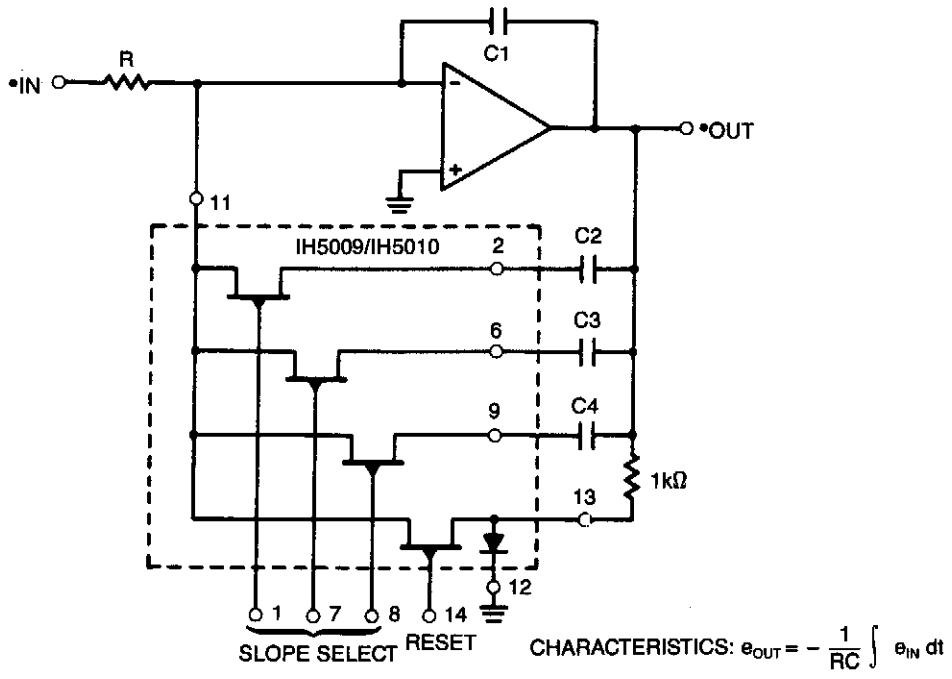
45

Mathematical Circuits

The sources of the following circuits are contained in the Sources section, which begins on page 668. The figure number in the box of each circuit correlates to the entry in the Sources section.

Fast Binary Adding Circuits
Programmable Slope Integrator
Multiplying Precise Commutating Amp

PROGRAMMABLE SLOPE INTEGRATOR



INTERSIL

Fig. 45-1

By using analog switch IH5009/IH5010 to select various capacitors, a variable slope integrator can be had. If $C_3 = 2(C_2)$ and $C_4 = 4(C_2)$, seven different slopes can be obtained if binary information is fed to pins 1, 7, and 8 of the analog switch.

FAST BINARY ADDING CIRCUITS

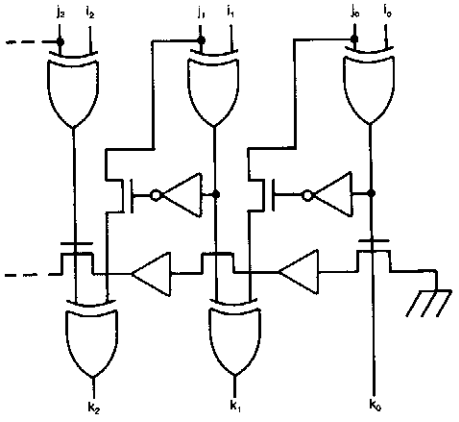


Fig. 45-2(a)

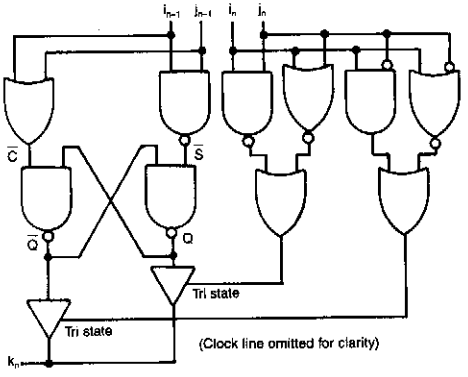


Fig. 45-2(b)

