

CHAPTER 6

Other Modes of Operation

In Chapter 1 several classes of sequential circuits are discussed, and here we survey them as well as some others in greater detail. They differ with regard to the circumstances under which input changes are permitted, and in terms of internal structure. Objectives vary from maximizing average speed to minimizing design problems.

6.1 Circuits that Generate Completion Signals

These circuits, referred to in Section 1.6 as Muller circuits (also called *speed-independent* circuits), send back signals to the input source indicating when they are ready to accept new input states. In cases in which the time required to process an input is subject to large variations, dependent perhaps on the particular input or on the internal state, the potential increase in the average input rate is substantial.

Such an approach is possible only when the input source is under the control of the system or when a system-controlled buffer may be placed between the input source and the circuit. When the circuit must respond immediately to uncontrollable input changes, as is the case in many real-time applications, Muller circuits are not feasible. Perhaps their most natural environment would be in a digital computer, which is essentially a closed system after its memory has been loaded with the program and data for a given problem. It is also possible to use the completion-signal philosophy in parts of a system or to govern communications among the major blocks of a system, while using other approaches within certain of its subsystems.

As is made evident later, the same restriction on stray-delay locations that is imposed in Section 4.4 is necessary here, namely, that the delays in the lines be small compared with those in the gate outputs. However, whereas in all our earlier discussions we assume that there are known upper bounds on all stray-delay values (an obvious necessity in computing the minimum allowable spacing between input changes), in this section that assumption is not needed,

and furthermore, delay elements are not used. We assume here that stray delays may be pure *or* inertial, although some workers in this field postulate inertial delays.

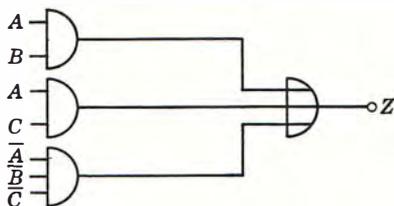
The design of combinational circuits is discussed first, and then the method is extended to include SOC sequential circuits. The final subsection deals with the problem of constructing networks of such systems. It should be noted at the outset that the methods to be presented in this section reflect only one of many approaches and that our treatment of this topic is much less comprehensive than that accorded to the subjects of the preceding chapters.

6.1.1 Combinational Circuits with Completion Signals

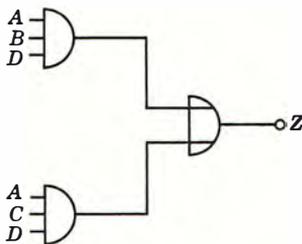
Probably the first case in which the completion-signal concept found application was that of the parallel binary adder, an iterative combinational circuit. It was realized that, depending on the numbers being added, the time required for the outputs to respond varies over a very wide range. When a carry bit must “ripple” through the entire circuit, the delay is a maximum. In cases in which there is no carry, the delay may be less than the maximum by a factor of about n for an n -bit adder. If the input numbers are randomly chosen, the *average* time required varies approximately as $\log_2 (5n/4)$ (a formula given by Hendrickson [HE-1]). However, if the adder does not return some signal indicating when it has completed each computation, then it is necessary to allow the maximum time in every case.

In order to avoid this considerable waste of time, adders have been developed that generate completion signals so that they can process data at their average rate instead of at a rate limited by the worst-case situation. Such circuits may require as many as twice the number of components as a simple adder and are actually slower in operation; nevertheless, in an appropriate environment, they offer significant overall gains in efficiency. The technique employed in such adders [GIL-1] has been generalized to encompass a broad class of iterative circuits [WAI-1], but instead of discussing this technique, we present a method applicable to the realization of *any* combinational function. The two techniques are, however, related to some extent. Until further notice, it is understood that stray delays are confined entirely to gate outputs, line delays being zero.

We began by describing a condition that constitutes the principal obstacle to the successful design of Muller circuits, namely, the *delay hazard*. Consider Fig. 6.1, which depicts a two-stage AND-to-OR circuit realizing the function $\bar{A}\bar{B}\bar{C} + AB + AC$ without logic hazards (see Table 6.1a for a K -map of the function showing the product terms used). The input sequence 011, 111, 101 should produce the output sequence 0, 1, 1. Suppose that we observe both



(a)



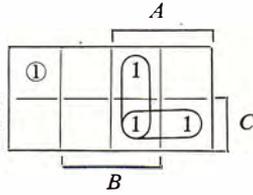
(b)

Figure 6.1 (a) A circuit with a 0101 delay hazard for input changes $011 \rightarrow 111 \rightarrow 101$. (b) A circuit with a 01010 delay hazard for input changes $0111 \rightarrow 1111 \rightarrow 1110$.

the input terminals (A , B , and C), and the output Z in order to determine when the effects of the first input change have ended and the circuit is ready for the second change. One might think that, after Z has changed from 0 to 1 in response to the change in ABC from 011 to 111 (the first change), the circuit is ready for the second change. But if this assumption is acted upon, the result could be a transient error. The initial input change excites both the AB - and the AC -AND gates. Suppose there is a relatively large delay associated with the AC gate. Then the AB gate would go on first and the OR gate might then respond to this signal *before* the AC gate goes on. If we then permit the second input change (to 101), the result might be to turn off the AB gate and the OR gate (hence Z) while the AC gate is still struggling to emit a 1-signal. When this gate finally does go on, Z again switches to 1. Hence the result is a Z -sequence of 0101 instead of 011. (Note that this process is quite similar to that described in Section 4.2 for generating a dynamic-hazard transient after a simultaneous change of several input variables.)

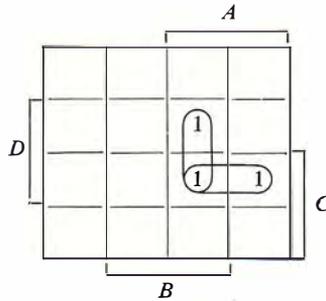
In general, we define a *delay hazard* as a condition in a circuit free of logic hazards such that a sequence of two consecutive input changes $I_1 \rightarrow I_2 \rightarrow I_3$

Table 6.1a



(a) Function realized by Fig. 6.1a.

Table 6.1b



(b) Function realized by Fig. 6.1b.

can produce either of the following output sequences

1. $f(I_1), f(I_2), \overline{f(I_2)}, f(I_3)$, where $f(I_2) = f(I_3)$
2. $f(I_1), f(I_2), f(I_3), f(I_2), f(I_3)$, where $f(I_2) \neq f(I_3)$.

Figure 6.1a and Table 6.1a illustrate the first possibility, and Fig. 6.1b and Table 6.1b illustrate the second (assuming pure stray delays).

If bounds on the stray-delay values are known, then we can wait for a sufficient period after each input change to permit the effects of that change to reach *all* gate outputs before initiating the next change. If such bounds are not known or if we are unwilling to “assume the worst” in each case, then safe operation may require monitoring gates interior to the circuit, as well as the input and output signals. Note, however, that the delay-hazard trouble results from the fact that several of the OR-gate inputs go on during the course of the transitions. Once one of them goes on it is impossible to determine from the output signal whether or not another one is on. Hence we cannot tell if the circuit has *fully* absorbed the input change. If the circuit is such that only *one* input to the output OR gate is energized during a given transition, then we are able to tell from the output when the circuit has reached a stable

state after an input change. A dual situation exists with respect to OR-to-AND circuits. Here the problem arises when more than one AND-gate input can be 0 so that the first input changing to 0 masks the status of the other inputs.

The overall objective is to design the circuit so that the input states may be changed as quickly as the circuit can respond to the changes, and so that the sequence of output states is appropriately related to the sequence of input states. In cases in which there are several output variables, the output state during transitions may, of course, assume values intermediate between consecutive desired states.

Our approach is to force the input to alternate between a state corresponding to a significant input state, called a *data word*, and a fixed *spacer word*. These words are so chosen that:

1. It is easy to determine when the input is a data word and when it is a spacer.
2. No data word is in the region spanned by the spacer and some other data word.

The second condition permits matters to be so arranged that exactly one data word occurs between consecutive spacers. The outputs can be similarly coded, although it is not necessary as we show later.

There are many kinds of codes satisfying the above conditions, only one of which, the *double-rail* method, is presented here. Each signal x is transmitted in double-rail fashion on two lines labeled x_0 and x_1 . When $x = 0$, x_0 is set equal to 1 and x_1 is set to 0. The condition $x = 1$ is represented by $x_0 = 0$ and $x_1 = 1$. The spacer state is coded by setting all x_0 - and x_1 -signals equal to 0, for every variable. Under no conditions are both x_0 and x_1 allowed to equal 1. It is convenient to refer to x_1 as x and x_0 as \bar{x} even though it must be borne in mind that, since both of these signals may simultaneously equal 0, they are not actually complements of one another.

