OCTOBER 1967

BUILDING-BLOCK HANDBOOK

by

The Engineering Staff

of

PHILCO-FORD CORPORATION
Microelectronics Division
Blue Bell, Pennsylvania • 19422

PRINTED IN U.S.A.

CONTENTS

Section 1

INTRODUCTION

You can design your own large-scale arrays with Philco-Ford MOS Building Blocks, by following the simple rules explained in this Handbook. The procedure is similar to breadboarding conventional integrated circuits. You merely lay out decals of the Building Blocks* on scaled paper representing the chip area of the array and draw single-line interconnections between the blocks. Thus you retain complete control of your own system design.

Each Building Block is a predesigned integrated-circuit layout for a particular logic function, having completely specified input-output characteristics. The versatile Philco-Ford Family includes a wide variety of logic functions, capable of implementing any logic system.

The Building-Block approach offers many advantages. For example:

- *A continually expanding library of proven, predesigned circuits.* These circuits have been widely used in the mass-produced Philco-Ford Standard Family as well as in numerous custom arrays. They have been optimized for yield. Therefore, Building-Block arrays are manufactured quickly and in volume, with confidence that they will perform as predicted.

- *Quick turnaround.* The time required for the design of individual circuits is eliminated.

- *Efficient use of silicon area* – in contrast to a 'Master Slice' concept which wastes silicon with superfluous transistors and interconnect area.

- *Greatly reduced cost* – a natural result of all the above advantages. For example, the use of predesigned, pre-evaluated circuits eliminates the engineering required to custom-design circuit functions for each array, as well as the possible need for rework.

- *User design.* The system engineer carries his own design from concept to manufacture. Thus he has more design flexibility and greater proprietary protection.

This handbook explains step by step (1) how to partition a Building-Block system and lay out each part on a silicon chip, optimizing chip area and array performance; (2) how to predict power supply requirements and speed; (3) how to develop a good functional test; and (4) how to specify package and array performance. Section 2 explains the fundamentals of MOS transistor and circuit operation. Section 3 describes the logic and circuit contents of each Building Block, together with a few of their many possible applications. Section 4 briefly discusses the factors affecting cost in a Building-Block system. Section 5 describes the layout procedure with detailed examples. Section 6 explains how to calculate speed and power. Section 7 describes the many in-process and end-of-line tests performed as a matter of routine on all integrated circuits and arrays at Philco-Ford. In addition, it explains how to develop a functional test and describes the computer program available at Philco-Ford for generating tests. Section 8 describes in detail the steps required to order and manufacture a Building-Block array.

The Appendices include a sample specification sheet (A); a list of the contents of Building-Block kits available from Philco-Ford (B); package information (C); and data sheets for each of the Building Blocks (D).

NOTE: Be sure to register your Handbook so that we can provide you with data sheets and information as new Building Blocks become available. Large-scale integration is an evolutionary technology, and the library of Building Blocks available to achieve it will therefore be continually expanding.

*or draw in penciled outlines

Section 2

DESIGN FUNDAMENTALS

In this section, we will summarize the operation of the MOS transistor; show what it looks like physically; explain how it is used in the Building Blocks both as an inverter and as a load "resistor", and describe its operation in Building-Block circuits.

2.1 DEVICE FUNDAMENTALS

Fig. 2.1a shows a somewhat simplified* three-quarter view of a single MOS transistor cut out of a silicon chip containing other devices and the interconnections between them. The transistor consists of a block of N type silicon called the "body" or "substrate", into which P^+ impurities are diffused in two parallel strips called, respectively, the "source" and the "drain". In operation, an induced P region known as the "channel" connects source and drain. The entire surface of the chip is covered by silicon dioxide, which is etched down over the P regions and the channel area. A metal gate is deposited over the channel area, separated from it by the oxide. Metal supply lines contact the source and drain through a small cutout in the oxide.

Fig. 2.1b is a stylized top view of the transistor in Fig. 2.1a, showing the metal (in outline only) on top of the chip, the P regions (cross-hatched) beneath the surface, and the cutouts (black) allowing contact between metal and P regions. The circuit symbol for the transistor is shown in Fig. 2.1c.

If the gate, source, and body are grounded and a negative voltage applied to the drain, the drain-to-body PN junction is reverse-biased. Since this isolates the drain from the source, no current will flow between the two. If a negative voltage is applied to the gate, electrons are repelled from the

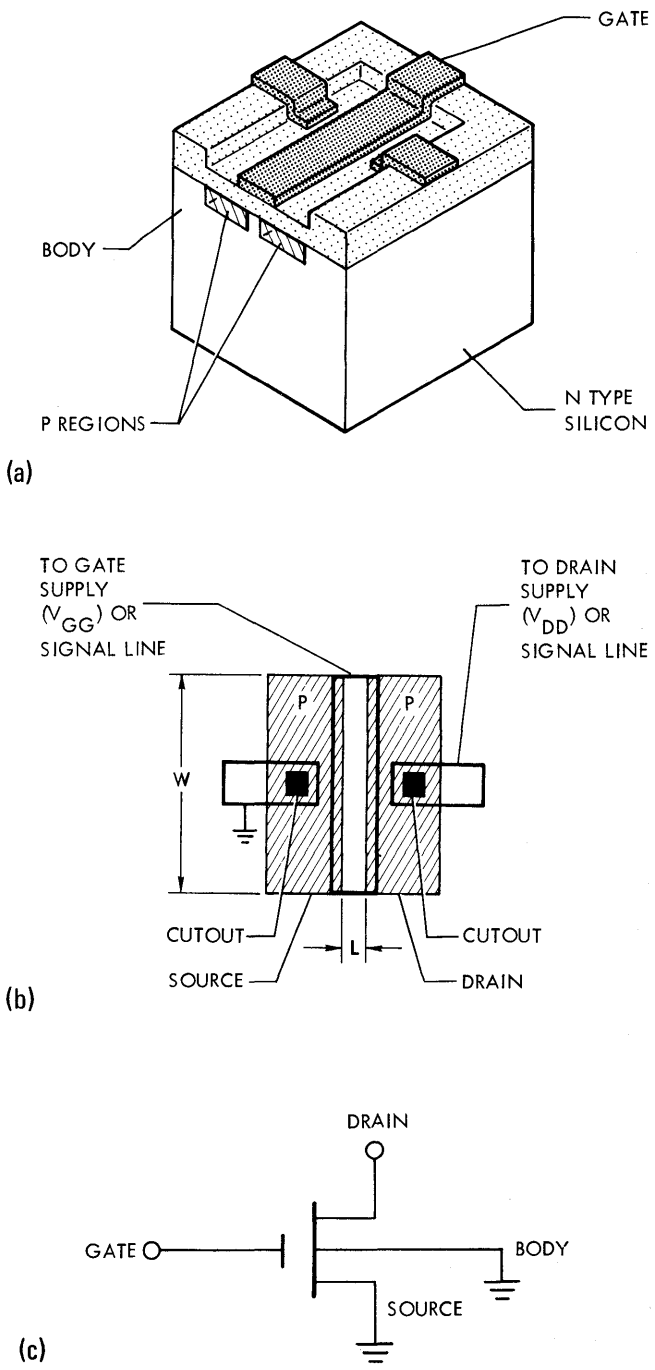


Fig. 2.1 The MOS Transistor: (a) Three-Quarter View; (b) Stylized Top View; (c) Schematic Diagram.

*For more detailed information on MOS device fundamentals, see "MOS MONOLITHIC SUBSYSTEMS: A REVOLUTION IN MICRO-ELECTRONICS," available from Philco-Ford.

surface region of the silicon immediately beneath the gate, and holes are attracted to it. When the gate-to-source voltage reaches a certain threshold, called V_T (-4 to -6 V in Building-Block transistors), this surface region changes or "inverts", becoming P-type instead of N-type. This inverted region, called the "channel", provides a path for conduction between source and drain. If the drain is more negative than the source, current will flow between the two. However, if the gate voltage is less than V_T, no current can flow, thus the maximum "0" level is fixed by the minimum V_T. As the gate voltage becomes more negative than V_T, the channel depth increases, causing the ON resistance to decrease.

In the stylized view of the MOS transistor shown in Fig. 2.1b, the distance from source to drain is labeled "L", to signify channel length. The dimension labeled "W" is the width of the thin-oxide region and therefore the width of the channel.

2.2 CIRCUIT FUNDAMENTALS

2.2.1 The MOS as a Load "Resistor"

A device with a ratio of L:W>>1 has a high ON resistance and thus can be used as a load "resistor". The resistance is determined not only by its dimensions, however, but also by the difference between its gate voltage and its source voltage: The larger this voltage difference, the lower the ON resistance. As the source voltage of the load falls toward V_{DD} during turn-off, its ON resistance therefore increases. Thus the turn-off waveform is considerably slower than that of a simple RC network. To minimize this effect, the Building-Block load transistors are operated from a gate supply (V_{GG}) twice as negative as the drain supply (V_{DD}).

2.2.2 Logic Levels

Fig. 2.2 shows the circuit diagram of a Building-Block inverter stage with a low-current MOS transistor (T_2) as the load. Since the inverter (as well as the load) is turned on by a negative gate voltage, it is convenient to define a logic "1" to be a negative voltage near V_{DD} and a logic "0" to be at or near ground.

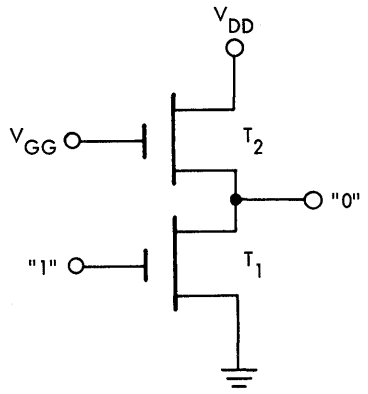


Fig. 2.2 Schematic Diagram of Building-Block Inverter Stage.

The dimensions of the inverter are such that its ON resistance with a "1" level on its gate is small. Under these conditions, therefore, the inverter is effectively a closed switch in series with a load resistor (see Fig. 2.3).

With the specified "0" level on its gate, the inverter is effectively an open switch (see Fig. 2.4), since leakage currents are very small.*

2.2.3 Load Capacitance

In Fig. 2.5, two Building-Block inverter stages are shown. The load capacitance C_L consists of (1)

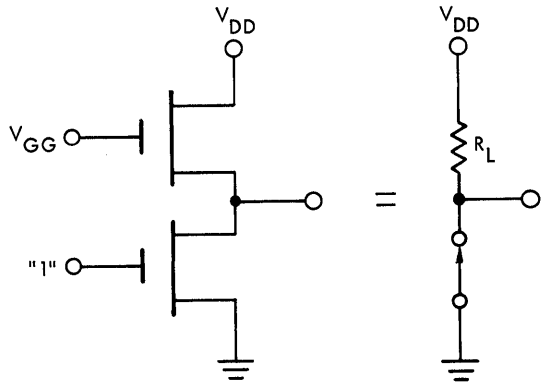


Fig. 2.3 Inverter with "1" on its Gate and Equivalent Circuit.

*Leakages need not be considered at all when designing with strictly dc Building Blocks. However, minimum clock frequency in the BB-403 and BB-404 (the one-bit shift registers) is a function of leakage (see Sections 2.2.5 and 3.2.3).

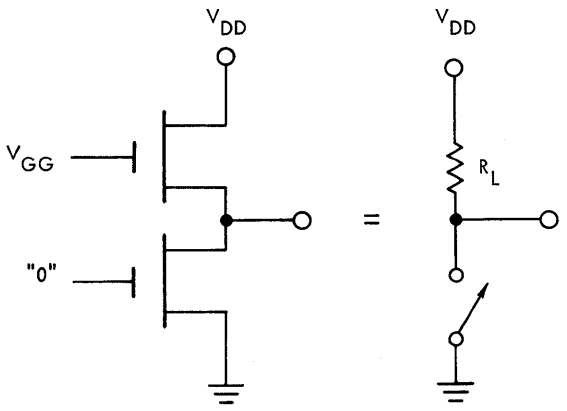


Fig. 2.4 Inverter with "0" on its Gate and Equivalent Circuit.

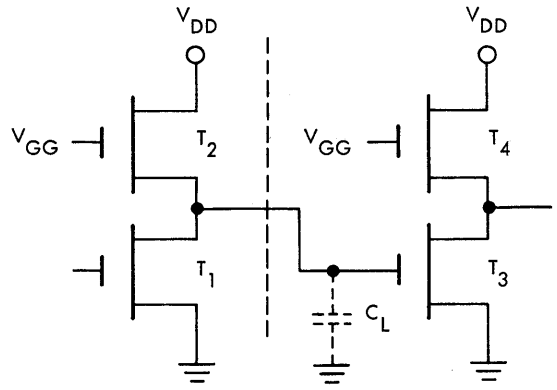


Fig. 2.5 Two Building-Block Inverter Stages.

PN-junction capacitance between the drain of T_1 and the grounded body; (2) the capacitance of the interconnections (see Section 2.2.3); and (3) capacitance between the metal gate and the grounded source of T_3.

Because of the extremely high gate-oxide impedance ($>10^{14}\Omega$), the load capacitance of one stage is effectively isolated from the next (see Fig. 2.6). Thus the RC time constant for charging the load capacitance is determined only by the capacitance itself and the load resistance R_L. Discharge time through the low ON resistance of the inverter is much less than the charge time.

2.2.4 Interconnecting Lines

Interconnecting lines may be made either of (1) metal deposited on the silicon dioxide surface of the chip or (2) buried P regions. Fig. 2.7a is a stylized top view showing a metal line (in outline only) on the surface of the chip and two P region lines (cross-hatched), beneath the surface. A cutout permitting contact between the metal line and one of the buried P lines is shown in black. Fig. 2.7b shows a cross-sectional view along the A-A cut in Fig. 2.7a.

Since the oxide layer is thinner in the crossover region than elsewhere, the capacitance between the metal and the P region passing under it is greater than the capacitance between the metal and the substrate. Procedures for minimizing crossovers are described in Section 5.3, and methods of computing resistance and capacitance of interconnecting lines are given in Section 6.1.1.

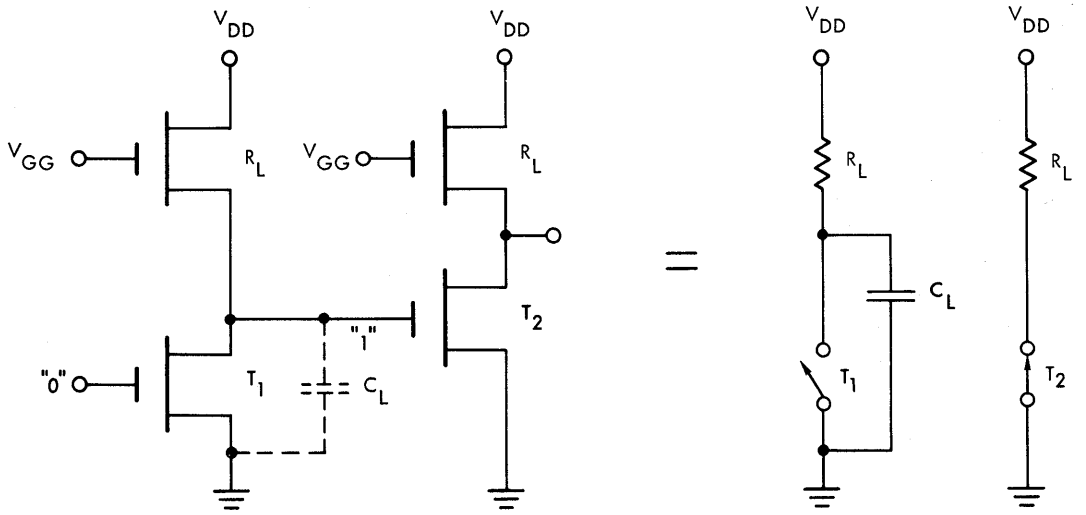


Fig. 2.6 Isolation of Input from Output Stage.

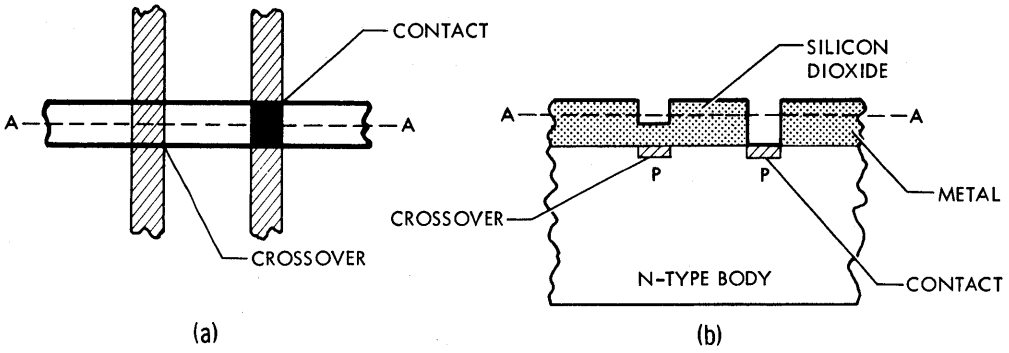


Fig. 2.7 P and Metal Interconnect Lines: (a) Stylized Top View; (b) Cross-Section.

2.2.5 The MOS Transistor as a Coupling Device

The MOS transistor can also be used to couple data from one inverter stage to the next, as shown in Fig. 2.8.*

Since coupling is identical through the two stages, we will consider only the first stage. When the clock ϕ_1 appears at the gate of coupling transistor T_3, the voltage level at the drain of inverter T_1 is coupled to the gate of T_4. The source of the coupling transistor is considered to be the side that is, at least initially, at the more positive voltage level – in this case, the side attached to the gate of T_4. Unless the voltage difference between the gate of T_3 and its source is equal to or greater than the threshold voltage, T_3 will be off. Thus, as the source rises toward a "1", the ON resistance of T_3 will approach an open circuit unless ϕ_1 is much more negative than the "1" level. For this reason, clock voltages equal to V_{GG} are normally used for this coupling mode.

The clock ϕ_1 must remain on for a time at least equal to that of the propagation delay through T_3 and T_4. In the BB-403 and BB-404, the maximum propagation delay is 500 ns at 125°C. Thus the maximum clock frequency is 1 MHz at the high-temperature extreme. When ϕ_1 turns off and ϕ_2 turns on, the voltage level at the gate of T_4 is stored on C_{L1}, gradually leaking off through the source-to-body PN junction of T_3. In the BB-403 and BB-404, the time required for the "1" level stored on the gate capacitance to rise above the amount that can be tolerated is 0.05 ms at 125°C. Thus the minimum clock frequency is 10 kHz at the high-temperature extreme.

*This coupling mode is used in the BB-403, and the BB-404, the one-bit shift registers discussed in Section 3.2.3.

2.3 CIRCUIT PERFORMANCE

2.3.1 Speed

The speed of the Building-Block circuits is effectively limited only by the charge and discharge time of the load capacitance. Speeds of at least 500 kHz can easily be obtained for most Building-Block systems; by careful attention to layout (see Section 5), speeds in the MHz range can often be achieved.

2.3.2 Fan-Out

Since current drive is needed only to charge and discharge the load capacitances, fan-out is limited only by the system speed requirement. Because of the small capacitances within the chip, very high fan-outs (10 or more) can be obtained without seriously affecting speed. However, when an inverter fans out to a large number of gates on a second chip, and in addition the interconnect capacitance is large, an output buffer (the BB-020 or BB-021) is usually necessary to meet speed requirements. This device consumes considerable silicon area.

2.3.3 Noise

Noise inside a Building-Block chip is not a problem because the Building Blocks are designed for high noise immunity.

"Zero" Level Noise Immunity. A minimum gate threshold of -4 V is required before any Building-Block transistor will begin to turn on.

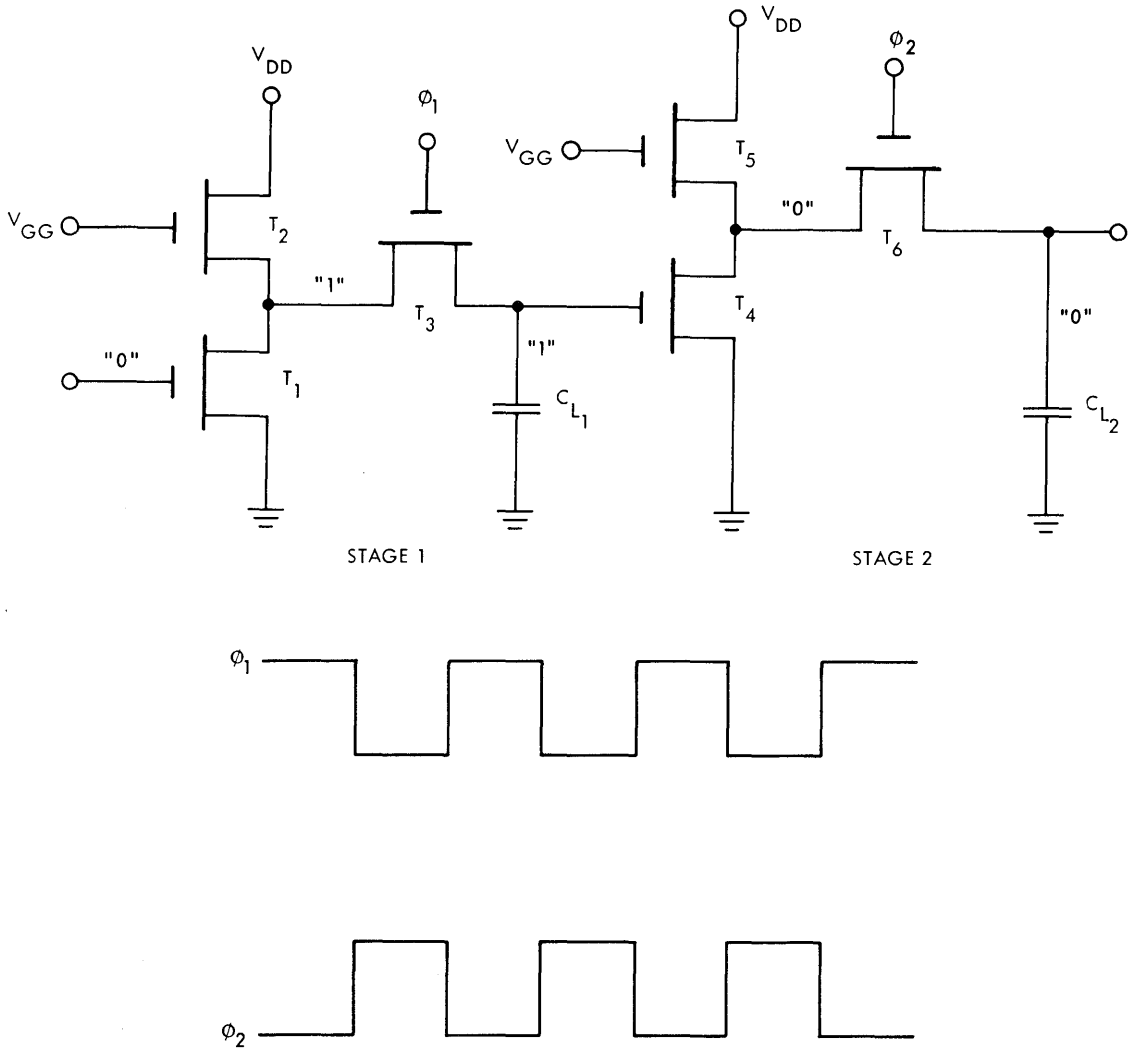


Fig. 2.8 The MOS as a Coupling Device.

Until the gate becomes considerably more negative than this threshold, the output (drain) voltage remains above the "1" level. The Building Blocks are designed for a maximum "0" level out of -2 V. Thus a "0" level noise immunity of considerably more than 2 V is built into the design.

"One" Level Noise Immunity. The Building Blocks are designed for a minimum "1" level out of -10 V. The maximum threshold voltage required to turn on any Building-Block transistor is -6 V. Thus the "1" level noise immunity is at least 4 V.