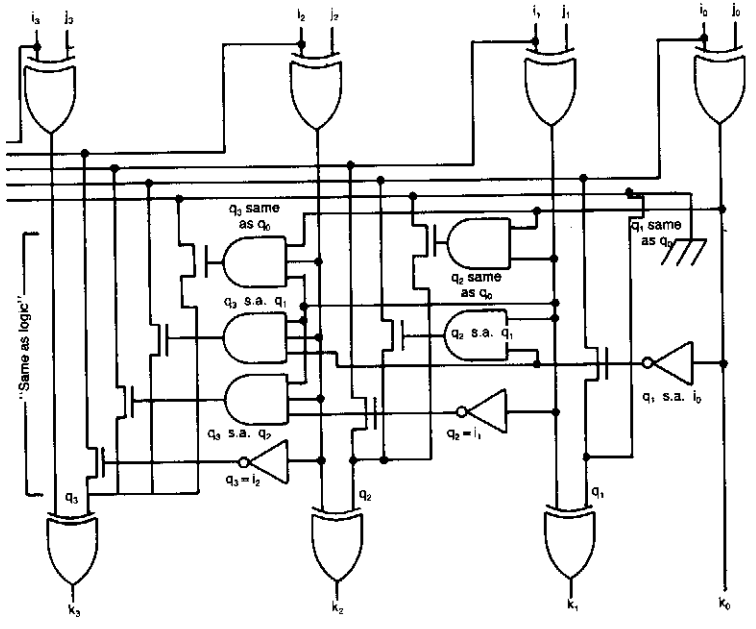


Fig. 45-2(c)

FAST BINARY ADDING CIRCUITS (*Cont.*)

Some circuits that add binary numbers have problems with time delay caused by carry propagation. This has been partially solved by the carry look-ahead adder. However, because of the complexity of this scheme, the carry look-ahead logic usually covers no more than 4 bits, and a ripple carry is implemented between the carry look-ahead blocks.

The Daniels Adder avoids these problems by presenting a scheme where carry bits are not used at all in the process of binary addition. It is based on recognition patterns, which exist with the binary addition truth table.

The addition is described by the following two sets of equations:

$$\begin{aligned}
&\text{if } i_{n-1}=j_{n-1} \qquad q_n=i_{n-1}\\
&\text{if } i_{n-1}=j_{n-1} \qquad q_n=q_{n-1}\\
&\text{if } i_n=j_n \qquad k_n=q_n\\
&\text{if } i_n=j_n \qquad k_n=q_n
\end{aligned}$$

with the boundary condition that $q_{-1}=0$, where i_n, j_n, and k_n are the bit of binary weight 2^n (n^{th} bit) of the addend, summand, and sum respectively, q_n is an intermediate variable and q_n is the inverse of q_n.

