SUMMARY

This paper describes a number of computing elements whose accuracy is considerably higher than that of comparable analog devices and whose speed of operation is considerably faster than that of comparable digital circuits. These elements employ both analog and digital computing techniques. The input and output signals of these hybrid computing devices are presented by two quantities, the most significant part by a parallel binary number and the least significant part by an analog voltage.

The hybrid multiplier consists of a relatively simple digital multiplier, three D/A converters, a very simple analog multiplier, and a summing and comparison circuit. An accuracy of 1 part in $10^4$ and a bandwidth of 1 KHz is possible with a relatively simple circuit. The accuracy can be expanded theoretically as high as desired at the cost of additional equipment.

The hybrid integrator consists of an accumulating register, a variable amplitude sawtooth generator, a simple analog integrator and a summing and comparison circuit. Its dynamic range and accuracy are limited only by the number of digits used in the R-register.

The hybrid function generator consists of a diode decoding matrix, a diode translation matrix, two D/A converters and a summing and comparison circuit. A repeatability of 1 part in $10^4$ and a bandwidth of 1 KHz is easily possible.

There are no A/D converters, no external storage means, no timing circuits (except clock frequency for integrator), no controls to adjust, and no temperature-sensitive circuits in a system using these hybrid computing elements.

INTRODUCTION

Analog computers are limited to the solution of problems requiring a speed of operation of several hundred cycles, and an accuracy not higher than 0.1% of full scale. In an analog computer each computing element performs, in general, one mathematical operation, and there exists thus a one to one relationship between the complexity of a mathematical problem and the complexity of the computer. We say, the analog computer operates "in parallel".

Digital computers are inherently sequential machines. They perform every mathematical operation through repeated addition or subtraction. The time required to perform a mathematical operation is in a function of the time required for one addition and on how many additions are required. Besides, a digital computer performs a large number of mathematical operations in sequence. The larger this number the slower must be the repetition period. Generally speaking, a digital computer is limited to the solution of problems in which the input variables change slowly with respect to time.

The analog and the digital computer are therefore limited to the two distinct fields of computation. When the solution of a certain mathematical problem requires an accuracy higher than 0.1% and a computation speed higher than several hundred cycles neither the analog nor the digital computer is capable of performing this task economically. Hence, a new computer is required for this class of problems.

It is the purpose of this paper to describe a number of combined analog/digital computing elements which have an accuracy-speed product several orders higher than comparable analog circuits. These elements employ both analog and digital computing techniques, and associated with them are hybrid (partly digital, partly analog) input and output signals.

DESCRIPTION OF A HYBRID COMPUTING SYSTEM

An analog computing system is characterized by the fact that all computing elements, such as adders, multipliers, function generators, integrators, etc., accept analog input voltages and produce analog output voltages.

Accordingly, in a hybrid computing system all computing elements must accept hybrid input signals and provide hybrid output signals. A hybrid signal is defined as being partly digital and partly analog. The more significant part of the signal is represented by a digital signal $X_D$ in the form of a binary or decimal number or some other digital code. The less significant part is represented by an analog quantity $X_A$, usually a dc voltage. In this paper the digital signal has been chosen to consist of a three-bit binary number and the analog signal of a dc voltage which varies between 0 and 1.25V, though in many instances it may be more convenient to use other values. If the variables of this hybrid system are related with the variables of a 10V analog system the magnitudes of the $2^0$, $2^1$, $2^2$ bits of the binary number representing the digital portion are 1.25V, 2.5V and 5.0V, respectively.
A magnitude of an input signal $X$ of 6.685V, would therefore be presented as the binary number $B_{101}$ and the dc voltage of $+5$V. As $X$ increases to 7.499V the analog voltage $X_A$ increases to 1.250V. When $X$ becomes 7.501V $X_A$ increases first to 1.252V, then returns to zero, and increases to .001V. As $X_A$ returns to zero the binary number $X_D$ must increase by a magnitude equal to 1.25V, i.e., the 2nd digit, in order to maintain the magnitude of the total signal constant. It is important that this change-over will be executed as quickly as possible.

Hybrid computing elements employing DDA (Digital Differential Analyser) techniques in combination with analog computing techniques have been developed by the National Bureau of Standards. It will be shown later that such a system not only requires more components but its slewing time is limited by its "unit step" digital operation. In addition, if a DDA computer once makes a mistake, e.g. due to power failure or transients in the system, it would carry that mistake, until the computation is restarted from the initial conditions.

The hybrid computing system described employs only pure digital computing techniques (in addition to the analog techniques). Since all digital operations are carried out in parallel its slewing rate is considerably better. In addition, any remedy failure of the input signals or the equipment has only a momentary effect on the output signal.