This code clearly meets our criteria:

1. It is easy to determine when the input is a data word (one of each pair of input lines has a 1-signal) and when it is a spacer (*all* input lines have 0-signals).
2. When the input is changing from a spacer to a data word, none of the intermediate states is a data word (assuming no signal that is supposed to stay at 0 temporarily goes on.) The same is true for transitions from data words to spacers.

Consider now a system of the form shown in Fig. 6.2, assuming that the Z_i - and \bar{Z}_i -outputs are generated by two-stage AND-to-OR logic. Suppose that at the start the source emits a spacer (all x_i and \bar{x}_i set to 0) and that this

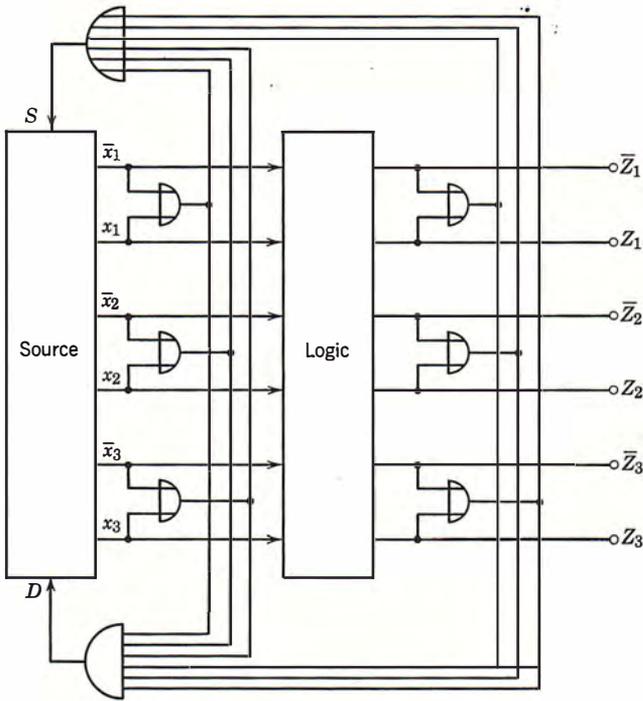


Figure 6.2 Block diagram of a combinational circuit generating return signals.

situation persists until all AND gates and OR gates in the logic block emit 0-signals. Then S , the output of the upper OR gate, and D , the output of the lower AND gate, both become 0. These signals are interpreted by the source as a request for a data word, which it proceeds to supply. The effect then is that 1-signals are generated by certain of the AND gates in the logic block, eventually causing, for each i , either Z_i or \bar{Z}_i to go on. When this process is complete, the output corresponds to a data word and eventually D goes on. Meanwhile S is also turned on. The source interprets $S = D = 1$ as a request for a spacer and complies by turning off all x_i - and \bar{x}_i -signals. This in turn extinguishes all signals in the logic circuit, causing the output to enter the spacer state and switching S and D to 0. When this occurs, a new data word is supplied by the source and the entire process is repeated.

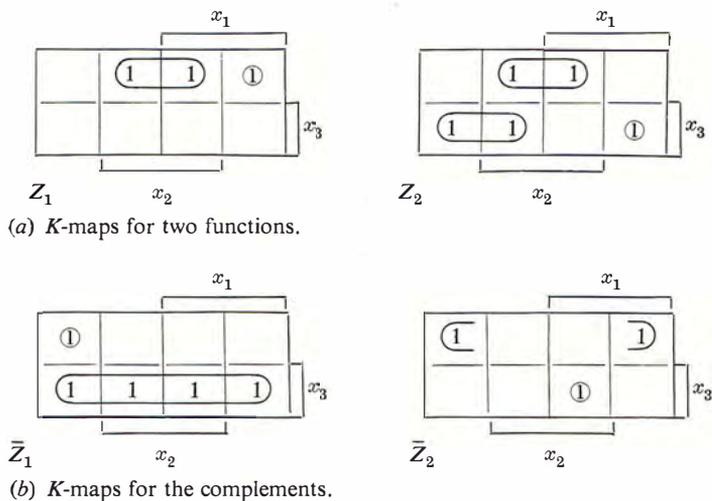
Let us now examine the situation in further detail to determine under what circumstances correct operation is assured. Suppose that, for a particular data input, some Z_i is supposed to go on and that two of the AND gates feeding the Z_i -OR gate go on. If the delay at the output of one of the AND gates is very

long, then the 1-signal may not get through that delay until after the next spacer and the next data input are produced. This condition constitutes a delay hazard and may cause an incorrect output to occur. To avoid it, we impose the restriction that the logic be so designed that, for any data input and for any i , exactly one AND gate in the circuits generating Z_i or \bar{Z}_i be allowed to go on. If this is the case, then when S and D both go from 1 to 0, it must follow that there are no 1's lurking in any stray delay in the logic since the only AND gates that were on must have gone off. When a data input is fed to the logic, exactly one AND gate eventually goes on for each output pair Z_i and \bar{Z}_i . Note that there is no possibility of a false transient occurring anywhere (that is, there are no combinational hazards) since there are no inverters in the circuit. (It is *assumed* that the source operates as specified, never emitting spurious signals.)

The requirement that only one AND gate at a time be allowed to go on for each output pair means that each 1-point of each output function must be covered by exactly one product term. (The same is true for the complement of each function.) This means that in general it is not possible to realize all of the functions with sums of prime implicants, a factor that tends to increase the amount of logic circuitry needed. However, since as pointed out earlier, combinational hazards are not a factor, no special gates are needed to avoid them (see Section 4.2), hence leading to some reduction in the number of gates that would otherwise be required.

As an example, consider the functions mapped in Table 6.2a. The complements are shown in Table 6.2b. Selected product terms are marked in these

Table 6.2



tables, leading to the expressions

$$\begin{aligned} Z_1 &= x_2\bar{x}_3 + x_1\bar{x}_2\bar{x}_3, \\ \bar{Z}_1 &= \bar{x}_1\bar{x}_2\bar{x}_3 + x_3, \\ Z_2 &= x_2\bar{x}_3 + \bar{x}_1x_3 + x_1\bar{x}_2x_3, \\ \bar{Z}_2 &= \bar{x}_2\bar{x}_3 + x_1x_2x_3. \end{aligned}$$

The resulting circuit is shown in Fig. 6.3. It might be instructive for the reader to trace through in detail the operation of the circuit for the typical input sequence

$$001 \rightarrow 110 \rightarrow 111.$$

The specified response is

$$01 \rightarrow 11 \rightarrow 00.$$

In our circuit, the generated input sequence, including spacers, is

$$x_1\bar{x}_1x_2\bar{x}_2x_3\bar{x}_3 = 000000 \rightarrow 010110 \rightarrow 000000 \rightarrow 101001 \rightarrow 000000 \rightarrow 101010,$$

and the response, including spacers, is

$$Z_1\bar{Z}_1Z_2\bar{Z}_2 = 0000 \rightarrow 0110 \rightarrow 0000 \rightarrow 1010 \rightarrow 0000 \rightarrow 0101.$$

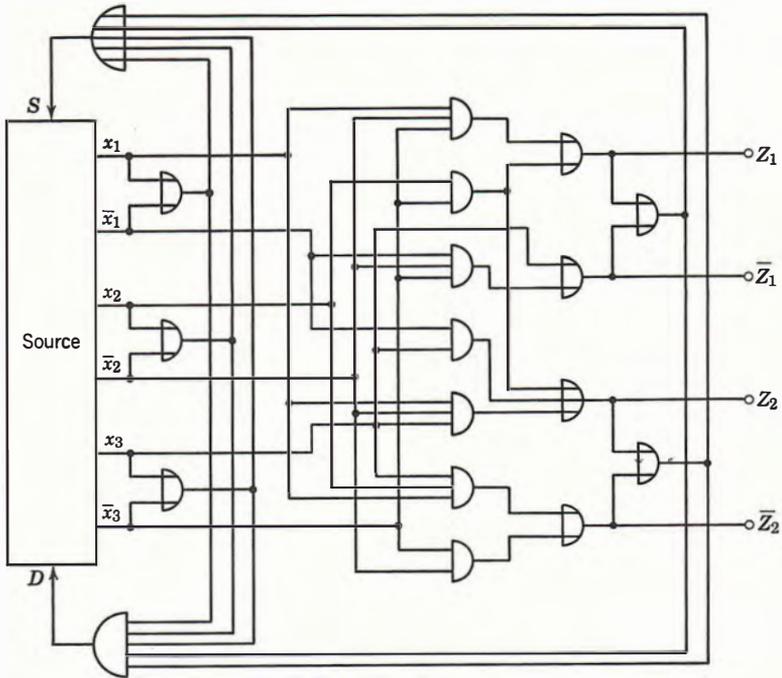


Figure 6.3 Circuit realizing Table 6.2.

