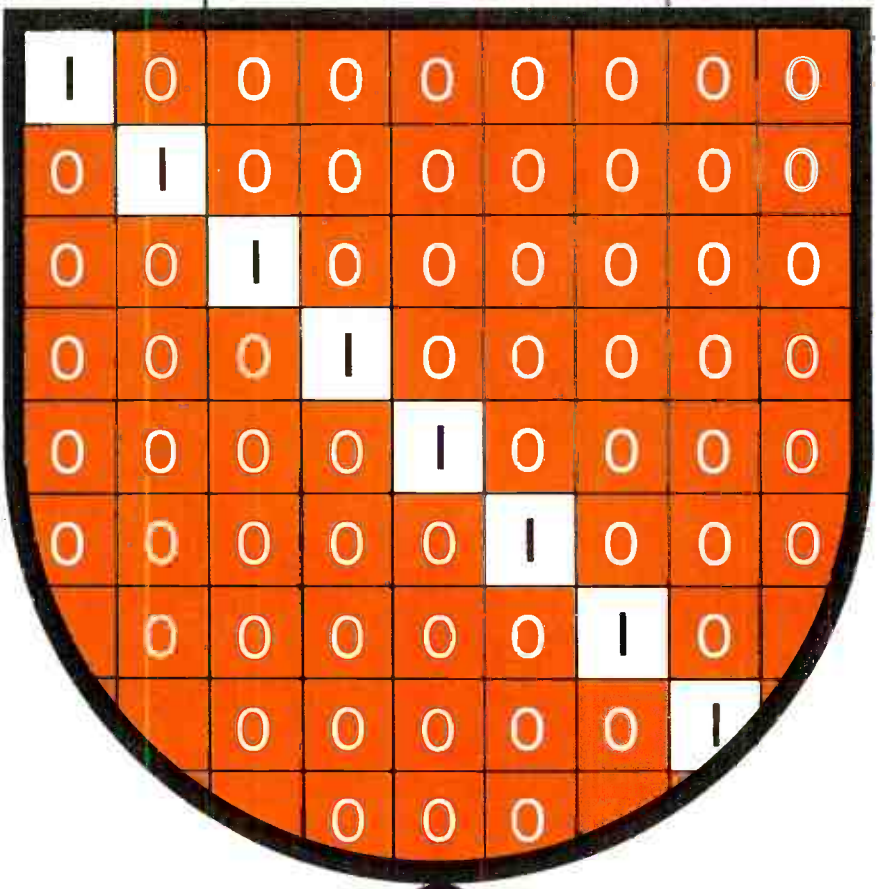
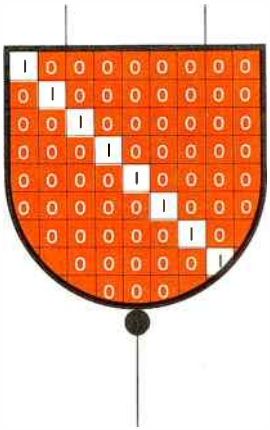
The Lenkurt®

DECEMBER 1969

DEMODULATOR



BINARY LOGIC AND PCM



Aristotle, in 330 B.C., to explain his philosophies, developed a logic system dealing with statements that were either true or false. In 1847, George Boole reduced Aristotle's logic to a mathematical shorthand that has become a universal logic language.

Binary logic is a way of thinking that can be applied to the design of *any* system where the "inputs" and "outputs" are just on-off actions. The invention of transistors led to the development of a series of logic modules capable of performing basic binary logic functions in electronic systems.

The complex PCM (pulse-code modulation) system can be broken down into subsystems whose inputs and outputs are simply on-off actions. This subsystem equipment is then designed using the principles of binary logic and implemented with corresponding logic modules.

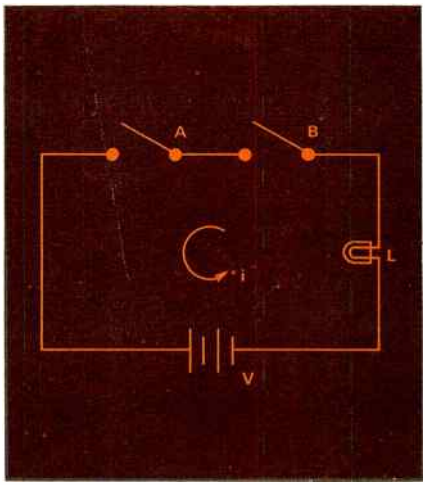


Figure 1.

Logic Modules

Basic logic modules are called AND gates, OR gates, and INVERTERS. These modules can be combined to obtain NAND and NOR gates. Logic building blocks called flip-flops can be made from these gates.