From the generalized form of a hybrid computing element in Fig. 1 it can be seen that all inputs and the output consist of a digital part and an analog part. The digital input signal $X_D$ is accepted by a digital computing circuit, while the analog input signal $X_A$ is accepted by an analog computing circuit. There is also one (or more) computing circuits which accept both analog and digital inputs. Since the digital output must have as many digits as the input, the additional digits must be converted back into an analog voltage. The outputs from the D/A converter, the Dig./Anal. computing circuit, and the analog computing circuit are summed in the analog adder. If the analog sum is larger than a certain threshold voltage the comparison circuit subtracts the threshold voltage from the analog sum and provides a "carry" signal which is added to the digital signal. This augmented digital signal is then the digital portion of the output signal. The output from the analog adder is the analog portion of the output signal.

With the number of blocks shown in the diagram of Fig. 1 a hybrid computing element seems to be a complicated and costly device and one begins to wonder whether the increase in cost and complexity justifies the improvement in performance. However, it will be shown that most of these blocks consist of relatively simple electronic circuits.

THE HYBRID MULTIPLIER

One of the best examples on how simple and how cheaply a hybrid computing element can be built is the hybrid multiplier. The circuit described here differs from the NBS version in that the digital input signals are binary numbers and not pulse rates. This eliminates the need for the $X$ and $Y$-registers, in which the input variables are integrated. Without this integration considerable higher bandwidth is possible at the same clock frequencies. Further, instead of the common $R$-register, which allows multiplication only by a unity step, the device to be described employs a parallel binary multiplier.

The operation of the hybrid multiplier shown in Fig. 2 is based on the multiplication of two sums, namely,

$$XY = (X_D + X_A) \times (Y_D + Y_A) = X_D Y_D + X_D Y_A + X_A Y_D + X_A Y_A$$

From the equation above it can be seen that the product $XY$ is now expressed as the sum of four products. At first this may not seem to be a very elegant solution but it will soon become apparent that the circuits required for the individual multiplier elements are relatively simple and that conventional circuitry can be used. From the block diagram in Fig. 2 it can be seen that the first multiplier element is a simultaneous binary multiplier, described in the literature. The second and the third multiplier elements accept one digital and one analog signal and thus digital to analog (D/A) converters can be used. The fourth multiplier element can be a simple, one-quadrant analog multiplier. The outputs from all but the digital multiplier elements are in analog form and must be added in a conventional analog summing device. The number of digits at the output of the digital multiplier is the sum of the digits of the two input signals. In order that the form of the output signal is compatible with the form of the input signals the output signal should have the same number of digits as the input signal. This can be easily accomplished when the output signal is scaled, i.e., when the binary point is shifted and when the least significant digits are converted back into an analog signal. The latter conversion is achieved by means of the D/A converter #5. The scaling of the output signal will be explained in greater detail on hand of the specific example below.

The real advantage of this multiplier can be appreciated only if the magnitudes of the accuracies required from each multiplier element are described. To simplify this description it is useful to make the assumption, that both input and output signals vary in magnitude between the equivalent of 0 and 100V.

Before describing the scaling operation in the multiplier for the binary system it is convenient to describe it first for the decimal system. Assume therefore further that the digital multiplier accepts one decimal digit on each input and provides two decimal digits at the output, and that the analog portion of the variable is represented by a dc voltage varying between 0 and 100V. If we desire now
to multiply, e.g. 9.2 x 8.6 the output from the first multiplier should be 9 x 8 = 72; from the second multiplier the output should be 9 x .6 = 5.4; from the third multiplier the output should be 8 x .2 = 1.6, and from the last multiplier the output should be .2 x .6 = .12. When summed this gives the correct value of 79.12. In order to represent the output signal with only one digit it is necessary to scale it down, i.e., to move the decimal point one place to the left and convert everything on the right of the decimal point into an analog signal. When this analog signal is added to the analog sum derived previously and the whole sum is then attenuated by a factor of ten we have a total analog output signal of (5.4 + 1.6 + 0.12 + 2.0) x 0.1 = .912. Together with the single decimal output (7) from the digital multiplier the total output is then 7 + .912 = 7.912, which is the product 79.12.

The magnitudes of the output signals of each of the multipliers in this example above is inversely proportional to the accuracies required from the various multiplier elements. If it is, e.g., desired that the overall multiplier be accurate to 0.01%, then the D/A converter/multiplier must perform to an accuracy of only 0.1% and the analog multiplier must be accurate only to 1%.

Returning now to the actual multiplier, where a variable is represented by a three-digit binary number and a dc voltage varying between 0 and 1.25V, and using the same numbers as in the example above, gives X = 9.2V = 6.75V + 0.45V = B111 + 0.45V and Y = 8.6V = 7.5V + 1.1V = B110 + 1.14V. The product XY then becomes (B111 + 0.45V) x (B110 + 1.14V) = B101010 + (8.75 x 0.45V) + (1.14V x 0.45V) = B101010 + 13.149V which is, for B 100000 equal to 50V, and B 010000 = 25V, etc., equal to 66.25V + 13.149V = 79.147V. Comparing this with the results obtained above for the decimal system, it can be seen that it is the same result. Proper scaling now requires that the six digit binary number B 101010 is reduced to B101000 and that the rest, B10, is converted back to analog. The voltage equivalent of B 0110 is in our example 3.125V and thus the total analog voltage increases to 16.620V. If now the binary point is shifted three places to the left and if the analog voltage is reduced by a factor of 10 we obtain B 101 and 1.662V which is by our definition 6.25V + 1.662V = 7.922V, the desired product XY divided by 10.