It is possible to dispense with the output spacers if we are willing to use FF's at the output terminals. This technique, to be illustrated next, is of particular interest since the basic principle involved is applied shortly to the synthesis of sequential circuits.

As shown in Fig. 6.4, a single SET-RESET-FF is used for each output variable. For each data input either the SET or the RESET terminal of the Z_i -FF is energized, depending upon whether Z_i is specified at 1 or 0 for that input state. The logic block generating these signals is identical to the logic block of Fig. 6.2. When a given FF has received its excitation signal and responded to it, then the OR gate associated with that FF goes to 1. When the input corresponds to a data word (coded in double-rail form as before) and all output FF's have received excitations and responded, then the D -AND gate goes on. An OR gate is used to sense the presence of an input spacer and the fact that all FF excitations have returned to 0, thereby generating the S -signal, which in conjunction with the D -signal, controls the input source as discussed earlier.

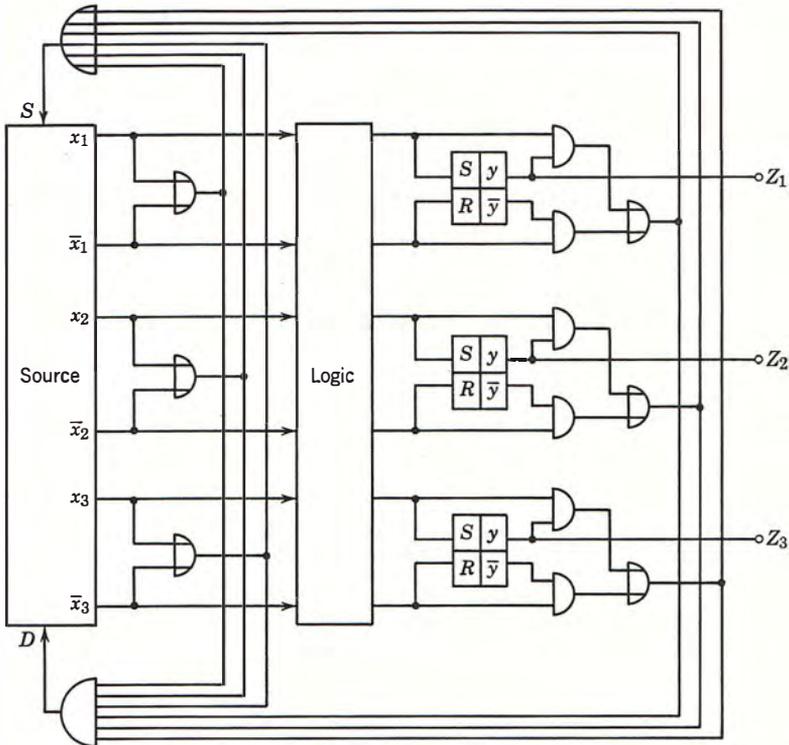


Figure 6.4 Circuit without output spacers.

In the case of a circuit of the form of Fig. 6.4 realizing the functions of Table 6.2, it may be seen that the input sequence

$$x_1x_2x_3 = 001 \rightarrow 110 \rightarrow 111$$

results as before in the source producing

$$x_1\bar{x}_1x_2\bar{x}_2x_3\bar{x}_3 = 000000 \rightarrow 010110 \rightarrow 000000 \rightarrow 101001 \rightarrow 000000 \rightarrow 101010.$$

Now, however, the resulting output Z_1Z_2 is

$$01 \rightarrow 11 \rightarrow 00.$$

Note that when, as in this example, several outputs are required to change for one transition, we can expect intermediate outputs to occur, depending on the stray-delay values. Thus the actual system output might be

$$01 \rightarrow 11 \rightarrow 01 \rightarrow 00 \quad \text{or} \quad 01 \rightarrow 11 \rightarrow 10 \rightarrow 00.$$

Let us now briefly summarize the results thus far. We begin with everything in a quiescent state (all gates emitting 0's); then an input word is fed to the circuit by the source. Regardless of how long the stray delays are or how unevenly distributed they are (recall that line delays are zero), the input is held constant until the outputs have assumed correct values. A signal is then sent back to the source indicating that a spacer signal may now be sent in to clear the circuit. When the spacer signal has been sent and the circuit is clear of all 1-signals, a request for the next relevant input state is sent back to the source.

Thus far it has been assumed that all line delays are zero. Now let us examine the extent to which this assumption can be relaxed. When nonzero delays are allowed in connecting wires (or in gate *inputs*—see Subsection 4.3), a signal change at a node (which may be an input terminal or a gate output terminal) that directly feeds several different points may affect these points at different times. This can lead to malfunction if any input change can occur and penetrate to a different terminal of a gate that is still receiving, at other terminals, signals derived from a previous input state. Such a situation can always be avoided if the line delay between connected pairs of nodes never exceeds the delay through the two stages of logic that generate the *S*- and *D*-signals. A conservative rule is that the largest line delay should never exceed half of the smallest gate delay. The relationship involved resembles that discussed in connection with the delayfree realizations of Section 4.4.

A very useful characteristic of the circuits we are discussing is that a wide class of circuit faults cause a prompt halt in system operation. Suppose that, at some node in one of these circuits, the signal becomes fixed at 0 or at 1. When, in the course of normal operation, the signal at this node is supposed

to assume the opposite value, the effect is as if an infinitely large delay existed in the output of the gate (or possibly the source) that generates the signal at this node. This would freeze the input at the existing state. Thus, for example, if a 1-signal became permanently established at any node in Fig. 6.3, the S -signal would remain fixed at 1, and so the circuit would be unable to request the next data input. If some node became "stuck at 0," then as soon as an input state occurred for which a 1 at this node became essential in order to produce a 1-signal at some Z_i or \bar{Z}_i , the latter terminal would also become stuck at 0 and hence the next spacer input would never be requested. Clearly such behavior is a boon to a maintenance man.

We close this section by pointing out that there is no known method for realizing a source circuit with elements having unbounded stray delays. Such circuits are not difficult to design if components with bounded delays are available.

6.1.2 Sequential Circuits with Completion Signals

The concepts involved in the combinational-circuit model depicted in Fig. 6.4 can serve as the basis for synthesizing SOC sequential functions by simply feeding back to the input side of the logic block the outputs of some of the FF's as shown in Fig. 6.5. These y -FF's then serve as state devices; whereas the others, the Z -FF's, remain as output devices. The key problem is to generate satisfactory row assignments.

If an STT assignment is used, then, after an input change, the termination of any resulting change in the internal state is indicated unambiguously when all of the FF's are in states corresponding to their excitations (the principle exploited in Fig. 6.4). It remains, however, to restrict the class of row assignments further in order to avoid delay hazards.

As is the case for combinational-circuit synthesis, the Z_i or Y_i circuits in the logic block should be so designed that no more than one of the AND gates feeding any one of the output OR gates is 1 during the course of any transition. This means that not only must every 1-point in each Z - and Y -function be covered by exactly one product term, but in addition if there is an $i \rightarrow j$ transition in an input column in which Y_k (or Z_k) has 1-points for rows i and j , then a *single* product term should cover both of these 1-points in the Y_k (or Z_k) realization. But, as pointed out in Section 3.3, the Liu assignment has the property that, in each input column, a unique y -subcube is associated with each destination set, and this subcube is disjoint from all of those associated with the other destination sets of the same column. Since the Y -excitations are constant within each destination set, it follows that, for each Y -function, a set of disjoint product terms can cover the 1-points of the function in such a manner that, for any transition within that column, at most

