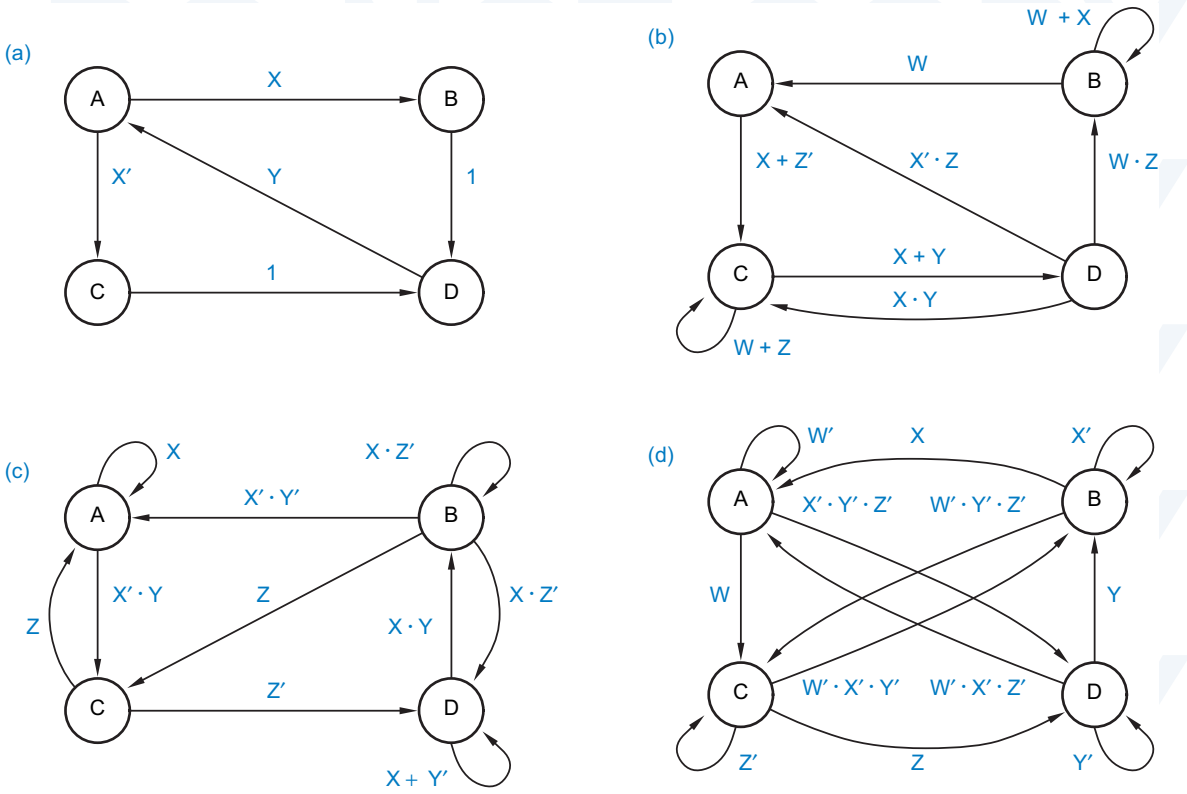


Figure X7.20

7.21 Synthesize a circuit for the state diagram of Figure 7-64 using six variables to encode the state, where the LA–LC and RA–RC outputs equal the state variables themselves. Write a transition list, a transition equation for each state variable as a sum of p-terms, and simplified transition/excitation equations for a realization using D flip-flops. Draw a circuit diagram using SSI and MSI components.

7.22 Starting with the transition list in Table 7-18, find a minimal sum-of-products expression for Q2*, assuming that the next states for the unused states are true don't-cares.

7.23 Modify the state diagram of Figure 7-64 so that the machine goes into hazard mode immediately if LEFT and RIGHT are asserted simultaneously during a turn. Write the corresponding transition list.

Exercises

7.24 Explain how metastability occurs in a D latch when the setup and hold times are not met, analyzing the behavior of the feedback loop inside the latch.

