stackrel{?}{=} S_1$

S_3	S_1
S_3	S_0
S_1	S_1

$S_0 \neq S_3$ (already has X)

"X out" S_3/S_1, S_3/S_2, S_4/S_3, S_5/S_3

Step 4 – Equal States

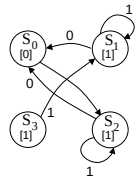
Equal state: $S_4 = S_5 = S_2$, $S_0 = S_6$

Can't tell yet: $S_1 - S_4$, $S_1 - S_5$

 Copyright © 1999 The Regents of the University of California

2 STATE MINIMIZATION (cont.)

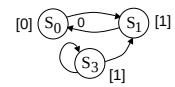
2.4 Minimized State Diagram – Pass 2 Step 1✓ Step 2✓ Step 3 X



	S0	S1	S2
S1	x		
S2	x	S0 S0 S2 S1	
S3	x	S3 S0 S1 S1	S3 S0 S1 S2

$S_2 = S_1$

Final result:

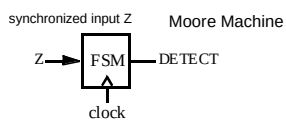


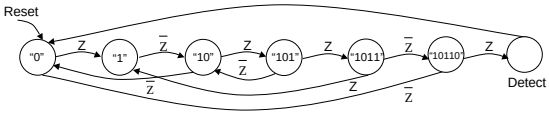
 Copyright © 1999 The Regents of the University of California

3 HIERARCHICAL FSM: Divide and Conquer

Motivation: 2 simple FSM instead of one more complex

Example: Detect 101101 sequence (non-overlapping)

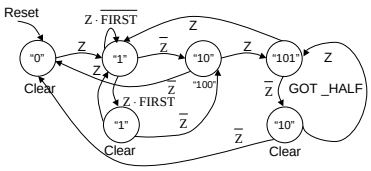


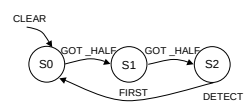


 Copyright © 1999 The Regents of the University of California

3 HIERARCHICAL FSM (cont.)

Two "simpler" machines (Moore/Moore) a bit hairy

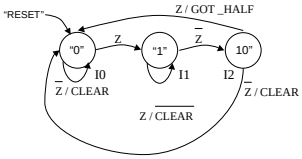


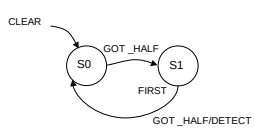


 Copyright © 1999 The Regents of the University of California

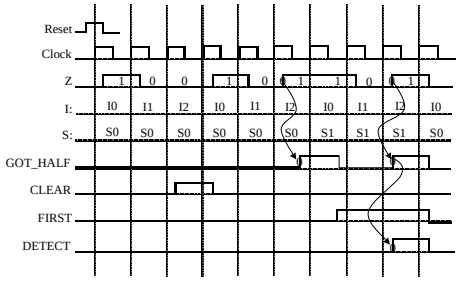
3 HIERARCHICAL FSM (cont.)

Two simpler machines – Mealy/Mealy





 Copyright © 1999 The Regents of the University of California

3 HIERARCHICAL FSM (cont.)

Check with timing diagram – 100101101