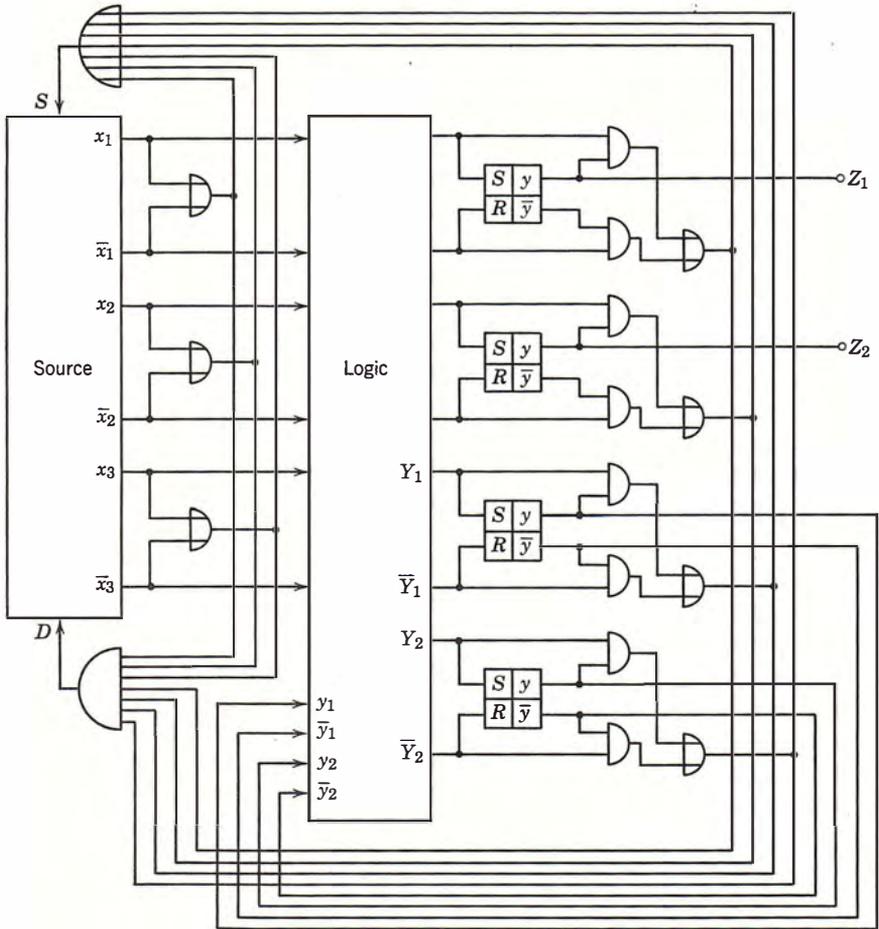


Figure 6.5 Sequential-circuit model.

one of the product terms is ever on. Since the Z -functions are constant within each destination set, they are also realizable in a similar manner.

Thus a satisfactory design is possible in the form of Fig. 6.5 provided that we use a Liu assignment and realize the Y - and Z -functions with maximal subcubes *within* each column, but with no multiple coverage of any 1-point. As is the case with the combinational circuits of the form of Fig. 6.4, it is necessary to generate Z_i -, \bar{Z}_i -, Y_j -, and \bar{Y}_j -signals for every i and j and for every data input. When the input spacer is on, all Y -, \bar{Y} -, and Z -, and \bar{Z} -signals are specified to be 0.

As an example, consider Table 6.3 a flow matrix with a Liu assignment.

Table 6.3 Flow Matrix with Liu Assignment

	x_1x_2			y_1	y_2	y_3
	00	01	11			
1	●, 0	2, 0	①, 0	0	0	0
2	3, 0	②, 0	②, 0	1	0	0
3	③, 0	5, 1	4, 0	1	0	1
4	1, 0	5, 1	④, 0	0	0	1
5	⑤, 1	⑤, 1	4, 0	0	1	1

The FF excitation expressions are derived by setting $S = Y$ and $R = \bar{Y}$, using product terms covering maximal subcubes within each column. Within each expression, the terms are chosen so as to be mutually disjoint.

$$S_1 = \bar{x}_2y_1 + \bar{x}_1x_2\bar{y}_3 + x_1x_2y_1\bar{y}_3,$$

$$R_1 = \bar{x}_1\bar{x}_2\bar{y}_1 + \bar{x}_1x_2y_3 + x_1\bar{y}_1 + x_1y_1y_3,$$

$$S_2 = \bar{x}_2y_2 + \bar{x}_1x_2y_3,$$

$$R_2 = \bar{x}_1\bar{x}_2\bar{y}_2 + \bar{x}_1x_2\bar{y}_3 + x_1,$$

$$S_3 = \bar{x}_2y_1\bar{y}_2 + \bar{x}_2y_2 + x_2y_3,$$

$$R_3 = \bar{x}_2\bar{y}_1\bar{y}_2 + x_2\bar{y}_3,$$

$$Z = \bar{x}_2y_2 + \bar{x}_1x_2y_3,$$

$$\bar{Z} = \bar{x}_1\bar{x}_2\bar{y}_2 + \bar{x}_1x_2\bar{y}_3 + x_1.$$

The above approach can be modified if desired to produce output changes separated by spacers analogously to the system for combinational circuits shown in Fig. 6.2. The modification can be accomplished by omitting the output FF's, and generating Z - and \bar{Z} -signals as before with OR gates to indicate when each output signal has been produced (see Fig. 6.6).

Combinational hazards and sequential hazards are no problem in circuits designed as described above, and the same "error-stop" feature described in the last subsection is also a property of these circuits.

6.1.3 Networks of Circuits

In a complex digital system, it may be desirable to interconnect a number of combinational and sequential circuits of the type treated in the two preceding subsections. Such connections necessitate the modification of the circuits to provide storage for their output states and a means for responding to requests for spacers or new outputs.

A simple chain of circuits is shown in Fig. 6.7, in which the leftmost circuit receives inputs from a source that it controls with S - and D -signals as discussed earlier. Each circuit interior to the chain processes data received from the circuit on its left and passes on the results as inputs to the circuit on its

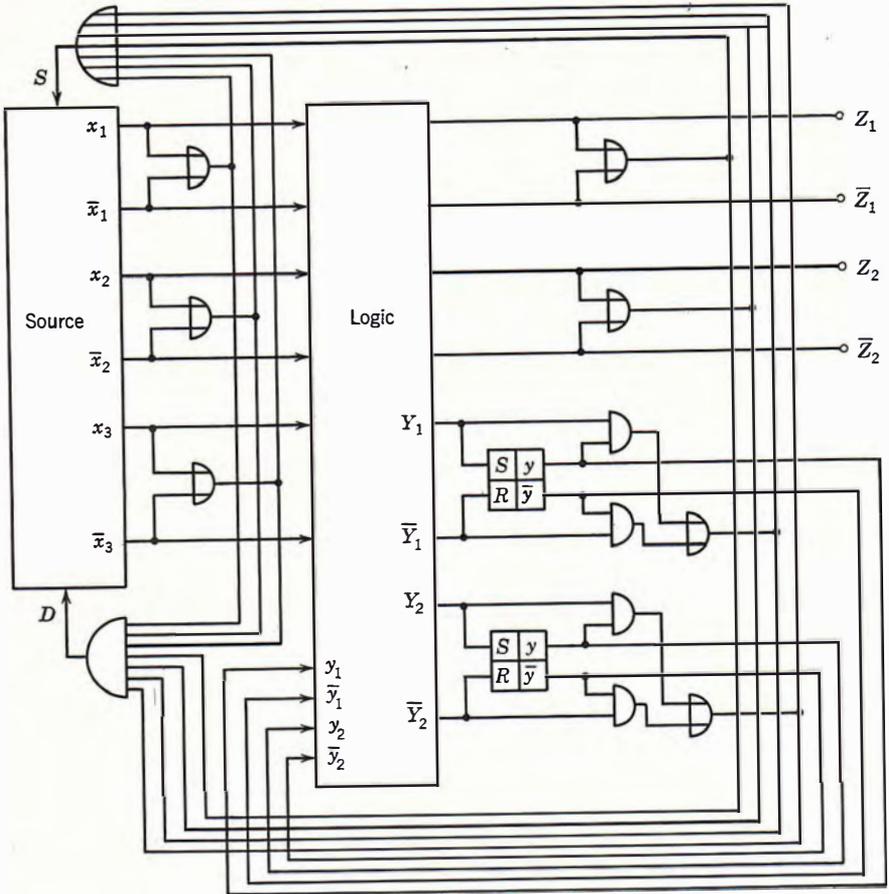


Figure 6.6 Sequential-circuit model with spacer output.

right. When the circuit on the right requests a spacer or a data input, the request is relayed to the circuit on the left. The receptor on the right end of the chain absorbs the overall result of the system and calls for spacers and data according to its own operating rate. We may think of the receptor as an output device such as a printer, and the source as a buffer memory.

A circuit requests a spacer (or data) when its output is data (or a spacer). This request is acted upon when the circuit to its left is emitting a spacer (or data) *and* the circuit to its right is requesting a spacer (or data). Thus, in a long chain, the effects of several different input states can appear at different circuits buffered by circuits with spacer outputs. In no case can the results of a later input overtake and thus garble the results of an earlier input.

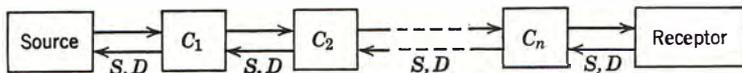


Figure 6.7 A cascade of circuits.

A combinational-circuit model suitable for inclusion in such a chain is shown in Fig. 6.8 which is based on Fig. 6.2 with FF's added for each Z_i and \bar{Z}_i . When the $(i + 1)$ st module requests a spacer ($SD = 10$) and the input from the $(i - 1)$ st module is a spacer, the upper AND gate is activated, causing all of the i th module output FF's to be reset, thus generating an output spacer. When this has occurred (note that the \bar{y} -outputs cannot go on until the S -signals go to 0), the lowest AND gate is energized, resetting the request FF Q , which then sends a request for data ($SD = 01$) signal to the $(i - 1)$ st module.