Noise problems between chips in a Building-Block system can be eliminated simply by insuring that the specified maximum "0" and minimum "1" levels (including any noise voltages) are provided at the inputs of the chips.

2.3.4 Power

Power dissipation in a Building Block chip is unlikely to exceed package ratings even at the high temperature extreme of 125°C. The only significant power considerations are (1) power supply drain and (2) the effect of heating on the load resistance and hence on charge time. The latter is considered in Section 6.2, and the speed-power tradeoff is discussed in Section 6.3.

Section 3

BUILDING BLOCKS

The Building-Block family includes all the elements used in a dc system – basic combinatorial logic and storage elements – plus a wide variety of combined functions available in single blocks to conserve chip area. In addition, the family includes two one-bit shift-register elements: a dc element and a dynamic element, which make use of the coupling technique described in Section 2.2.5.

In this section we will describe the operation of the basic MOS NOR and NAND gates and discuss some of the available blocks which incorporate these functions. Then we will describe the operation of the master-slave flip-flop, show how it can be connected to form a ripple-through counter and a shift register, and discuss the clock-generator element used with it. Finally we will describe the operation of the one-bit shift-register elements.

A List of Building Blocks, with circuit and logic diagrams and Boolean expressions, is included as Table 3.1 at the end of this section.

3.1 COMBINATORIAL LOGIC

3.1.1 Basic NOR Gate

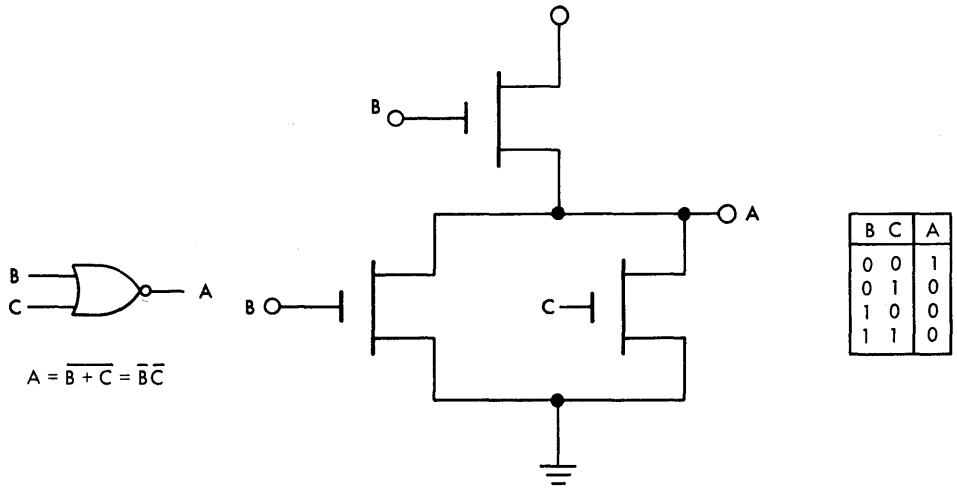
Logic and circuit diagrams of the two-input NOR gate (the BB-002) are shown in Fig. 3.1, together with a truth table and Boolean expressions governing its operation. A "1" level* on the gate of either inverter (i.e., at B or C) causes that inverter to turn on and pulls the output (A) to "0". As shown by the truth table, the output A can be at a "1" level only when B and C are both "0".

NOR gates with two to six inputs are available in separate blocks.

3.1.2 Basic NAND Gate

The logic and circuit diagrams of the two-input NAND gate (the BB-010) are shown in Fig. 3.2, to-

*As pointed out in Section 2.2.2, logic "1" is defined as a negative voltage (near V_{DD}) and logic "0" is at or near ground.



B	C	A
0	0	1
0	1	0
1	0	0
1	1	0

Fig. 3.1 Basic NOR Gate: Logic Diagram, Circuit Diagram and Truth Table.

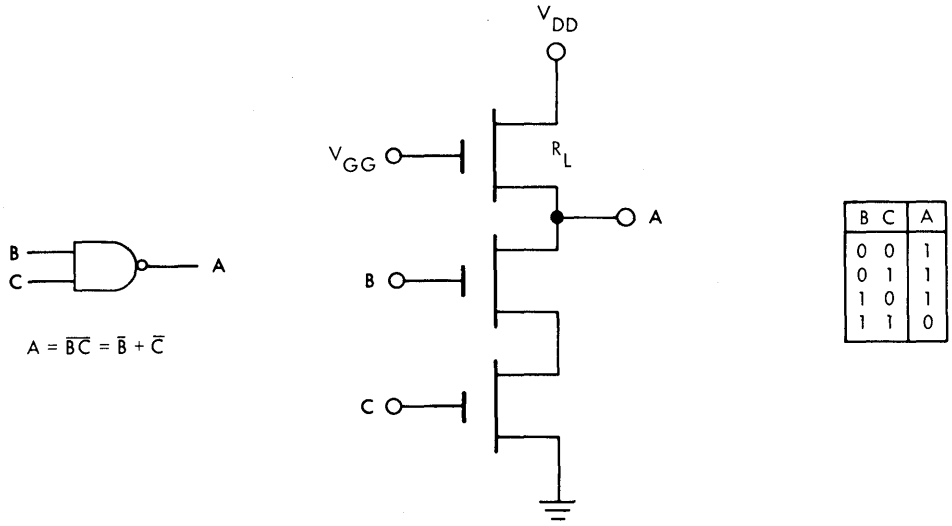


Fig. 3.2 Basic NAND Gate: Logic Diagram, Circuit Diagram and Truth Table

gether with a truth table and Boolean expressions. With a "0" level on either B or C, that inverter is off. Thus no current flows and A is at a "1" level. With a "1" at both inputs, A is "0".

3.1.3 Combined Gating Functions

Basic NOR and NAND circuits are combined to form a wide variety of OR-NAND and AND-NOR gates, each available in a separate block.

The EXCLUSIVE-OR function is also available in a single block, the BB-101.

As shown in Fig. 3.3, two BB-101's, a BB-203 (2/2-Input AND-NOR), and a BB-001 (inverter), can be interconnected to form a full adder.

3.1.4 Output Buffers

Although any of the Building Blocks can be used at outputs, two special buffers, the BB-020 and BB-021 are provided to meet heavy current requirements. Both blocks have an output at the top as well as at the bottom to permit easy access to output pads (see Section 5).

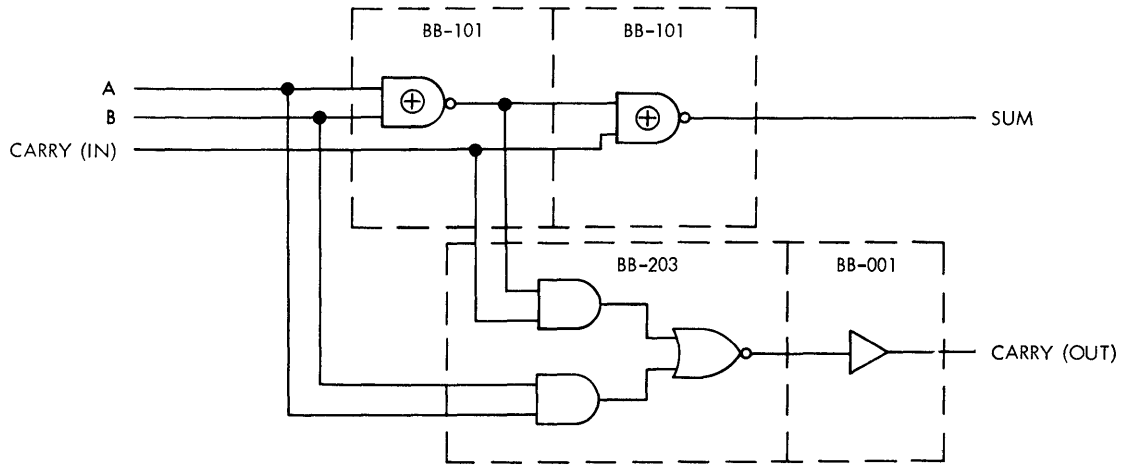


Fig. 3.3 Full Adder.

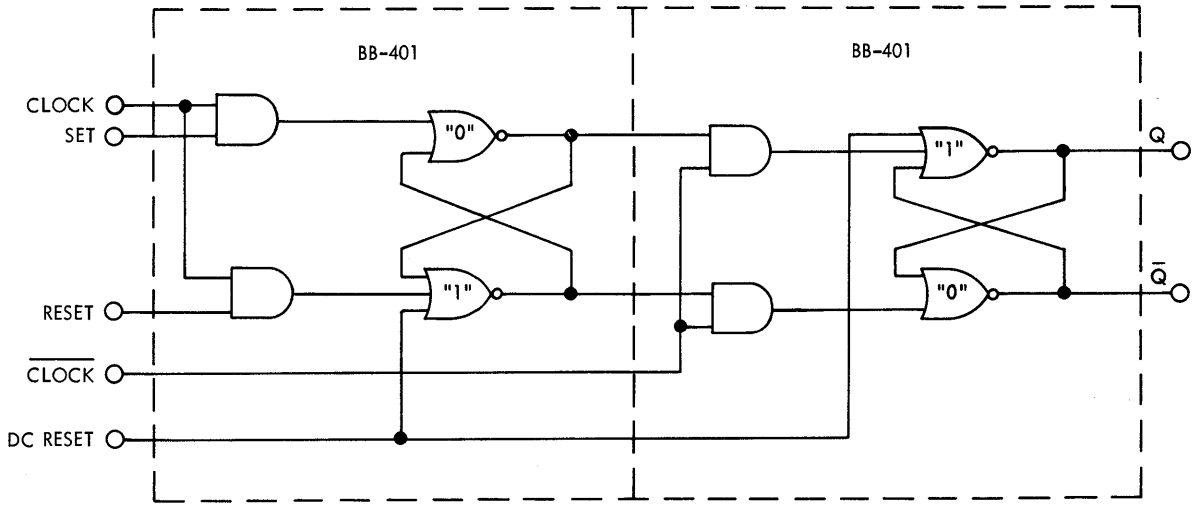
3.2 STORAGE ELEMENTS

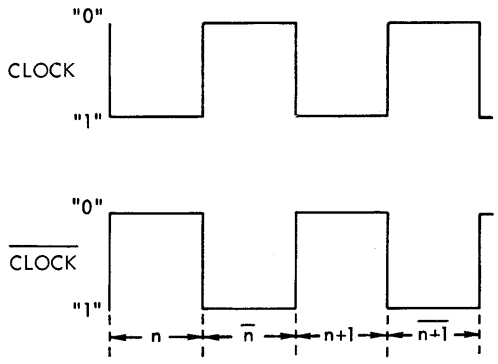
3.2.1 Master-Slave Flip-Flop

A master-slave flip-flop can be constructed by interconnecting two BB-401 blocks as shown in Figure 3.4.

A four-stage ripple-through binary counter formed by cascading master-slave flip-flops is shown in Fig. 3.5, together with a tabulation of the counting sequence. Input pulses are fed into a master-slave clock-generator element (discussed below) and passed through the counter from one stage to the next. Consider stage 2, for example. When the clock input of this stage goes to a "1", a "1" is stored in the master flip-flop because the SET input is negative and the RESET is "0". The slave (output Q) goes to a "1" when the $\overline{\text{clock}}$ input goes negative (i.e., when stage 1 returns to a "0"). Since the SET input of stage 2 is now "0" and the RESET is "1", the master changes to "0" when the clock input goes negative, and the slave follows on the subsequent negative $\overline{\text{clock}}$ input. Thus, each stage changes state when the Q output of the preceding stage goes from "1" to "0".

Since the stages of this counter change state sequentially, there is a finite time during the change when the counter may contain a spurious count. For example, a count of 2 is momentarily present between counts 3 and 4. No special precautions are needed for frequency division since the output is taken only from the final stage. However, when outputs are taken from two or more stages and fed to a logic gate, critical counts should be clocked through the gate.

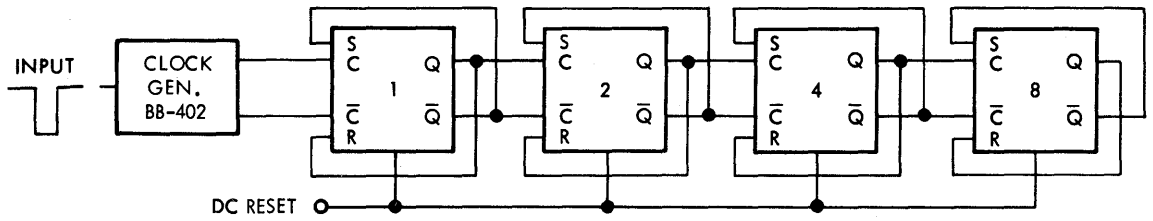




TRUTH TABLE FOR MASTER-SLAVE OPERATION

INPUTS AT CLOCK TIME n+1 SET	RESET	DC RESET	OUTPUT (Q) AT $\overline{\text{n+1 CLOCK}}$ TIME
0	0	0	$Q_{\overline{n}}$
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	X

Fig. 3.4 Logic Diagram, Truth Table and Clocks for Master-Slave Flip-Flop.



8	4	2	1
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0

Fig. 3.5 Four-Stage Binary Ripple-Through Counter Using Master-Slave Flip-Flops.

A shift register can be constructed with master-slave flip-flops as shown in Fig. 3.6. When the clock goes to a "1", the master flip-flop is either set or reset, depending on the data input. This state is passed to the slave when $\overline{\text{clock}}$ goes to a "1", and to the master of the next stage on the subsequent clock signal.

3.2.2 Master-Slave Clock-Generator Element (BB-402)

Logic and schematic diagrams of the master-slave clock-generator element (the BB-402) are shown in the List of Building Blocks. This element is driven by a square wave to generate clock and $\overline{\text{clock}}$ outputs which can never be simultaneously in the same state. The number of flip-flops it can drive is limited only by the speed requirements of the circuit.*

This element cannot be used with the one-bit shift register stages discussed below because the maximum "1" level it provides is V_{DD} (the outputs being taken from the drain of an MOS inverter). As explained in Section 2.2.5, a voltage approaching V_{GG} is required to clock a coupling transistor of the type used in the one-bit shift register stages.

3.2.3 One-Bit Shift Registers (BB-403 and BB-404)

Logic and schematic diagrams of the one-bit shift register (the BB-403) are shown in Fig. 3.7. Since this element requires three externally generated clocks, it consumes considerable space in interconnect. However, the block itself is 70 percent smaller than the master-slave flip-flop discussed above. For three or more register stages, therefore, the BB-403 is usually the better choice.

The action of the coupling transistors T_1 and T_4 is described in Section 2.2.5. Clock phases ϕ_1 and ϕ_2 are complementary. The data input is switched to the gate of T_3 by ϕ_1, and stored on C_{G1} when ϕ_1 turns off and ϕ_2 comes on. After a time equal to the propagation delay through the register, the data state appears at the output. Clock ϕ_3 then turns on, permitting feedback through T_7, which latches the flip-flop into a dc state until the next data input is clocked through ϕ_1.

*Methods of reducing propagation delay through critical nodes are discussed in Section 6.1.3.

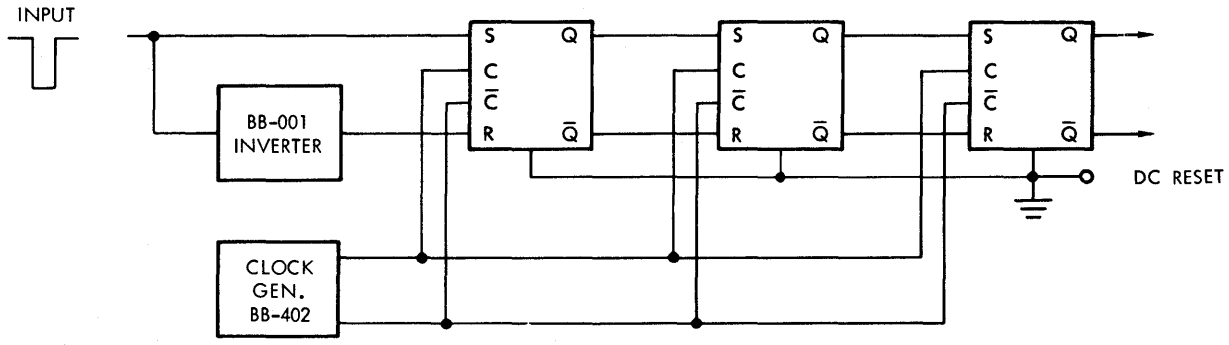


Fig. 3.6 Master-Slave Shift Register.

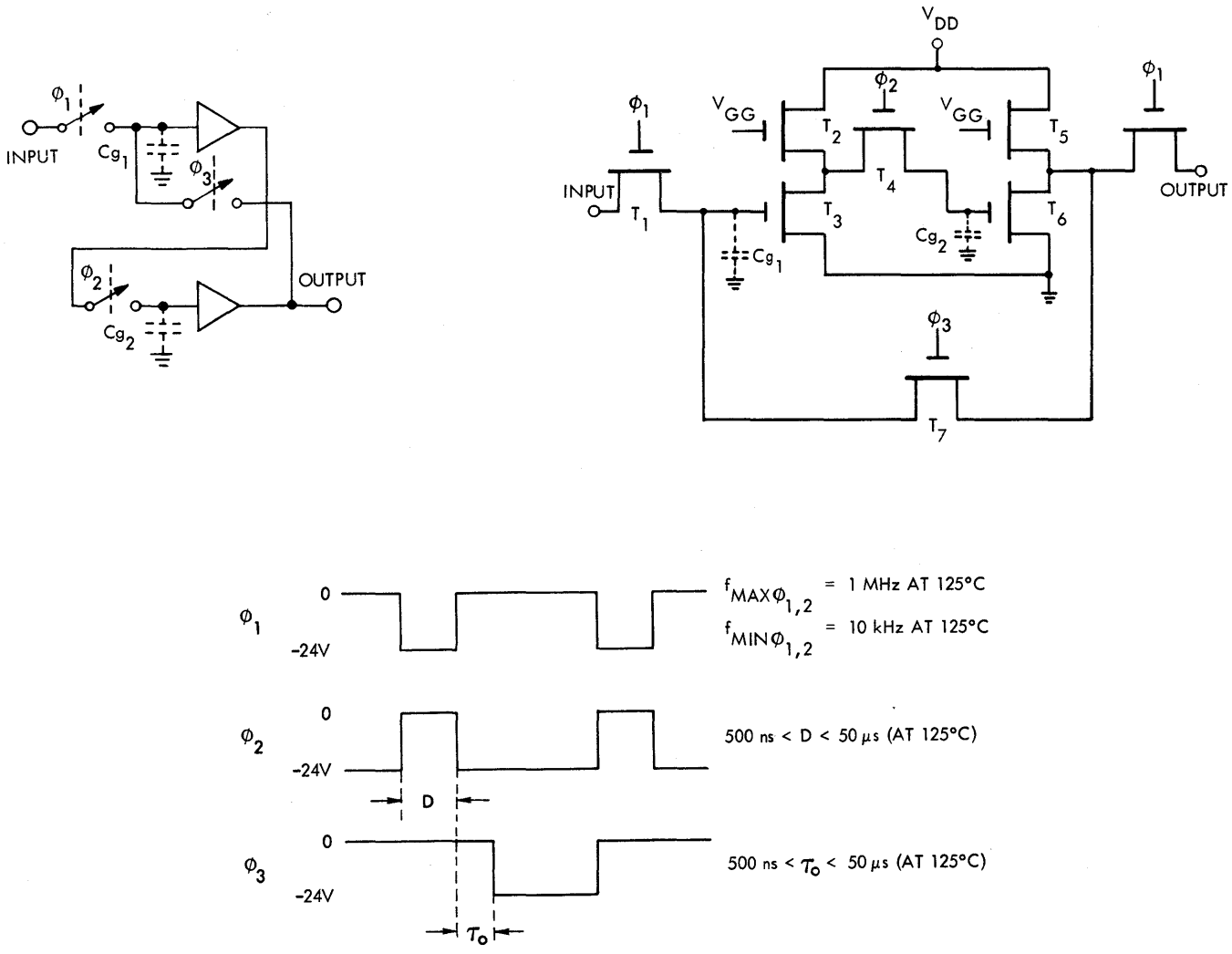


Fig. 3.7 One-Bit DC Shift Register Stage (BB-403) and Clock Waveforms.

Clocks ϕ_1 and ϕ_2 should never be on at the same time since this would permit a change at the input to be clocked prematurely through the register. On the other hand, ϕ_2 may turn off for a brief period before ϕ_1 comes on because the data will be stored on C_{G2} for as long as 50 μs before leaking off through the source-to-body PN junction of T_4 (see Section 2.2.5).

The minimum ON time of either ϕ_1 or ϕ_2 is 500 ns – determined by the propagation delay through the register. Because of the flip-flop latching, there is no upper limit to the time that ϕ_2 may be on. However, its maximum OFF time (and hence the maximum ON time of ϕ_1) is 50 μs – determined by the rate at which data leaks off C_{G2}.

The time τ_o between the disappearance of ϕ_1 and the appearance of ϕ_3 may range from 500 ns to 50 μs at 125°C. The minimum length of τ_o is determined by the propagation delay through the register. The maximum is fixed by the length of time data can be stored on C_{G1} before leaking off. When ϕ_1 and ϕ_2 are operated at rates above approximately 100 kHz, ϕ_3 need not be used at all since C_{G1} will store the input data longer than the entire OFF time of ϕ_1.

In this case, the dynamic shift-register element, the BB-404, shown in Fig. 3.8 should be used since it is somewhat smaller. This element is identical in circuit operation to the BB-403 except that ϕ_3 is eliminated and the load resistors are clocked by ϕ_1 and ϕ_2.* Considerations in laying out this block and the BB-403 are discussed in Section 5.2.

3.3 CUSTOM DESIGNS

To meet certain system requirements, it may be necessary to custom-design one or more blocks (or even an entire chip). This service can be readily obtained from Philco-Ford.

*For detailed information on the circuit operation of this element, see Philco-Ford Application Note No. 401, "pL5R100 Dynamic MOS Shift Register," by James L. Cortright.

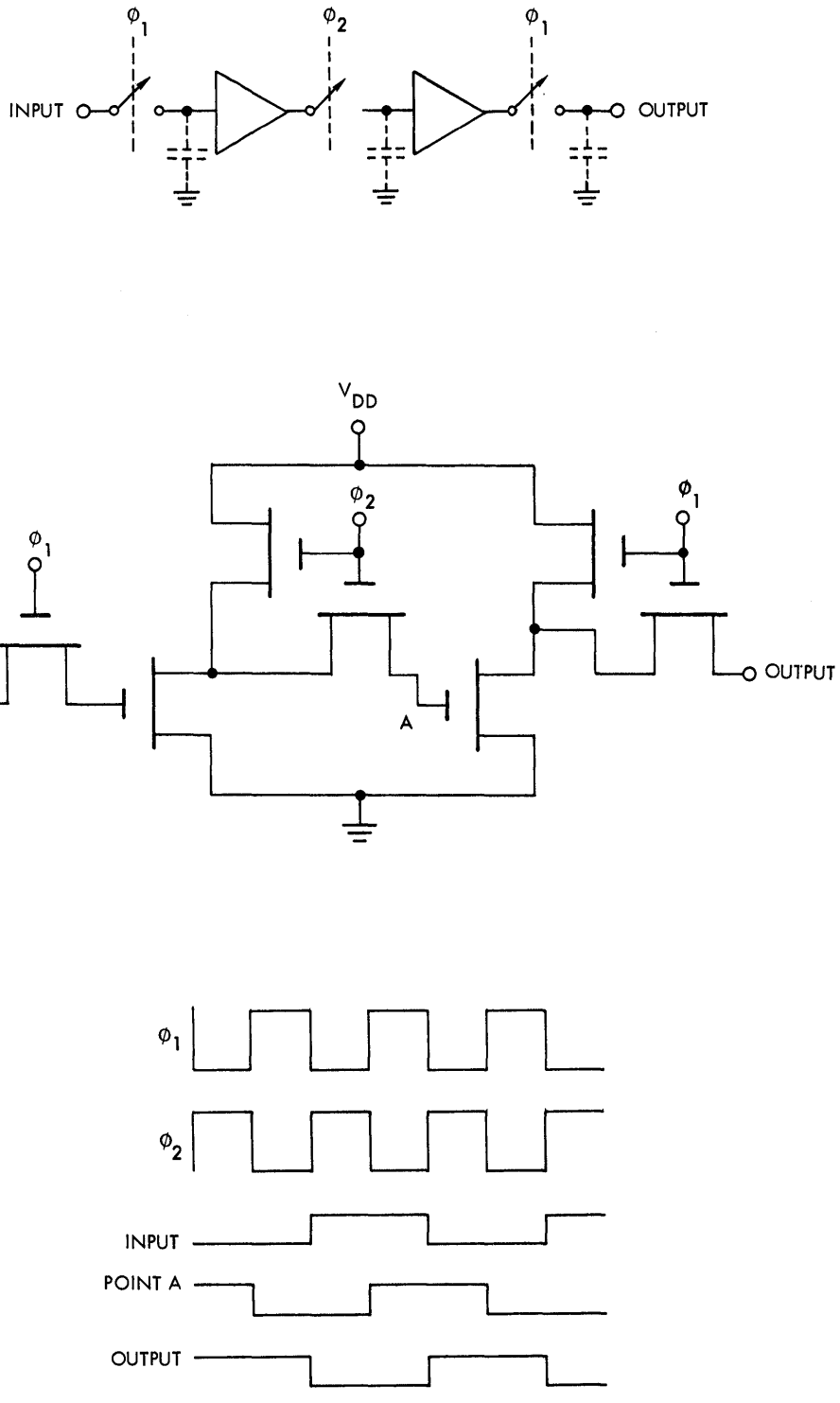


Fig. 3.8 One-Bit Dynamic Shift-Register Stage (BB-404) and Clock Waveforms.