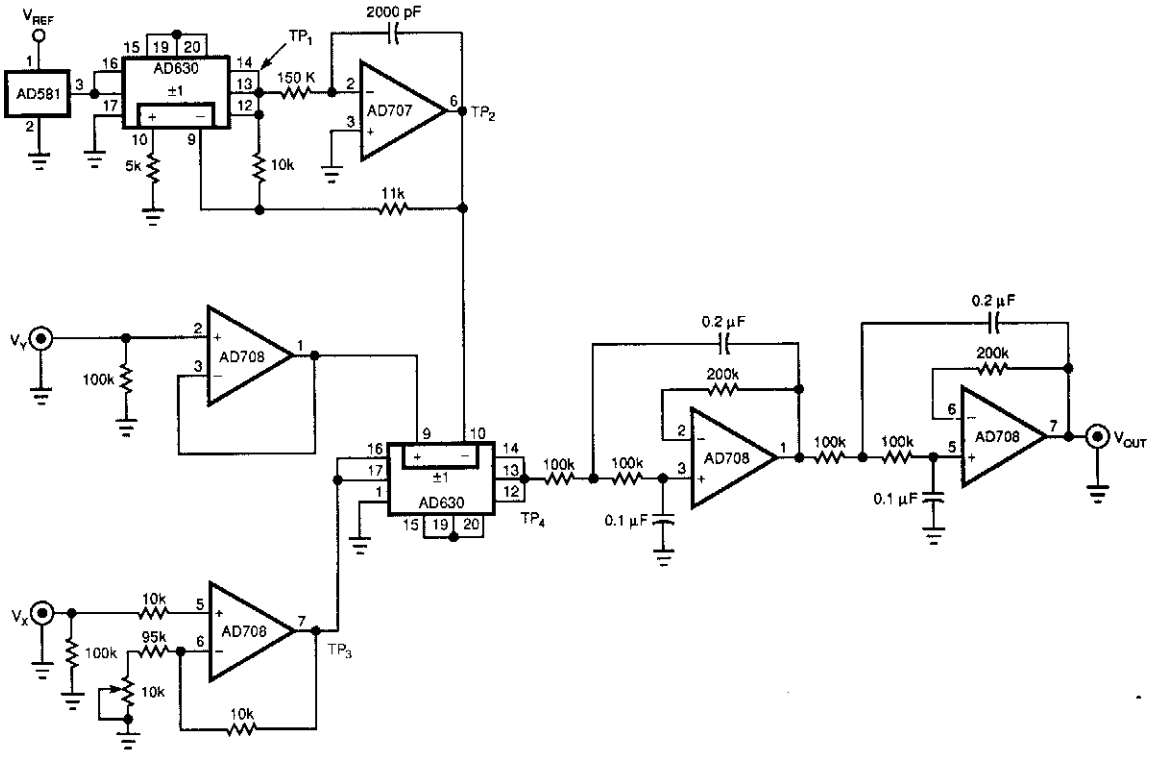
The value of the sum is (depending upon i_n and j_n) either the same as or the inverse of (depending upon i_{n-1} and j_{n-1} a0, a1, or the inverse of the $(n-1)^{\text{th}}$ bit of the sum. Figure 45-1(a) shows the logic diagram of the ripple through implementation of the adder.

Because each stage calculates whether its value of the intermediate variable q_n is the same as the previous stage's value (q_{n-1}) in parallel, it is possible to devise simple "same as" logic that does not have the complexity drawback of carry look-ahead logic and can be carried over any number of bits (Fig. 45-1(b)). A 32-bit adder built in this way will result in 11-gate delays (no gate having more than 4 inputs).

Especially compact and efficient is the pipelined implementation (Fig. 45-1(c)), which can produce the sum at a rate of 3-gate delays/bit.

The high-speed adder circuits can be used on gate arrays or full-custom ICs to implement fast calculation of addresses or data values. Because of their compact nature, they also use less space on the silicon than conventional adders do.

MULTIPLYING PRECISE COMMUTATING AMP



EDN

Fig. 45-3

By using a pulse-width-height modulation technique, this circuit implements a 0.015%-accurate multiplier. The circuit's output equals $V_X V_Y/10$. An AD581 voltage reference, an AD630 commutating amplifier, and an integrator comprising an AD707 op amp, 2000-pF capacitor, and 150-kΩ resistor first generate a precision triangle wave. For a given state of the AD630's output ($+V_{REF}$ at TP_1, for example) the integrator ramps until its output reaches -11 V. Then, TP_1 changes state and the integrator begins ramping toward $+11$ V. The triangle wave's period is $4.4RC$ or 1.32 ms, where R and C are the values of the integrator components.

The circuit uses a second AD630, driven by the variable V_X to compare the triangle waveform at TP_2 to the signal at V_Y. The duty cycle, $T_1 + T_2$, at the output of this second commutating amplifier is:

$$T_1 = \frac{2RC(11 - V_Y)}{10}$$

and

$$T_2 = \frac{2RC(11 + V_Y)}{10}$$

During T_1, the voltage at TP_4 equals $-1.1V_X$. During the remaining period, T_2, the pulse height will equal $+1.1V_X$. V_{OUT} is the average, obtained by low-pass filtering, of this T_1 and T_2 combined waveform and equals:

$$V_o = \frac{-1.1V_XT_1 + 1.1V_XT_2}{T_1 + T_2} = \frac{V_XV_Y}{10}$$

You can use a higher bandwidth filter and a higher carrier frequency to build a faster multiplier.