The receipt of a data request ($SD = 01$) from the right activates the logic block (the D -signal is an input to each AND gate in that block), and if the Q -FF is in the RESET state, then eventually a data input will be fed to the logic from the left, setting various i -module FF's. When one member of each pair of Z_i -FF's reaches the SET state, the AND gate activating the SET terminal of the Q -FF goes on, causing a spacer request to go to the $(i - 1)$ st module. Thus a sequence of inputs from the left is processed and the results transmitted to the right on request, with spacers intervening.

Next we consider how the sequential-circuit configuration of Fig. 6.6 can be modified so as to be compatible with the combinational circuits just discussed in systems such as that shown in Fig. 6.7. What is needed (see Fig. 6.9) is a pair of FF's for each output variable with the circuits needed to detect spacer outputs (as in Fig. 6.8.) A FF is used to record whether a spacer or data output occurred last, and another FF indicates whether or not all of the internal state FF's are both excited and stable (that is, for each FF, $S = 1$ and the FF is set, or $R = 1$ and the FF is reset). These two FF's control the request FF Q , which is set whenever both are set and reset whenever both are reset. Spacer or data inputs are requested from the left when Q is in the set or reset states, respectively. The S -signal from the right controls the resetting of the output FF's, and the D -signal controls all of the AND gates in the logic as in the case of the circuit of Fig. 6.8.

In cases in which a module feeds several other modules, the S - and D -signals from *all* of these modules must feed the AND gates, so that a change between spacer and data in the output of a module can occur only when requested by all of the successor modules.

The techniques described above indicate possible solutions to the problems discussed. In order to obtain speed-independent operation (that is, circuitry that functions properly regardless of the relative and absolute operating

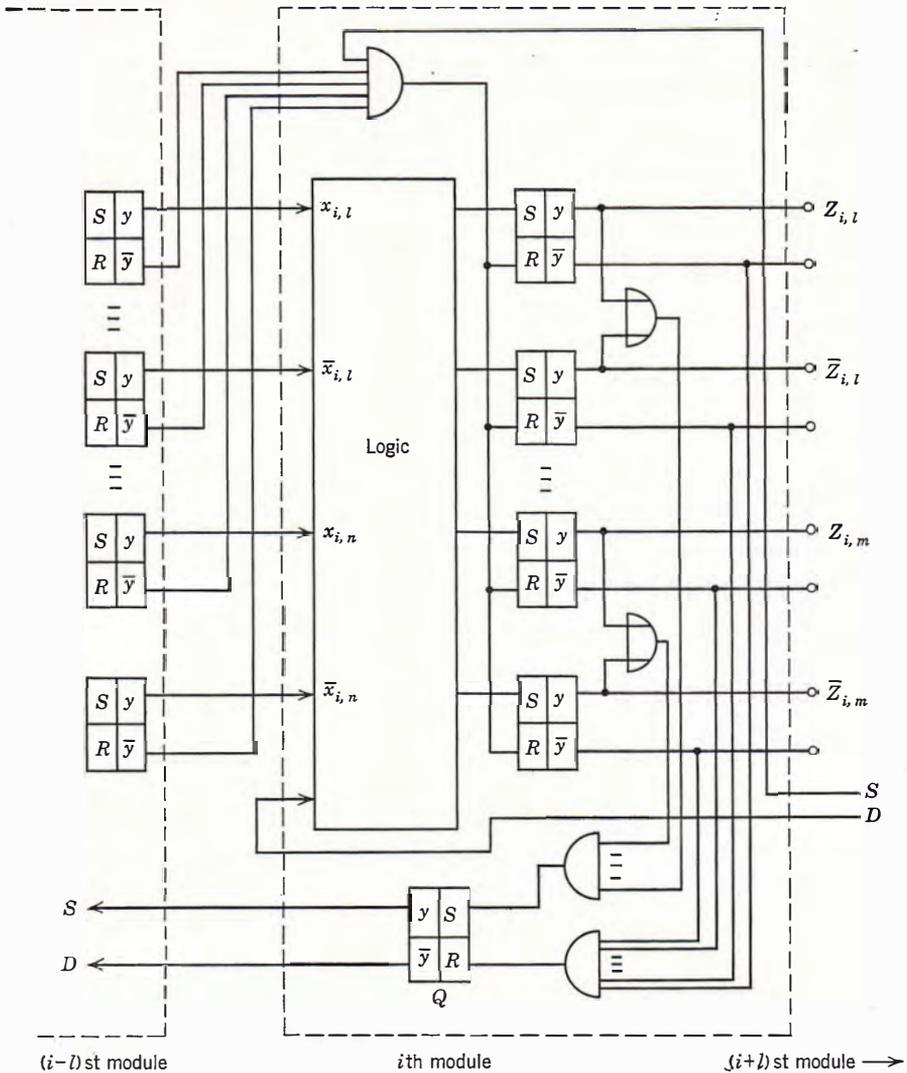


Figure 6.8 Combinational-circuit module that can be used in a network.

speeds of the components used), a substantial price is indicated in terms of the number of components required (a factor of perhaps 3), and since the logic paths have been lengthened by the addition of the various completion detection circuits, the maximum operating speed is decreased. However, if long chains of circuit modules are required, with significant variations in operation times, *effective* operating speeds may be increased, and there may also be gains

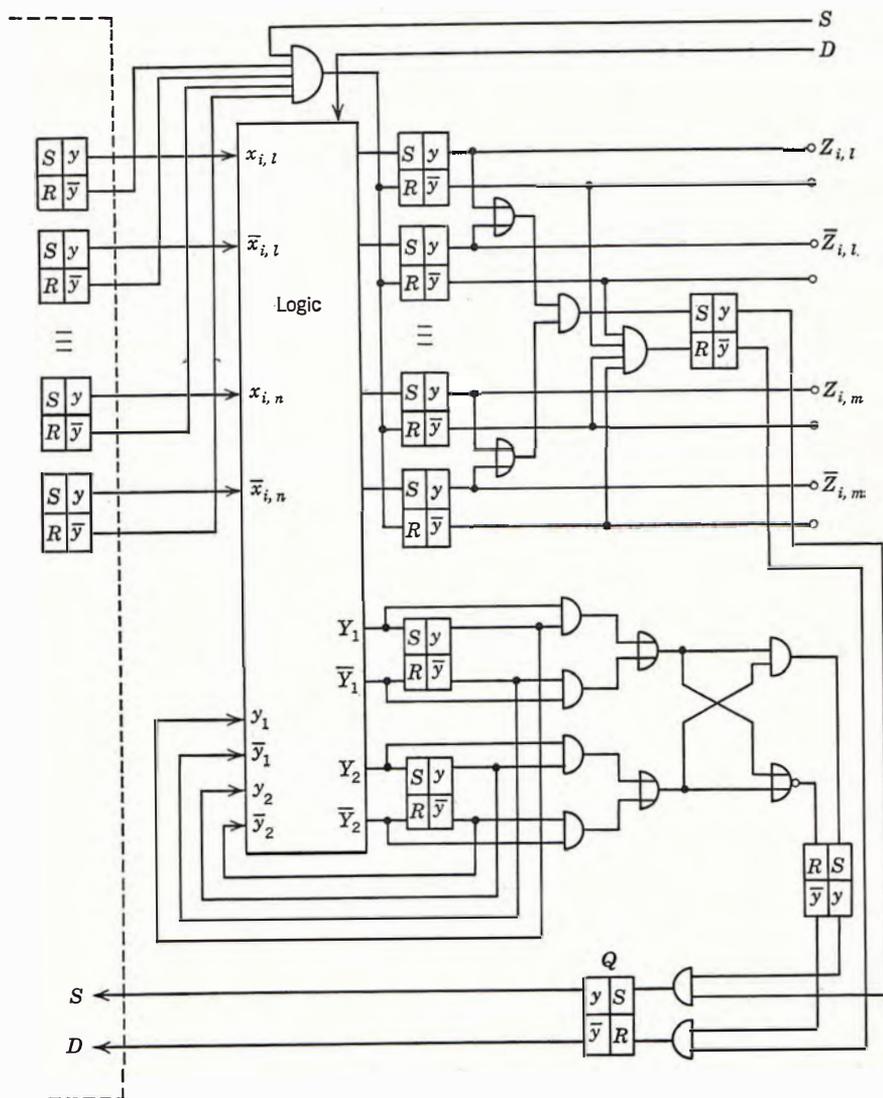


Figure 6.9 Sequential-circuit module that can be used in a network.

in reliability and maintainability. It may yet be found that this mode of operation can be implemented with simpler circuitry.

The theory of speed-independent circuits has been largely concerned with the realization of autonomous functions (no input variables) and so we show how the ideas presented here can be applied to this problem. Table 6.4a

describes an autonomous function, in which it is assumed that the starting state is 1. In this example, the specified output consists of a transient sequence followed by a periodic sequence.

Our approach is to convert the table to a two-column table that generates the desired output when the input alternates. In our example, Table 6.4b is the expanded version of Table 6.4a, having two columns and twice as many rows. The expanded table is of the SOC type considered earlier and so is realizable by a module of the form of Fig. 6.9. Starting in 1-0 of Table 6.4b and changing x back and forth between 0 and 1 produces the same output sequence described by the given table. If the outputs are produced as in the model of Fig. 6.5, then the simulation will be exact in that no spacers will separate consecutive data outputs. What is needed to complete the design is a

Table 6.4a Table for an Autonomous Function

1	2, 00
2	3, 01
3	4, 11
4	5, 01
5	6, 00
6	7, 10
7	4, 11

Table 6.4b Expanded Version of a

	x	
	0	1
1	①, 00	1', 00
1'	2, 01	①', 00
2	②, 01	2', 01
2'	3, 11	②', 01
3	③, 11	3', 11
3'	4, 01	③', 11
4	④, 01	4', 01
4'	5, 00	④', 01
5	⑤, 00	5', 00
5'	6, 10	⑤', 00
6	⑥, 10	6', 10
6'	7, 11	⑥', 10
7	⑦, 11	7', 11
7'	4, 01	⑦', 11

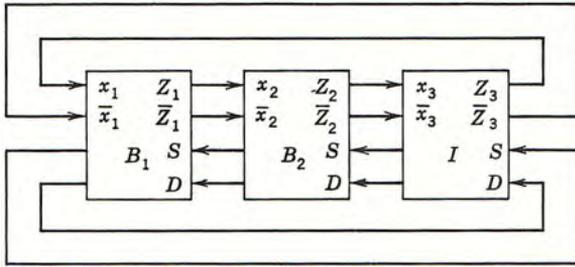


Figure 6.10 Circuit generating 010101 ···.

means for generating the alternating x -signal with modules of the type under discussion. This can be accomplished with a closed chain of three simple combinational-circuit modules as shown in Fig. 6.10. Each circuit feeds its output to the next circuit and receives from it the S - and D -signals that we have discussed. Locally, each module behaves precisely as a module interior to the cascade of Fig. 6.7 producing data and spacer outputs under the same constraints. The I -block is an inverter ($Z_3 = \bar{x}_3$), and each of the B -blocks simply transmits the received input to the output ($Z_1 = x_1$ and $Z_2 = x_2$). An examination of this system, assuming that initially two outputs are spacers and the other output 0 or 1, will show that one module at a time changes its output (to or from a spacer) and that, at each module, consecutive data outputs alternate between 0 and 1. (There is some resemblance to the behavior of the circuit shown in Fig. 5.9.) For example, if $Z_1 = \bar{Z}_1 = Z_2 = \bar{Z}_2 = Z_3 = 0$ and $\bar{Z}_3 = 1$, then I will be requesting (and currently receiving) a spacer from B_2 , B_2 will be requesting data from B_1 , and B_1 will be requesting (and receiving) data from I . Hence only B_1 is in a position to change its output, and it will be to the data output $Z_1\bar{Z}_1 = 01$.

The output of any one of the modules of the circuit in Fig. 6.10 can be connected to the input of a sequential-circuit module (in the form of Fig. 6.9) realizing Table 6.4*b* as shown in Fig. 6.11. The S - and D -signals transmitted back from the sequential-circuit module control the release of inputs to that module, and so the system operates according to the rules presented earlier.

6.2 Pulse-mode and Synchronous Circuits

The advantages and disadvantages of pulse-mode circuits and, by extension, synchronous circuits are sketched briefly in this section.

Consider the flow matrix of Table 6.5, describing a function with three input states (I_1 , I_2 , and I_3) and two output states (O_1 and O_2). The function is to be realized by a pulse-mode circuit with three input terminals and two output terminals. An I_i -input is represented by a pulse on the I_i -terminal,

References

Although no claim to completeness is made, the references listed here include the principal works dealing with the subject matter of this book, as well as a number of books and papers of collateral interest. Direct references are made to this material in the sections on "sources" that conclude each chapter, and there are also a few references made in the body of the text.

Several introductory texts on switching circuit theory which deal at most marginally with asynchronous circuits may be brought to the reader's attention. They are (in no particular order) by Prather [PRA-1], Marcus [MAR-4], Torng [TOR-1], Wood [WOO-1], Krieger [KRI-1], Phister [PHI-1], Humphrey [HUM-1], Bartee, Lebow, and Reed [BAR-1], and Hill and Peterson [HI]. Moore [MO-1] has edited an interesting collection of papers in the field, and books edited by McCluskey and Bartee [McC-5] and by Biorci [BIO-1] contain a number of interesting papers, both tutorial and otherwise. Some books treating "finite automata," which are clocked sequential circuits considered in terms of relations between input and output sequences, are by Booth [BOO-1], Hennie [HENN-1], Gill [GILL-1], Ginsburg [GIN-4], and Harrison [HARR-1]. The first two are the most up-to-date treatments.

The most important journal with respect to papers on switching theory is the *IEEE Transactions on Computers* (*IEEE-TC*, formerly *IEEE-TEC* and *IRE-TEC*), with the *Journal of the Association for Computing Machinery* (*JACM*) in second place. Other important sources, particularly for early publication of results are the *Proceedings of the Annual Symposia on Switching and Automata Theory*.

- AR-1 D. B. Armstrong, A. D. Friedman, and P. R. Menon, "Design of asynchronous circuits assuming unbounded gate delays," Bell Laboratories internal memorandum, 1967. (To appear in *IEEE-TC* in 1969.)
- AR-2 D. B. Armstrong, A. D. Friedman, and P. R. Menon, "Realization of asynchronous sequential circuits without inserted delay elements," *IEEE-TC*, C-17, No. 2, 129-134 (February 1968).
- BAR-1 T. C. Bartee, I. L. Lebow, I. S. Reed, *Theory and Design of Digital Machines*, McGraw-Hill, New York, 1962.

- BE-1 J. Beatty and R. Miller, "Some theorems for incompletely specified sequential machines with applications to state minimization," *Proc. 3rd Ann. Symp. Switching Circuit Theory, Logical Design*, 123-136 (September 1962).
- BE-2 J. Beatty and R. Miller, "An approach to state minimization for incompletely specified sequential machines," *IBM Research Report, RC-1055* (September 1963).
- BIO-1 G. Biorci, ed., *Network and Switching Theory, A NATO Advanced Study Institute*, Academic Press, New York, 1968.
- BOO-1 T. L. Booth, *Sequential Machines and Automata Theory*, John Wiley & Sons, New York, 1967.
- BRE T. H. Bredt and E. J. McCluskey, "Analysis and synthesis of control mechanisms for parallel processes," *Symposia on Parallel Processing Systems and Applications*, Naval Postgraduate School, Monterey, California, June 25-27, 1969.
- BR-1 J. A. Brzozowski, "Some problems in relay circuit design," *IEEE-TEC, EC-14*, No. 4, 630-634 (August 1965).
- BR-2 J. A. Brzozowski and S. Singh, "Definite asynchronous sequential circuits," *IEEE-TEC, C-17*, No. 1, 18-26 (January 1968).
- CAG-1 W. B. Cagle, R. S. Menne, R. S. Skinner, R. E. Staehler, and M. D. Underwood, "No. 1 ESS logic circuits and their application to the design of the central control," *BSTJ*, **43**, 2055-2095 (September 1964).
- CAL-1 S. H. Caldwell, *Switching Circuits and Logical Design*, John Wiley & Sons, New York, 1958.
- EI-1 E. B. Eichelberger, "Sequential circuits synthesis using hazards and delays," Ph.D. Dissertation, Dept. of Electrical Engineering, Princeton University, March 1963.
- EI-2 E. B. Eichelberger, "Sequential circuit synthesis using input delays," *Proc. 4th IEEE Annual Symp. on Switching Circuit Theory and Logical Design, S-156*, 105-116 (September 1963).
- EI-3 E. B. Eichelberger, "Hazard detection in combinational and sequential switching circuits," *IBMJ*, **9**, No. 2, 90-99 (March 1965).
- EL-1 B. Elspas, J. Goldberg, R. A. Short, and H. S. Stone, "Investigation of propagation-limited computer networks," Final Report—Phase II, Stanford Research Institute, July 1965.
- FE-1 D. Ferrari and A. Grasselli, "A cellular structure for sequential networks," *IEEE Conference Record of 8th Annual Symposium on Switching and Automata Theory*, 210-225 (October 1967).
- FRD-1 A. Friedes, "State reduction and state assignments for asynchronous sequential machines," Ph.D. Dissertation, Department of Electrical Engineering, Columbia University, 1967.
- FRM-1 A. D. Friedman, "Feedback in synchronous and asynchronous sequential switching circuits," Ph.D. Dissertation, Department of Electrical Engineering, Columbia University, 1965.
- FRM-2 A. D. Friedman, "Feedback in asynchronous sequential circuits," *IEEE-TEC, EC-15*, No. 5, 740-749 (October 1966).
- FRM-3 A. D. Friedman, "Feedback in synchronous sequential circuits," *IEEE-TEC, EC-15*, No. 3, 354-367 (June 1966).
- FRM-4 A. D. Friedman, R. L. Graham, J. D. Ullman, "Universal single transition time asynchronous state assignments," *IEEE-TC, C-18*, No. 6, 541-547 (June 1969).

- FRM-5 A. D. Friedman and P. R. Menon, "Synthesis of asynchronous sequential circuits with multiple input changes," *IEEE-TEC*, C-17, No. 6, 559-566 (June 1968).
- GE-1 G. B. Gerace and G. Gestri, "State assignments for reducing the number of delay elements in sequential machines," *Information and Control*, 10, No. 3, 223-253 (March 1967).
- GE-2 G. B. Gerace and G. Gestri, "Sequential machines with less delay elements than feedback paths," June 1967, unpublished (submitted to *TEC*).
- GILB-1 E. N. Gilbert, "Gray codes and paths on the n -cube," *BSTJ*, 37, 815-826 (May 1958).
- GILC-1 B. Gilchrist, J. H. Pomerine, and S. Y. Wong, "Fast carry logic for digital computers," *IRE-TEC*, EC-4, 133-136 (December 1955).
- GILL-1 A. Gill, *Introduction to the Theory of Finite State Machines*, McGraw-Hill, New York, 1962.
- GIM-1 J. F. Gimpel, "A reduction technique for prime implicant tables," *IEEE-TEC*, EC-14, no. 4, 542-552 (August 1965).
- GIN-1 S. Ginsburg, "A synthesis technique for minimal-state sequential machines," Natl. Cash Register Co. internal report, July 1958.
- GIN-2 S. Ginsburg, "On the reduction of superfluous states in a sequential machine," *JACM*, 6, 259-282 (April 1959).
- GIN-3 S. Ginsburg, "A technique for the reduction of a given machine to a minimal-state machine," *IRE-TEC*, EC-8, No. 3, 346-355 (September 1959).
- GIN-4 S. Ginsburg, *An Introduction to Mathematical Machine Theory*, Addison-Wesley, Reading, Mass. 1962.
- GR-1 A. Grasselli and F. Luccio, "A method for minimizing the number of internal states in incompletely specified sequential networks," *IEEE-TEC*, EC-14, No. 3, 350-359 (June 1965).
- GR-2 A. Grasselli, "Minimal closed partitions for incompletely specified flow tables," *IEEE-TEC*, EC-15, No. 2, 245-249 (April 1966).
- GR-3 A. Grasselli and F. Luccio, "A method for combined row-column reduction of flow tables," *Proc. 7th Ann. Symp. Switching Theory, Automata* (September 1966).
- HAL-1 A. D. Hall, Jr., "Introduction to the theory of speed independent asynchronous switching circuits," Report No. 50, Digital Systems Laboratory, Princeton University, July 1966.
- HAL-2 A. D. Hall, Jr., "Synthesis of double rank sequential circuits," Report No. 53, Digital Systems Laboratory, Princeton University, Dec. 1966.
- HAL-3 A. D. Hall, Jr., and F. S. Acton, "Scheduling university course examinations by computer," *CACM*, 10, No. 4, 235-238 (April 1967).
- HAL-4 A. D. Hall, Jr., "Treatment of delays in asynchronous networks," Ph.D. Dissertation, Dept. of Electrical Engineering, Princeton University, 1966.
- HAMG-1 R. W. Hamming, "Error detecting and correcting codes," *BSTJ*, 29, 147-160 (April 1950).
- HAML-1 D. Hammel, "Ideas on asynchronous feedback networks," *Proc. 5th Ann. Symp. Switching Theory, Logical Design, Princeton University, November 11-13, 1964*, pp. 4-11.
- HARR-1 M. A. Harrison, *Introduction to Switching and Automata Theory*, McGraw-Hill New York, 1965.
- HART-1 J. Hartmanis and R. E. Stearns, *Algebraic Structure Theory of Sequential Machines*, Prentice-Hall, Englewood Cliffs, N.J. 1966.

- HAZ-1 B. Hazeltine, "Encoding of asynchronous sequential circuits," *IEEE-TEC*, EC-14, 727-729 (October 1965).
- HEND-1 H. C. Hendrickson, "Fast high-accuracy binary parallel addition," *IRE-TEC*, EC-9, No. 4, 465-469 (December 1960).
- HENN-1 F. C. Hennie, *Finite State Models for Logical Machines*, John Wiley & Sons, New York, 1968.
- HI F. J. Hill and G. R. Peterson, *Introduction to Switching Theory and Logical Design*, John Wiley & Sons, New York, 1968.
- HUF-1 D. A. Huffman, "The synthesis of sequential switching circuits," *J. Franklin Institute*, 257, No. 3, 161-190 (March 1954); and No. 4, 275-303 (April 1954). Also reprinted in [MO-2].
- HUF-2 D. A. Huffman, "A study of the memory requirements of sequential switching circuits," Technical Report No. 293, Research Laboratory of Electronics, MIT, March 14, 1955.
- HUF-3 D. A. Huffman, "Design of hazard-free switching circuits," *JACM*, 4, 47-62 (January 1957).
- HUF-4 D. A. Huffman, unpublished paper delivered at Princeton University Symposium on Switching Theory, Princeton, N.J. January 1962.
- HUM-1 W. S. Humphrey, Jr., *Switching Circuits with Computer Applications*, McGraw-Hill, New York, 1958.
- KA-1 R. Y. Kain, "Synthesis of up-down counters," *IEEE-TEC*, EC-16, No. 2, 146-151 (April 1967).
- KE-1 W. Keister, A. E. Ritchie, and S. H. Washburn, *The Design of Switching Circuits*, D. Van Nostrand, Princeton, N.J., 1951.
- KI-1 L. L. Kinney, "A characterization of some asynchronous state assignments," *IEEE Conf. Rec. of Ninth Ann. Symp. on Switching and Automata theory*, pp. 20-27, Oct. 1968.
- KI-2 L. L. Kinney, "Decomposition of asynchronous sequential switching circuits," Ph.D. dissertation, Electrical Engineering Dept., University of Iowa, February 1968.
- KL-1 M. Kliman and O. Lowenschuss, "Asynchronous electronic switching circuits," 1959 *IRE National Conv. Rec.*, Pt. 4, 267-271.
- KRI-1 M. Krieger, *Basic Switching Circuit Theory*, The Macmillan Co., New York, 1967.
- LA-1 C. G. Langdon, Jr., "Analysis and synthesis of asynchronous circuits under different delay assumptions," Ph.D. dissertation, Electrical Engineering Dept. (also *Technical Report TR67-14*), Syracuse University, October 1967.
- LA-2 G. G. Langdon, Jr., "Delay-free asynchronous circuits with constrained line delays," *IEEE-TC*, V.C.-17, 1131-1143 (December 1968).
- LE-1 S. B. Lerner, "Hazard correction in asynchronous sequential circuits," *IEEE-TEC*, EC-14, No. 2, 265-267 (April 1965).
- LIU-1 C. N. Liu, "A state variable assignment method for asynchronous sequential switching circuits," *JACM*, 10, 209-216 (April 1963).
- MAL-1 G. A. Maley and J. Earle, *The Logic Design of Transistor Digital Computers*, Prentice-Hall, Englewood Cliffs, N.J., 1963.
- MAR-1 M. P. Marcus, "Relay essential hazards," *IEEE-TEC*, EC-12, 405-407 (August 1963).
- MAR-2 M. P. Marcus, "Derivation of maximal compatibles using Boolean algebra," *IBMJ*, 8, No. 5, 537-538 (November 1964).
- MAR-3 M. P. Marcus, "Cascaded binary counters with feedback," *IEEE-TEC*, EC-12, 361-364 (August 1963).

- MAR-4 M. P. Marcus, *Switching Circuits for Engineers*, Prentice-Hall, Englewood Cliffs, N.J., 1962.
- MAY-1 R. H. Mayne, "The binary counter: a novel logical design," unpublished Bell Laboratories memorandum, October 1958.
- MAY-2 R. H. Mayne, "A binary counter with associated parity bit," unpublished Bell Laboratories memorandum, Nov. 1958.
- McC-1 E. J. McCluskey, "Fundamental mode and pulse mode sequential circuits," *Proc. IFIP Congress 1962, Munich Germany, Information Processing*, North-Holland Publishing Co., Amsterdam, pp. 725-730.
- McC-2 E. J. McCluskey, "Minimum-state sequential circuits for a restricted class of incompletely specified flow tables," *BSTJ*, **4**, 1759-1768 (November 1962).
- McC-3 E. J. McCluskey, "Transient behavior of combinational logic networks", in *Redundancy Techniques for Computing Systems*, Spartan Books, pp. 9-46, 1962.
- McC-4 E. J. McCluskey, *Introduction to The Theory of Switching Circuits*, McGraw-Hill New York, 1965.
- McC-5 E. J. McCluskey and T. C. Bartee, eds., *A Survey of Modern Switching Circuit Theory*, McGraw-Hill, New York, 1962.
- McN-1 R. McNaughton, "Badly timed elements and well timed nets," Moore School Report No. 65-02, University of Pennsylvania, June 10, 1964.
- ME-1 G. H. Mealy, "A method for synthesizing sequential circuits," *BSTJ*, **34**, 1045-1079 (September 1955).
- MLM-1 J. Millman and H. Taub, *Pulse, Digital, and Switching Waveforms*, McGraw-Hill, New York, 1965.
- MLR-1 R. E. Miller, *Switching Theory*, Vol. I: *Combinational Circuits*, John Wiley & Sons, New York, 1965.
- MLR-2 R. E. Miller, *Switching Theory*, Vol. II: *Sequential Circuits and Machines*, John Wiley & Sons, New York, 1965.
- MO-1 E. F. Moore, "Gedanken experiments on sequential machines" in [SH-1].
- MO-2 E. F. Moore, *Sequential Machines: Selected Papers*, Addison-Wesley, Reading, Mass., 1964.
- MU-1 D. E. Muller and W. S. Bartky, "A theory of asynchronous circuits I," Report No. 75, University of Illinois, Digital Computer Laboratory, Nov. 1956.
- MU-2 D. E. Muller and W. S. Bartky, "A theory of asynchronous circuits II," Report No. 78, University of Illinois, Digital Computer Laboratory, March 1957.
- MU-3 D. E. Muller and W. S. Bartky, "A theory of asynchronous circuits," *Proc. of an Int. Symp. on the Theory of Switching*, vol. 29, Annals of the Computation Laboratory of Harvard University, Harvard University Press, pp. 204-243, 1959.
- MU-4 D. E. Muller, "The general synthesis problem for asynchronous digital networks," *IEEE Conference Record of Eighth Ann. Symp. Switching, Automata Theory*, 71-82 (October 1967).
- MU-5 D. E. Muller, "Treatment of transition signals in electronic sequential circuits by algebraic methods," *IRE-TEC*, **EC-8**, No. 3, 401 (September 1959).
- MU-6 D. E. Muller, "Asynchronous logics and application to information technology," *Proc. of a Symp. on the Application of Switching Theory in Space Technology*, Stanford University Press, March 1962.
- NA-1 R. Narasimhan, "Minimizing incompletely specified sequential switching functions," *IRE-TEC*, **EC-10**, No. 3, 531-532 (September 1961).
- PA-1 M. C. Paull and S. H. Unger, "Minimizing the number of states in incompletely specified sequential switching functions," *IRE-TEC*, **EC-8**, No. 3, 356-367 (September 1959).

- PA-2 M. C. Paull and G. Waldbaum, "A note on state minimization of asynchronous sequential functions," *IEEE-TEC EC-16*, No. 1, 94-97 (February 1967).
- PA-3 M. C. Paull, "The minimization of the number of states of a completely specified sequential function having restrictions on its allowable input sequences," unpublished Bell Telephone Laboratories memorandum, 1966.
- PHI-1 M. Phister, *Logic Design of Digital Computers*, John Wiley & Sons, New York, 1956.
- PRA-1 R. E. Prather, *Introduction to Switching Theory: A Mathematical Approach*, Allyn and Bacon Inc., Boston, 1967.
- RE-1 I. S. Reed, "Some remarks on state reduction of asynchronous circuits by the Paull-Unger method," *IEEE-TEC, EC-14*, No. 2, 262-265 (April 1965).
- RIC-1 R. K. Richards, *Arithmetic Operations in Digital Computers*, D. Van Nostrand, Princeton, N.J., 1955.
- RO-1 J. E. Robertson, "Problems in the physical realization of speed independent circuits," *Proc. 2nd AIEE Symp. on Switching Circuit Theory and Logical Design, Detroit, Michigan, S-134*, 106-108 (October 1961).
- SA-1 G. Saucier, "Encoding of asynchronous sequential networks," *IEEE-TEC, EC-16*, No. 3, 365-369 (June 1967).
- SH-1 C. E. Shannon and J. McCarthy, eds. *Automata Studies*, Princeton University Press, 1956.
- SIM-1 J. C. Sims, Jr., and H. J. Gray, "Design criteria for autosynchronous circuits," *Proc. Eastern Joint Computer Conf., Dec. 3-5, 1958*, pp. 94-99.
- SIN-1 S. Singh, "Asynchronous sequential circuits with feedback," Dept. of Electrical Engineering, University of Ottawa, Technical Report, No. 68-9, May 1968.
- SMI-1 R. J. Smith, II, J. H. Tracey, W. L. Schoeffel, and G. K. Maki, "Automation in the design of asynchronous sequential circuits," *Proc. Spring Joint Computer Conf.*, pp. 55-60, 1968.
- TA-1 C. J. Tan, "Note on general race-free assignment for asynchronous circuits," unpublished note, Columbia University Dept. of Electrical Engineering, February 19, 1967.
- TA-2 C. J. Tan, "State assignment for asynchronous sequential machines," *Bell Telephone Laboratory Report* (August 1967).
- TA-3 C. J. Tan, "Synthesis of asynchronous sequential switching circuits," Ph.D. Dissertation Dept. of Electrical Engineering, Columbia University, 1969.
- TA-4 C. J. Tan, "Computer aided design of asynchronous sequential switching circuits," *First Ann. Houston Conf. on Circuits, Systems and Computers*, U. of Houston (May 22, 23, 1969).
- TA-5 C. J. Tan, P. R. Menon, A. W. Friedman, "Simplification and decomposition of asynchronous sequential circuits," to appear in *IEEE-TC* (September 1969).
- TOR-1 H. C. Torng, *Introduction to the Logical Design of Switching Systems*, Addison-Wesley, Reading, Mass. 1964.
- TR-1 J. H. Tracey, "Internal state assignments for asynchronous sequential machines," *IEEE-TEC, EC-15*, No. 4, 551-560 (August 1966).
- UN-1 S. H. Unger, "A theorem on feedback in sequential switching circuits," Quarterly Progress Report, Research Laboratory of Electronics, MIT (April 15, 1955).
- UN-2 S. H. Unger, "A study of asynchronous logical feedback networks," Technical Report No. 320, Research Laboratory of Electronics, MIT (April 1957). Also Sc.D. Dissertation, Dept. of Electrical Engineering, MIT, 1957.
- UN-3 S. H. Unger, "Hazards and delays in asynchronous sequential switching circuits," *IRE Trans. on Circuit Theory, CT-6*, 12-25 (March 1959).

274 References

- UN-4 S. H. Unger, "Simplification of state tables," *AIEE Fall Meeting, Chicago* (October 1960). Reprinted in [McC-5], pp. 145-170.
- UN-5 S. H. Unger, "Flow table simplification—some useful aids," *IEEE-TEC, EC-14*, No. 3, 472-475 (June 1965).
- UN-6 S. H. Unger, "A row assignment for delay-free realizations of flow tables without essential hazards," *Proc. 7th Ann. Symp. on Switching and Automata Theory, Berkeley California, October, 27, 1966*. Also in *IEEE-TC, C-17*, No. 2, 146-158 (February 1968).
- UN-7 S. H. Unger, "Timing considerations in clocked switching circuits," Quarterly Progress Report, Research Laboratory of Electronics, MIT, pp. 113-115 (January 16, 1956).
- WAI-1 W. M. Waite, "The production of completion signals by asynchronous, iterative networks," *IEEE-TEC EC-13*, 83-86 (April 1964).
- WAR-1 W. H. Ware, "The logical principles of a new kind of binary counter," *Proc. IRE*, **41**, No. 10, 1429-1437 (October 1953).
- WOO-1 P. E. Wood, Jr., *Switching Theory*, McGraw-Hill, New York, 1968.
- YOE-1 M. Yoeli and S. Rino, "Application of ternary algebra to the study of static hazards," *JACM*, **11**, No. 1, 84-97 (June 1964).
- YOU-1 D. H. Younger, "Minimum feedback arc sets for a directed graph," *IEEE Trans. on Circuit Theory, CT-10*, 238-245 (June 1963).