7.25 What is the minimum setup time of a pulse-triggered flip-flop such as a master/slave J-K or S-R flip-flop? (*Hint:* It depends on certain characteristics of the clock.)

7.26 Describe a situation, other than the metastable state, in which the Q and /Q outputs of a 74x74 edge-triggered D flip-flop may be noncomplementary for an arbitrarily long time.

7.27 Compare the circuit in Figure 7.27 with the D latch in Figure 7-12. Prove that the circuits function identically. In what way is Figure 7.27, which is used in some commercial D latches, better?

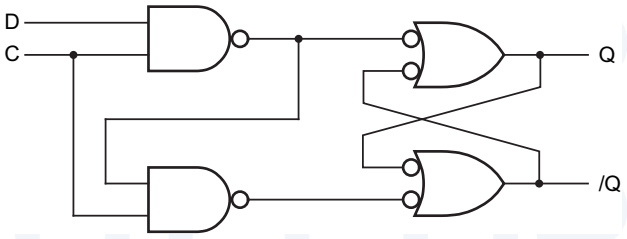


Figure X7.27

7.28 Suppose that a clocked synchronous state machine with the structure of Figure 7-35 is designed using D latches with active-high C inputs as storage elements. For proper next-state operation, what relationships must be satisfied among the following timing parameters?

t_{Fmin}, t_{Fmax}	Minimum and maximum propagation delay of the next-state logic.
t_{CQmin}, t_{CQmax}	Minimum and maximum clock-to-output delay of a D latch.
t_{DQmin}, t_{DQmax}	Minimum and maximum data-to-output delay of a D latch.
t_{setup}, t_{hold}	Setup and hold times of a D latch.
t_H, t_L	Clock HIGH and LOW times.

7.29 Redesign the state machine in Drill 7.9 using just three inverting gates—NAND or NOR—and no inverters.

7.30 Draw a state diagram for a clocked synchronous state machine with two inputs, INIT and X, and one Moore-type output Z. As long as INIT is asserted, Z is continuously 0. Once INIT is negated, Z should remain 0 until X has been 0 for two successive ticks and 1 for two successive ticks, regardless of the order of occurrence. Then Z should go to 1 and remain 1 until INIT is asserted again. Your state diagram should be neatly drawn and planar (no crossed lines). (*Hint:* No more than ten states are required).

7.31 Design a clocked synchronous state machine that checks a serial data line for even parity. The circuit should have two inputs, SYNC and DATA, in addition to CLOCK, and one Moore-type output, ERROR. Devise a state/output table that does the job using just four states, and include a description of each state's meaning in the table. Choose a 2-bit state assignment, write transition and excitation

equations, and draw the logic diagram. Your circuit may use D flip-flops, J-K flip-flops, or one of each.

7.32 Design a clocked synchronous state machine with the state/output table shown in Table 7.32, using D flip-flops. Use two state variables, Q1 Q2, with the state assignment A = 00, B = 01, C = 11, D = 10.

Table X7.32

	X		
S	0	1	Z
A	B	D	0
B	C	B	0
C	B	A	1
D	B	C	0
	S*		

7.33 Repeat Exercise 7.32 using J-K flip-flops.

7.34 Write a new transition table and derive minimal-cost excitation and output equations for the state table in Table 7-6 using the "simplest" state assignment in Table 7-7 and D flip-flops. Compare the cost of your excitation and output logic (when realized with a two-level AND-OR circuit) with the circuit in Figure 7-54.

7.35 Repeat Exercise 7.34 using the "almost one-hot" state assignment in Table 7-7.

7.36 Suppose that the state machine in Figure 7-54 is to be built using 74LS74 D flip-flops. What signals should be applied to the flip-flop preset and clear inputs?