However, the proper code for 7.922V would be B110 + 0.412 and not B 101 + 1.662V, because by definition the analog voltage can vary only between 0 and 1.25V. It can be seen that if 1.25V is subtracted from 1.662V the remainder is exactly 0.412, and that V 101 is just one binary digit smaller than B110. The "carry" operation, so familiar in digital computers, must also be performed here. This requires that whenever the summed analog voltage is larger than 1.25V, a "one" must be added to the binary number representing the more significant part of the output variable, and 1.25V must be subtracted from the summed analog voltage representing the least significant part of the output variable. The comparison circuit in the block diagram of Fig. 2 performs this comparison and switching operation. A carry signal originating at the comparison circuit is sent to the appropriate adder circuit in the digital multiplier element. It also provides the 1.25V which must be subtracted from the summed analog voltage.

The advantage of this multiplier will become obvious only when the circuit simplicity of the individual multiplier elements is shown.

The Parallel Binary Multiplier in Fig. 3 accepts two input signals with three binary digits each and provides a six-digit output. It performs the same algebraic operation as a human operator, multiplying two binary numbers. Binary multiplication is achieved by adding the digits in each column of the array of binary numbers, which is obtained by writing the multiplicand and once for every "one" in the multiplier, but always shifted one digit to the left for each increasing digit of the multiplier.

In the circuit of Fig. 3 the "writing" is done by producing coincidences between the specific digits of the multiplicand and that of the multiplier. Only when coincidence occurs the #and# gate produces a "one". The "ones" in each column are summed by conventional adding circuits. The number of components required by parallel binary multiplier rises sharply with the number of digits used at the inputs. With three digits for both Xn and Yn, the circuit required only 9 gates (2 diodes each) and 9 half adders; if the two inputs have 4 digits each, 16 gates and 16 half adders are needed. In general the circuit requires n² gates and n½ half adders, some buffers (emitter followers) and some additional gating between the half adders to permit full adder operation.

The Digital to Analog (D/A) Converter/Multiplier For a three-digit input the D/A converter/multiplier in Figure 4 comprises only three complementary transistor voltage switches and three binary-weighted precision resistors. One side of all switches is always connected to ground and the other to the analog input variable X₃ or Y₃. When the switches are energized by the control lines of the digital input variable the current flowing to the summing point through the three resistors is proportional to the product X₃Y₃ or X₃Y₃.

The Analog Multiplier It has been mentioned before that the accuracy required from the analog multiplier needs to be only approximately ±1% of full scale in a system with a total accuracy of ± 0.01%, and with 3-digit binary numbers.

An extremely simple version of such a multiplier is shown in Figure 5. Its operation is based on the triangular wave integration principle.
A triangular wave is biased to the level of the first input variable X. This biased waveform is then clipped at +Y and -Y, the second multiplica-
tion variable, to produce a trapezoidal wave train. When this wave train is subjected to low-pass fil-
tering the resultant output voltage is proportion-
al to the product XY.

Biasing of the triangular wave is obtained by returning the secondary of transformer T-1 to the potential representing X. Clipping of the trian-
gular wave is achieved by feeding the triangular wave to the bases of a complementary transistor voltage switch. This switch consists of one pnp and one npn transistors, connected at the emitters. The collector of the pnp transistor is connected to -Y, the collector of the npn transistor to +Y. The voltage at the emitters of these two transis-
tors is then the biased triangular wave accurately limited to +Y and -Y. Low-pass filtering is done by cascaded RC stages.

The Summing and Comparison Circuit. For ease of understanding, the summing and comparison circuits have been shown in Figure 2 as two blocks, in reality, however, they are combined in one circuit. In essence, this circuit is a simplified version of the partial A/D converter with only one binary digit.

The single flip-flop shown in Figure 6 is supposed to be set when Vp, the output voltage of the d-c amplifier, becomes more negative than 1.25V, and it is supposed to reset when Vp becomes more positive than ground. The output of the flip-flop is used to provide the "CARRY" signal to the add-
ers in the digital multiplier and to energize the transistor voltage switch, which connects +1.25V to the summing point.

Overall Performance of the Multiplier. The static accuracy of this multiplier is limited only by the complexity and finally by the cost of the circuit, in particular, of the digital multiplier. Theoretically, as in any digital computer, this multiplier can be made as accurate as desired, by using more digits for the more significant part of the variable. Practically, however, it may be most economical to operate the hybrid multiplier with an accuracy of 1 part in 10^4 to 10^5.