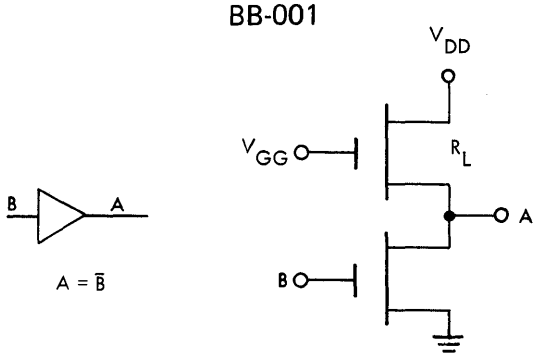
TABLE 3.1

LIST OF BUILDING BLOCKS

I. COMBINATORIAL LOGIC

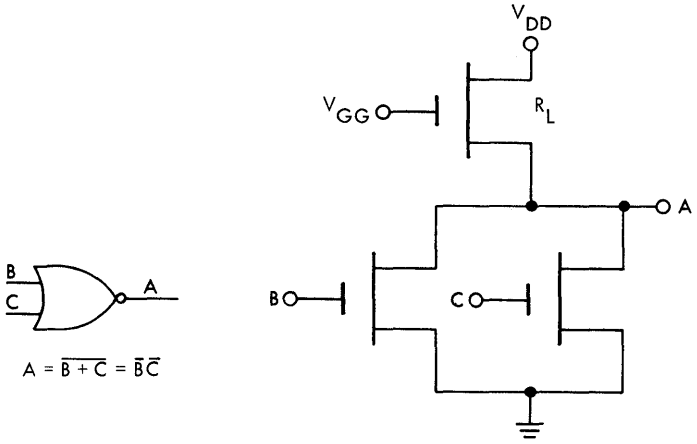
A. Inverter

BB-001



B. NOR Gates

BB-002 (Two-Input NOR)



BB-003 (Three-Input NOR)

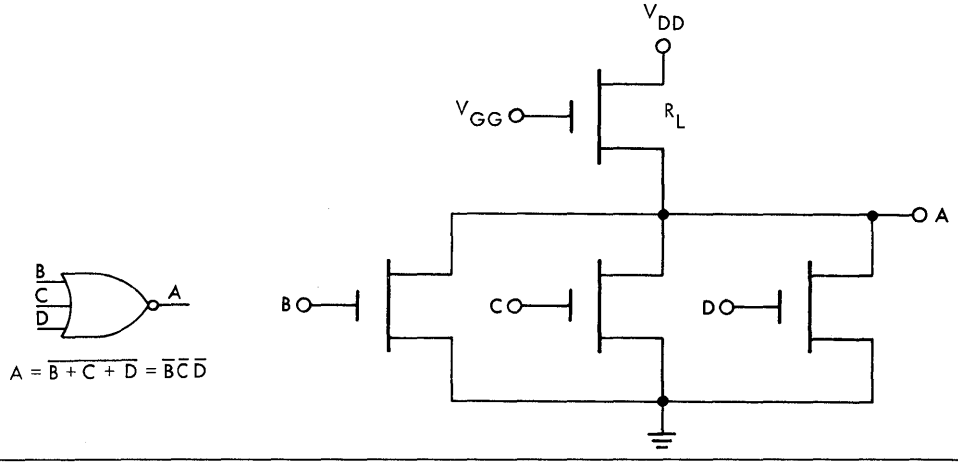
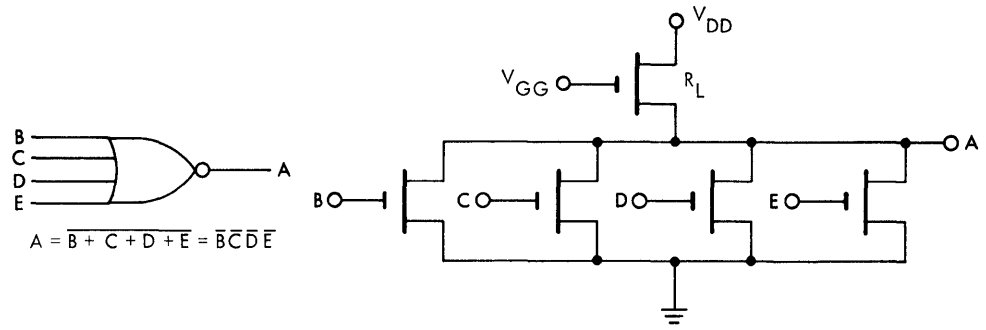


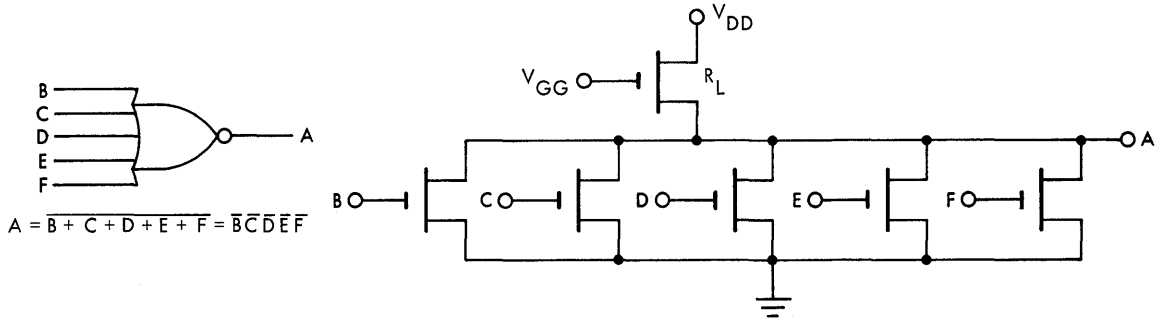
TABLE 3.1

LIST OF BUILDING BLOCKS (Continued)

BB-004 (Four-Input NOR)



BB-005 (Five-Input NOR)



BB-006 (Six-Input NOR)

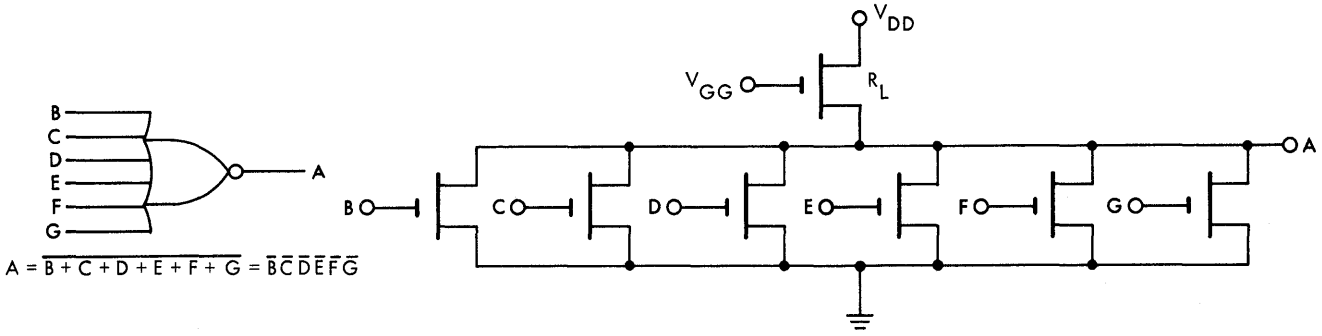
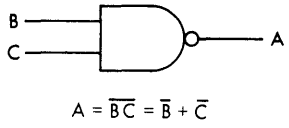


TABLE 3.1

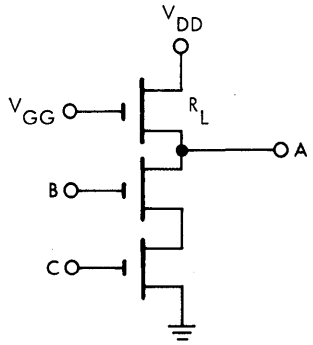
LIST OF BUILDING BLOCKS (Continued)

C. NAND Gate

BB-010 (Two-Input NAND)

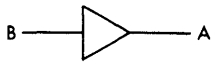


$A = \overline{BC} = \overline{B} + \overline{C}$



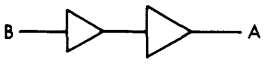
D. Output Buffers

BB-020 (Single Output Buffer)



$A = \overline{B}$

BB-021 (Double Output Buffer)



A = B

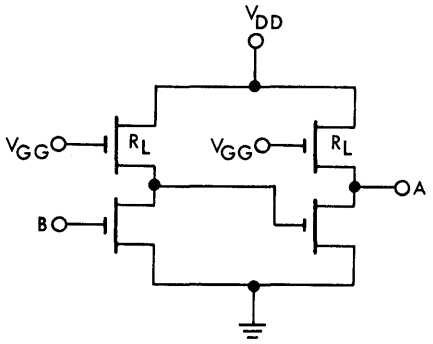
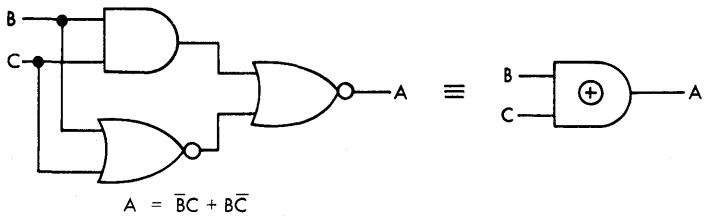


TABLE 3.1

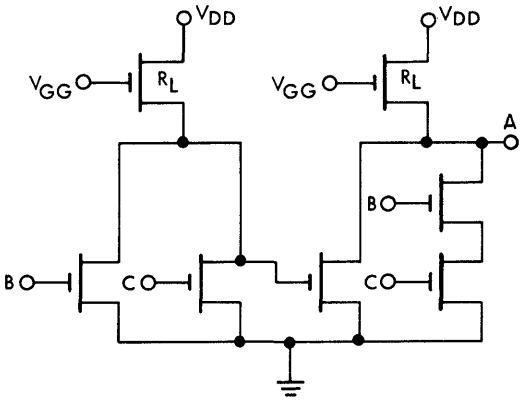
LIST OF BUILDING BLOCKS (Continued)

E. Exclusive OR

BB-101

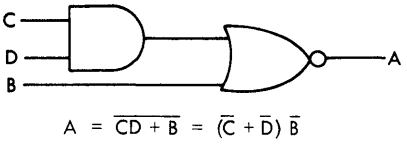


$A = \overline{B}C + B\overline{C}$

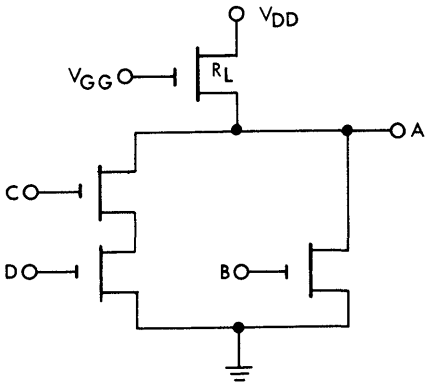


F. AND-NOR Gates

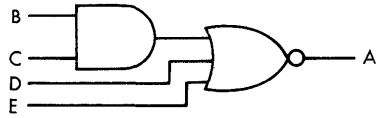
BB-201 (Two-Input AND/One-Input NOR)



$A = \overline{CD + B} = (\overline{C} + \overline{D})\,\overline{B}$



BB-202 (Two-Input AND/Two-Input NOR)



$A = \overline{BC + D + E} = (\overline{B} + \overline{C})\overline{DE}$

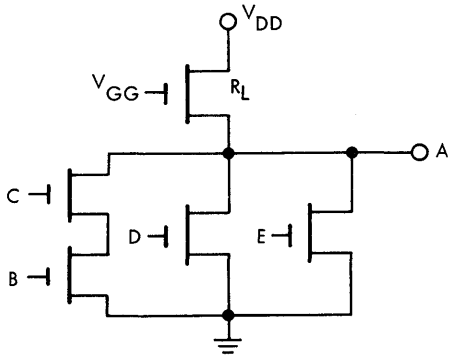
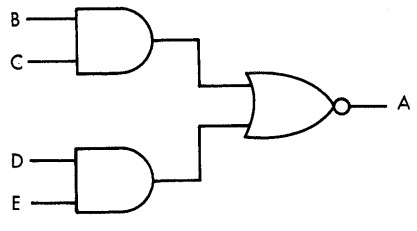
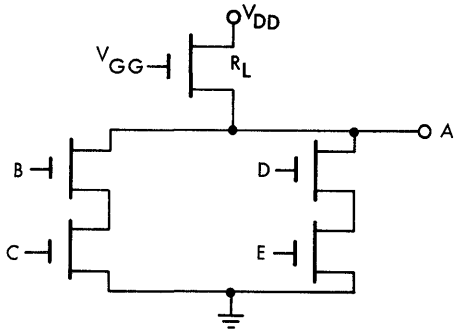


TABLE 3.1

LIST OF BUILDING BLOCKS (Continued)

BB-203 (Two/Two-Input AND-NOR)

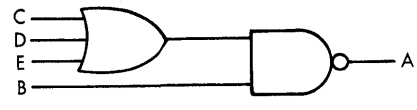


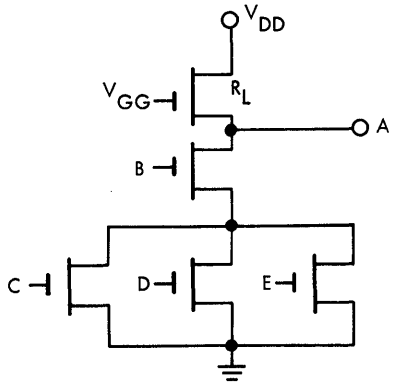


$A = \overline{BC + DE} = (\bar{B} + \bar{C})(\bar{D} + \bar{E})$

G. OR-NAND GATES

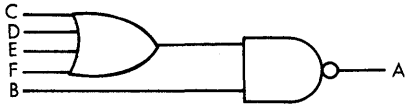
BB-301 (Three-Input OR/One-Input NAND)

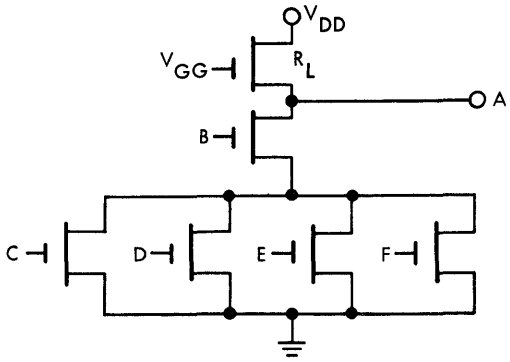




$A = \overline{B(C + D + E)} = \bar{B} + \bar{C}\bar{D}\bar{E}$

BB-302 (Four-Input OR/One-Input NAND)



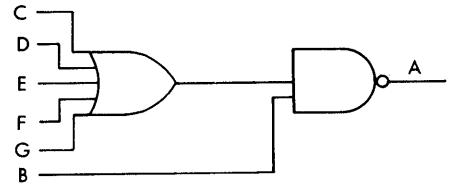


$A = \overline{B(C + D + E + F)} = \bar{B} + \bar{C}\bar{D}\bar{E}\bar{F}$

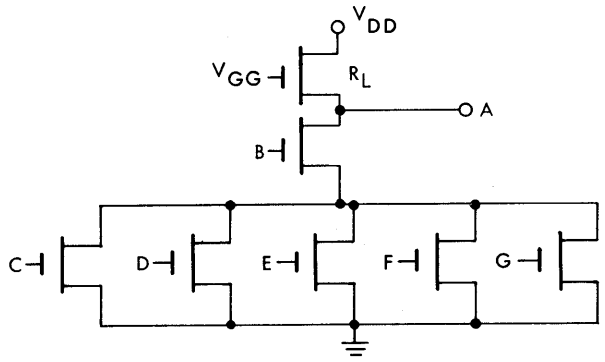
TABLE 3.1

LIST OF BUILDING BLOCKS (Continued)

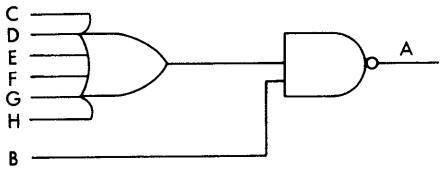
BB-303 (Five-Input OR/One-Input NAND)



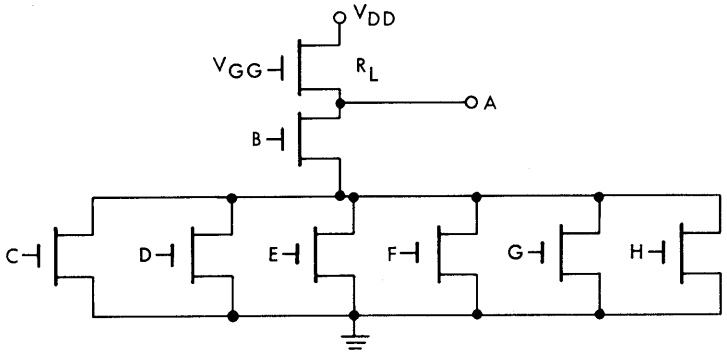
$A = \overline{B(C + D + E + F + G)} = \bar{B} + \bar{C}\bar{D}\bar{E}\bar{F}\bar{G}$



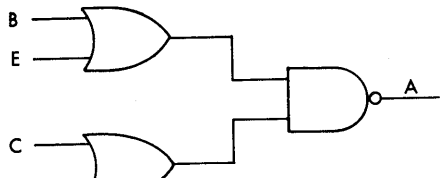
BB-304 (Six-Input OR/One-Input NAND)



$A = \overline{B(C + D + E + F + G + H)}$
$= \bar{B} + \bar{C}\bar{D}\bar{E}\bar{F}\bar{G}\bar{H}$



BB-305 (Two/Two-Input OR-NAND)



$A = \overline{(B + E)\,(C + D)} = \bar{B}\bar{E} + \bar{C}\bar{D}$

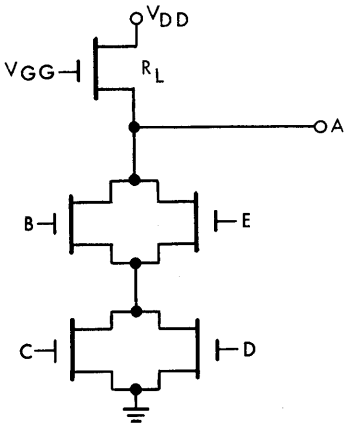
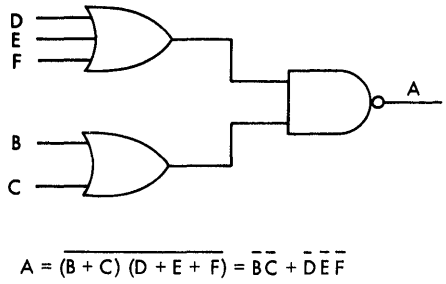


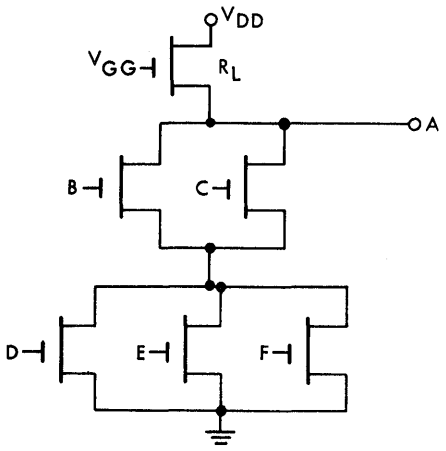
TABLE 3.1

LIST OF BUILDING BLOCKS (Continued)

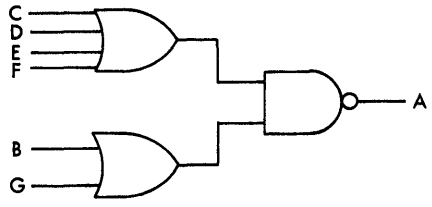
BB-306 (Three/Two-Input OR-NAND)



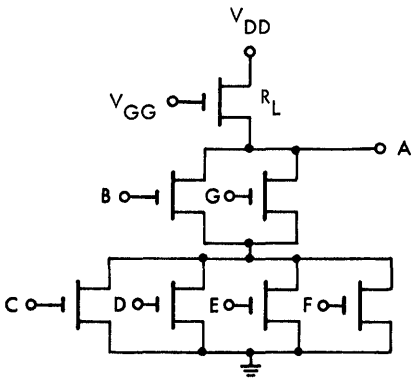
$A = \overline{(B + C)(D + E + F)} = \bar{B}\bar{C} + \bar{D}\bar{E}\bar{F}$



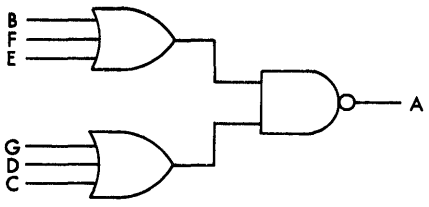
BB-307 (Four/Two-Input OR-NAND)



$A = \overline{(B + G)(C + D + E + F)} = \bar{B}\bar{G} + \bar{C}\bar{D}\bar{E}\bar{F}$



BB-310 (Three/Three-Input OR-NAND)



$A = \overline{(B + F + E)(G + D + C)} = \bar{B}\bar{F}\bar{E} + \bar{G}\bar{D}\bar{C}$

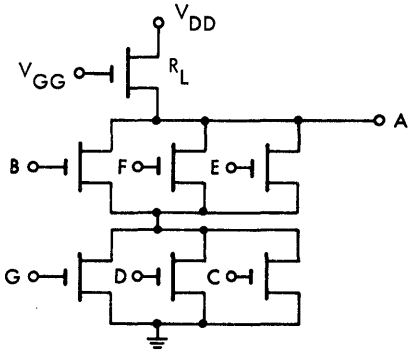
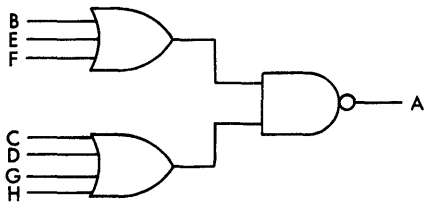
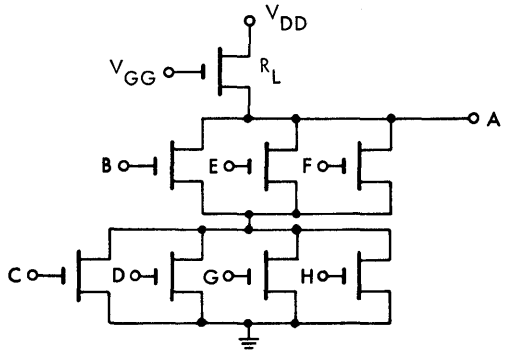