AND, OR, and INVERTER

Consider the circuit with two switches (A and B) connected in series, a voltage supply (V), and a light bulb (L) shown in Figure 1.

The light will be on, if, and only if, switch A *and* switch B are closed. The logic AND gate gets its name from this simple circuit analogy.

The "switching" circuit described above can be implemented with relays or diodes as well as with switches. All these circuits are cumbersome for the logic designer, so shorthand logic symbols have been developed. The logic symbol for the AND gate is shown in Figure 2.

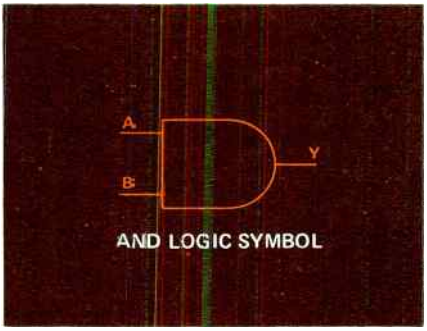


Figure 2.

Using a truth table and representing an "on" condition as a "1", an "off" condition as a "0", and the AND gate output as "Y", the combination of states for an AND gate is graphically displayed (Figure 3).

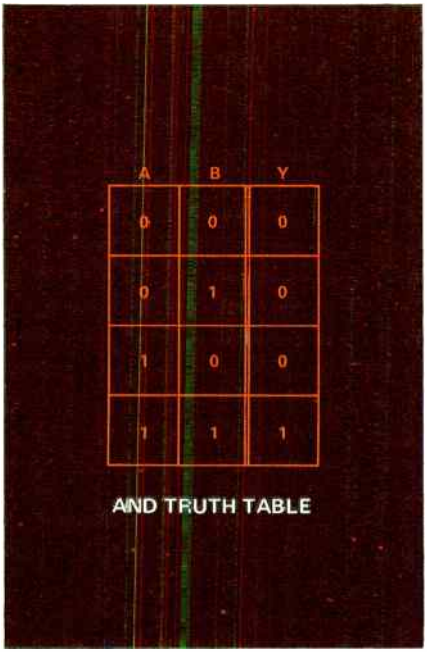


Figure 3.

Using the "switching" circuit analogy again, put the two switches in parallel rather than in series (Figure 4).

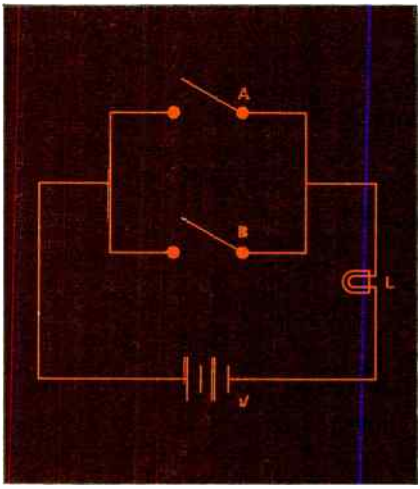


Figure 4.

With this arrangement, the current flows in the circuit and the bulb is on, if switch A *or* switch B *or* both are closed. The logic OR gate gets its name from this type of an arrangement. Figure 5 shows the logic symbol and truth table for an OR gate.

Logic functions can be implemented with diodes for electronic applications. But, diodes have two weaknesses. First, the output of diode AND and OR gates is attenuated. Second, diode gates are passive elements and unable to drive a network of gates.

Common emitter transistors have the ability to amplify a signal. By putting a transistor at the output of the diode AND and OR gate circuitry, the attenuated signal is restored to its original level. Because transistors are

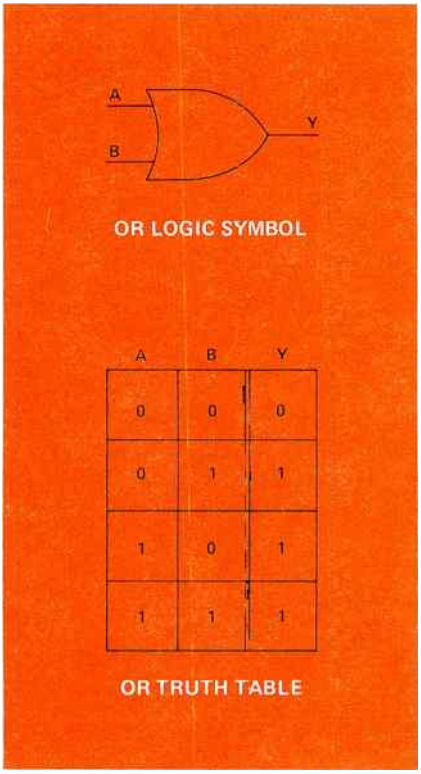


Figure 5.