In contrast to the digital computer the resolu-
tion of these hybrid circuits is very high, due to the analog portion of the variable, and is limi-
ted only by the steps occurring during switchover.

The dynamic range of this multiplier is deter-
ded by the noise in the circuit, which consists
mainly of the drift in the d-c amplifier, the volt-
age drop across the transistor voltage switches
and the integrated effect of the transients occur-
ing during the switchover. In general this noise is below 1mV and thus in a 10V full scale system, the dynamic range would be 10,000 or higher.

The frequency response of such a hybrid multi-
plier would also be relatively high since the
bandwidth of the individual multiplier elements
may be made high. It is expected that this multi-
plier should be able to handle frequencies above
100KHz.

THE HYBRID INTEGRATOR

The hybrid integrator described in this paper differs from the NBS version because it employs pure digital techniques instead of the DDA tech-
niques. This eliminates the need for an additional register in which the input signal is integra-
ted. Without this integration higher bandwidth is possible. In order to assure that the output ap-
proached a continuous function, the NBS integra-
tor employs a resettable integrator, whereas the
integrator to be described uses a variable-ampli-
tude sawtooth generator. The replacement of the resettable integrator by a variable-amplitude saw-
tooth generator is a major circuit simplification.

If the independent variable X, which is repre-
sented by a digital value Xp and by an analog
value Xa, is integrated with respect to time, the integral becomes

\[ Y = \int_{0}^{t} \left( X_p X_n \right) dt = \int_{0}^{t} \left( X_p + \frac{1}{T} \right) X_n dt \]  

(2)

Further, if the time is divided into equal inter-
vals of duration \( \Delta t \) and if the digital quanti-
ty is permitted to change its value only at the
beginning of \( \Delta t \), then the integral can be expressed as

\[ Y = \frac{1}{T} \sum_{i=1}^{N} \left( X_{pi} \Delta t + X_{pi} \left( t - (n-1) \Delta t \right) \right) \]  

(3)

Where \( X_{pi} \) is the value of \( X_p \) at the i-th inter-
val of \( \Delta t \), and \( X_{pi} \) is its value in the n-th inter-
val.

In Fig. 7 the integral is represented as the
area under the curve \( X(t) \) from the time \( t = 0 \) to an arbitrary \( t \). In the graph and in the equa-
tions above it has been assumed that the initial
value of \( Y \) is zero. The three terms within the
brackets of equation (3) correspond to the three
areas depicted in Fig. 7. Area 1 is the integral of the digital part of \( X \) between \( t = 0 \) and \( t \), are-
a 2 is the integral of \( X_p \) between \( (n-1)\Delta t \) and \( t \), area 3 is the integral of the ana-
log part of \( X \) between \( t = 0 \) and \( t \).
The digital parts of the input and output signals of the Link hybrid computing elements are parallel binary signals, and therefore no input register is required. The Link hybrid integrator in Fig. 3 consists of three parts which produce the three signals that correspond to the three areas in Fig. 7.

In the digital part of the integrator Xp is added into the R-register at the beginning of each \( \Delta t \). The contents of the R-register is a \( n \)-digit binary number representing area 1, where \( n \) is a function of the dynamic range desired. \( k \) of the \( n \)-digits (in Fig. 8 the constant \( k \) is assumed to be 3) represent directly the digital output of the integrator. The remaining \( n-k \) digits are converted back into the analog voltage \( V_A \).

In the hybrid part of the integrator a variable amplitude sawtooth generator provides the voltage \( V_{2} \) which is proportional, to both \( X_{p} \) and to \( t \). The variable amplitude sawtooth generator can, but need not be a D/A converter and a resettable integrator, if \( X_{p} \) is kept constant during the time interval \( \Delta t \). The output from the resettable NBS integrator is nothing but a sawtooth with variable amplitude and constant period \( \Delta t \). It is therefore suggested for the Link hybrid integrator that only a D/A converter be used and that the reference voltage \( V_{p} \) be replaced by a sawtooth wave with a constant amplitude and period. The output from the D/A converter will then be also a sawtooth wave with an amplitude proportional to \( X_{p} \). The resettable integrator in the NBS circuit is a rather complicated device which consists of an analog integrating amplifier and a pulse switching circuit to discharge the integrating capacitor in as short a time as possible. The replacement of it simplifies the integrator circuit significantly. The reference sawtooth used in the Link integrator needs to be generated only once for a number of integrators.

In the third part the analog portion \( X_{3} \) of the input variable \( X \) is integrated into the voltage \( V_{3} \) by a conventional integrating amplifier.

The three voltages \( V_{i1}, V_{i2}, V_{3} \) are summed to produce the analog output voltage of the integrator \( V_{YA} \), which is also compared with an upper or lower threshold voltage \( V_{TH} \). When \( V_{YA} > + V_{TH} \) the comparator produces a "carry" signal which adds \( +1 \) to the contents of the R-register and subtracts \( V_{TH} \) from \( V_{YA} \); when \( V_{YA} \leq - V_{TH} \) the carry is removed from the R-register.

The R-register accepts the \( X_{p} \) input lines and adds the value of \( X_{p} \) to the binary number stored already in the register. There are several methods of achieving this task. Fig. 9 illustrates a straightforward version of such a circuit, which consists of a parallel-binary adder and a \( n \)-digit binary counter. The parallel adder sums \( X_{p} \) with the outputs from the last three digits of the counter. Its output is connected to the inputs of the least significant stages of the counter in order to "set" or "reset" the flip-flops. To assure that the register output is changing only at the beginning of each \( \Delta t \) the digit lines of \( X_{p} \) are gated with the clock frequency by means of conventional "And" circuitry.

The Variable Amplitude Sawtooth Generator must generate a sawtooth wave with constant period and an amplitude proportional to \( X_{p} \). As mentioned before this can be accomplished by connecting a sawtooth wave with constant period and constant amplitude into a conventional D/A converter as shown in Fig. 4.

The D/A converter attenuates the reference sawtooth wave accurately to an amplitude which is directly proportional to \( X_{p} \). The number of stages required for the D/A converter is identical with the number of digits used for \( X_{p} \).

Since there exists a multitude of circuits which generate a sawtooth wave with constant period and constant amplitude this circuit will not be described here.

The Analog Integrator is an operational amplifier with an integrator capacitor in its feedback path, however, it can be just as well only a simple RC integrator, if more digits are used for \( X_{p} \).

The Summing and Comparison Circuit must add the input voltages \( V_{i1}, V_{i2}, V_{3} \) and compare this sum with a positive and with a negative threshold voltage \( V_{TH} \) in exactly the same fashion as was described for the hybrid multiplier. The "carry" signal produced by the comparison circuit must be added to the contents of the R-register.

**THE LINEAR-SEGMENT HYBRID FUNCTION GENERATOR**

Function generators in which the output signal is related to the input signal in any arbitrary fashion usually employ linear-segment approximation principles. In order to design such a function generator the designer must know the values of the function at certain fixed points of the input variable, and the slope of the function between any two adjacent fixed points.

In present-art linear-segment diode function generators, a reverse-biased diode becomes conductive when the input signal becomes equal to or larger than a certain fixed value, which is referred to as breakpoint. The amount of reverse bias on the diode is equal to the value of the function at the breakpoint. Each diode that has been made conducting decreases the slope of the function for a particular segment. It can therefore be said that the linear-segment diode function generator stores the value of the function for each breakpoint and the value of the slope between any two adjacent breakpoints. In a 100V computing system the accuracy with which a diode function generator can retrace the linear-segment curve is of the order of 0.1%. This requires
that the circuit has been carefully set up, which is generally a very tedious and time-consuming job. When the diode function generator must operate in a 10V computing system its repeatability is considerably worse, due to the inherent noise or threshold level of the diodes.

The function generator to be described in this paper employs also, the linear-segment principle, but it stores as parallel binary numbers the value of the function and the value of the slope. In contrast to the biased-diode function generator in which all operations are performed by analog computing elements, the function generator to be described performs only the linear operation of summing with a dc amplifier; all other operations are performed with digital or hybrid computing elements.

The combination of analog and digital computing techniques lends itself very well to the generation of arbitrary functions of one or two variables. Such hybrid function generators exhibit unusually high repeatability combined with a relatively high speed of operation.

Compared with the biased-diode function generator the following advantages can be listed for the hybrid function generator:

1. The repeatability is at least one order higher.
2. The speed of operation is the same.
3. The circuit is largely insensitive to temperature variations, since diodes and transistors are used only as digital switching elements.
4. The generator has no controls to adjust and no biases to be set.
5. The function generator can be set up or changed from one function to another by inserting pins in a patchboard or by inserting a punched card. Which of these methods is used depends entirely on the intended application.

A linear-segment function generator can only approach the curve \( Y = f(x) \) in Fig. 10. The eight breakpoints \( P_0, P_1, P_2, \) etc. on the curve have the abscissas \( x_0, x_1, x_2, \) etc. and the ordinates \( f(x_0), f(x_1), f(x_2) \) etc., respectively. The eight equally-spaced values of \( x \) correspond to the eight values of the three-digit binary number representing \( x_D \). The slope between any two breakpoints is

\[
\frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} = \Delta f(x_i) = \Delta f(x_i) \quad (1)
\]

In order to find the value of \( y = f(x) \) for a value of \( x \) that lies between two adjacent values of \( x_i \), i.e., between \( x_i \) and \( x_{i+1} \) an increment must be added to the value of the function at \( x_i \). This increment is the product of the slope of the function between these two breakpoints and the increment of \( x_i \), defined generally as \( x_i^+ \) or as \( x_i^- \) in the hybrid computing system. The value of the function for any magnitude of the input variable can be expressed for any linear-segment approximation as

\[
f(x) = f(x_i) + x^+ \Delta f(x_i) \quad (5)
\]

Every computing element described in this paper is required to accept hybrid input signals and to provide hybrid output signals. The final version of the hybrid function generator will also satisfy this requirement. However, it is convenient to describe at first the function generator with hybrid input and analog output as shown in Fig. 11. A function generator of this kind, but in combination with a partial A/D converter was disclosed previously \(^2\). At that time it was desired to have a function generator with analog input and analog output. This necessitated the use of a partial A/D converter, which reduced the dynamic performance of the function generator. In a computing system where all variables are represented by hybrid signals no A/D converter is required and the function generator is thus capable of operating with relatively high input frequencies.

The function generator in Fig. 11 comprises a diode decoding matrix, a diode translation matrix, two digital to analog (D/A) converters and a dc amplifier.

The diode decoding matrix converts the three-digit binary numbers representing the digital portion \( x_D \) of the input variable \( x \) into eight control lines. Only one of these eight control lines is energized at any one time.

Each of these eight control lines is connected to one horizontal bus wire of the diode translation matrix. The diode translation matrix provides two parallel binary numbers as outputs if one of its eight inputs is activated. The \( k \) digit number originating in section \#1 represents the value of the function \( f(x_D) \) for a specific digital value \( x_D \), the \( m \) digit number originating in section \#2 represents the value of the slope \( f(x_D) \) in the \( x_D \) segment.

Each of the two D/A converters accepts one of the two binary numbers as digital inputs and provides an output which is proportional to the product of the digital signal and the reference potential. The number representing \( f(x_D) \) is converted into an analog voltage by D/A converter \#1. Its reference voltage \( V_{ref} \) and its output voltage is thus \( V_{ref} f(x_D) \). The number repre-
resenting the slope \( \Delta f(x_D) \) is converted into an analog voltage by D/A converter \#2 to which \( V_R \), the voltage representing the analog portion of the input signal, is connected as reference voltage. The output from the second converter is thus \( V_{RA} \Delta f(x_D) \).

When the outputs from the two D/A converters are summed in a conventional DC operational amplifier, an output voltage is obtained which magnitude is proportional to the desired value of the function, i.e.

\[
V_D(x) = V_R f(x_D) + V_{RA} \Delta f(x_D)
\]  
(6)

It has therefore been shown that the function generator in Fig. 11 is capable of generating arbitrary functions of one variable. Although the output is an analog voltage, there are certain instances where this function generator may be used also in a hybrid computing system. If, however, the function generator must drive another hybrid computing element, it must provide an output signal which is also in the hybrid form.

A function generator with hybrid input and hybrid output signals is illustrated in Fig. 12. The diode decoding matrix, the diode translation matrix and D/A converter \#2 are identical to those shown in Fig. 11. For any specific \( x_D \) input, the two sections of the diode translation matrix provide two binary numbers as outputs; sections \#1 provides a \( k \)-digit binary number representing \( f(x_D) \), while section \#2 provides a \( m \)-digit binary number representing \( \Delta f(x_D) \). D/A converter \#2 accepts the number \( \Delta f(x_D) \) and multiplies it with \( V_{RA} \), the voltage representing the analog portion of the input variable \( x_D \), in the same manner as described before, the \( k \)-digit binary number for \( f(x_D) \) is now split into the three most-significant digits and into the \( k-3 \) least significant digits. The latter digits are connected to D/A converter \#1, which converts them into an analog voltage \( V_{RA} f_3(x_D) \). The outputs from the two D/A converters are summed again by a standard DC operational amplifier.

The output from the DC amplifier constitutes the analog portion \( V_{RA} \) of the hybrid output signal. This analog voltage is then compared with a threshold voltage in the comparison and switching circuit. If \( V_{RA} > V_{th} \), a voltage is fed back to the summing point of the dc amplifier to subtract \( V_{th} \) from \( V_{RA} \), and at the same time a digital signal is provided which indicates that \( V_{RA} \) has exceeded the threshold voltage. When \( V_{RA} \leq V_{th} \), the feedback signal and the digital signal are removed.

A parallel binary adder sums the three most significant digits from section \#1 of the diode translation matrix with the digital (carry) signal from the comparison circuit. The output from the adder circuit is another three digit parallel binary number which represents the more significant part of the output variable \( y \).

With this hybrid form of output representation the repeatability and dynamic range of the function generator can be, theoretically, extended as high as desired by using more digits.

Most of the circuitry used in the linear-segment hybrid function generators is conventional and therefore emphasis is put to the general circuit form, to special components used, to the number of subcircuits used and to their interconnections. Only where the circuits differ from the basic conventional design will they be described in detail.