TABLE 3.1

LIST OF BUILDING BLOCKS (Continued)

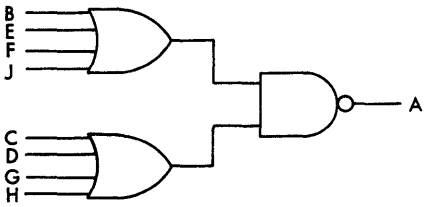
BB-311 (Four/Three-Input OR-NAND)

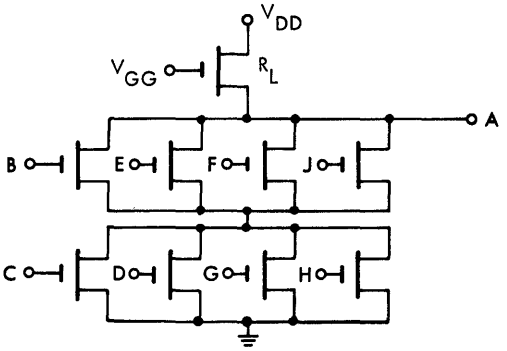




$A = \overline{(B + E + F)(C + D + G + H)} = \bar{B}\bar{E}\bar{F} + \bar{C}\bar{D}\bar{G}\bar{H}$

BB-314 (Four/Four-Input OR-NAND)

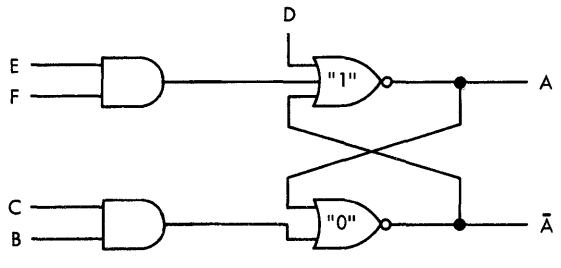




$A = \overline{(B + E + F + J)(C + D + G + H)} = \bar{B}\bar{E}\bar{F}\bar{J} + \bar{C}\bar{D}\bar{G}\bar{H}$

II. STORAGE ELEMENTS

BB-401 (Half Master-Slave Flip-Flop)
(May also be used as a Set-Reset Flip-Flop)



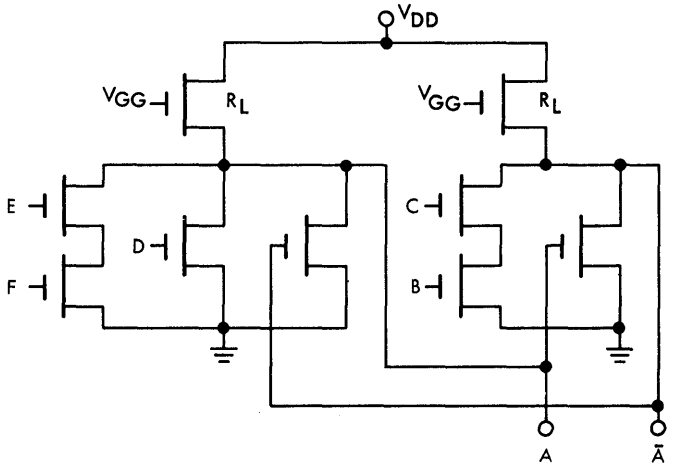


TABLE 3.1

LIST OF BUILDING BLOCKS (Continued)

BB-402 (Master-Slave Clock Generator)

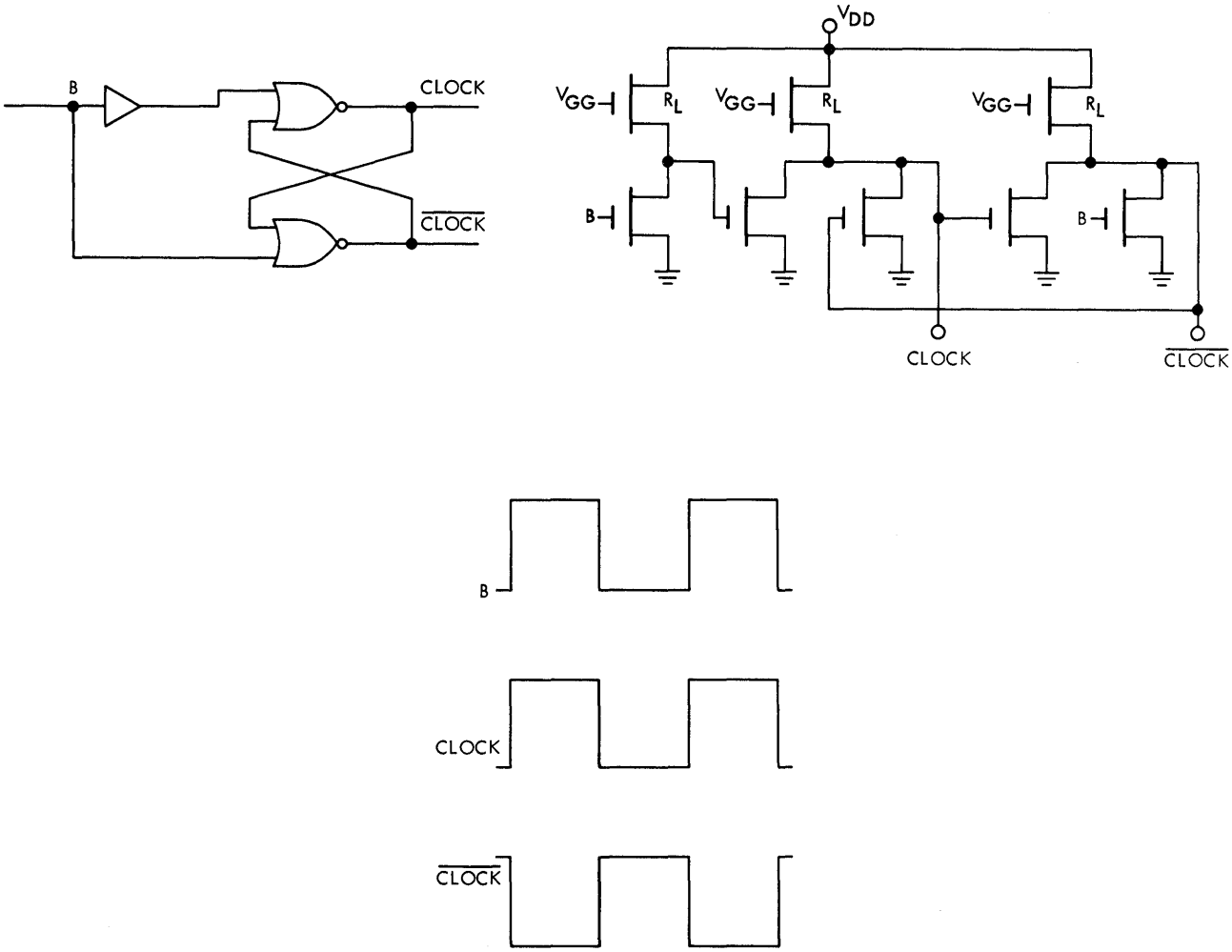
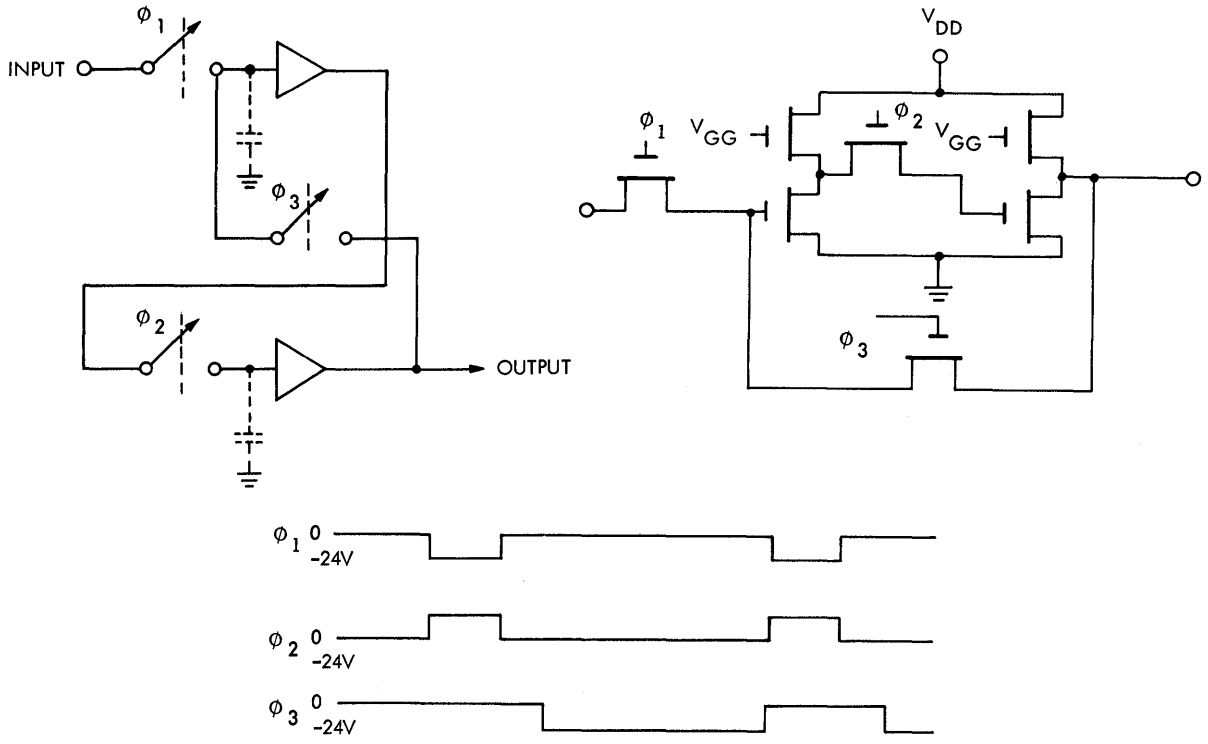


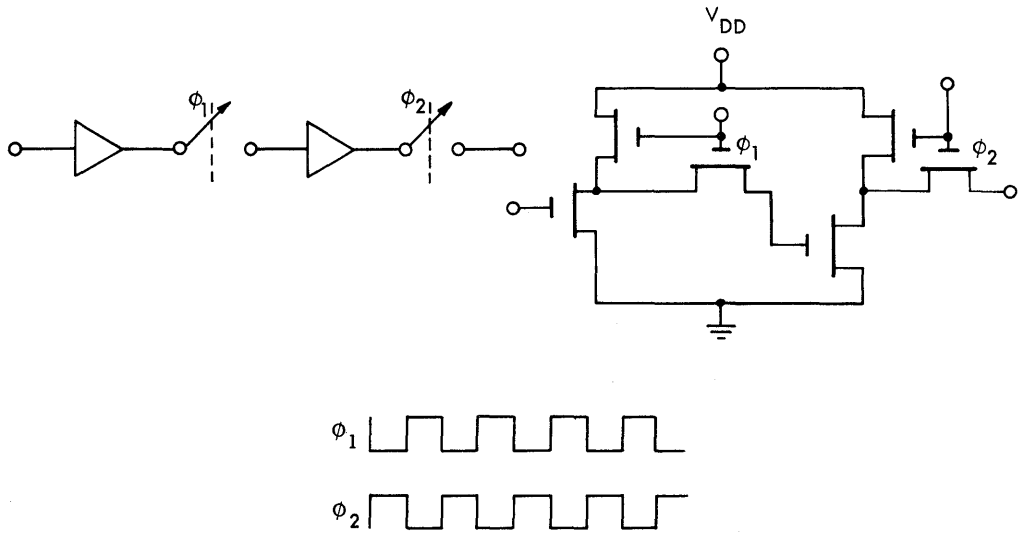
TABLE 3.1

LIST OF BUILDING BLOCKS (Continued)

BB-403 (One-Bit Dc Shift Register)



BB-404 (One-Bit Dynamic Shift Register)



Section 4

PARTITIONING THE MOS BUILDING-BLOCK SYSTEM

In partitioning any MOS system, the principal considerations are pin limitation and chip area. The pin limitation is fundamental. Obviously no chip should be designed with more inputs and outputs than are available on the package. From the standpoint of cost, however, chip area is usually the primary concern. As shown in Fig. 4.1, the production cost of a chip increases nearly exponentially with its area. (The range of costs for any given chip area is due to other variables, such as package, testing, area utilization, etc.)

In this section, we will describe two simple partitioning tradeoffs that can be used to minimize area in a Building-Block system. Then we will discuss the relationship between the number of transistors in a logic function and the estimated chip area it will occupy. Finally, we will present a flow chart of the steps necessary to design and partition an MOS Building-Block system.

4.1 TRADEOFFS FOR MINIMIZING CHIP AREA

The area cost of inputs and outputs is high because each one requires a 120 μ X 120 μ bonding pad. When, as often happens, a large output buffer is also required (see Section 3.1.4), the area cost is even higher. Therefore, to conserve silicon area, the number of inputs and outputs should be minimized.

One method of accomplishing this is shown in Fig. 4.2. Instead of partitioning the logic as shown in Fig. 4.2a, it actually saves area to use an extra block, as in Fig. 4.2b, and generate the complement within the second chip. This eliminates one output and one input, saving at least two pad areas and possibly also an output buffer.

To keep interconnections within one chip rather than letting them run from chip to chip, redundancy may be desirable in some cases. Figure 4.3 shows how a decoder might be used redundantly in a Building-Block system to minimize leads. To choose between the two alternatives, you would need to calculate and compare the area of the extra decoder, in the one case, against that of the extra inputs and outputs, in the other.

4.2 ESTIMATING CHIP SIZE

It is impossible to predict precisely how much chip area will be consumed in interconnections between

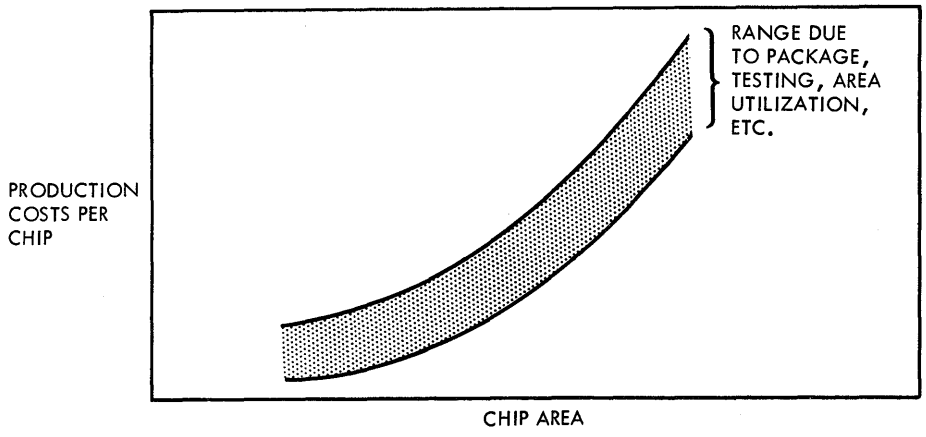


Fig. 4.1 Graph of Relationship Between Production Costs per Chip and Chip Area.

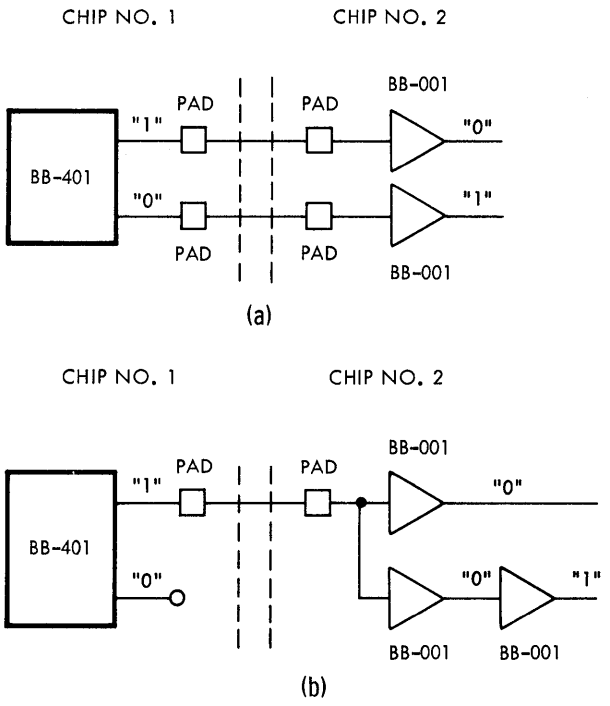


Fig. 4.2 Trading Building Blocks for Leads.

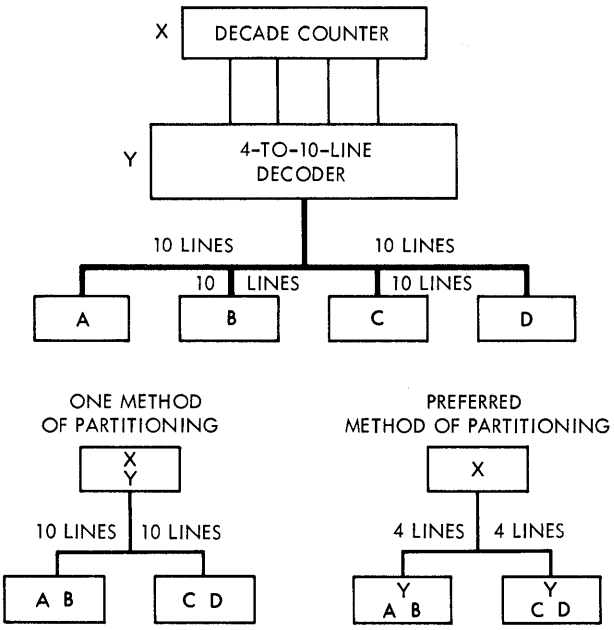


Fig. 4.3 Minimizing Area by Redundancy.

Building Blocks. Different logic functions vary widely in this respect. A decoder, for example, will probably occupy far less space than a feedback type of control unit having the same device count. As a rough guideline in partitioning, however, we can assume a chip density of one transistor* per 31,250 square microns, including scribe lines (see Section 5), interconnections, pads, and other "waste" area. If the layout is carefully planned, this is a reasonably good rule of thumb for logic of medium complexity. As a rough estimate, then, if the Building Blocks you put on your chip contain a total of 100 transistors, you would expect the chip area to be about 3,125,000 square microns (5,000 square mils).

4.3 SUGGESTED STEPS IN PARTITIONING A BUILDING-BLOCK SYSTEM

In light of the above discussion, we can outline the following steps for optimum partitioning of an MOS Building-Block system (see also flow chart in Fig. 4.4):

1. Study the Boolean expressions in terms of the available Building Blocks and regroup them if necessary. For example, the expression

   $$ABCDE = \overline{Z}$$

   cannot be implemented in negative logic (true = -V) with MOS Building Blocks. However, reexpressed by DeMorgan's Theorem in the form

   $$\overline{A} + \overline{B} + \overline{C} + \overline{D} + \overline{E} = Z$$

   it can be implemented with the BB-005 five-input NOR gate.

2. Choose one of the available packages (see Appendix C).

3. Partition the logic, bearing in mind the economic advantages of reducing area and the above guideline for estimating chip size.

4. Lay out the logic on each chip by one of the methods described in Section 5.

5. If one of the partitioned sections cannot be laid out on a single chip of the maximum size (see Table 5.1, item 5.1.4), divide it into two or more subsections and lay it out again.

*By "transistor," we mean any kind of MOS transistor – whether an inverter, load, or coupling device.

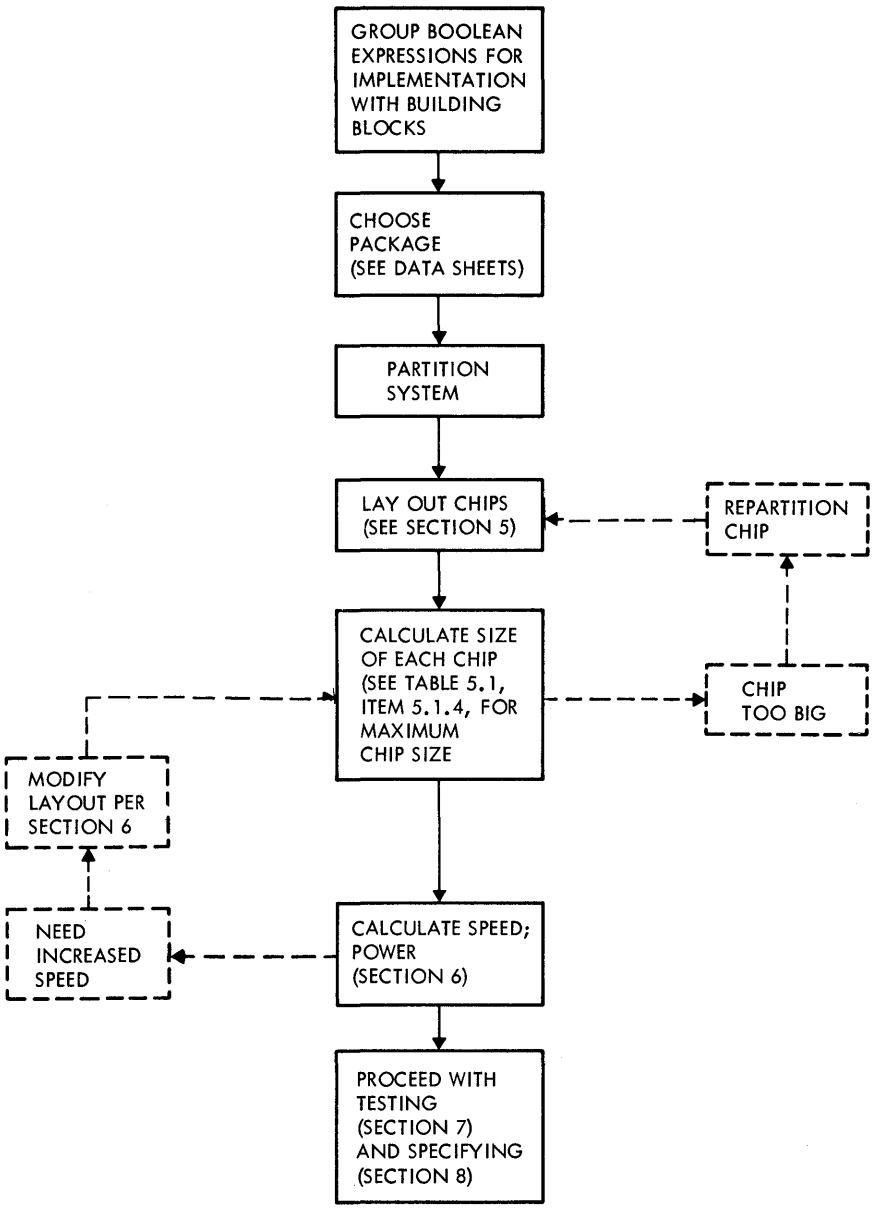


Fig. 4.4 Flow Chart of Steps in Partitioning and Laying Out a Building-Block System.

6. Calculate the speed and power as explained in Section 6. The power dissipation is unlikely to exceed the package rating. If you require higher speed through any portion of the system, you will need to modify the layout by one of the methods discussed in Section 6.1.3.

Section 5

LAYOUT

Laying out a Building-Block chip is easy. You merely place scaled (250:1) decals of the required Building Blocks one after another in rows on scaled (1 mm = 4 μ) paper; then draw interconnecting lines according to the rules given in Table 5.1.* If you prefer not to use decals, you can draw your own scaled outlines of the Building Blocks (as shown in the data sheets) in rows on the grid paper.