7.37 Write new transition and excitation tables and derive minimal-cost excitation and output equations for the state table in Table 7-6 using the "simplest" state assignment in Table 7-7 and J-K flip-flops. Compare the cost of your excitation and output logic (when realized with a two-level AND-OR circuit) with the circuit in Figure 7-56.

7.38 Repeat Exercise 7.37 using the "almost one-hot" state assignment in Table 7-7.

7.39 Construct an application table similar to Table 7-10 for each of the following flip-flop types: (a) S-R; (b) T with enable; (c) D with enable. Discuss the unique problem that you encounter when trying to make the most efficient use of don't-cares with one of these flip-flops.

7.40 Construct a new excitation table and derive minimal-cost excitation and output equations for the state machine of Table 7-8 using T flip-flops with enable inputs (Figure 7-33). Compare the cost of your excitation and output logic (when realized with a two-level AND-OR circuit) with the circuit in Figure 7-54.

7.41 Determine the full 8-state table of the circuit in Figure 7-54. Use the names U1, U2, and U3 for the unused states (001, 010, and 011). Draw a state diagram and explain the behavior of the unused states.

7.42 Repeat Exercise 7.41 for the circuit of Figure 7-56.

7.43 Write a transition table for the nonminimal state table in Figure 7-51(a) that results from assigning the states in binary counting order, INIT–OKA1 = 000–110. Write corresponding excitation equations for D flip-flops, assuming a minimal-cost disposition of the unused state 111. Compare the cost of your equations with the minimal-cost equations for the minimal state table presented in the text.

7.44 Write the application table for a T flip-flop with enable.

7.45 In many applications, the outputs produced by a state machine during or shortly after reset are irrelevant, as long as the machine begins to behave correctly a short time after the reset signal is removed. If this idea is applied to Table 7-6, the INIT state can be removed and only two state variables are needed to code the remaining four states. Redesign the state machine using this idea. Write a new state table, transition table, excitation table for D flip-flops, minimal-cost excitation and output equations, and logic diagram. Compare the cost of the new circuit with that of Figure 7-54.

7.46 Repeat Exercise 7.45 using J-K flip-flops, and use Figure 7-56 to compare cost.

7.47 Redesign the 1s-counting machine of Table 7-12, assigning the states in binary counting order (S0–S3 = 00, 01, 10, 11). Compare the cost of the resulting sum-of-products excitation equations with the ones derived in the text.

7.48 Repeat Exercise 7.47 using J-K flip-flops.

7.49 Repeat Exercise 7.47 using T flip-flops with enable.

7.50 Redesign the combination-lock machine of Table 7-14, assigning coded states in Gray-code order (A–H = 000, 001, 011, 010, 110, 111, 101, 100). Compare the cost of the resulting sum-of-products excitation equations with the ones derived in the text.

7.51 Find a 3-bit state assignment for the combination-lock machine of Table 7-14 that results in less costly excitation equations than the ones derived in the text. (*Hint:* Use the fact that inputs 1–3 are the same as inputs 4–6 in the required input sequence.)

7.52 What changes would be made to the excitation and output equations for the combination-lock machine in Section 7.4.6 as the result of performing a formal multiple-output minimization procedure (Section 4.3.8) on the five functions? You need not construct 31 product maps and go through the whole procedure; you should be able to "eyeball" the excitation and output maps in Section 7.4.6 to see what savings are possible.

7.53 Synthesize a circuit for the ambiguous state diagram in Figure 7-62. Use the state assignment in Table 7-16. Write a transition list, a transition equation for each state variable as a sum of p-terms, and simplified transition/excitation equations for a realization using D flip-flops. Determine the actual next state of the circuit, starting from the IDLE state, for each of the following input combinations on (LEFT, RIGHT, HAZ): (1,0,1), (0,1,1), (1,1,0), (1,1,1). Comment on the machine's behavior in these cases.

7.54 Suppose that for a state SA and an input combination I, an ambiguous state diagram indicates that there are two next states, SB and SC. The actual next state SD for this transition depends on the state machine's realization. If the state machine

is synthesized using the V* = Σp-terms where V* = 1) method to obtain transition/excitation equations for D flip-flops, what is the relationship between the coded states for SB, SC, and SD? Explain.

