A Modular, High-Speed Serial Pipeline Multiplier for Digital Signal Processing

GARY L. BALDWIN, SENIOR MEMBER, IEEE, BERNARD L. MORRIS, DAVID B. FRASER, AND ANGELO R. TRETOLA, MEMBER, IEEE

Abstract—**This paper describes the design and implementation of a 44 Mbit/s serial pipeline multiplier that exploits an efficient algorithm with a novel circuit architecture. The multiplier, intended for use with signed-magnitude coefficients and two's complement data of arbitrary length, produces products automatically rounded and truncated to the same length as incoming data. The circuit's design focuses on the bit-cell, a unit of circuitry associated with one bit of the coefficient word, from which multipliers of arbitrary complexity may be constructed. A practical realization of this multiplier contains four bit-cells, each of which dissipates 20 mW, as well as all associated data, coefficient, and control registers necessary for its operation. The total power dissipation for the chip is 140 mW. The physical implementation of the multiplier employs buried-collector bipolar devices and two-level aluminum metallization to obtain a compact chip 120 mil^2. Descriptions of the circuit's arithmetic architecture, design, performance, and use are given in detail.**

I. INTRODUCTION

DIGITAL multipliers continue to occupy a position of major importance in the design and development of general- and special-purpose computers. As one of the key elements in digital signal processors and digital filters, the multiplier has traditionally hampered the consideration of the digital approach to signal processing owing to high cost, poor speed-power performance, and inefficient packaging. Recent advances in the realization of digital multipliers as monolithic integrated circuits [1]-[4], [14] have done much to reduce multiplier delay times and power consumption, and the size and cost of digital signal processors as well. But the demand for lower cost, higher efficiency realizations of digital filters [5] requires continued improvements in the performance of digital multipliers. This paper describes the design and implementation of a 44 Mbit/s digital multiplier that exploits an efficient multiplication algorithm with a novel circuit architecture. The multiplier is similar to one described by Kane [3] but employs a different algorithm, a different technology, and a more compact design to achieve higher speeds at a lower power dissipation.

While systems organized with parallel data (for which a parallel multiplier may be better suited) have been used extensively in computer hardware, the serial approach to digital filter organization is efficient in highly multiplexed serial applications such as the processing of telephone voice and data signals [6]. However, this approach normally requires high-speed multipliers to make the filters economical. The circuit described here is specifically intended for such an application. It takes advantage of the fact that, in a digital filter with fixed coefficients, multiplicands need not appear in the two's complement format. This sacrifice in generality, which is not a serious one in many applications, reduces the complexity of the multiplier and increases its speed. The improved performance exhibited by this circuit also results, in part, from a design in which products are automatically rounded to the same length as the incoming data. This feature reduces the multiplier's latency in comparison to double-precision serial multipliers [2], [4].

The high-speed multiplier is described below in terms of one bit-cell, a unit of circuitry associated with one bit of the coefficient word. A multiplier of arbitrary complexity (within limits set by chip size and yield) can then be implemented by cascading bit-cells to a length equal to the number of bits in the coefficient. As a sample realization of the circuit, four bits of the multiplier are combined with all necessary data and control registers in a single chip.

The multiplier described here was designed for use in digital filters whose two's complement (TC) data are organized serially (least significant bit first) and whose signed magnitude (SM) coefficients are available either in serial or parallel. An overview of the algorithm used to perform this TC/SM multiplication is given in Section II. The adaptation of the algorithm into block diagram form and the need for a modified, final bit-cell are described in Section III. Section IV contains the details of the circuit design and consideration of speed-limiting elements of the bit-cell. The multiplier was fabricated in a high-speed bipolar process, using a combination of emitter-function-logic [7], stacked current switches, and two-level metallization. The features of the process are summarized in Section V. The ultimate performance achieved and a comparison with other approaches to serial multiplication are given in Section VI.

II. TC/SM MULTIPLICATION ALGORITHM

This multiplier uses a modified shift-and-add algorithm described by Lyon [8]. The algorithm can be understood best by referring to [8], but in order to enhance the understanding of circuitry presented below, a brief explanation of the algorithm is given with the aid of Fig. 1.

The most significant two bits of the data word, shown in

Manuscript received December 27, 1977.
G. L. Baldwin is with Bell Laboratories, Holmdel, NJ 07733.
B. L. Morris is with Bell Laboratories, Allentown, PA 18105.
D. B. Fraser and A. R. Tretola are with Bell Laboratories, Murray Hill, NJ 07974.

0018-9200/78/0600-0400$00.75

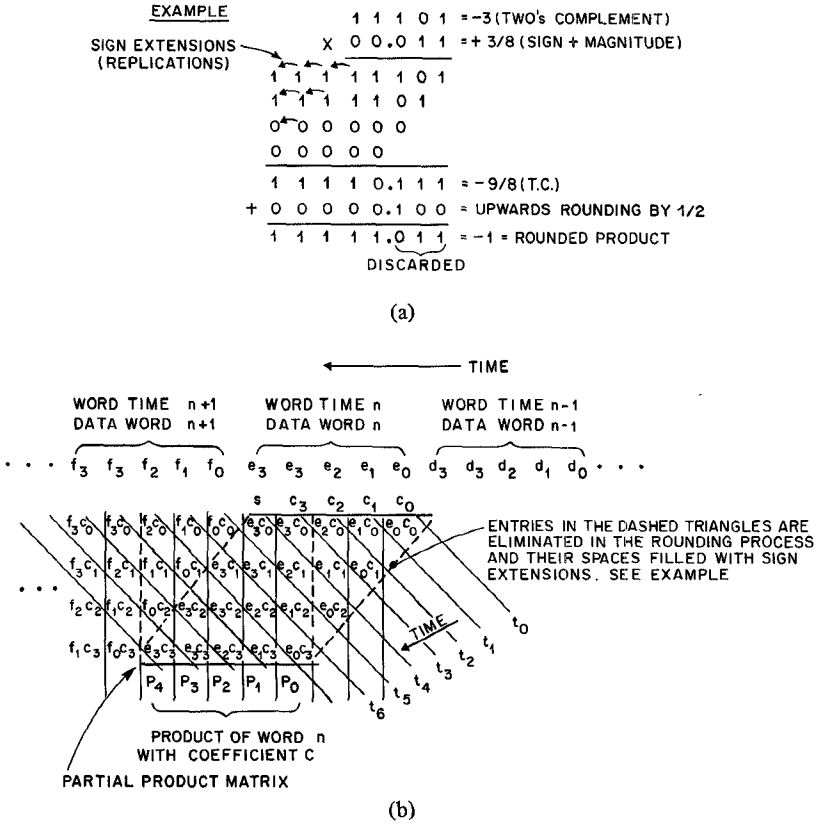


Fig. 1. (a) Example of two's complement/signed magnitude multiplication. (b) General form of multiplication, showing serial data and partial product formation as a function of time.

general form in Fig. 1(b), are the sign bit and the extended (replicated) sign bit. An in-range product always produces equal sign and extended sign bits. The values of $|C|$ need not be restricted to 2, as indicated in Fig. 1, but this restriction is common in many digital filter realizations and in other applications.

Multiplication using this algorithm proceeds in much the same way as multiplication by hand except for the following important difference. Sign extensions fill the high-order positions of partial products, as shown in Fig. 1(a), to prevent internal overflows during multiplication. The low-order bits, those below the binary point, are discarded after the addition of one-half to round off the product to the nearest integer.

The generalized multiplication shown in Fig. 1(b) indicates the times during which entries in the partial product matrix are formed. If each horizontal line in the matrix is associated with one cell of the multiplier, then during each time slot a given cell first forms a one-bit partial product (e.g., $e_i c_j$) and then performs the pairwise addition of its partial product, the partial sum from the preceding cell, and the carry retained from the preceding addition. In order to permit nonzero sign extensions that occur with negative TC data, the multiplier must discard the low-order partial product terms [dashed triangle in Fig. 1(b)] that might interfere with sign extensions from preceding multiplications. The multiplier performs this elimination automatically, as explained in the next section, while it retains the carries that might propagate from additions within the dashed triangle. This close packing of the partial products allows one $n \times m$ multiplication every n clock periods (for serial n-bit data) by starting a new multiplication before finishing the previous one.

A feature of the algorithm not evident from the example in Fig. 1 is the need for a two's complementation of the final product for negative coefficients. This requirement destroys the modularity of the multiplier, but as shown in Section IV, the bit-cell can be modified very easily to provide this function. The multiplier must also inhibit any sign extensions associated with the final bit-cell (associated with the most significant magnitude bit of the coefficient) since carries from the final partial sum are not required in the correct product [i.e., $f_0 c_3$ is not replaced by $e_3 c_3$ in Fig. 1(b)].

III. Multiplier Block Diagram

The bit-cell of the TC/SM multiplier follows from a straightforward adaptation of the algorithm presented in the previous section. The multiplier cell described by Lyon [8] has been modified to the form shown in Fig. 2 for this implementation. The full adder has been broken into sum (Σ) and carry ($\overline{C}$) circuits to emphasize the coefficient ($\overline{M}$) connection. An optional delay element inserted after the adder "pipelines" the multiplier by restricting the propagation of the sum to one bit-cell. This approach increases the operation speed since the formation of a new partial product need not wait for sums to propagate the full length of the multiplier.

The control signal R in Fig. 2 gates coefficients into their proper location, blocks the propagation of sums discarded

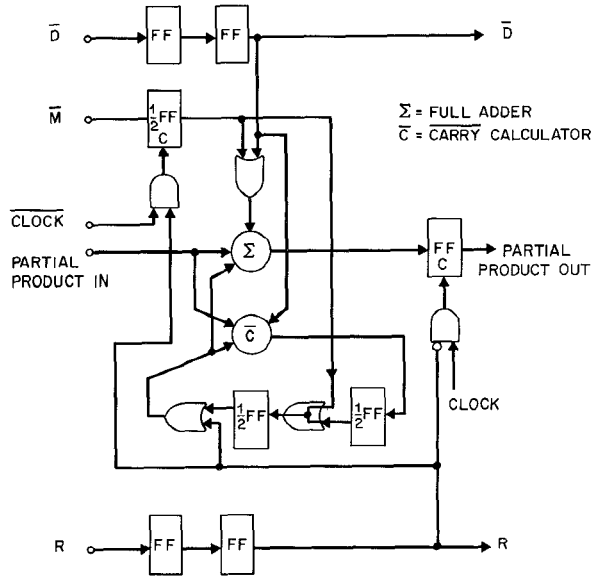


Fig. 2. Bit-cell with data, coefficient, and control registers.

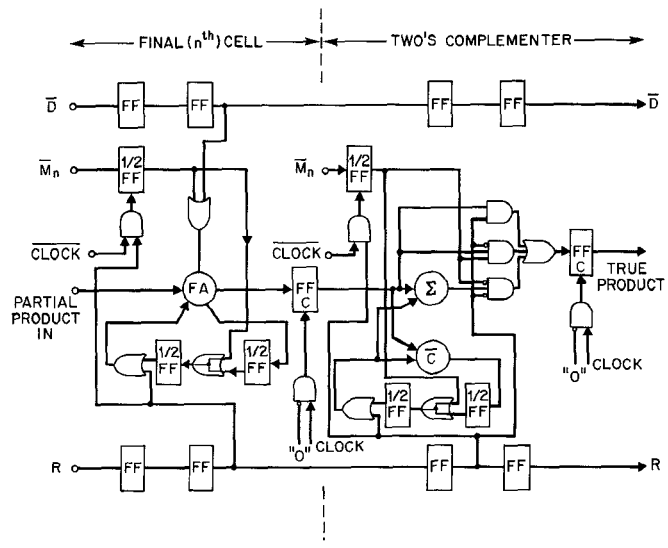


Fig. 3. Final cell and two's complementer formed from two normal bit-cells.