Procedures for laying out both the standard blocks and the one-bit shift-register blocks (the BB-403 and BB-404) are discussed in this section. Simple methods of calculating line resistance and capacitance are presented in Section 6.1.1.

Although a functioning chip can be laid out without even considering the order in which blocks feed into each other, the number of crossovers – and often the chip area as well – can be minimized by grouping together, insofar as possible, blocks that interconnect. Therefore, block order is discussed in this section, and the blocks required to implement a given logic diagram are arranged by two different methods.

5.1 LAYOUT PROCEDURE AND RULES

Fig. 5.1 shows a decal of the BB-001 Inverter Block with metal power supply lines (V_{DD}, V_{GG}, and ground) running through it at right angles to its longitudinal axis, and P-region signal lines running out the bottom. In accordance with layout convention (see Table 5.1), metal lines are drawn solid and P lines dashed. The decal is scaled 250:1 and dimensioned in the actual block size.

*Both decals and grid paper are provided by Philco-Ford in kits (See Appendix B for kit contents). Additional decals and paper can be obtained by contacting the Marketing Services Department of Philco-Ford.

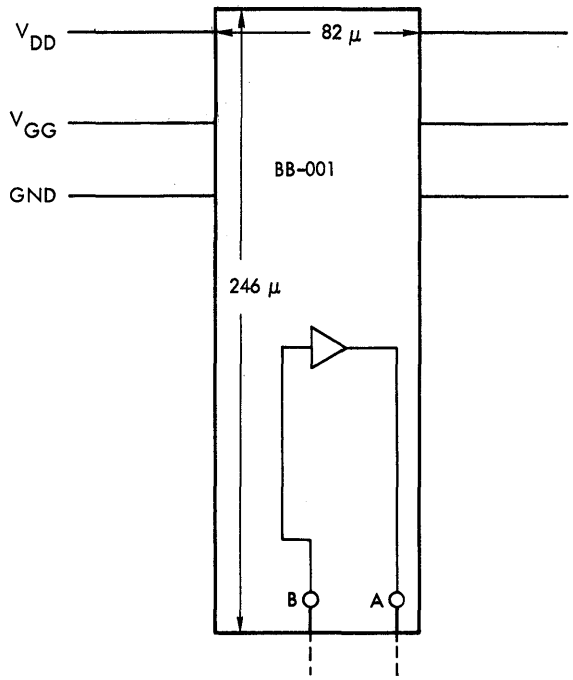


Fig. 5.1 Block Outline of BB-001.

The other DC Building Blocks are identical to the BB-001 except in width and in the number of P-region signal lines that extend from the bottom. Thus they can be arranged in rows, as shown in Fig. 5.2. Vertical signal lines from the blocks connect with horizontal lines that run parallel to the blocks. Horizontal lines are all metal (drawn solid), whereas the vertical lines consist of P-regions (dashed lines) for a short distance out of each block, as well as in crossover areas, where they must be isolated from the metal lines (see also Section 2.2.4 on crossovers). Elsewhere, vertical lines should be metal, if possible, to minimize line resistance and capacitance. Ground and V_{DD} lines must be of continuous metal; V_{GG} lines should also be of continuous metal if possible. These directions are repeated in Table 5.1, Section 5.1.8.

TABLE 5.1

LAYOUT RULES

5.1.1 Drawing Conventions

NOTE: All lines shall be parallel to the grid lines on the layout paper.

5.1.1.1 **Metal lines:** Solid

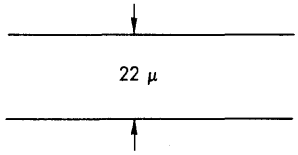
5.1.1.2 **P-region lines:** Dashed

5.1.1.3 **Interconnection:** Dot

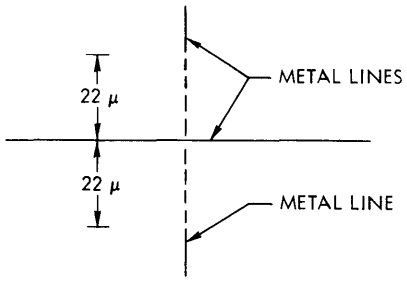
5.1.1.4 **Crossover:** No Dot

5.1.2 Spacings Between Lines and Blocks

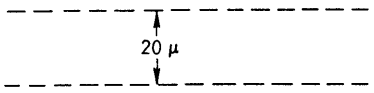
5.1.2.1 **Distance between parallel metal lines:** 22 μ



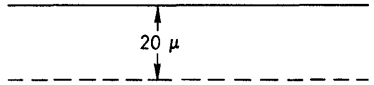
5.1.2.2 **Distance between perpendicular metal lines:** 22 μ



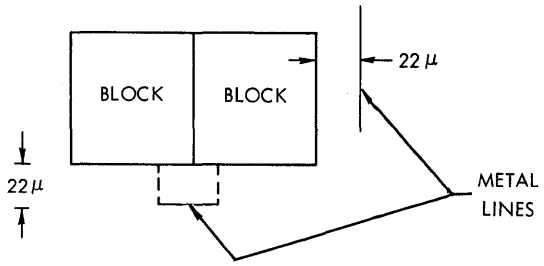
5.1.2.3. **Distance between P lines:** 20 μ



5.1.2.4 **Distance from metal to P line:** 20 μ



5.1.2.5 **Distance from Building Block to nearest metal line:** 22 μ



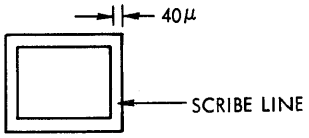
(Hence minimum length of P signal line from Block: 22 μ)

TABLE 5.1

LAYOUT RULES (Continued)

5.1.3 Scribe Line

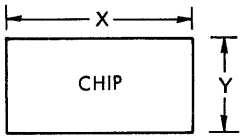
5.1.3.1 Width: 40 μ



5.1.3.2 **Distance from inside edge of scribe line to adjacent bonding pad, Building Block, P or metal line:** 50 μ

5.1.4 Chip Dimensions

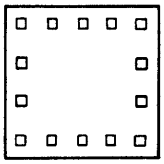
5.1.4.1 **Maximum Aspect Ratio:**
X : Y 1.67



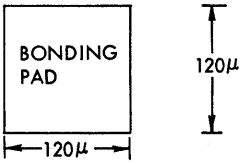
5.1.4.2 **Maximum Size:**
See Appendix C

5.1.5 Bonding Pads

5.1.5.1 **Placement:** Pads must be placed around all four sides of the chip. For optimum layout, they should be evenly spaced.



5.1.5.2 **Pad dimensions:** 120 μ X 120 μ



5.1.5.3 **Distance from pad to anything:** $\geqslant$ 50 μ

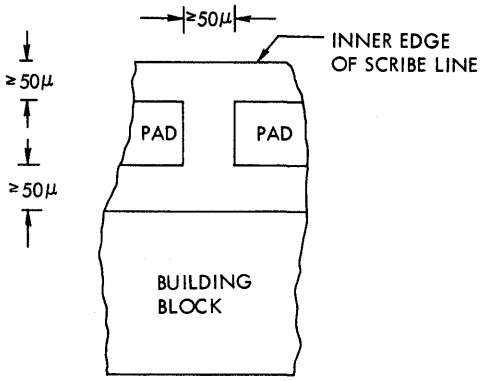
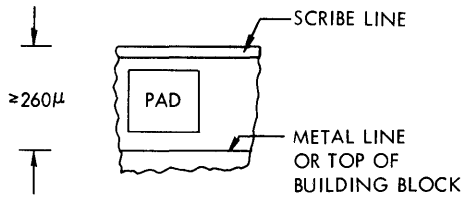


TABLE 5.1

LAYOUT RULES (Continued)

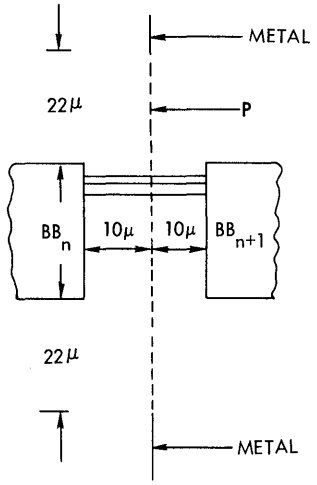
5.1.6 Margin at Chip Periphery: $\geqslant$ 260 μ



5.1.7 Rules for Placement of Lines Between Building Blocks*

5.1.7.1 **Distance from P line to either block:** 10 μ

5.1.7.2 **Line between blocks shall be P**



*This should never be done with V_{DD}, V_{GG}, ground or clock lines.

5.1.8 General Rules

5.1.8.1 Use metal lines rather than P lines wherever possible.

5.1.8.2 V_{DD} and ground lines must be of continuous metal.

5.1.8.3 No metal lines should be placed between pads and scribe line.

5.1.8.4 Horizontal lines are all metal; vertical lines are P for a distance of 22 μ out of each block, in crossover areas, and between blocks. Elsewhere, vertical lines should be metal.

5.1.8.5 Bonding pads should be labeled (V_{DD}, V_{GG}, signal inputs, outputs, etc.).

5.1.8.6 If you draw in the outlines of the Building Blocks instead of using decals, identify each block by part number (BB-001, etc.).

5.1.8.7 Pads may be placed in any convenient order.

5.1.8.8 The dimensional integrity of your layout is extremely important because the accuracy of the photographic masks depends on it. Therefore, the paper on which it is drawn should be handled with care to avoid creasing or tearing.

5.1.8.9 For instructions in layout of BB-403 and BB-404, see Section 5.2.

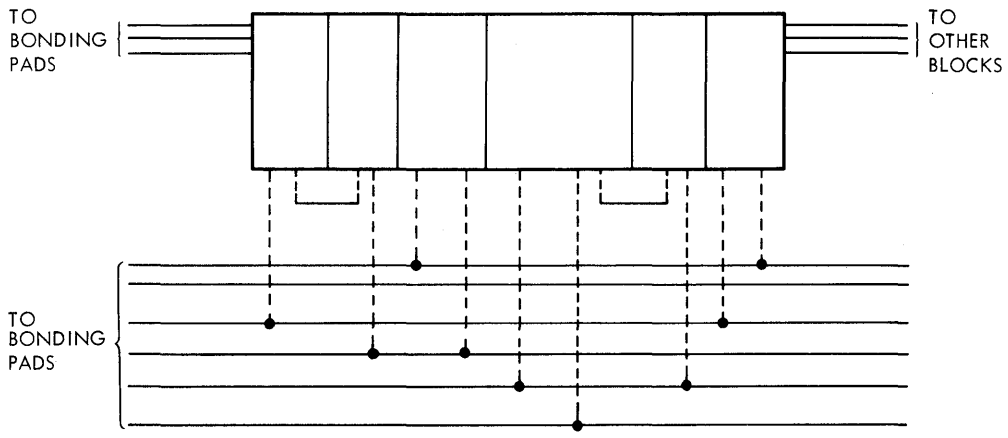


Fig. 5.2 Block Interconnect Pattern.

As shown in Fig. 5.3, the interconnect lines run out to large metal pads at the periphery of the chip. Wires bonded to these pads connect the circuitry to the package leads.

The dimensions of the chip and the placement of pads on it are specified in Table 5.1. The maximum aspect ratio (item 5.1.4.1) is fixed by manufacturing requirements: If a chip is too elongated, it may break in dicing or assembly. As stated under item 5.1.5.1, pads must be placed around all four sides of the chip. Otherwise, the wires connecting the pads to the package leads might cross over each other. For optimum layout, the pads should be evenly spaced. As stated under 5.1.6, a border of 260 μ must be left along each edge of the chip: 40 μ for the scribe line,* 50 μ from the scribe line to the pad, 120 μ for the pad itself, and 50 μ from the pad to the nearest interconnect line or Building Block.

The size of the chip depends on the package (see Appendix C). In the 14- and 22-lead flatpacks, for example, neither dimension should be longer than 2,500 μ. The maximum size, therefore, is 2,500 μ X 2,500 μ. In the TO-5 can, the maximum chip size is 1,250 μ X 1,250 μ.

*A safety margin left for the scribing and dicing operations.

To illustrate the spacing rules, we will calculate the Y dimension necessary to accommodate the layout shown in Fig. 5.3.

Starting from the top of the chip:

From chip edge to nearest metal line (item 5.1.6)	260 μ
Space consumed by 4 metal lines (item 5.1.2.1)	88 μ
Length of blocks (see data sheets)	246 μ
From blocks to first metal line (item 5.1.2.5)	22 μ
Space consumed by 9 metal lines	198 μ
Length of blocks	246 μ
Space consumed by 4 metal lines	88 μ
From last metal line to chip edge	260 μ
Total Y Dimension	1,408 μ

When there are too many Building Blocks to be placed in two rows without unduly elongating the chip, then the layout shown in Fig. 5.4 is recommended instead of the one discussed above.

In both layouts, metal lines extend around the rows of blocks to pads on the periphery of the chip (except in the case of output buffers, which have an output at the top). It is also possible to connect a

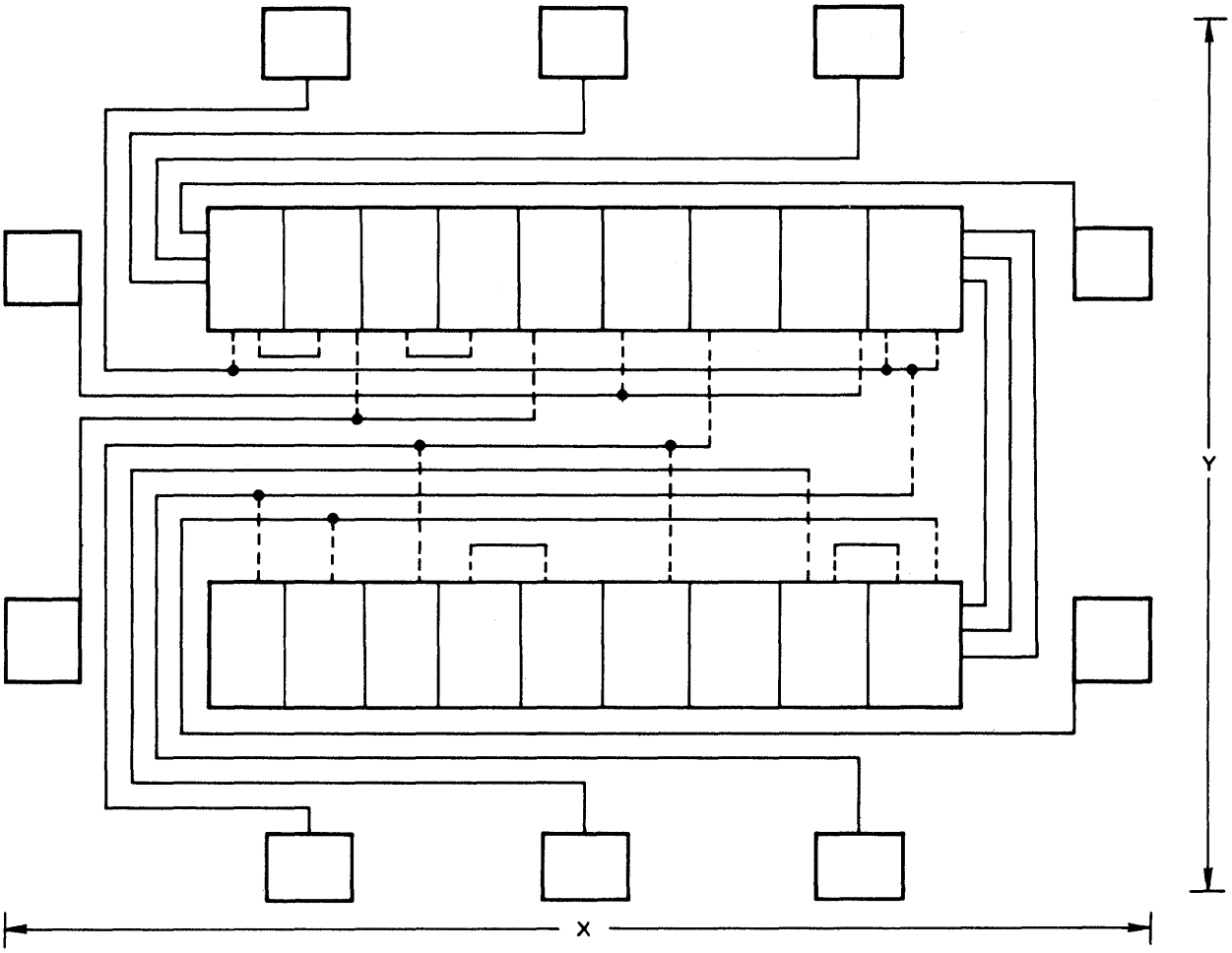


Fig. 5.3 Block Layout with Bonding Pads at Periphery of Chip

signal line to a pad by running a P line between two Building Blocks, as shown in Table 5.1, item 5.1.7.* (Of course, this will increase the overall length of the row of blocks and therefore may not save chip area.)

You should label the pads on your layout: V_{DD}, ground, signal inputs and outputs, etc. Pads may be placed in any convenient order.

If you draw in the outlines of the Building Blocks instead of using decals, you should identify each block by number: e.g., BB-001, BB-002, etc.

*This should never be done with V_{DD}, V_{GG}, ground or clock lines.

Caution: The dimensional integrity of your layout is extremely important because the accuracy of the photographic masks depends on it. Therefore, the paper on which it is drawn should be handled with care to avoid creasing or tearing.

5.2 LAYOUT OF ONE-BIT SHIFT REGISTERS

The decal of the BB-403, the one-bit dc shift register, is shown in Fig. 5.5. Note that both clock and signal lines are attached to the bottom of the block. In general, to avoid clock-to-signal-line crossovers, you should not place a set of these blocks in the center of a row of other blocks. The clock amplitudes are at least twice as large as the internal signal levels, and rise times are usually rapid; thus internal crosstalk or capacitive coupling can be minimized by avoiding crossovers. In addition, to avoid distorting the clock waveform you should use

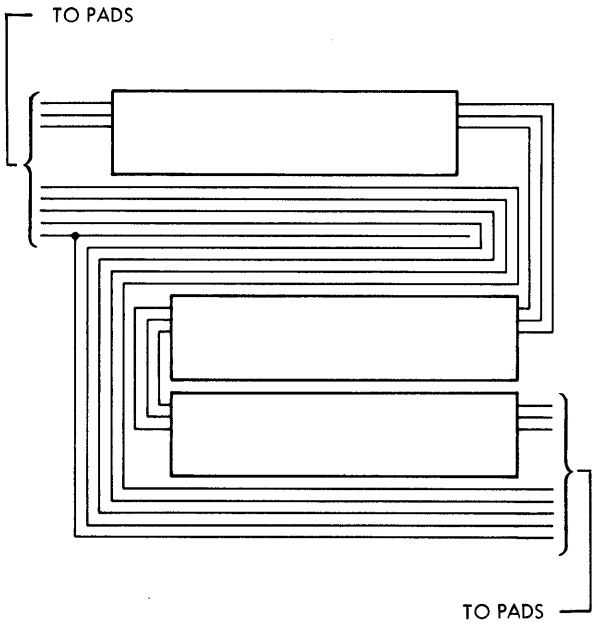


Fig. 5.4 Layout Suggested as Alternative to Very Long Rows of Blocks.

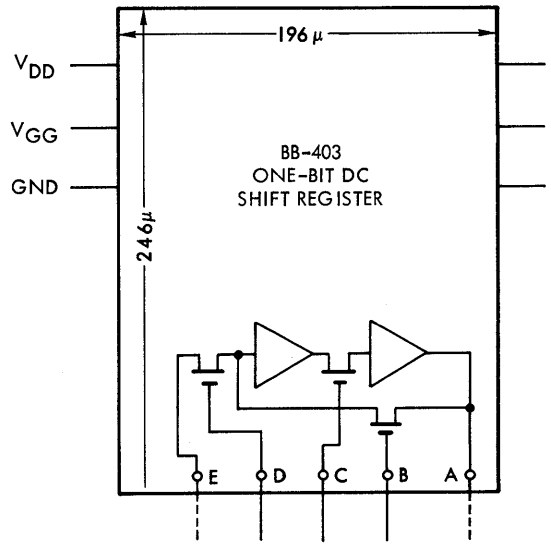


Fig. 5.5 Block Outline of BB-403, the One-Bit DC Shift Register.

continuous metal for the clock lines insofar as possible. (See Section 6.1.1.2 for resistances of interconnecting lines.)

The same considerations apply also to the BB-404 (the dynamic shift register) shown in Fig. 5.6. However, since the clock lines run through this

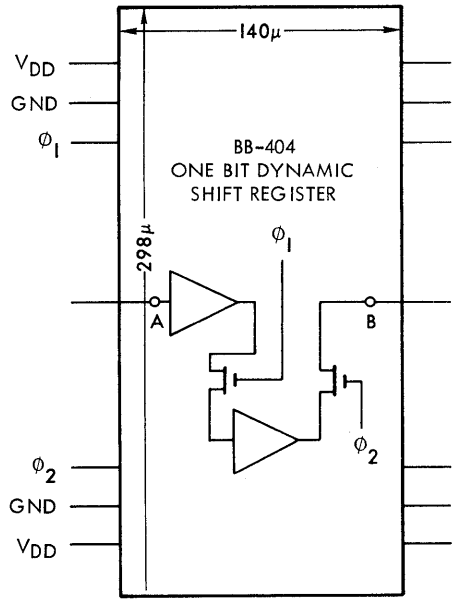


Fig. 5.6 Block Outline of BB-404, the One-Bit Dynamic Shift Register.

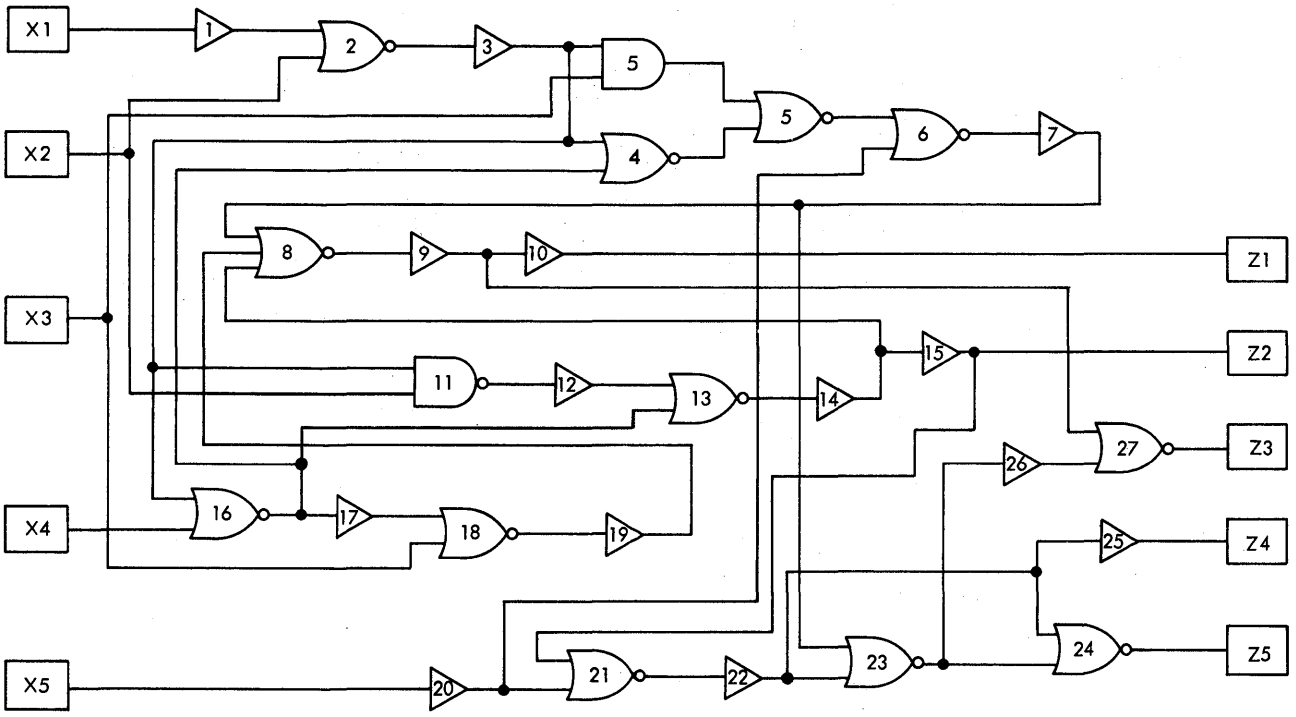
block, parallel to the power supply lines, interconnection is simplified. Also these blocks can be plugged directly into each other without running signal lines from one to another.

