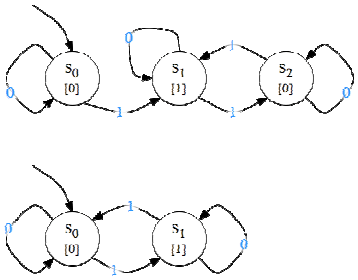
Lecture 20

- FSM simplification

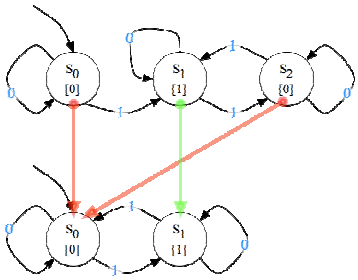
FSM minimization

- Two simple FSMs for odd parity checking



Collapsing states

- Can collapse S_0 and S_2 into one state



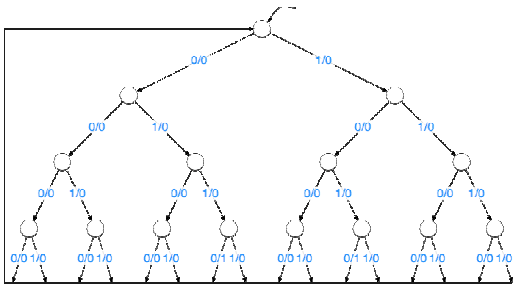
FSM minimization

- Row matching
  - Easier to do by hand
  - Misses minimization opportunities
- Implication chart
  - Guaranteed to find the most reduced FSM
  - More complicated algorithm (but still relatively easy to write a program to do it)

Simple problem

- Design a Mealy machine with a single bit input and a single bit output. The machine should output a 0, except once every four cycles, if the previous four inputs matched one of two patterns (0110, 1010)
- Example input/output trace:
  in: 0010 0110 1100 1010 0011 …
  out: 0000 0001 0000 0001 0000 …

… and a simple solution



Find matching rows

		Next State		Output	
Input Sequence	Present State	X=0	X=1	X=0	X=1
Reset	S_0	S_1	S_2	0	0
0	S_1	S_3	S_4	0	0
1	S_2	S_5	S_6	0	0
00	S_3	S_7	S_8	0	0
01	S_4	S_9	S_{10}	0	0
10	S_5	S_{11}	S_{12}	0	0
11	S_6	S_{13}	S_{14}	0	0
000	S_7	S_0	S_0	0	0
001	S_8	S_0	S_0	0	0
010	S_9	S_0	S_0	0	0
011	S_{10}	S_0	S_0	1	0
100	S_{11}	S_0	S_0	0	0
101	S_{12}	S_0	S_0	1	0
110	S_{13}	S_0	S_0	0	0
111	S_{14}	S_0	S_0	0	0

Find matching rows

		Next State		Output	
Input Sequence	Present State	X=0	X=1	X=0	X=1
Reset	S_0	S_1	S_2	0	0
0	S_1	S_3	S_4	0	0
1	S_2	S_5	S_6	0	0
00	S_3	S_7	S_8	0	0
01	S_4	S_9	S_{10}	0	0
10	S_5	S_{11}	S_{12}	0	0
11	S_6	S_{13}	S_{14}	0	0
000	S_7	S_0	S_0	0	0
001	S_8	S_0	S_0	0	0
010	S_9	S_0	S_0	0	0
011	S_{10}	S_0	S_0	1	0
100	S_{11}	S_0	S_0	0	0
101	S_{12}	S_0	S_0	1	0
110	S_{13}	S_0	S_0	0	0
111	S_{14}	S_0	S_0	0	0

Merge matching rows

		Next State		Output	
Input Sequence	Present State	X=0	X=1	X=0	X=1
Reset	S_0	S_1	S_2	0	0
0	S_1	S_3	S_4	0	0
1	S_2	S_5	S_6	0	0
00	S_3	S_7	S_8	0	0
01	S_4	S_9	$S_{10'}$	0	0
10	S_5	S_{11}	$S_{10'}$	0	0
11	S_6	S_{13}	S_{14}	0	0
000	S_7	S_0	S_0	0	0
001	S_8	S_0	S_0	0	0
010	S_9	S_0	S_0	0	0
011 or 101	$S_{10'}$	S_0	S_0	1	0
100	S_{11}	S_0	S_0	0	0
110	S_{13}	S_0	S_0	0	0
111	S_{14}	S_0	S_0	0	0

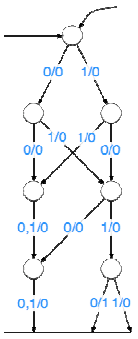
Merge until no more matches

		Next State		Output	
Input Sequence	Present State	X=0	X=1	X=0	X=1
Reset	S_0	S_1	S_2	0	0
0	S_1	S_3	S_4	0	0
1	S_2	S_5	S_6	0	0
00	S_3	$S_{7'}$	$S_{7'}$	0	0
01	S_4	$S_{7'}$	$S_{10'}$	0	0
10	S_5	$S_{7'}$	$S_{10'}$	0	0
11	S_6	$S_{7'}$	$S_{7'}$	0	0
Not (011 or 101)	$S_{7'}$	S_0	S_0	0	0
011 or 101	$S_{10'}$	S_0	S_0	1	0

Final state transition table

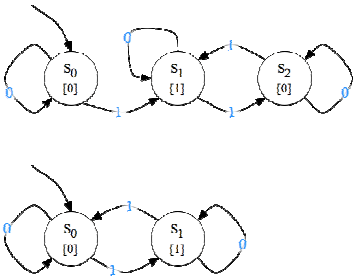
		Next State		Output	
Input Sequence	Present State	X=0	X=1	X=0	X=1
Reset	S_0	S_1	S_2	0	0
0	S_1	$S_{3'}$	$S_{4'}$	0	0
1	S_2	$S_{4'}$	$S_{3'}$	0	0
00 or 11	$S_{3'}$	$S_{7'}$	$S_{7'}$	0	0
01 or 10	$S_{4'}$	$S_{7'}$	$S_{10'}$	0	0
Not (011 or 101)	$S_{7'}$	S_0	S_0	0	0
011 or 101	$S_{10'}$	S_0	S_0	1	0

Final FSM

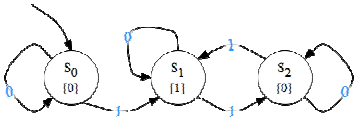


Row matching not optimal

- Again, the two FSMs for odd parity checking



No rows match!



Present State	Next State		Output
	X=0	X=1	
S_0	S_0	S_1	0
S_1	S_1	S_2	1
S_2	S_2	S_1	0