The Diode Decoding Matrix. This matrix is of conventional design and consist of \( 2^s \) \( \text{AND} \) gates, each having \( s \) diodes. For example in Figure 11, and \( \text{Section 1} \), the number \( s = 3 \), Therefore the required matrix must have \( 8 \) \( \text{AND} \) gates, each with \( 3 \) diodes. Because the outputs from the diode decoding matrix must be able to drive the diode translation matrix they are passed through conventional emitter followers.

The Diode Translation Matrix. This matrix is an array of horizontal and vertical bus wires. There are \( 2^s \) horizontal wires which accept as inputs the signals on the \( 2^s \) control (or output) lines from the diode decoding matrix. An input signal can exist only at one horizontal wire at a time. For each input the translation matrix must provide as output two parallel binary numbers. For this reason the vertical wires are split up into two sections, each of which has as many vertical wires as there are digits in the binary number. Section number one, which is to represent the value of the function \( f(x_D) \), at one of the \( 2^s \) breakpoints must have \( k \) digits; section \#2, which is to represent the slope of the function \( \Delta f(x_D) \) between the breakpoints \( x_D \) and \( x_{D+1} \), has \( m \) digits. The numbers \( m \) and \( k \) depend on the accuracy desired from the circuit. Each vertical line has therefore a specific binary significance.

In order to have a specific binary number as output from each of the two sections, it is necessary to provide a low impedance path between the particular horizontal line that has been energized and the appropriate vertical lines. If, e.g., the number desired is 011000... only connections to the second and the third most significant digits are required, for all the "1" (ones) in the binary number.

The connections required must be such that a low impedance between a horizontal and a vertical line exists only when the particular horizontal line is energized, for all the other energized horizontal lines a high impedance must exist.

There are several possible ways in which two lines can be connected with each other; still fulfilling the requirements set forth above. However, if speed of operation and cost must be considered the semiconductor diode seems to be the most favorable component for this job.
The operation of the matrix will be understood better by referring to Figure 13 where a circuit is shown, which solves the function $f(x) = x^2$. This particular function has been chosen because it requires only a few diodes in the translation matrix, and because the performance of the function generator is simple to check.

The circuit in Figure 13 is used for the actual breadboard, for which there are 8 control lines and thus 8 horizontal lines in the translation matrix. Since $f(x_0)$ and $f(x_2)$ are represented by an 11-digit binary number, there are thus 11 vertical lines in each section of the matrix. For convenience these vertical lines are "weighted" from $2^0$ for the most significant digit to $2^9$ for the least significant digit. The input signal to the horizontal lines are $+15V$ when the line is energized, and $-15V$ when the line is not energized. The breakpoints had been chosen as follows: $x_0 = 0$, $x_1 = 1$, $x_2 = 2$, $x_3 = 3$, etc. For these values of $x$, the values of the function are $f(x_0) = 0$, $f(x_1) = 1$, $f(x_2) = 4$, $f(x_3) = 9$, etc., and the values of the slope are $\Delta f(x_0) = 1$, $\Delta f(x_1) = 3$, $\Delta f(x_2) = 5$, $\Delta f(x_3) = 7$, etc. The diode translation matrix which can also be looked upon as a memory must therefore be loaded with the binary numbers for the values of $f(x_0)$ and $f(x_1)$. Since they are all integers the digits $2^0$ to $2^9$ are not required. When the maximum number of diodes on one vertical line is eight, one is conducting and seven are cutoff. The ratio between the forward impedance and the cutoff impedance of the individual diodes should be at least 20 times higher than the 7:1 ratio mentioned above, to obtain a signal to noise ratio larger than 10:1. Even the cheapest diodes on the market will fulfill this requirement. The signals on the vertical lines are the outputs of the translation matrix, which must control the switches in the D/A converters. The amplitude of these output signals must thus be $15V$, and the output impedance must be low enough to guarantee a current of 1mA. All diodes on one vertical line form an "OR" gate. In order to provide $-15V$ at the output it is necessary to return the "OR" gate to a negative potential.

The Digital to Analog (D/A Converter) used with the linear-segment hybrid function generator is based on the voltage decoder with ladder network II, which is described by A. Susskind. The voltage generators shown in the above mentioned reference are replaced with complementary transistor voltage switches which connect either zero or the reference potential to the ladder network. The basic characteristic of a ladder network is such that the current contribution from each stage is half the current contribution of the previous or more significant stage.

The number of stages of the ladder circuit is determined by the number of digits used in section $\delta$ or $\delta'$ of the diode translation matrix. There is no theoretical limit to the number of stages in the ladder network; a practical limit is the noise level in the system. Because the signal to noise ratio on these transistor switches is higher than 10,000:1, the D/A converter is capable of operating essentially with the same accuracy as the accuracy of the resistors used in the ladder network.

The Summing and Comparison Circuit shown in Fig. 12 for the hybrid function generator is identical in hardware and performance as that described for the hybrid multiplier and illustrated in Fig. 6.

The DC Operational Amplifier used for summing and comparison is operating most of the time as a simple summing amplifier with a 1:1 feedback ratio. But, when $V_{YA}$ exceeds the upper or lower threshold voltage the input to this amplifier is changed rapidly and the output is required to follow as fast as possible. Since, however, $V_{YA}$ must be accurate only to approximately 1/10, (for three-digit hybrid signals) the requirements on drift stability and linearity are not extremely high. Summing up the dc amplifier should provide a gain of about 10&sup2; and a bandwidth of 10KC or higher.

The Performance of the Hybrid Function Generator. Both the static and dynamic performance of the hybrid function generator are limited by the performance of the D/A converters, by the summing and comparison circuit and by how many digits are used in the digital portion of the input and output variable.

With the circuit of Fig. 12 a static repeatability of 0.01% of full scale output has been measured, but better repeatability is possible, even with three digits, since the analog portion of the output signal alone can be made to be accurate to 0.05%.

For any linear-segment function generator repeatability expresses how closely the circuit approximates a given curve. The latter can therefore be used only when reference is made to a specific curve and when the number and position of the breakpoints is defined.

Dynamic performance results are not available yet, but it is expected that the circuit will operate with input frequencies of several kilocycles. The major limitation on bandwidth is placed upon the circuit by the dc amplifier, used for summing and comparison, which is required to change its output signal from the threshold voltage back to zero in as short a time as possible.
ADDITION OF HYBRID SIGNALS

In analog computers the addition of two or more signals can be easily performed by connecting the signals over suitable resistors to the summing point of a dc operational amplifier.

In digital computers the addition of two n-digit binary numbers requires n "full adders" if the operation is to be performed in parallel.

Consequently the addition of two hybrid signals, each consisting of a k-digit binary number and a dc-voltage, requires k "full adders" (K<sub>k</sub>n) and a dc operational amplifier. Since the sum of the two analog signals may exceed the full-scale value a comparison circuit is also needed to provide the "carry" to the digital adder.

CONCLUSION

The hybrid computing elements described in this paper, with their expandable accuracy and relatively high speed of operation, are believed to be most useful in the solution of problems requiring an accuracy one or two orders higher than that of comparable analog circuitry and a speed of operation in the kilo cycles region.

A breadboard has been built for each of the hybrid circuits described in order to confirm the basic principle of operation. Although only the hybrid function generator has been subjected to thorough static testing a number of basic problems have come up, which require further attention.

The speed of operation of all hybrid computing elements is largely determined by the speed of the comparison circuit. More specifically it is the DC amplifier used with that circuit, which must change its output from full scale to zero during a transition from analog to digital, and vice versa. If the amplifier output returns to zero after the digital number has changed, then the total output is for a short time too large, if it returns before, the output is too small.

When the digital and analog outputs are appropriately combined and displayed this overlapping or ambiguity shows up as a positive or a negative spike, the duration of which is equal to the time difference between the analog and digital operations. Since the frequency of these spikes is high compared to the basic operating frequency it should be possible to filter them out. However, the integrated transients produce a change in the amplitude of the output signal and thus an error, whose magnitude can not be neglected. In closed-loop systems both the unfiltered transients and the discontinuity of output signals during the switchover have adverse effect on the stability characteristic.

Another problem to be analyzed is how to handle variables with both positive and negative polarities. In the multiplier, e.g., it would be most advantageous to represent the input variable in the "signed magnitude" code in order to keep the digital circuit as simple as possible. All circuits involving adders, on the other hand, prefer numbers represented in the "two's" complement form. Therefore some compromise solution must be found.

A third problem is that of scaling or multiplying by a constant. In analog circuits this is easily achieved by changing the scaling or summing resistors. In hybrid computers it may require as much as a simplified multiplier circuit.

These, and probably many more, problems must be solved before a hybrid computing system with the elements described will become a useful tool.

REFERENCES

Fig. 1. Generalized form of a hybrid computing element.

Fig. 2. Block diagram of the hybrid multiplier.
Fig. 3. Parallel binary multiplier for two 3-digit numbers.
Fig. 4. Three-digit D/A converter/multiplier.

Fig. 5. Two-quadrant analog multiplier.
Fig. 6. Summing and comparison circuit.

Fig. 7. Hybrid integration illustration.
Fig. 8. Block diagram of the hybrid integrator.

TOTAL OUTPUT FROM R REGISTER

Fig. 9. Block diagram of the R-register.
Fig. 10. Linear segments approximate the curve \( Y = f(x) \).

Fig. 11. Linear-segment hybrid function generator with hybrid input and analog output.

\[ V_0(X) = V_R f(X_D) + V_{XA} \Delta f(X_D) \]
**Fig. 12.** Linear-segment hybrid function generator with hybrid input and output.

**Fig. 13.** Diode translation matrix wired for \( f(x) = x^2 \)