5.3 OPTIMUM ORDER OF BLOCKS

As stated above, a functioning chip can be designed without considering the order in which the blocks are placed. In this case, however, the number of crossovers – and probably also the area required for interconnects – will be larger than necessary. The order can be improved just by placing the blocks "by eye" more or less in the order of logic flow, but to minimize the number of crossovers, a matrix method is usually necessary. As a comparison, we will lay out a set of logic blocks by the two approaches.

5.3.1 Method 1: Arrangement "By Eye"

Let's assume that the hypothetical logic diagram in Fig. 5.7 is to be placed on one chip. The Building Blocks (see parts list in Fig. 5.7) required to implement the logic should be numbered more or less in the order in which the logic functions feed into each other. Then, as shown in Fig. 5.8, they are arranged serially – 1, 2, 3, 4, etc., from left to right in the first row, and from right to left in the second row, and interconnect lines are drawn in.



1. BB-001
2. BB-002
3. BB-001
4. BB-002
5. BB-201
6. BB-002
7. BB-001
8. BB-003
9. BB-001
10. BB-001 or BB-020
11. BB-010
12. BB-001
13. BB-002
14. BB-001
15. BB-001 or BB-020
16. BB-002
17. BB-001
18. BB-002
19. BB-001
20. BB-001
21. BB-002
22. BB-001
23. BB-002
24. BB-002
25. BB-001 or BB-020
26. BB-001
27. BB-002

Fig. 5.7 Hypothetical Logic Diagram to be Laid Out on a Chip.

5.3.2 Method 2: Arrangement by Simple Matrix (Hand Manipulation)

In the layout diagram in Fig. 5.8,* there are 66 crossovers. This number can be reduced by arranging the blocks so that each one is as close as possible to those with which it interconnects. To accomplish this, the logic diagram is subdivided into sections (see Fig. 5.9) which contain both pads and blocks. In fact, for this purpose, pads are considered to be the same as blocks. A section consists of a chain of blocks which terminates either (1) at an output which feeds into two or more blocks or (2) at an output pad. Each block (except the final one) in the section feeds into only one other block. However, a section may have an indefinite number of inputs. For example, Section A in Fig. 5.9 has two inputs – the input from the package lead to Pad X_1 (included in Section A) and an input from Pad X_2 (Section O). Section A incorporates Blocks 1, 2, and 3, which feed into each other serially, and terminates at the output of Block 3, which feeds into Blocks 4, 5, 11 and 16. Pad X_2 is considered a separate section because its output goes to two blocks – 2 and 11.

Notice that Section C appears to include two separate subsections. However, since Block 19 in one subsection actually feeds into Block 8 in the other, the two subsections may be treated as one.

The sections are arranged alphabetically in a matrix, as shown in Fig. 5.10, the outputs along the vertical axis at the left, and the inputs along the horizontal axis. Interconnections are indicated by X's. For example, Section A feeds into Sections B, E,

*Note that the relative length of the blocks has been considerably shortened in this diagram.

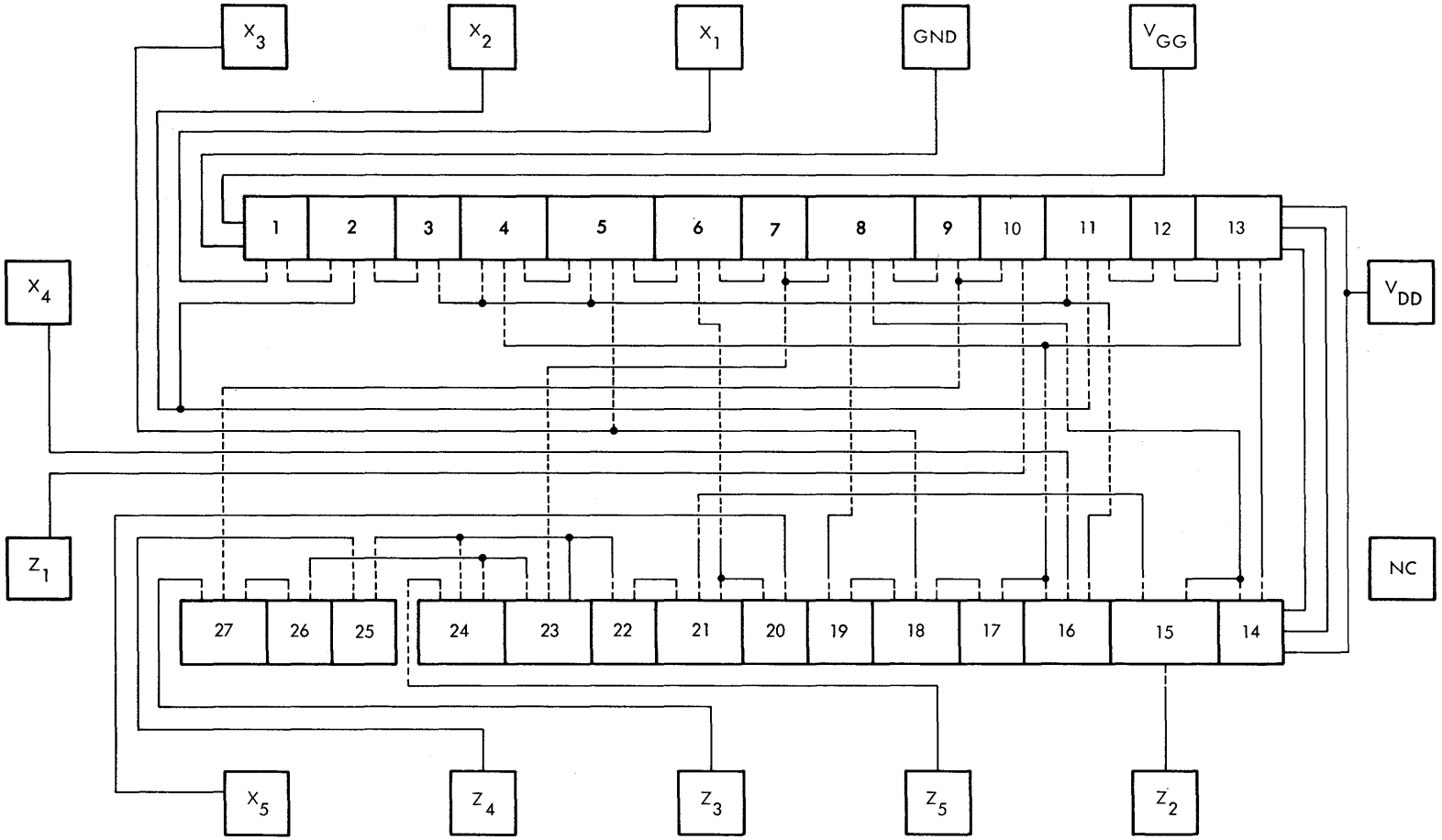


Fig. 5.8 Layout "By Eye" of Logic in Fig. 5.7.

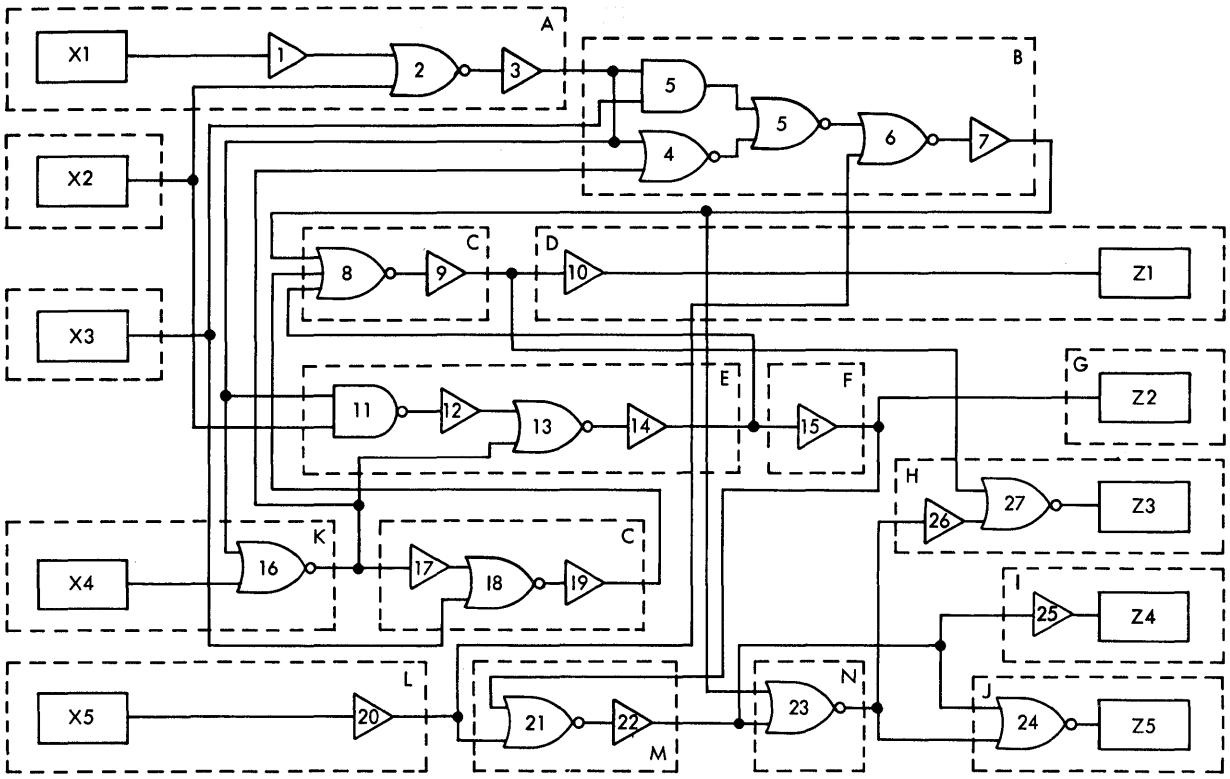


Fig. 5.9 Partitioning Logic for Layout by Simple Matrix.

and K, which are therefore marked with X's along the horizontal "A" line.

By examining the matrix in Fig. 5.10, you can see that the output from Section A interconnects with B, then crosses over lines extending from C and D before it interconnects with E. Five further crossovers (F, G, H, I, and J) occur before the line reaches K. To minimize crossovers, the X's should line up as close as possible to the diagonal, and on either or both sides of it. Two dashed lines are therefore drawn close to the diagonal to mark the region in which most of the X's should fall. A second matrix is then drawn, with the horizontal letters rearranged to bring as many X's together as possible. The vertical axis is rearranged in the new sequence. Usually more X's will fall within the dashed lines in the second matrix than in the first. The arrangement is then further improved by drawing a third or fourth matrix, if necessary.

Figure 5.11 shows an improved matrix, and Fig. 5.12 is a wiring diagram of the blocks laid out in this order. Note that there are only 44 crossovers in Fig. 5.12, about two-thirds the number incurred by using Method 1.

5.3.3 Method 3: Arrangement of Matrix by Computer

If the complexity of the logic prevents hand manipulation of the matrix, the sections can be placed in optimum order by computer at Philco-Ford.

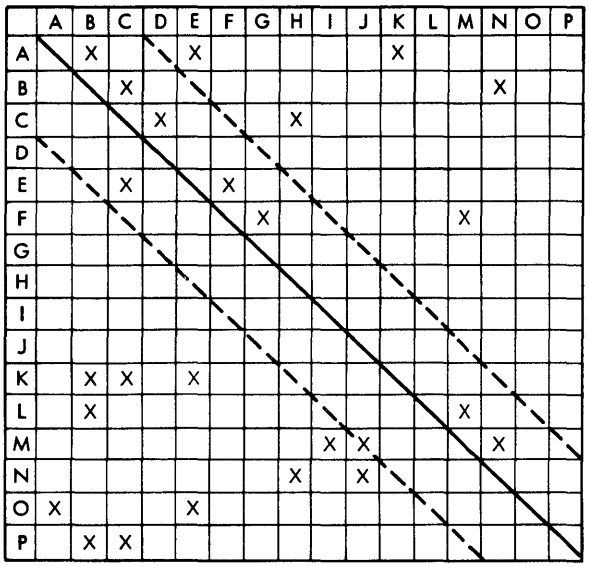


Fig. 5.10 Preliminary Matrix Showing Non-Optimized Crossover Pattern.

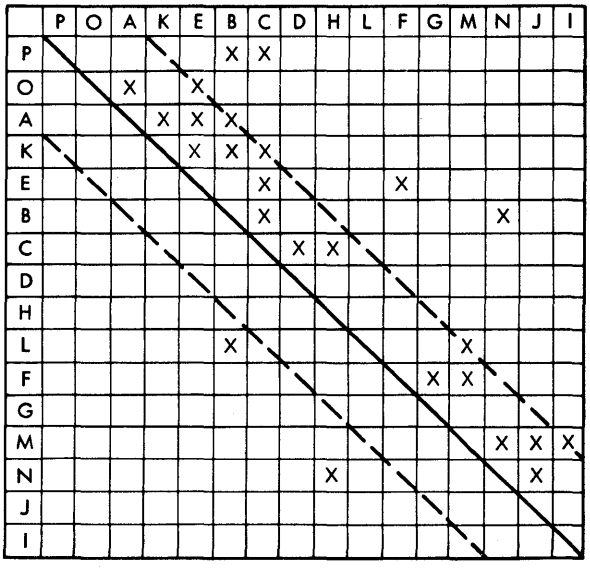


Fig. 5.11 Optimized Matrix Corresponding to Layout in Fig. 5.12.

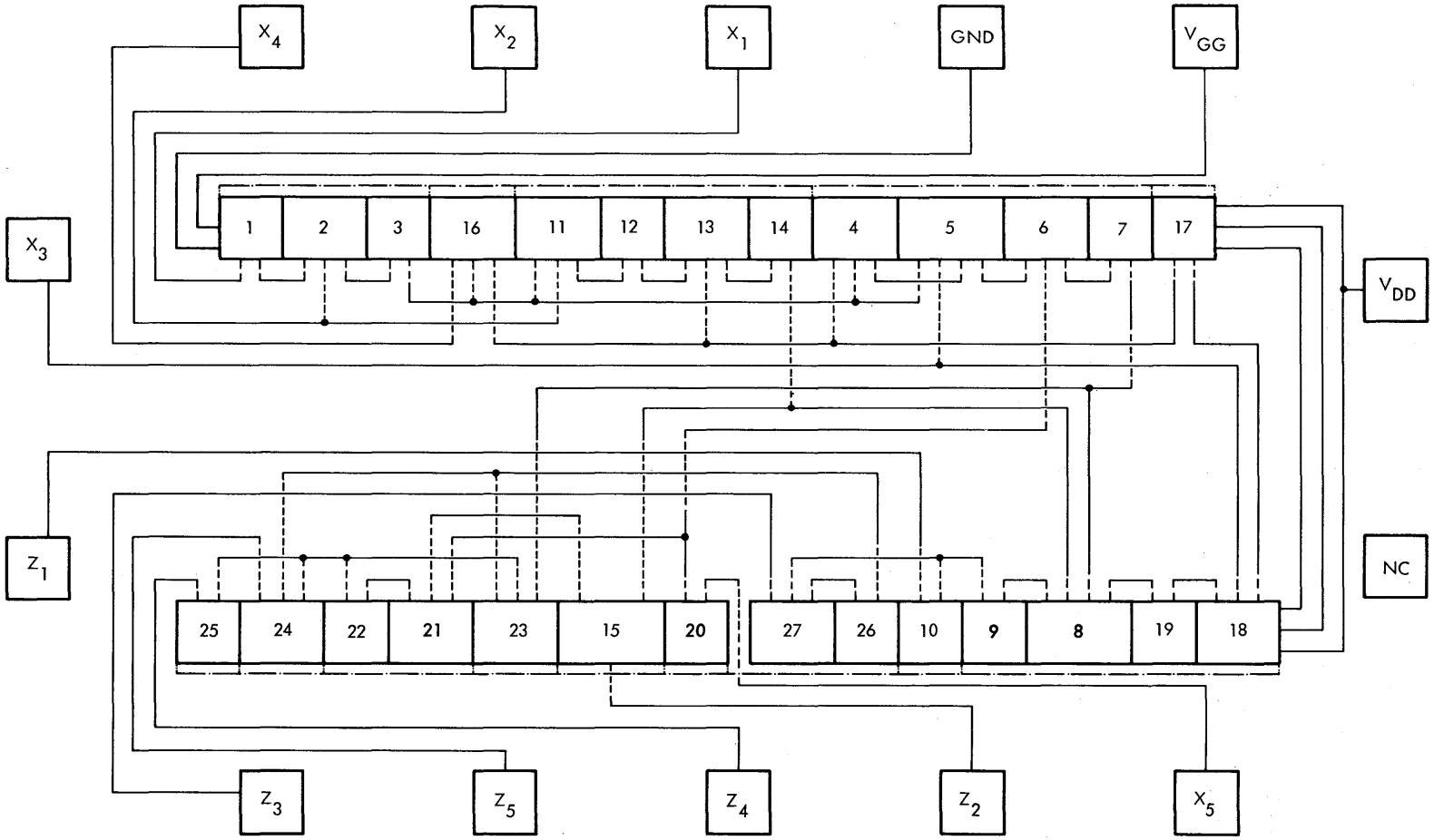


Fig. 5.12 Layout of Logic in Fig. 5.7 by Simple Matrix.

Section 6

PERFORMANCE CALCULATIONS

After the blocks have been laid out and interconnected, the next step is to determine the performance. In this section, we will consider propagation delay, power dissipation, and the effect of temperature on both.

6.1 PROPAGATION DELAY

6.1.1 General

Speed through a chain of Building-Block inverters is limited by the time required to charge the node capacitance through the high-resistance load transistor. As discussed in Section 2.2.1, the charging waveform is considerably slower with an MOS load than with a simple linear resistor. (The two waveforms are shown for comparison in Fig. 6.1. Note that the load source takes 3 time constants to reach a "1" level.)

The time required to discharge the load capacitance through the low-resistance inverter is considerably shorter than the time required to charge it through the load transistor. With a "1" level step voltage on the gate of a Building-Block inverter (Fig. 6.2), its drain rises to a "0" in a time equal to $0.3\,R_LC_L$, where R_LC_L is the time constant of the stage (referred to the load resistance for convenience).

For an internal stage, of course, the inverter gate waveform is the degraded turn-off waveform shown

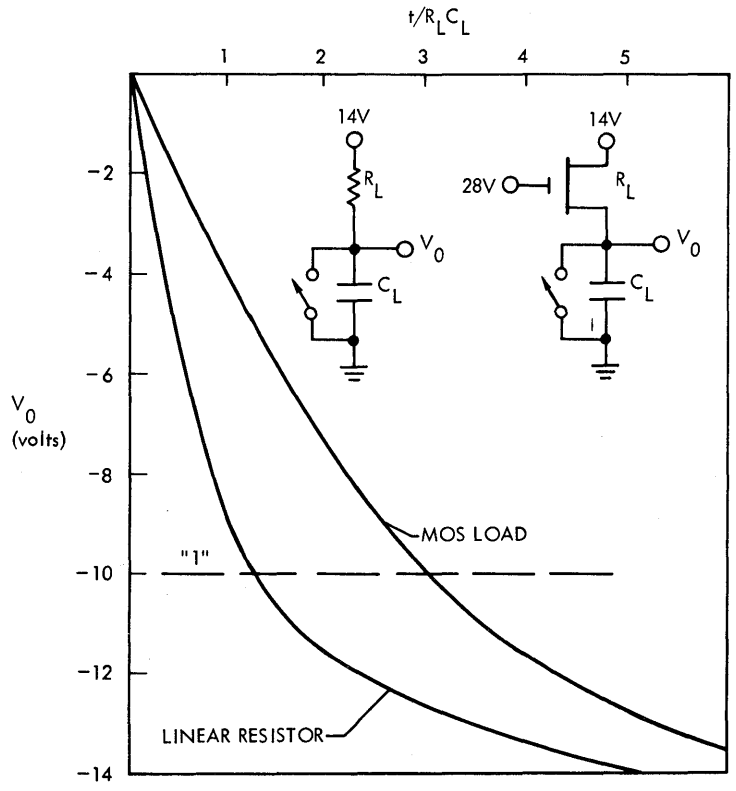


Fig. 6.1 Typical Turn-off Waveforms with MOS Load and Simple Linear Resistor.

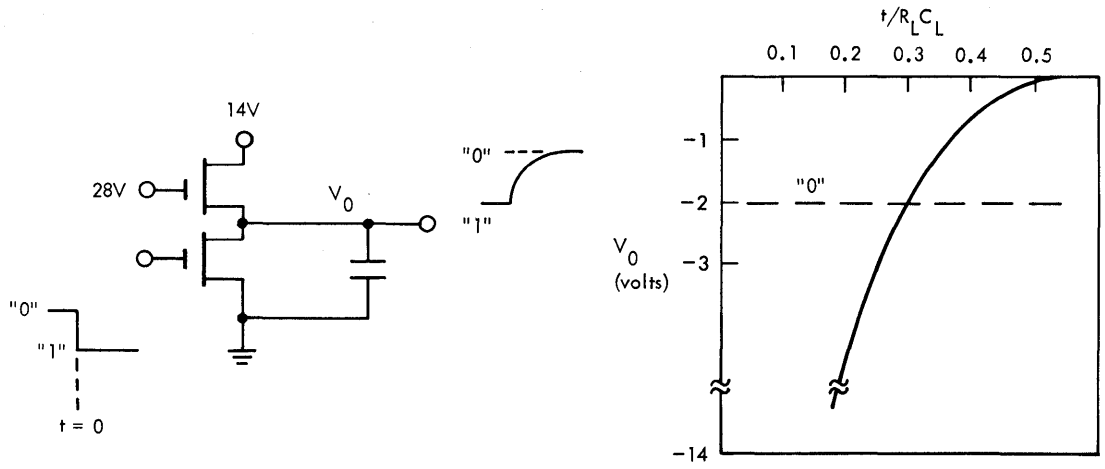


Fig. 6.2 Turn-on Waveform of MOS Inverter with Step Input.

in Fig. 6.1 rather than a step input as assumed above. Therefore, the inverter will start to turn on well before its gate reaches a "1" level. To obtain a simple method of calculating propagation delay through several cascaded stages, however, we will assume that an inverter remains off – with its drain at a "1" or below – while its gate is falling from a "0" to a "1". When its gate reaches a "1", its drain then starts to rise to "0", taking a time equal to $0.3\,R_LC_L$* to reach "0". We will assume further that an inverter remains on – with its drain at a "0" – while its gate is rising from a "1" to a "0". When its gate reaches "0", its drain then starts to fall toward V_{DD}, taking $3\,R_LC_L$* to reach a "1". Based on these assumptions, the propagation delay through a chain of inverters is

$$t_{pd} = 3\,R_LC_{L(off)} + 0.3\,R_LC_{L(on)} \qquad (6.1)$$

The propagation delay obtained by using Eq. (6.1) will always be longer than the worst case because each inverter actually starts to turn off (or on) before its gate reaches "0" (or "1"). Measured propagation delays are typically less than half as long as calculated delays.

*Where V_{DD} = 14 V and V_{GG} = 28 V. See Section 6.3 on speed-power tradeoff.

More precise propagation-delay calculations through a particular path can be obtained by computer techniques available at Philco-Ford. However, the worst-case results obtained by using Eq. (6.1) are usually adequate, unless speed is a prime consideration.