7.55 Repeat Exercise 7.54, assuming that the machine is synthesized using the V*′ = Σp-terms where V* = 0) method.

7.56 Suppose that for a state SA and an input combination I, an ambiguous state diagram does not define a next state. The actual next state SD for this transition depends on the state machine's realization. Suppose that the state machine is synthesized using the V* = Σp-terms where V* = 1) method to obtain transition/excitation equations for D flip-flops. What coded state is SD? Explain.

7.57 Repeat Exercise 7.56, assuming that the machine is synthesized using the V*′ = Σp-terms where V* = 0) method.

7.58 Given the transition equations for a clocked synchronous state machine that is to be built using master/slave S-R flip-flops, how can the excitation equations for the S and R inputs be derived? (*Hint:* Show that any transition equation, Qi* = expr, can be written in the form Qi* = Qi · expr1 + Qi′ · expr2, and see where that leads.)

7.59 Repeat Exercise 7.58 for J-K flip-flops. How can the "don't-cares" that are possible in a J-K design be specified?

7.60 Draw a logic diagram for the output logic of the guessing-game machine in Table 7-18 using a single 74x139 dual 2-to-4 decoder. (*Hint:* Use active-low outputs.)

7.61 What does the personalized license plate in Figure 7-60 stand for? (*Hint:* It's a computer engineer's version of OTTFFSS.)

7.62 Analyze the feedback sequential circuit in Figure 7-19, assuming that the PR_L and CLR_L inputs are always 1. Derive excitation equations, construct a transition table, and analyze the transition table for critical and noncritical races. Name the states, and write a state/output table and a flow/output table. Show that the flow table performs the same function as Figure 7-85.

7.63 Draw the logic diagram for a circuit that has one feedback loop, but that is *not* a sequential circuit. That is, the circuit's output should be a function of its current input only. In order to prove your case, break the loop and analyze the circuit as if it were a feedback sequential circuit, and demonstrate that the outputs for each input combination do not depend on the "state."

BUT flop
NBUT flop

7.64 A BUT *flop* may be constructed from a single NBUT gate as shown in Figure 7.64. (An NBUT *gate* is simply a BUT gate with inverted outputs; see Exercise 5.31 for the definition of a BUT gate.) Analyze the BUT flop as a feedback sequential circuit and obtain excitation equations, transition table, and flow table. Is this circuit good for anything, or is it a flop?

7.65 Repeat Exercise 7.64 for the BUT flop in Figure 7.65.

7.66 A clever student designed the circuit in Figure 7.66 to create a BUT gate. But the circuit didn't always work correctly. Analyze the circuit and explain why.

7.67 Show that a 4-bit ones'-complement adder with end-around carry is a feedback sequential circuit.

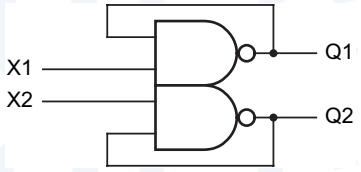


Figure X7.64

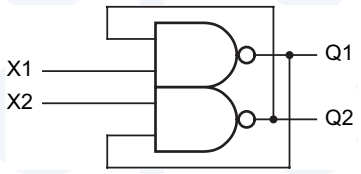


Figure X7.65

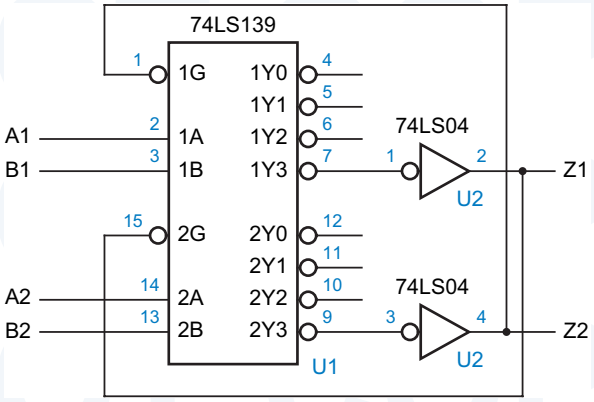


Figure X7.66

7.68 Analyze the feedback sequential circuit in Figure 7.68. Break the feedback loops, write excitation equations, and construct a transition and output table, showing the stable total states. What application might this circuit have?