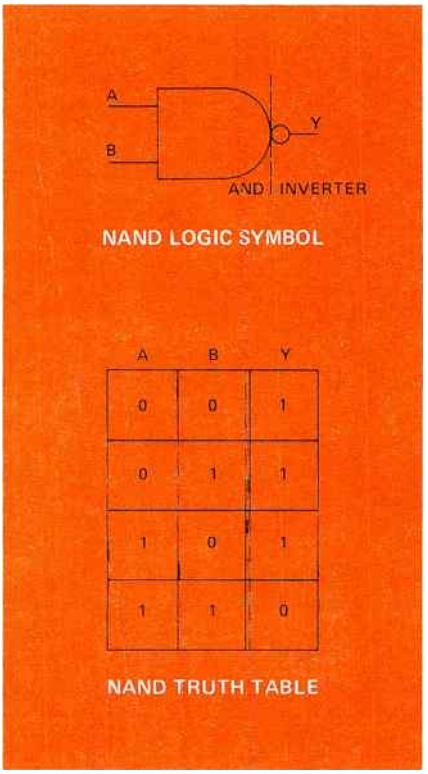


Figure 6.

active devices, they are capable of driving a network of logic functions.

As well as solving the inherent problems of diode gates, transistors perform the logic function of inversion. Regardless of the input signal state, the transistor output will be inverted (a "1" becomes a "0" and vice versa). Transistors are known therefore, as INVERTERS.

NAND and NOR

The combination of a logic AND gate and a transistor INVERTER is called a logic NAND gate (for NOT–AND) (Figure 6).

The output from a NAND gate will be negative, if, and only if, A and B are both positive.

A logic NOR gate is the combination of a logic Or and an INVERTER (for NOT–OR) (Figure 7).

If, and only if, both the NOR inputs are negative, the NOR output will be positive.

Integrated circuit technology has made NAND and NOR gates less expensive than the use of discrete components to construct NOT–AND and NOT–OR circuits.

For simplicity, the inputs to the logic gates have been limited to two,

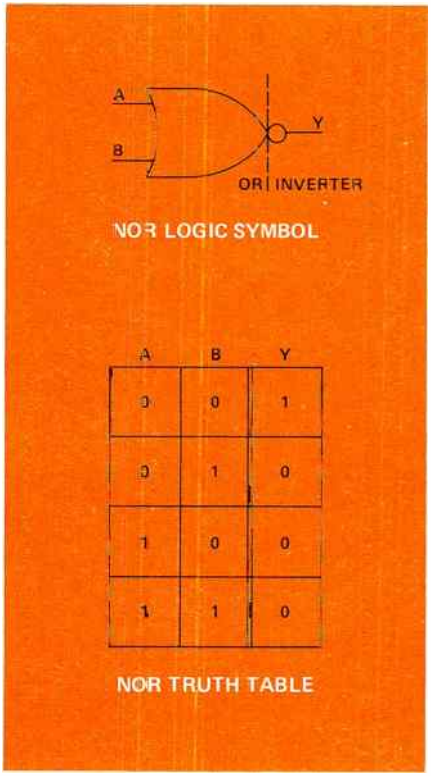


Figure 7.

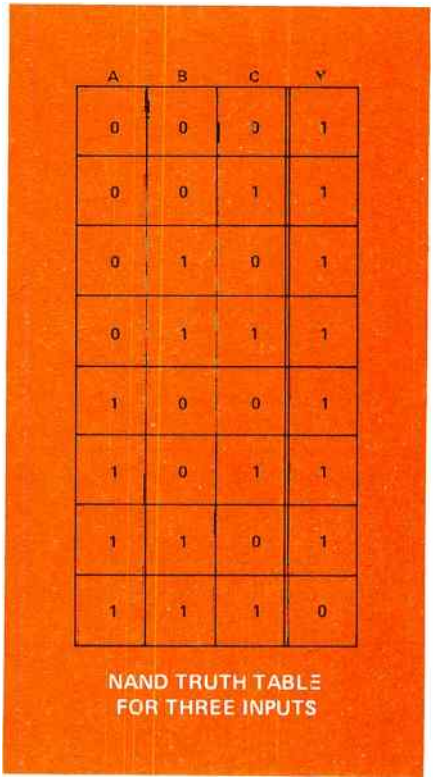


Figure 8.

but in practice, the gates can have more than two. The same logic rules prevail. For a NAND gate, the output will be negative, if, and only if, all the inputs are positive. Similarly, for a NOR gate, the output will be positive, if, and only if, all the inputs are negative. The truth table for a NAND gate with three inputs is shown in Figure 8.

Flip-Flops

One of the most common circuit building blocks formed from groups of logic gates is a flip-flop – widely used for storing a single bit of information. The popularity of flip-flops is due to the following factors:

1. They are available in integrated circuits or can be built from readily available discrete components.
2. They are fast acting – can be made to change states in as little as a few nanoseconds (depending upon the propagation delay of the logic family).
3. They are active devices.

Truth tables rather than circuitry will be used to explain flip-flops. The designer is interested more in what happens to his signal, than how it happens. Having selected his logic

modules from the same family (compatible power requirements, etc.), the designer works with a "black box" and its corresponding truth table. The flip-flops discussed are from the 930 DTL (Diode Transistor Logic) family used in the Lenkurt 91A PCM system.

The basic flip-flop is made of two NAND gates. It has two inputs (R and S) which determine what state ("0" or "1") the flip-flop will assume next, and two outputs (Q and $\overline{Q}$) which determine the flip-flop's present state. The Q and $\overline{Q}$ outputs from any flip-flop are always opposite states; if Q is "1", $\overline{Q}$ is "0" and vice versa.

The inputs and resulting outputs for an R–S flip-flop are shown in the truth table for a particular bit time (Figure 9). The output is a function of the flip-flop's outputs and its inputs at the previous bit time.

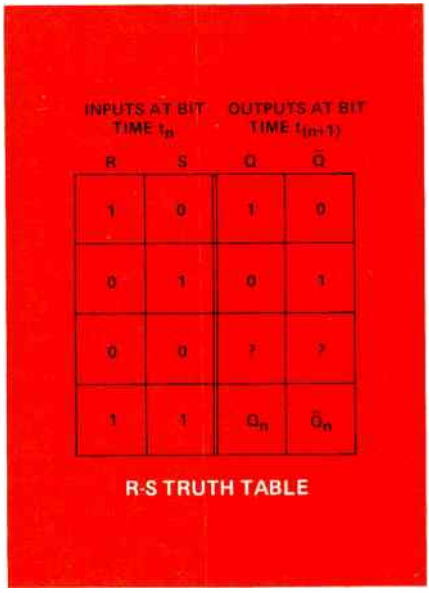


Figure 9.

If the R input is "1" and the S input is "0", the Q output will be "1". If the input states are reversed, the output states will also be reversed. If the input states are both "1", the output will be unchanged from what it was at the previous bit time. The "?" in the truth table indicates the output is undesirable, and therefore to be avoided, when the input states are simultaneously "0".

Although it is possible to design a circuit such that the input states are never simultaneously "0", it is also possible to use a J–K flip-flop which tolerates all possible input combinations (Figure 10).

Regardless of what the output was, it will change to the opposite state, when both inputs are "1". The output will be unchanged, if both J and K are "0".

The J–K flip-flop is essentially two R–S flip-flops in series. The J–K inputs affect the flip-flop only when synchronized with a clock pulse – a steady stream of signals used to allow the input voltages to reach their final value. The direct set and clear inputs (S_d and C_d), on the other hand, operate directly on the output without being synchronized with the clock pulse. The first R–S flip-flop reacts at time "1" as shown in Figure 11; the second R–S flip-flop at time "2"; while the direct set or clear can react at anytime.

If a "0" is applied at C_d, the J–K flip-flop is placed in the clear state (Q=0). If a "0" is applied at S_d, the J–K flip-flop is placed in the set state (Q=1). The S_d and C_d inputs dominate the output even if synchronized with the J–K inputs.

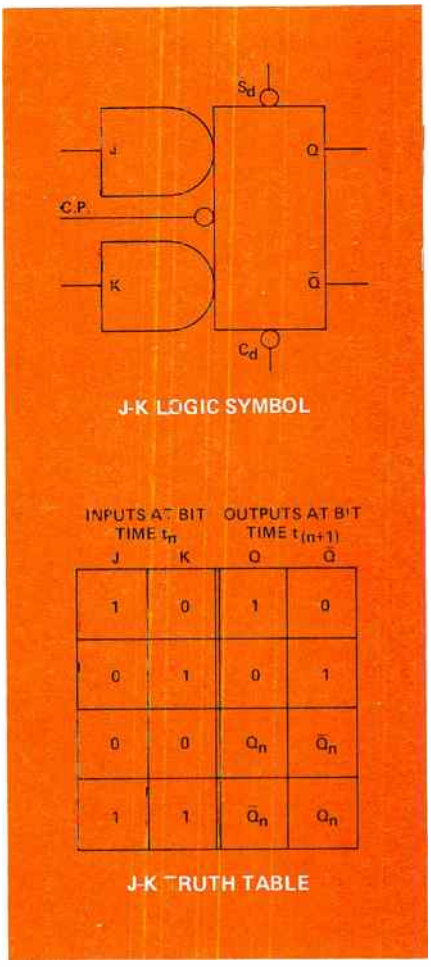


Figure 10.

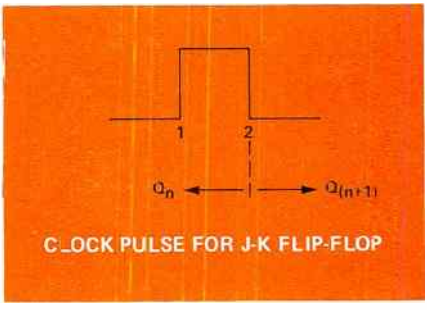


Figure 11.

Logic Modules for PCM Sampling

The sampler at both transmitting and receiving terminals in Lenkurt's 91A PCM system is basically a shift counter. This counter is the heart of the electronic mechanism that sequentially opens and closes the sampling gates for each channel – thereby multiplexing or demultiplexing the signals.

The number of stages in the shift counter is half the number of channels to be sampled; therefore, a 12 stage shift counter is needed to sample 24 channels.

Such a counter can assume 2^{12} = 4096 binary states. Only 24 states are required for sampling – the other 4072 states are undesirable and must be suppressed. The counter is operating properly when these undesirable states have been eliminated and the desired mode 1 operation (Figure 13) is sequentially sending out 24 separate pulses to the gates of 24 separate channels.

Mode 1 operation is accomplished by connecting 12 J–K flip-flops in series – one for each stage (Figure 14).

These flip-flops are driven by a clock pulse. With each clock pulse, the flip-flop state is shifted one stage to the right – the state of stage I at time t_1 will be the state of stage II at time t_2; etc. At stage XII, the Q output is fed back to the K input of stage I and $\overline{Q}$ is fed back to J of stage I. This "crossover" of output to input causes the state to reverse.

In a shift counter, the same state is shifted from one stage to the next with each clock pulse, reversing state when shifting from stage XII to stage I. Mode 1 fits this definition; there-

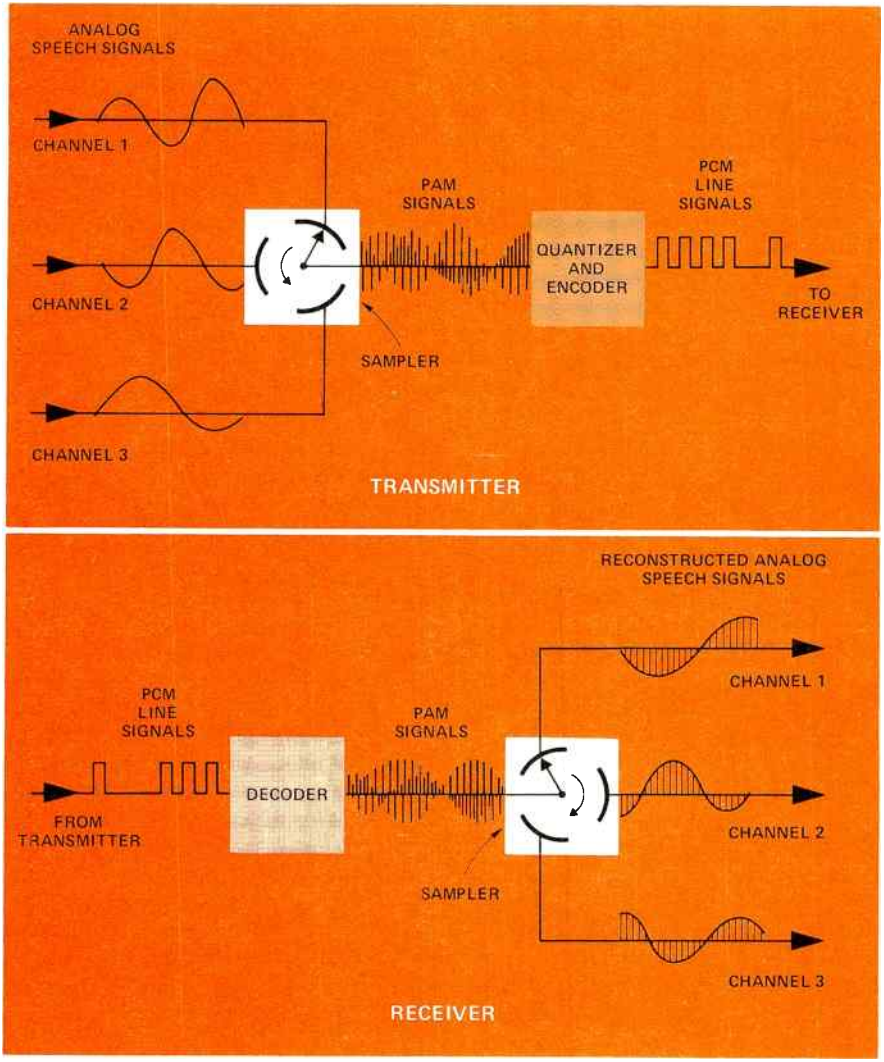


Figure 12. Simplified PCM system (only 3 of the 91A's 24 channels are shown).

fore, once the register assumes a mode 1 pattern, the counter will begin to cycle within this mode.

Applying power to the counter, the register may contain any one of the 4096 possible binary states (110001110000, for example). It is necessary to add either gate "X" *or* gate "Y", to suppress the undesirable states in the shift counter (Figure 14).

Gate "X" is actuated when both stage I and stage XII are in state "1", setting all the internal stages (II – XI) to "1". On the following clock pulse,

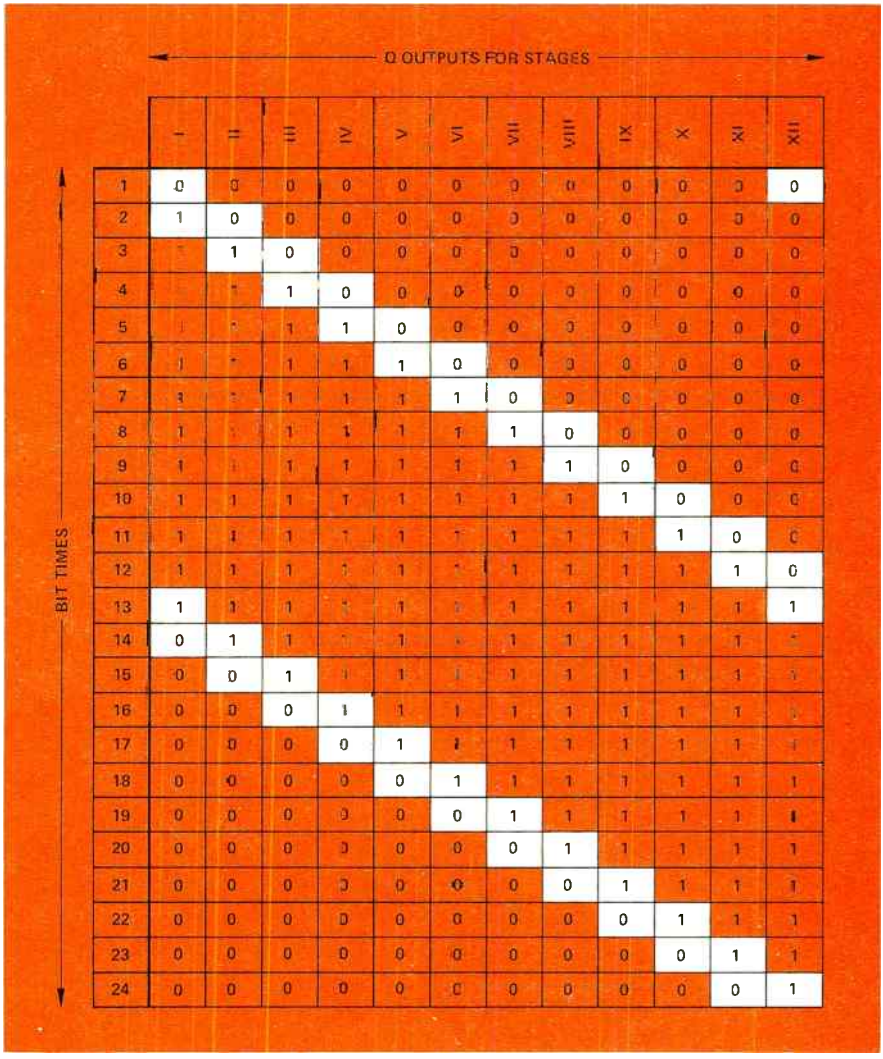


Figure 13. The 24 possible register patterns for mode 1.

stage I goes to "0" and all other stages are "1" as required for mode 1 operation. Figure 15 shows a sequence of patterns which starts with an arbitrary display when the power is applied and continues until the display matches mode 1.

If gate "Y" is used instead of gate "X", all the internal stages (II – XI) are set to "0" when stages I and XII are both "0".

For each of the 24 desirable states of the shift counter there is a readout gate made up of a two-input NAND

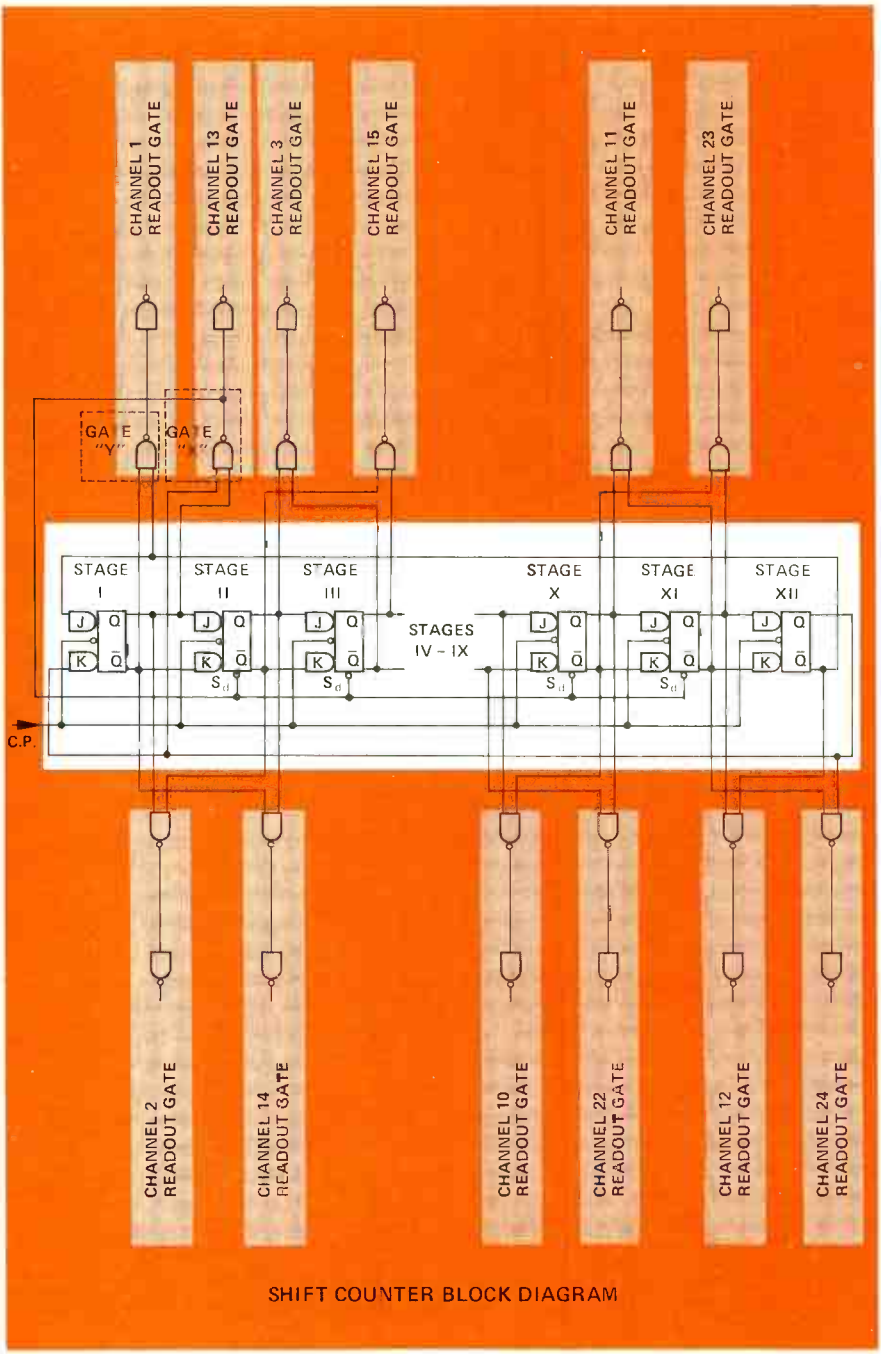

CHANNEL 1 READOUT GATE
CHANNEL 13 READOUT GATE
CHANNEL 3 READOUT GATE
CHANNEL 15 READOUT GATE
CHANNEL 11 READOUT GATE
CHANNEL 23 READOUT GATE
GATE "Y"
GATE "X"
STAGE I
STAGE II
STAGE III
STAGES IV - IX
STAGE X
STAGE XI
STAGE XII
J
K
Q
C.P.
CHANNEL 2 READOUT GATE
CHANNEL 14 READOUT GATE
CHANNEL 10 READOUT GATE
CHANNEL 22 READOUT GATE
CHANNEL 12 READOUT GATE
CHANNEL 24 READOUT GATE
SHIFT COUNTER BLOCK DIAGRAM

Figure 14.

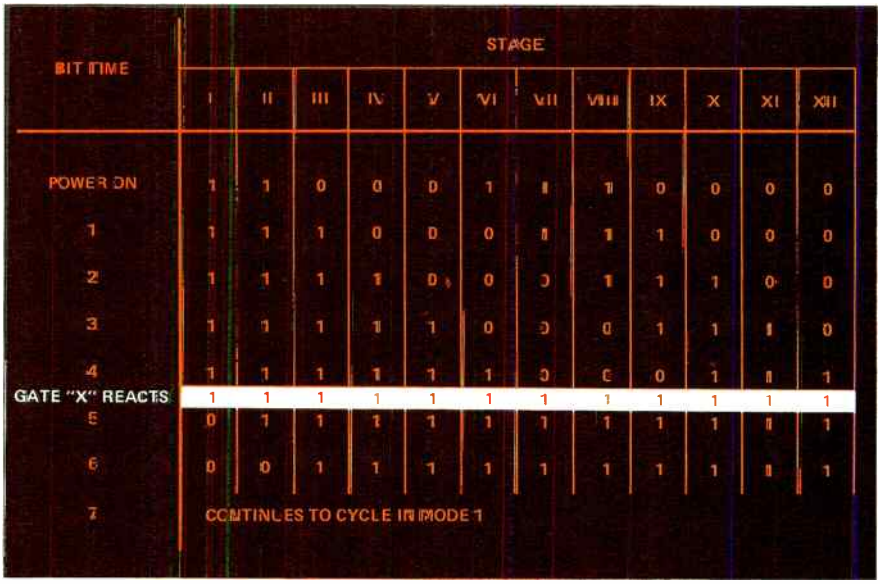


Figure 15.

gate followed by a single input NAND gate. (Figure 14)

The register state can be determined by knowing where there is a transition from "0" to "1" or vice versa, or by knowing that there is no transition (all "0's" or all "1's"). By comparing the outputs of adjacent stages (white areas in Figure 13), the register transition points can be determined. This comparison is done with the readout gate. For each bit time only one readout gate will be in state "1" – indicating the position of the transition point or the lack of any transition.

When a readout gate is in state "1", it opens the corresponding channel sampling gate. The cycling of the 24 readout gates for the shift counter successively opens and closes the sampling gates of each of the 24 multiplexed channels in the 91A PCM system.

Figure 14 shows that the "X" and "Y" gates used for mode suppression of the counter are required for readout – allowing the mode suppression without additional logic modules.

Binary logic and the 930 DTL modules are also utilized in the quantizing and coding equipment for Lenkurt's 91A PCM system.

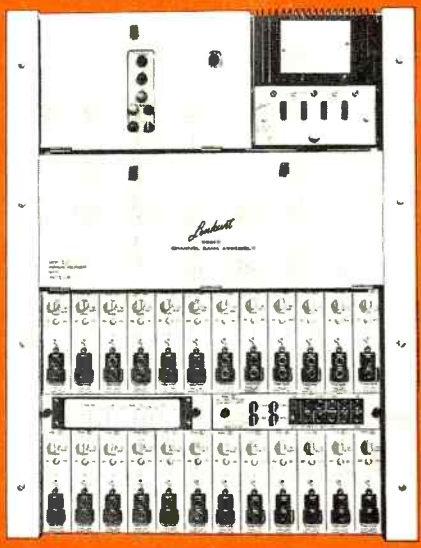
Framework for Expansion

PCM has achieved its present state in the communications industry because of efficient application of binary logic and the timely development of reliable, low cost logic modules.

Binary logic provides the framework for PCM's future expansion.

LENKURT ELECTRIC CO., INC.
SAN CARLOS, CALIFORNIA 94070

MR. E. A. GARCIA
CHIEF OPERATOR
% MILWAUKEE ROAD DEPOT
SAVANNA, ILL. 61074



Lenkurt's 91A PCM Cable Carrier System economically fills the need for low-cost exchange, EAS, and toll connecting trunk services on cable spans extending up to 50 miles or more. The system adds 24 toll-quality channels on two cable pairs – featuring complete end-to-end compatibility with the Western Electric T1 system. For additional information, write Lenkurt, Department C134.