6.1.1.1 Example. The time between the appearance of a "1" at the gate of the first stage in Fig. 6.3 and the appearance of a "0" at the drain of the third stage is:

$$t_{pd} = 0.3\,R_L(C_{L1} + C_{L3}) + 3\,R_LC_{L2}$$

where R_L is the same for all three stages.

Load resistance and capacitance in a Building-Block array are discussed below.

6.1.1.2 Load Resistance. The data sheets list load resistance at 25°C. However, this parameter increases with chip temperature. To determine propagation delay at temperatures above 25°C, therefore, the load resistance should be modified as explained in Section 6.2.3 below. In addition, any resistance in the interconnecting lines adds in series with R_L. As shown in Table 6.1, the resistance of metal lines is negligible, but that of P lines is 12.5 Ω/μ. Thus, a P line of 1,000 μ has a resistance greater than one-tenth that of the typical Building-Block load resistance.

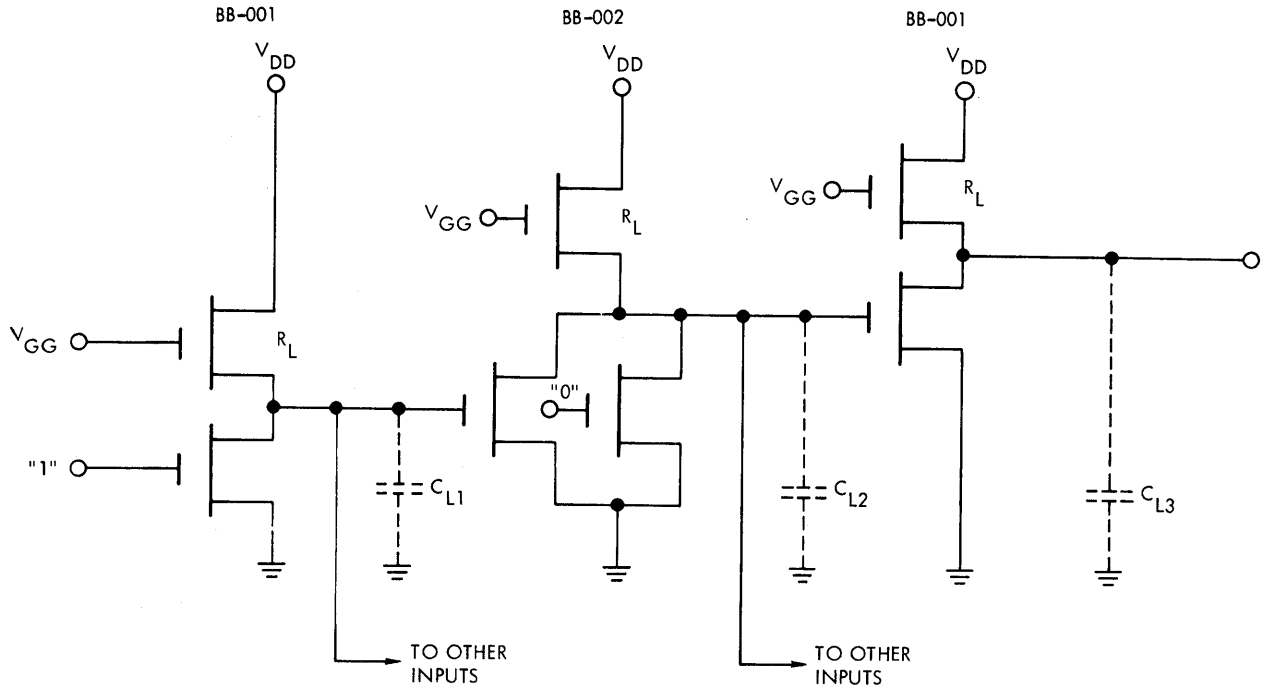


Fig. 6.3 Three Blocks in Cascade.

TABLE 6.1
INTERCONNECT RESISTANCE

Metal Lines	———
P Lines	12.5 Ω/μ

6.1.1.3 **Load Capacitance.** The capacitive load on a Building-Block output consists of the following:

1. The output capacitance of the block itself.
2. The capacitance of the interconnections between this block and any blocks it may drive.
3. The input capacitance of all the blocks which this block drives.

The input and output capacitances of each block are listed in the data sheets.

As discussed in Section 2.2.4, interconnect capacitance may consist of metal deposited on the silicon-dioxide surface of the chip or of buried P regions. In addition, the following must be added to the line capacitances:

1. *Crossover capacitance.* Whenever a metal line crosses over a P region, the crossover capacitance between the two must be added to the capacitance of both lines.
2. *Contact capacitance.* When a metal line contacts a P region, there is additional capacitance associated with the contact area. This also must be added to the line capacitance. These capacitances, plus the package capacitance and the capacitance between the metal bonding pad and ground, are itemized in Table 6.2.

Example: To Determine the Load Capacitance $C_{L(A)}$ of Block 1 in Fig. 6.4

The capacitance $C_{L(A)}$ of Block 1 in Fig. 6.4 consists of the following:

1. The output capacitance of Block 1 (listed in the data sheet).
2. The input capacitance associated with node C of Blocks 2 and 3 (also listed in the data sheet).
3. The capacitance of the P lines connecting Block 1 to the metal line and connecting the latter to Blocks 2 and 3.

TABLE 6.2

CAPACITANCES IN A BUILDING-BLOCK ARRAY

Capacitance	Location	Maximum	Units
C_{Pad}	From pad to ground	0.46	pF
$C_{Package}$	From pin to package	See appendix C	
	From pin to adjacent pin	See appendix C	
$C_{Crossover}$	From metal to P and from P to metal	0.04	pF
			
$C_{Contact}$	Between metal-to-P contact and ground	0.04	pF
			
C_P	From P line to ground	0.0007	pF/μ
C_{Metal}	From metal line to ground	0.00032	pF/μ

4. The capacitance of the metal line (the length is obtained from the dimensions of the BB-001 and BB-002 given in the data sheets).
5. The capacitance between the metal line and the two P lines that cross under it ($C_{Crossover}$).
6. The capacitance of the three contact areas between the metal and P lines ($C_{Contact}$).

These are listed below:

C_{out} of Block 1	0.34 pF
$C_{in(C)}$ of Block 2	0.38 pF
$C_{in(C)}$ of Block 3	0.38 pF
C_P (66 μ X 0.0007)	0.05 pF
C_{Metal} (136 μ X 0.00032)	0.04 pF
$C_{Crossover}$ (2 X 0.04)	0.08 pF
$C_{Contact}$ (3 X0.04)	0.12 pF
Total $C_{L(A)}$	1.39 pF

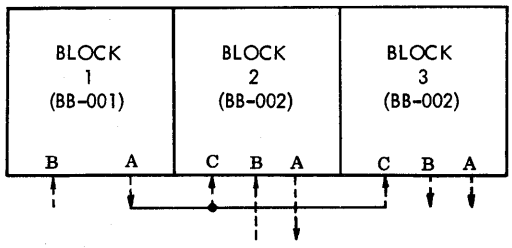


Fig. 6.4 Hypothetical Interconnect Pattern.

6.1.2 Propagation Delay Through Blocks with Internal Nodes

Propagation delay through the BB-101 and the BB-021, which have internal nodes, is calculated in exactly the same way as if the two stages were in separate blocks. For example, consider the BB-101 shown in Fig. 6.5. Suppose both inputs are at "0". When input B goes to a "1", the internal node (point F) goes to a "0" in a time equal to $0.3\,R_L C_{L(F)}$. When point F reaches a "0", transistor T starts to turn off, causing node A to go to a "1" in a time equal to $3\,R_L C_{L(A)}$. Thus the total propagation delay through the block in this case is

$$t_{pd} = 0.3\,R_L C_{L(F)} + 3\,R_L C_{L(A)}$$

Values of $C_{L(F)}$ and $C_{L(A)}$, as well as R_L, are listed in the data sheets.

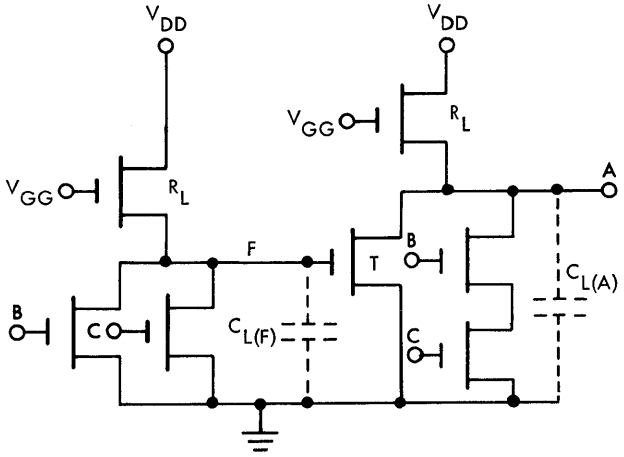


Fig. 6.5 Schematic Diagram of BB-101.

The propagation delay through blocks with positive feedback (e.g., the BB-401 and the BB-402) is considerably shorter than predicted by Eq. (6.1). Therefore, instead of giving the capacitances at the internal nodes for these blocks, the data sheets list the propagation delay itself. The data sheets for the one-bit shift registers (the BB-403 and the BB-404) also list propagation delay.

6.1.3 Increasing Speed

If the speed calculated through certain critical paths is marginal,* it can be increased by doing any (or all) of the following:

- Change the logic implementation to reduce the number of gates through which the signal must propagate.
- Rearrange the layout so that the Building Blocks in the critical path are adjacent. This reduces the lead lengths and the number of crossovers – hence the load capacitance – at each node in the critical path.
- Place a second block of the same type in parallel with a block that must drive a very large load capacitance. This will cut the load resistance in half.
- Use an output buffer at very high-capacitance nodes in the critical path. The load resistance of the buffer blocks is about one-tenth that of the other blocks.
- Increase the negative supply voltages (see Section 6.3 for allowable range and limitations).

If speed must be increased still further, arrangements can be made with Philco-Ford to customize part or all of the chip.

6.2 TEMPERATURE CONSIDERATIONS

6.2.1 Calculating Power Dissipation

The average power dissipated by each Building-Block at 25°C is listed in the data sheets. The total power dissipated by the array is calculated by determining the worst-case combination of blocks that can be on at a "0" level simultaneously and adding the stated power dissipations of these blocks.

*This problem is unlikely to occur in Building-Block arrays at clock rates below 500 kHz.

6.2.2 Effect of Power Dissipation on Chip Temperature

The maximum chip temperature T_C allowable in a Building-Block array is 160°C. This temperature is related to power dissipation by the following expressions:

$$T_C = P_D \theta_{CA} + T_A \quad \text{(for package without a heat sink)} \tag{6.2}$$

$$T_C = P_D(\theta_{CP} + \theta_{PS} + \theta_{SA}) + T_A \quad \text{(for package with a heat sink)} \tag{6.3}$$

where

T_C = chip temperature

P_D = power dissipation at 25°C

T_A = ambient temperature

θ_{CA} = thermal resistance from chip to ambient

θ_{CP} = thermal resistance from chip to package

θ_{PS} = thermal resistance from package to sink

θ_{SA} = thermal resistance from sink to ambient

(The values of θ_{CA} and θ_{CP} are given in the data sheet for each package.)

6.2.3 Effect of Chip Temperature on Load Resistance

The data sheets list R_L at 25°C; but, as shown by the graph in Fig. 6.6, R_L is directly proportional to the chip temperature T_C. Therefore, to calculate propagation delay at temperatures above 25°C, it is appropriate to use the corrected value of R_L obtained from this graph.

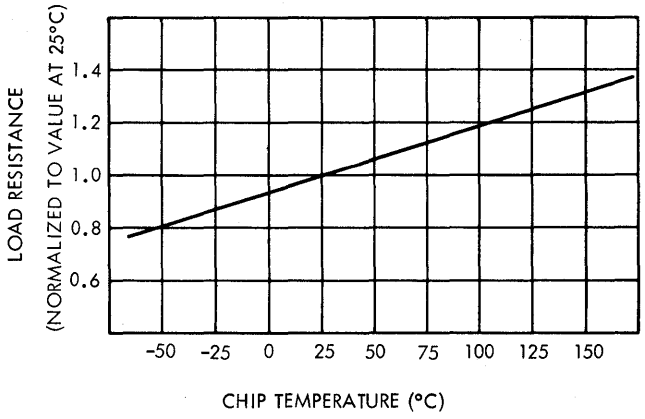


Fig. 6.6 Load Resistance versus Chip Temperature.

Example: Given the following conditions:

1. The total power dissipation for the array – obtained by adding the data sheet values for each ON block – is 100 mW. (This, of course, is the power dissipation at 25°C.)
2. The package has a thermal resistance θ_{CA} of 228°C/W from junction to ambient.
3. The specified ambient temperature T_A is 125°C.

Under these conditions, the chip temperature

$$T_C = P_D\theta_{CA} + T_A$$
$$= 100 \times 10^{-3} \times 228^{\circ}C/W + 125^{\circ}C$$
$$= 147.8^{\circ}C$$

According to the graph in Fig. 6.6, the load resistance at 147.8°C

$$R_{L147.8^{\circ}C} \approx 1.35\, R_{L25^{\circ}C}$$

Therefore, to determine propagation delay at 147.8°C in the above package (without a heat sink), the data sheet values of R_L should be multiplied by 1.35.

NOTE: The power dissipations listed in the data sheets are obtained from the formula:

$$(V_{DD} - V_{\text{“0”}})^2/R_L$$

where $V_{\text{“0”}}$ is the "0" level and R_L is the load resistance at 25°C. Because of the increase in R_L at array temperatures above 25°C, the power dissipation at these temperatures is less than the 25°C values listed in the data sheets. Therefore, T_C obtained from Eq. (6.2) or (6.3) will be larger than in reality. Normally this fact can be ignored. However, if the value of T_C thus obtained approaches the upper limit of 160°C, it is appropriate to iterate for a more accurate value: i.e., solve for T_C using the specified T_A and the 25°C value of power dissipation; then, from Fig. 6.6, determine the increase in R_L corresponding to this value of T_C, correct the power dissipation, and solve again for T_C.

6.3 SPEED-POWER TRADEOFF

The Building-Block designs have been optimized for a V_{DD} of -14 V and V_{GG} of -28 V. However, the speed-power tradeoff can be adjusted by varying these voltages. Bear in mind, however, that V_{GG} must be twice as negative as V_{DD} (for the reasons explained in Section 2.2.1), and that the allowable range for these two voltages is from -12 and -24 V, respectively, to -15 and -30V.

Making these voltages more negative will increase speed for the following reasons:

1. A more negative V_{DD} will cause the load capacitance to charge faster to the minimum "1" level of -10 V, and a more negative V_{GG} will decrease the load resistance.
2. A more negative voltage on the gate of the inverter will decrease the inverter resistance and hence the turn-on time.

Section 7

TESTING

To produce MOS arrays of consistently high quality, Philco-Ford manufacturing personnel exercise tight process control: Skilled operators follow written specifications and inspect every unit at critical points in the process. Quality Control personnel repeatedly sample-check their work. (See flow chart in Fig. 7.1.) These in-process tests are intended (1) to screen potential fallout units early in the manufacturing cycle and (2) to provide feedback so that slight process deviations can be immediately corrected.

After the package is sealed, every unit undergoes standard production environmental tests such as acceleration, temperature cycling, etc. (The flow chart in Fig. 7.2 lists the tests performed on flatpacks, for example.) These stresses are intended to bring out hidden defects in the package assembly and the chip itself, so that substandard units will be rejected by the electrical tests that follow.

The electrical tests include sample tests specified by the customer – e.g., propagation delay, power dissipation, etc. – and functional tests of all units. These 100% functional tests are custom-designed to verify the ability of every unit to perform the required logical functions. A well-written test is therefore a necessity.

There are two options available for setting up a test program:

- You may wish to generate the test program yourself. In this case, we will verify the outputs your test predicts and deliver your devices with the guarantee that they will perform as prescribed by your test.
- If the array is very complex, a computer-designed test is economically advantageous. A versatile computer-aid program, designed by Philco-Ford engineers, has been in use here for some time. You may therefore wish to take advantage of our experience in this area and let us generate your test program for you.

In this section, we will consider the problems involved in designing a good functional test and describe the Philco-Ford computer-aid program in some detail.

7.1 DESIGNING A FUNCTIONAL TEST

7.1.1 100% Testing: Economically Unfeasible for Large Arrays

A 100% functional test of an array verifies every possible combination of inputs. Even for a fairly simple array, such a test would be time-consuming to perform. For arrays containing internal memory elements, the task becomes virtually impossible. You would have to apply every possible input combination for every possible state of each memory element – plus all the input combinations required to test the combinatorial logic. Thus a binary function with 20 inputs and 20 memory elements would require at least $2^{20} \times 2^{20}$, or 2^{40} distinct tests. At a rate of 1 GHz it would take ten days to perform these tests on a single unit.

7.1.2 Alternative: Limited-Sequence Tests

Fortunately, such exhaustive testing is not essential. Carefully devised limited-sequence tests can assure a low failure rate with a high degree of confidence.

Depending on the type of array tested, a successful limited-sequence test might either:

- Test the input combinations the array will meet in actual use;
- Test the array only for its most likely failure modes.

The first approach guarantees proper operation provided the test is complete. Undoubtedly this is the best approach for functions like decoders, counters,

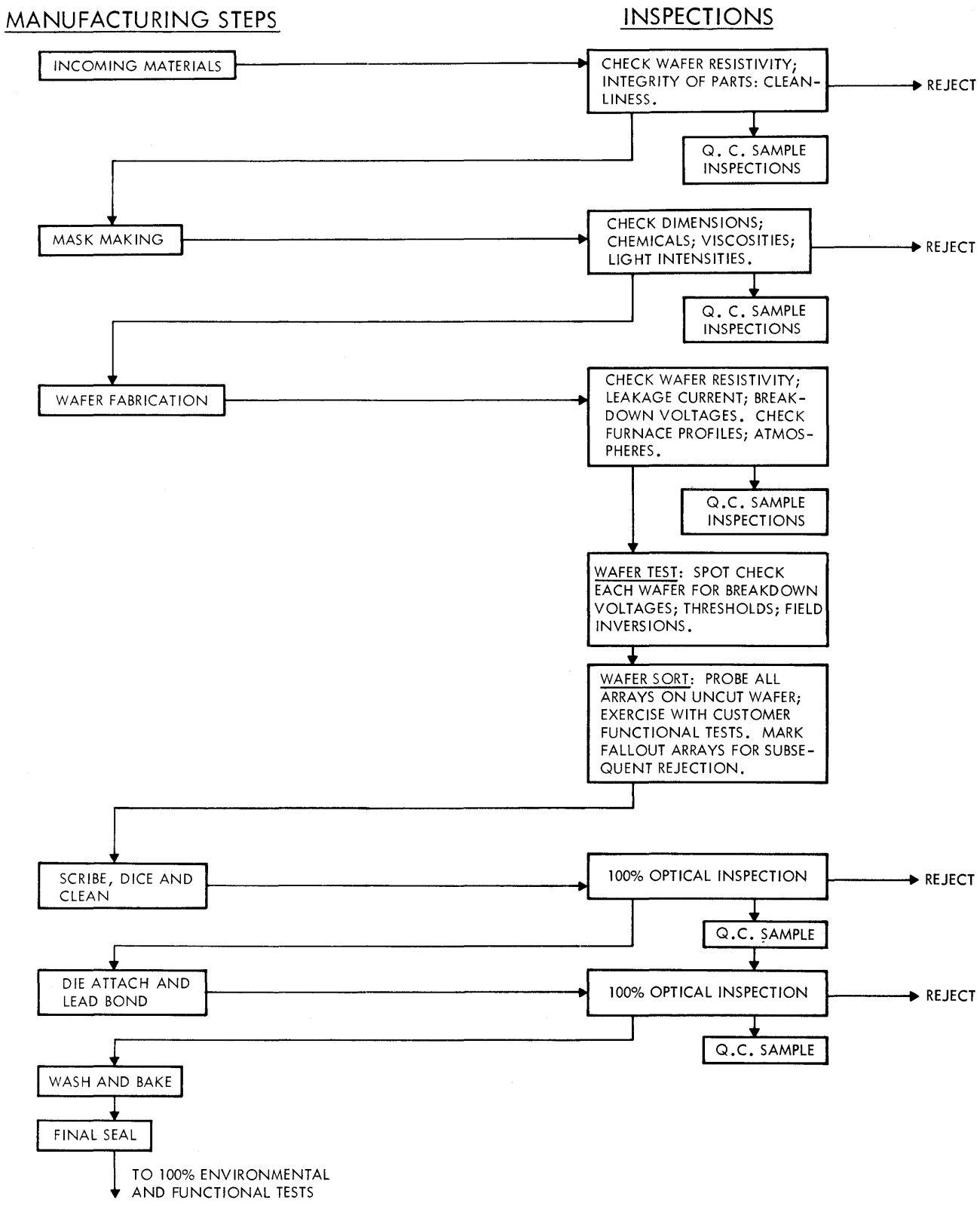


Fig. 7.1 Flow Chart of MOS Production.

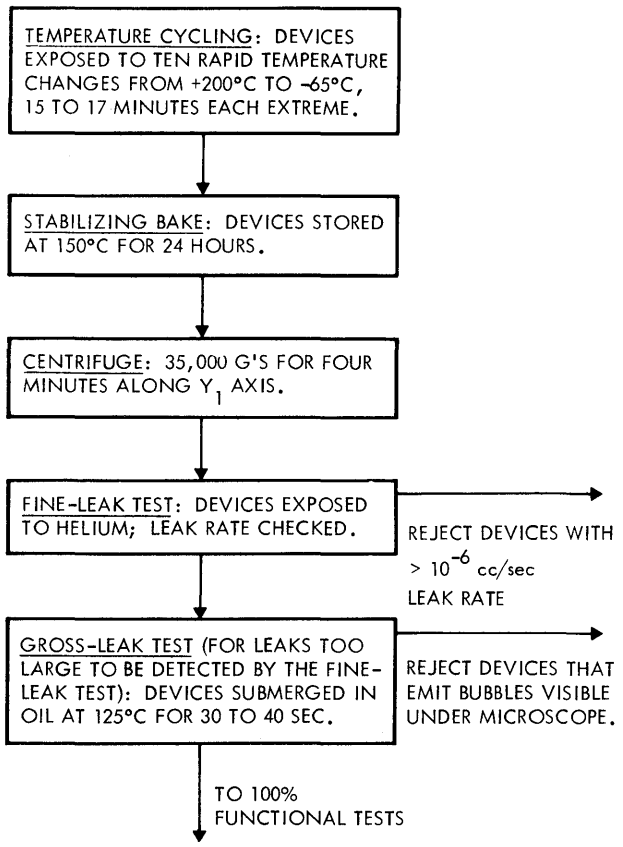


Fig. 7.2 100% Environmental Tests (Flatpacks).

etc., with a limited number of well defined inputs. However, for functions (such as control units) that may have many different input sequences, it is difficult to define all the input combinations that may occur in operation. Also the test program may be excessively long. The second approach is therefore preferable in such cases.

7.1.3 Philco-Ford Automated Test Procedure

Philco-Ford engineers have developed a computer program known as LOTGEN based on this second approach. This program can be used to generate a computer-devised test program that will:

- Test each signal line for its ability to transmit both a "1" and a "0" under proper conditions;
- Test each memory element for its abilitv to store a "1" and a "0" under proper conditions;
- Test each logic element for its most common failure modes.

Table 7.1 itemizes the tests a Philco-Ford computer-devised program will include, as well as the tests it will not include.

The failure modes that might occur under normal conditions are covered by this program. Only an extremely unlikely failure mode might cause an isolated unit to pass the test but function improperly in operation. For example, the OR gate is tested to insure that:

- It produces a "1" when a "1" is put on each input separately;
- It produces a "0" when all inputs are "0" simultaneously.