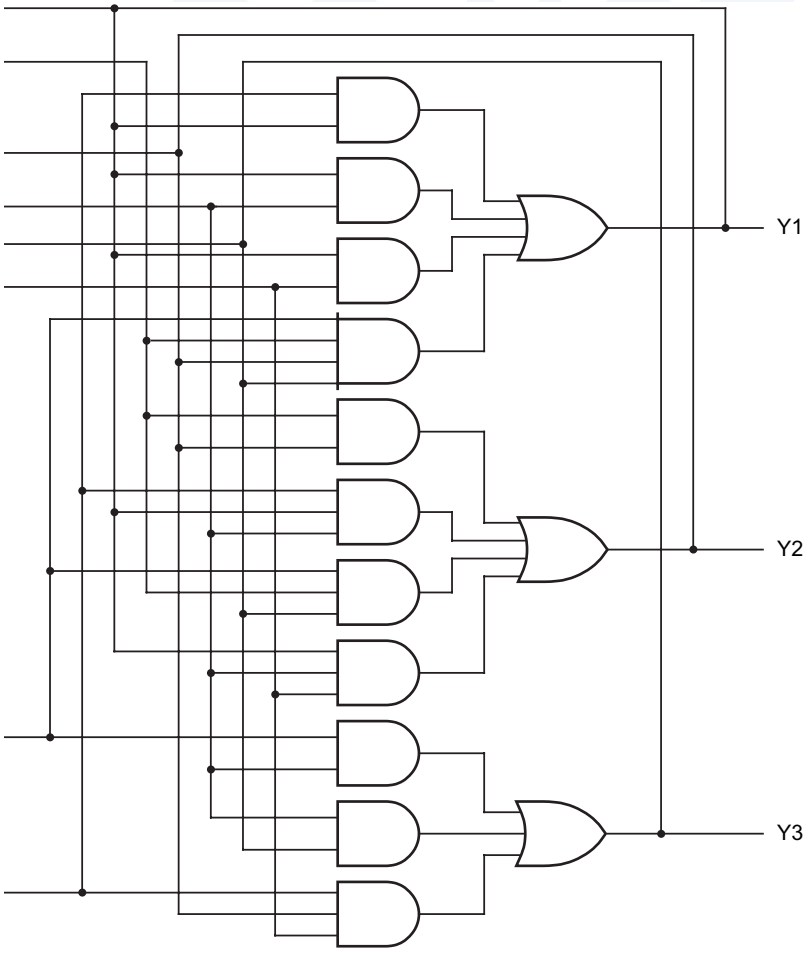
7.69 Complete the analysis of the positive-edge-triggered D flip-flop in Figure 7-86, including transition/output, state/output, and flow/output tables. Show that its behavior is equivalent to that of the D flip-flop in Figure 7-78.

7.70 We claimed in Section 7.10.1 that all single-loop feedback sequential circuits have an excitation equation of the form

$$Q^* = (\text{forcing term}) + (\text{holding term}) \cdot Q$$

Why aren't there any practical circuits whose excitation equation substitutes Q′ for Q above?

7.71 Design a latch with two control inputs, C1 and C2, and three data inputs, D1, D2, and D3. The latch is to be "open" only if both control inputs are 1, and it is to store



a 1 if any of the data inputs is 1. Use hazard-free two-level sum-of-products circuits for the excitation functions.

7.72 Repeat Exercise 7.71, but minimize the number of gates required; the excitation circuits may have multiple levels of logic.

7.73 Redraw the timing diagram in Figure 7-90, showing the internal state variables of the pulse-catching circuit of Figure 7-100, assuming that it starts in state 00.

7.74 The general solution for obtaining a race-free state assignment of 2^n states using 2^{n-1} state variables yields the adjacency diagram shown in Figure 7.74 for the $n = 2$ case. Compare this diagram with Figure 7-97. Which is better, and why?

7.75 Design a fundamental-mode flow table for a pulse-catching circuit similar to the one described in Section 7.10.2, except that the circuit should detect both 0-to-1 and 1-to-0 transitions on P.

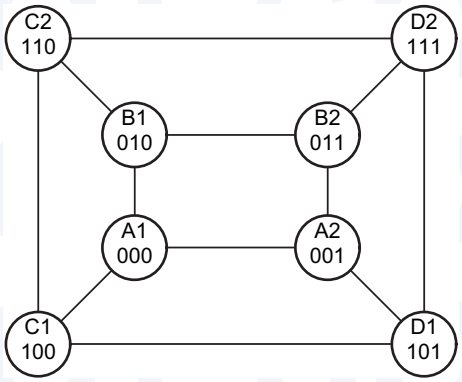


Figure X7.74

7.76 Design a fundamental-mode flow table for a double-edge-triggered D flip-flop, one that samples its inputs and changes its outputs on both edges of the clock signal.

7.77 Design a fundamental-mode flow table for a circuit with two inputs, EN and CLKIN, and a single output, CLKOUT, with the following behavior. A clock period is defined to be the interval between successive rising edges of CLKIN. If EN is asserted during an entire given clock period, then CLKOUT should be "on" during the next clock period; that is, it should be identical to CLKIN. If EN is negated during an entire given clock period, then CLKOUT should be "off" (constant 1) during the next clock period. If EN is both asserted and negated during a given clock period, then CLKOUT should be on in the next period if it had been off, and off if it had been on. After writing the fundamental-mode flow table, reduce it by combining "compatible" states if possible.

7.78 Design a circuit that meets the specifications of Exercise 7.77 using edge-triggered D flip-flops (74LS74) or JK flip-flops (74LS109) and NAND and NOR gates without feedback loops. Give a complete circuit diagram and word description of how your circuit achieves the desired behavior.

7.79 Which of the circuits of the two preceding exercises is (are) subject to metastability, and under what conditions?

7.80 For the flow table in Table 7-36, find an assignment of state variables that avoids all critical races. Additional states may be added as necessary, but use as few state variables as possible. Assign the all-0s combination to state A. Draw the adjacency diagram for the original flow table, and write the modified flow table and another adjacency diagram to support your final state-variable assignment.

7.81 Prove that the fundamental-mode flow table of any flip-flop that samples input(s) and change(s) outputs on the rising edge only of a clock signal CLK contains an essential hazard.

7.82 Locate the essential hazard(s) in the flow table for a positive-edge-triggered D flip-flop, Figure 7-85.

7.83 Identify the essential hazards, if any, in the flow table developed in Exercise 7.76.

Table 7-36

	X Y			
S	00	01	11	10
A	B	C	—	(A)
B	(B)	E	—	(B)
C	F	(C)	—	E
D	(D)	F	—	B
E	D	(E)	—	(E)
F	(F)	(F)	—	A
	S*			

7.84 Identify the essential hazards, if any, in the flow table developed in Exercise 7.77.

7.85 Build a verbal flip-flop—a logical word puzzle that can be answered correctly in either of two ways depending on state. How might such a device be adapted to the political arena?

7.86 Modify the ABEL program in Table 7-27 to use an output-coded state assignment, thereby reducing the total number of PLD outputs required by one.

7.87 Finish writing the test vectors, started in Table 7-35, for the combination-lock state machine of Table 7-31. The complete set of vectors should test all of the state transitions and all of the output values for every state and input combination.