during the rounding operation, and effects a sign extension to inhibit possible internal overflows. An R pulse occurs once per data word and assumes a logical value of 1 only during the clock period synchronous with the least significant bit of the data. Such timing information is usually available from elsewhere in a digital filter or may be circulated through the control register by presetting all but one flip-flop to 0. The modification of the carry-calculate circuit to produce the complement of the carry ($\overline{C}$) allows a simple OR connection between $\overline{C}$ and R after the carry-save flip-flop to preset $\overline{C}$ to 1 at the beginning of a multiplication.

Two key observations simplify the multiplier circuit and increase its speed. First, it should be noted that if the coefficient loaded into a bit-cell is zero, no carry will be generated from that cell during the course of a multiplication. Hence, the complement of the coefficient bit, $\overline{M}$ may be simply WIRED-ORed with $\overline{C}$ in the carry-save delay element. This simplification reduces the gate count in the $\overline{C}$ circuit. A second observation is that no carries propagate into a bit-cell nor is the partial sum saved when the coefficient bit first enters a cell. Hence, the coefficient may be loaded into the coefficient storage element during the second half of the clock period corresponding to the least significant bit of the partial sum. The coefficient $\overline{M}$ enters the carry-save flip-flop between the master and the slave section, as shown in Fig. 2. The coefficient storage element can be realized with a latch (one-half flip-flop), thereby reducing the amount of circuitry and eliminating the propagation delay of a flip-flop from the critical timing path.

The need for two's complementation to obtain the proper product for negative coefficients poses a problem for the TC/SM multiplier. Since an n-bit multiplier would normally be formed by cascading n identical stages, a modular bit-cell cannot be used as the final stage of the multiplier. However, the bit-cell can be reconfigured slightly to make a suitable, if somewhat inefficient, two's complementer. The approach adopted in the integrated realization of this multiplier uses an alterable second level of metal interconnection to form a two's complementer from the same bit-cell topology used elsewhere in the circuit. Because two-level metal also increases the packing density and eliminates highly capacitive diffused crossunders, the nonmodularity of the TC/SM algorithm is offset by improved performance and a more efficient use of silicon area.

The two's complementer proposed by Lyon [8] cannot be used directly in this design because the $\overline{C}$ circuit computes the incorrect input for the adder. However, the bit-cell can be reconfigured to do two's complementation by employing the algorithm given in the Appendix. Fig. 3 shows the modifcations necessary to form a two's complementer from a normal bit-cell. Also shown in Fig. 3 is the omission of the sign-extension in the "final" (n-lst) cell of an n-bit multiplier. An important feature of the design presented in Section IV is the flexibility that allows these modifications with only minor changes in the second layer metallization.

IV. Multiplier Circuit Design

The increase in efficiency obtained by including all data, coefficient, and control registers on the multiplier chip comes at the expense of a large number of D-type flip-flops in each bit-cell. The compact D storage element proposed by Skokan [7] offers excellent speed/power performance for serial data processing in the multiplier. This flip-flop, shown schematically in Fig. 4, not only serves as a basic building block of the multiplier, but also motivates the use of emitter-function-logic (EFL) elsewhere in the circuit. As demonstrated later in this section, an efficient high-speed multiplier results from combining the multiemitter inputs to the flip-flop with stacked current-steered logic in the collectors of the current switches—Q_9-Q_{12} in Fig. 4.