We assume it will also produce a "1" when two or more inputs are "1". But it is conceivable that in an isolated unit it might not do so. For example, suppose an extremely unlikely failure caused a two-input OR gate to behave like an EXCLUSIVE OR in one unit. Such a unit would pass the functional test but might function improperly in operation. Also, two failures occurring in different branches of the circuit might cancel each other out for all the input combinations used in the test but fail to do so in operation. Again, this would happen only rarely and in an isolated unit. The test therefore assures a low failure rate with a high degree of confidence.

7.3 MOS TESTER

The Philco-Ford MOS tester, shown in Fig. 7.3, is specially designed to test complex MOS arrays. Therefore the test program should be compatible with it. A block diagram of the instrument is shown in Fig. 7.4. The program card (Fig. 7.5) is a diode matrix which has 22 channels and can accommodate test words from 8 to 32 bits long.* Channel numbers correspond to package leads. At each bit time

*If more than 32 bits are required, additional matrices can be programmed.

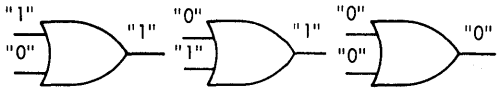
TABLE 7.1

LOTGEN TESTS

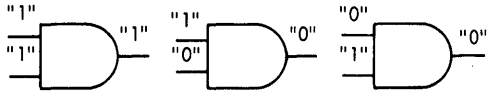
A Philco-Ford LOTGEN test will verify the following:

1. Each inverter will invert both a "1" and a "0".

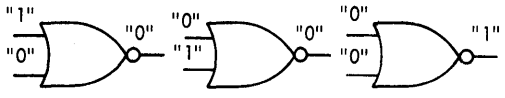
2. Each OR gate will produce a "1" with a "1" on 1 input; it will produce a "0" with "0's" on all inputs simultaneously.

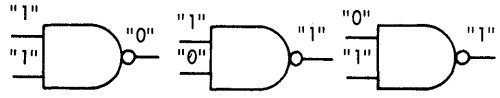


3. Each AND gate will produce a "1" with "1's" on all inputs simultaneously; it will produce a "0" with "1" on one input and "0" on the others.



4. NOR and NAND gates behave like OR and AND gates with inversion.



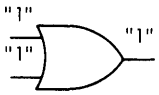


5. Each signal line will transmit a "1" and a "0" under proper conditions.

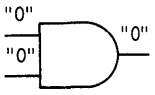
6. Each memory element will store a "1" and a "0" under proper conditions.

A Philco-Ford LOTGEN test will not verify the following:

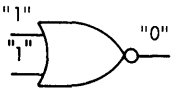
1. An OR gate will produce a "1" with "1's" on all inputs simultaneously.



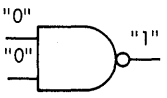
2. An AND gate will produce a "0" with "0's" on all inputs simultaneously.



3. A NOR gate will produce a "0" with "1's" on all inputs simultaneously.



4. A NAND gate will produce a "1" with "0's" on all inputs simultaneously.



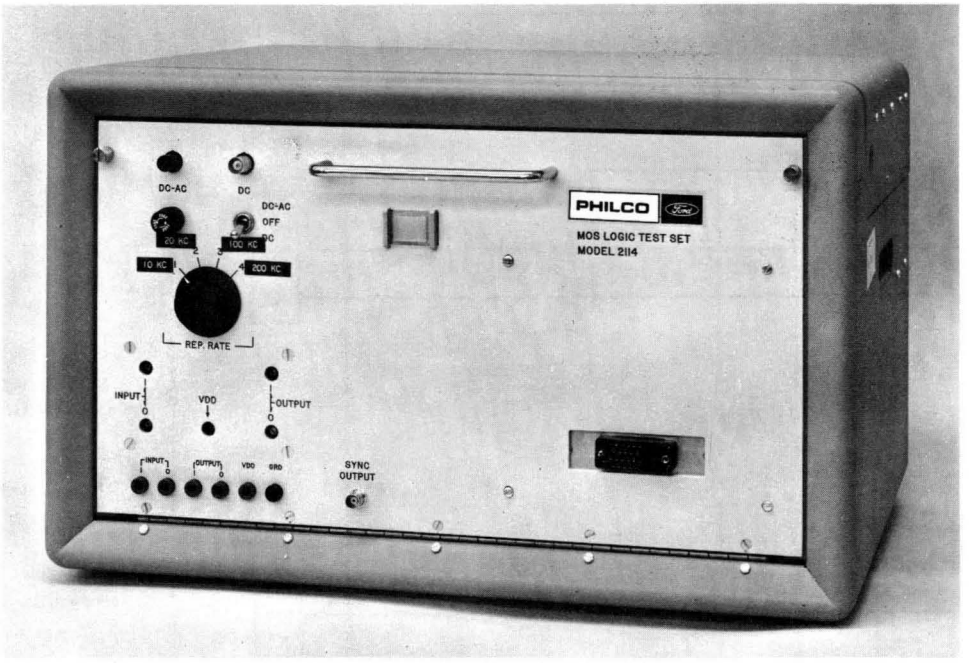


Fig. 7.3 Photo of MOS Tester.

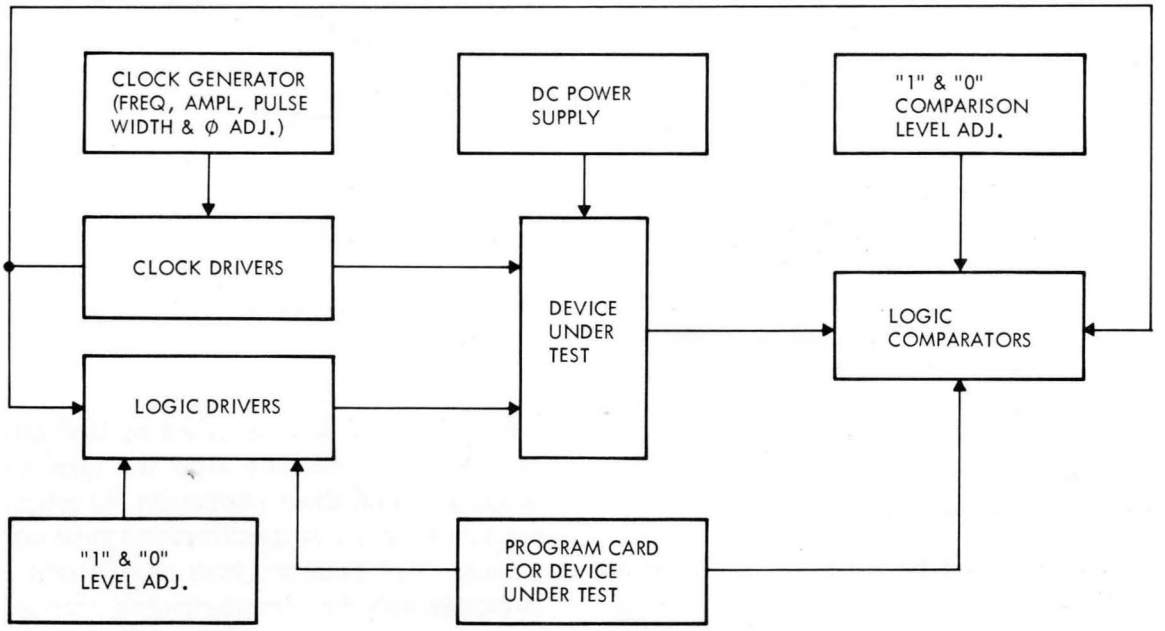


Fig. 7.4 Functional Block Diagram of MOS Tester.

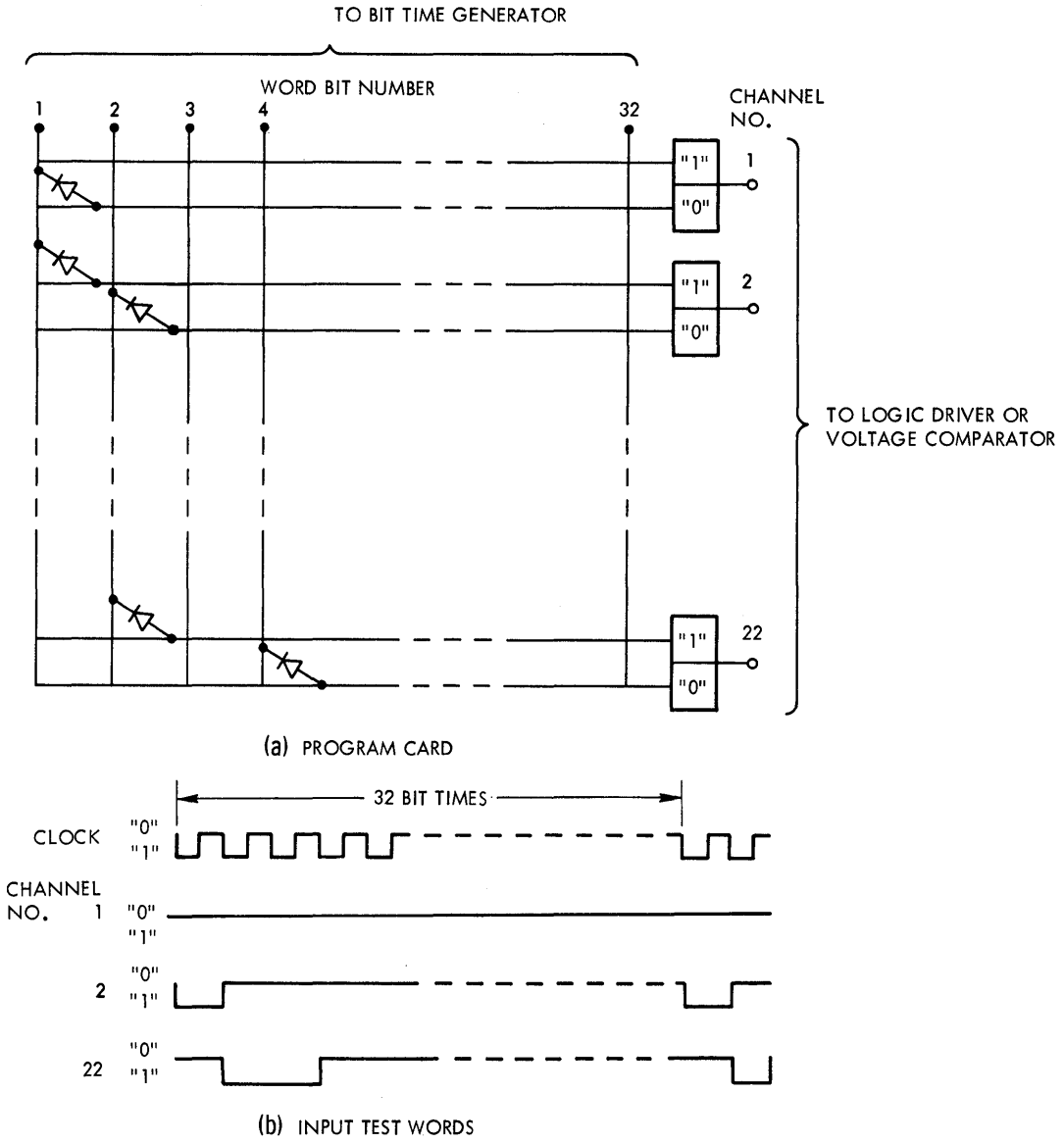


Fig. 7.5 Program Card and Input Test Words.

the clock is steered through one or more of the channels; then either (1) through a logic driver to one of the package input leads or (2) to a voltage comparator which is connected to one of the package output leads. If one of the device outputs does not correspond to the test word at a given bit time, the device is automatically rejected.

The test pattern should be prepared in the form shown in Fig. 7.6.

Building-block chips are normally tested at the worst-case "1" and "0" levels specified in the data sheets; at an operating temperature of 25°C; and at a clock rate of 10 to 250 KHz.

Other electrical tests specified by the customer are performed immediately after the functional tests. A specification sheet (Appendix A) provides space for you to write in specifications your system may require. For example, your propagation delay calculations may be sample-checked through one or more critical paths; rise and fall times, operating temperatures, power dissipation, etc., may be measured and guaranteed if necessary.

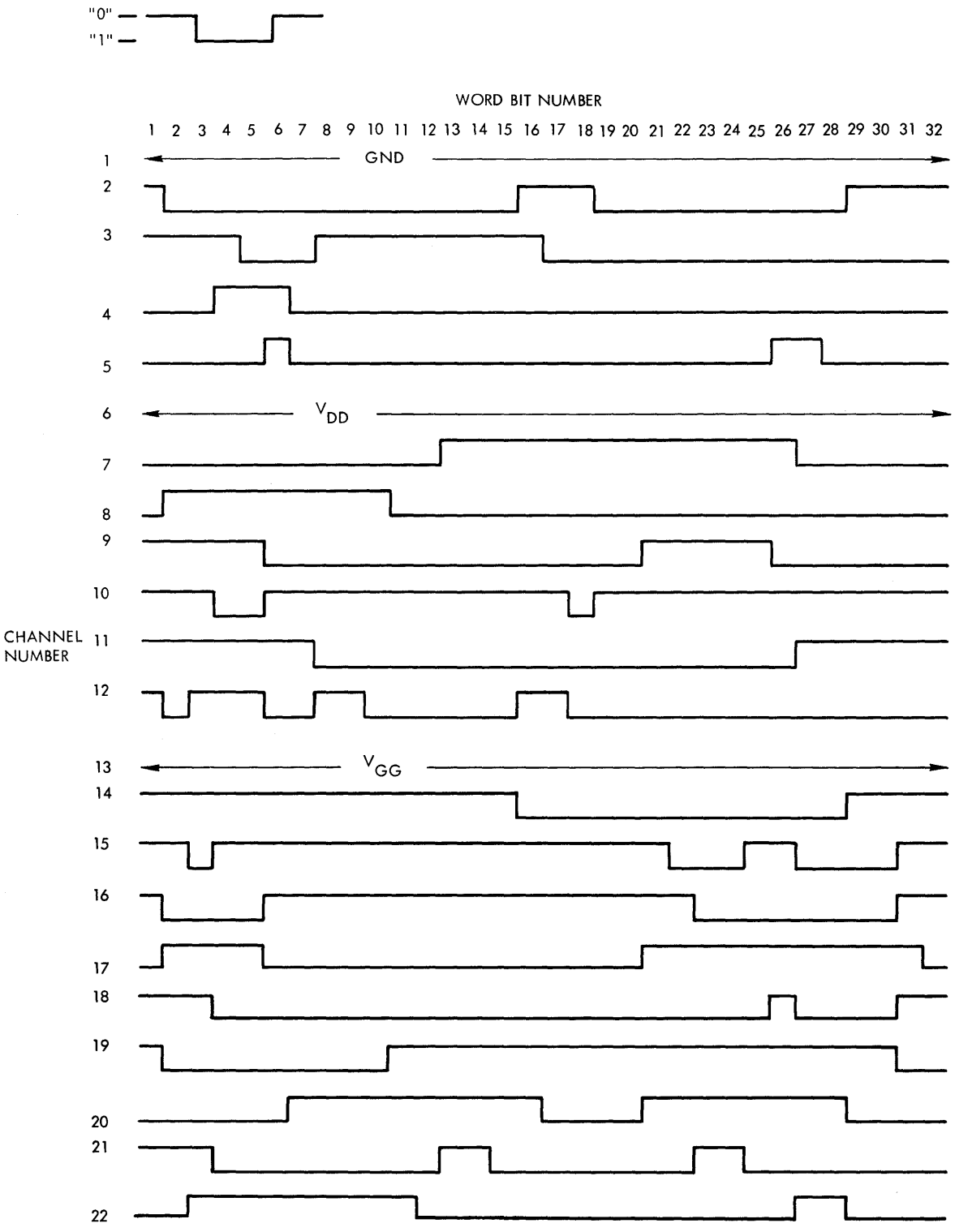


Fig. 7.6 Sample Program for Functional Test.

Section 8

ORDERING AND MANUFACTURING YOUR ARRAYS

In this section, we will briefly summarize (1) the steps you follow in ordering an array and (2) the steps we follow in manufacturing and shipping it to you.

For each array, you should provide us with the following:

1. A layout prepared according to the instructions in Section 5.
2. A test program (unless you wish us to generate one for you).
3. A specification sheet.

For your convenience, a form of specification sheet is attached as Appendix A. Additional copies may be obtained from Philco-Ford.

Under "Options" in the specification sheet, you should indicate whether you wish us (1) to calculate by computer (and guarantee) the propagation delay through the paths you indicate and/or (2) to generate the functional tests by computer. Under "Mechanical Specifications," indicate the type of package (e.g., 22-lead flatpack); how it should be marked; and whether or not you wish us to number the units serially.

Under "Environmental Specifications," indicate the operating temperature for which you have designed the array (per Section 6) and any special tests your system may require.

Under "Electrical Specifications," indicate the values of V_{DD} and V_{GG} and the calculated maximum power dissipation at 25°C. There are appropriate blanks under "Sample Performance Tests" to be filled in if you wish us to sample-check your propagation delay calculations through one or two critical paths. In addition, you may wish us to measure power dissipation, rise and fall times, or other performance characteristics. If so, indicate the test conditions.

The flow chart in Fig. 8.1 shows the procedure we follow on receipt of the above documents.

If you have prepared the test program, we will verify that the logic shown in your layout will produce the required outputs in response to the inputs you have indicated on the test program. If, on the other hand, you have elected to have us generate the test program, we will do so and submit it to you for approval. While the details of the test are being worked out, we will do the following:

1. We will check your layout to make sure it can be translated into a correctly functioning array.
2. We will verify your calculations for operating temperature and power dissipation.
3. If you wish us to calculate the propagation delay through one or more paths by computer, we will do so.
4. If you have chosen not to have us calculate propagation delay, but wish to verify your calculations by having us measure it through one or two critical paths, we will check your indicated calculations, using Eq. (6.1).

After these details have been completed and the test program agreed upon, we will manufacture your arrays, test them as agreed, and deliver them to you.

The cost to you will of course depend on (1) the extent of the customer interface involved; (2) whether or not we develop the test program and/or calculate propagation delay by computer; (3) the extent of testing required, etc. Firm prices and standard contract provisions will be provided at your request upon determination of the scope of support you require.

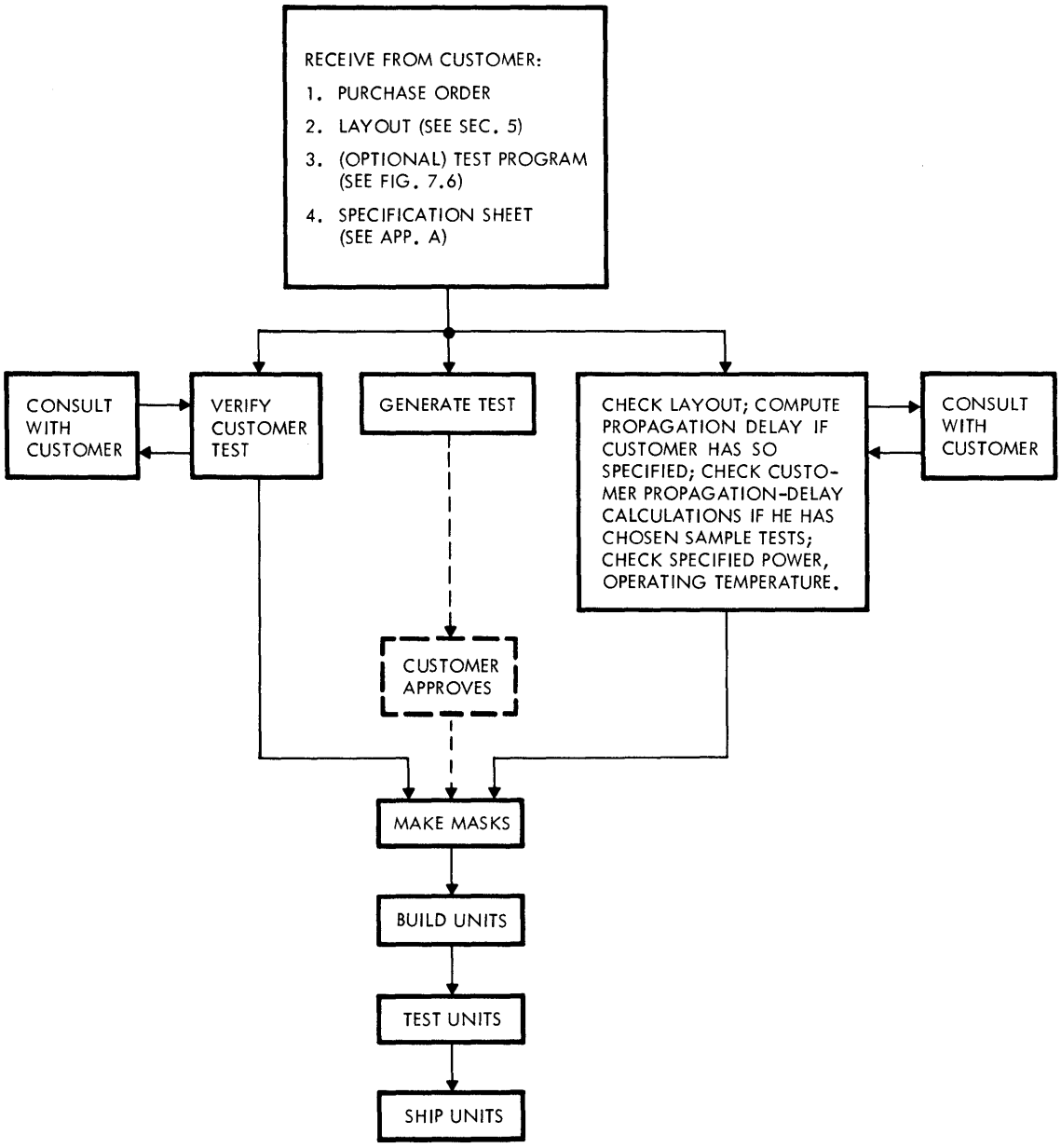


Fig. 8.1 Flow Chart Showing Procedure for Ordering, Manufacturing, and Shipping of Building-Block Arrays.

Appendix A

SAMPLE SPECIFICATION SHEET

SPECIFICATION SHEET

ARRAY *(identify by name and/or number)*_____

_____

OPTIONS

1. Propagation Delay at 25°C

   Philco-Ford to calculate by computer and guarantee:

   Yes____ *(if yes, fill out below)*

   No ____

   Pin____ to Pin____

   Pin____ to Pin____

2. 100% Functional Tests *(check one)*

   Philco-Ford to generate by computer: _____

   Use attached program: _____

MECHANICAL SPECIFICATIONS

Package type: _____

Marking:

Philco-Ford Product No. *(check if you wish this option)*: _____

Other 6-digit *(or less)* code identification: _____

Serialize units: Yes____ No ____

ENVIRONMENTAL SPECIFICATIONS

Operating temperature: _____

Storage temperature: -55°C to 150°C

ENVIRONMENTAL SPECIFICATIONS (Continued)

Tests:

Acceptance test temperature: 25°C ± 5°C

Standard environmental tests for indicated package

Other:

ELECTRICAL SPECIFICATIONS AT 25°C

V_{DD} _____ *(Range -12 V to -15 V; optimum -14 V ± 0.7 V)*

V_{GG} _____ *(Must equal 2 V_{DD}. Range -24 V to -30 V; optimum -28 V ± 1 V)*

Maximum Power _____

Maximum "0" level in: -3 V
Maximum "0" level out: -2 V
Minimum "1" level in: -9 V
Minimum "1" level out: -10 V
Input Leakage: 10 μA at 12V

Tests:

100% Functional Tests

ELECTRICAL SPECIFICATIONS AT 25°C (Continued)

Sample Performance Tests:

1. Propagation Delay: Yes ____ *(if yes, fill out below)*
   No ____

Path	Delay Calculated from Eq. (6.1)
Pin ____ to Pin ____	_____
Pin ____ to Pin ____	_____

Conditions

V_{DD}: 14 V ± 0.7 V

V_{GG}: 28 V ± 1 V

Temperature: 25°C

Source impedance of input pulse: 50 ohms *(nominal)*

Output load: 1 megohm in parallel with 10 pF

(Voltages on all input pins to be itemized on a separate sheet.)

2. Other: