Sequential logic design

Vikas Dhiman for ECE275

November 30, 2022

1 Objectives

1. Perform a state assignment using the guideline method
2. Reduce the number of states in a state table using row reduction and implication tables
3. Partition a system into multiple state machines

2 Full procedure for designing sequential logic circuit

1. Convert the word problem to a state transition diagram. Let the states be $S_0, S_1, S_2, \ldots, S_n$.
2. Draw state transition table with named states. For example,

Present State	Next State		Outputs	
	X = 0	X = 1	X=0	X=1
S_0	S_1	S_2	0	0
S_1	S_2	S_0	0	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

3. State reduction step: Reduce the number of required states to a minimum. Eliminate unnecessary or duplicate states.
4. State assignment step: Assign each state a binary representation. For example,

State name	State assignments ($Q_2Q_1Q_0$)
S_0	000
S_1	001
$\vdots$	$\vdots$

5. Draw State assigned transition table. For example,

Inputs (X_1X_0)	Present State (Q_1Q_0)	Next State ($Q_1^+Q_0^+$)	Outputs (Z_1Z_0)
0 0	00	01	0 0
0 0	01	10	0 0
$\vdots$ $\vdots$	$\vdots$	$\vdots$	$\vdots$ $\vdots$

   (a) Use excitation tables to find truth tables for the combinational circuits. For example, the excitation table for J-K ff is

Q	Q^+	J	K
0	0	0	d
0	1	1	d
1	0	d	1
1	1	d	0

3 State assignment by guideline method [1, Section 8.2.5]

3.1 State Maps

Example 1. *Draw a state map for a sequential assignment of the states*

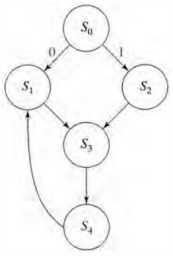


Figure 8.27 Five-state finite state machine.

3.2 Guideline method

Guideline method states that the following states should be adjacent in the state map according the following priorities:

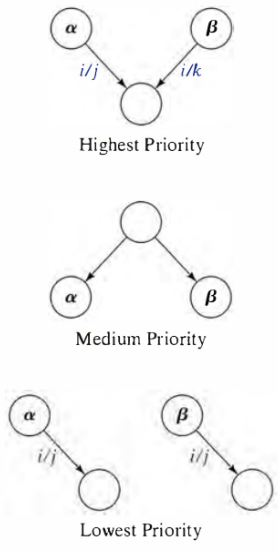


Figure 8.29 Adjacent assignment priorities.

Example 2. *A state transition table is given. Find optimal state assignment by using the guideline method.*

Input Sequence	Present State	Next State $X=0$	Next State $X=1$	Output $X=0$	Output $X=1$
Reset	S_0	S_1'	S_1'	0	0
0 or 1	S_1'	S_3'	S_4'	0	0
00 or 10	S_3'	S_0	S_0	0	0
01 or 11	S_4'	S_0	S_0	1	0

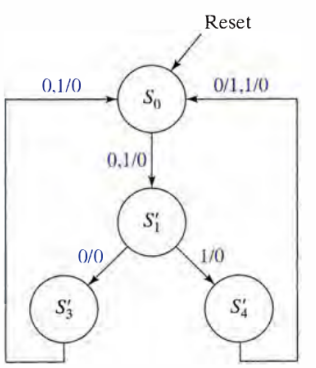
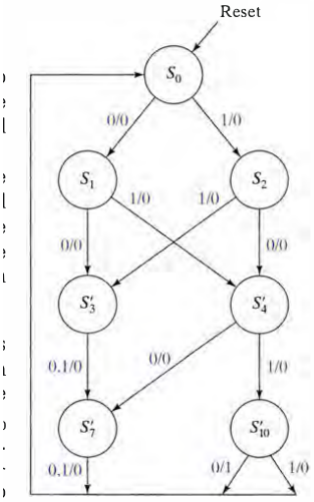


Figure 8.30 Reduced state diagram for 3-bit sequence detector.

Example 3. *Draw a Mealy FSM for detecting binary string 0110 or 1010. The machine returns to the reset state after each and every 4-bit sequence. Draw the state transition diagram on your own as practice problem. The state transition diagram is given here. Find optimal state assignment by using the guideline method.*



4 State reduction by implication chart

Example 4. *Design a Mealy FSM for detecting binary sequence 010 or 0110. The machine returns to reset state after each and every 3-bit sequence. For now the state transition table is given. Reduce the following state transition table*

Input Sequence	Present State	Next State		Output	
		$X=0$	$X=1$	$X=0$	$X=1$
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_0	S_0	0	0
01	S_4	S_0	S_0	1	0
10	S_5	S_0	S_0	0	0
11	S_6	S_0	S_0	1	0

4.1 Implication chart Summary

The algorithms for state reduction using the implication chart method consists of the following steps

1. Construct the implication chart, consisting of one square for each possible combination of states taken two at a time.
2. For each square labeled by states S_i and S_j, if the outputs of the states differ, mark the square with an X; the states are not equivalent. Otherwise, they may be equivalent. Within the square write implied pairs of equivalent next states for all input combinations.
3. Systematically advance through the squares of the implication chart. If the square labeled by states S_i, S_j contains an implied pair S_m, S_n and square S_m, S_n is marked with an X, then mark S_i, S_j with an X. Since S_m, S_n are not equivalent, neither are S_i, S_j.
4. Continue executing Step 3 until no new squares are marked with an X.
5. For each remaining unmarked square S_i, S_j, we can conclude that S_i, S_j are equivalent.

References

[1] Randy Katz and Gaetano Barriello. *Contemporary Logic Design.* Prentice Hall, 2004.

Ask for reference PDF if you need it.