The carry-save flip-flop differs from the basic storage element (Fig. 4) only in the use of a WIRED-OR connection of $\overline{M}$ between the master and slave, as shown in Fig. 2. The latch used to hold the coefficient $\overline{M}$ comprises one-half of a normal

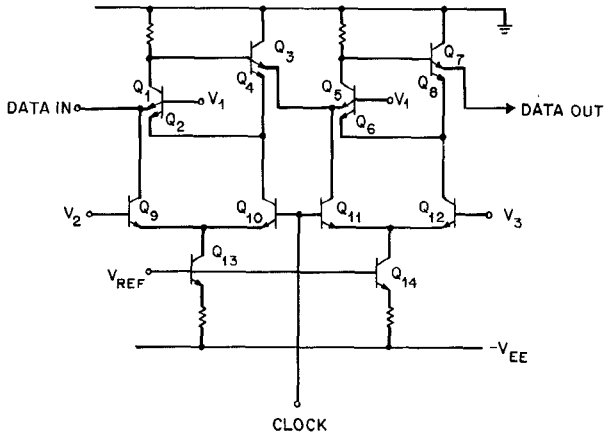


Fig. 4. *D*-flip-flop (after Skokan [7]).

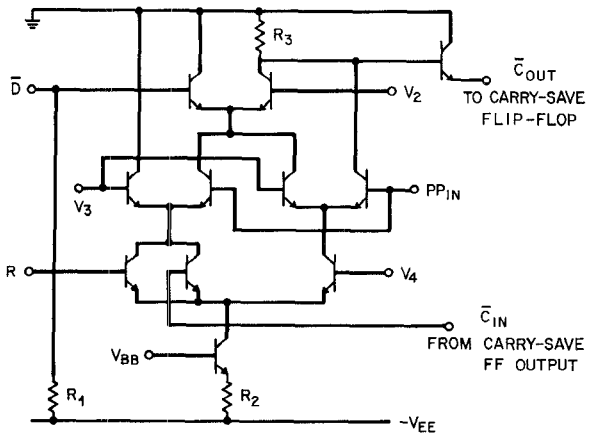


Fig. 6. Carry-calculate circuit (actually computes $\overline{C}$).

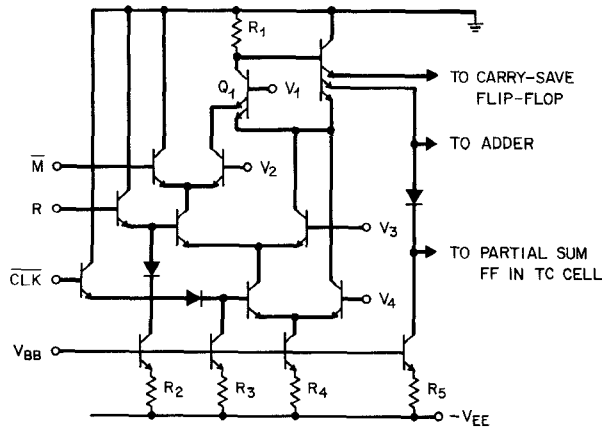


Fig. 5. Coefficient latch with data selector.

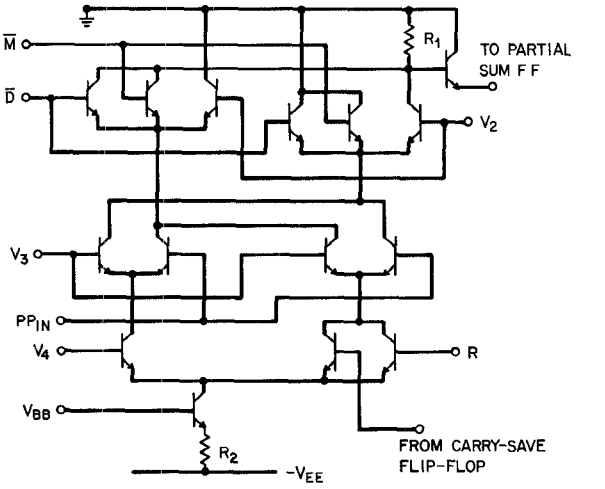


Fig. 7. Full adder.

flip-flop, as shown in Fig. 5. Inserting two levels of logic between Q_1 and the clocked current switch combines the data selectors and coefficient latch into a single element that uses only one current source. Because the coefficient is not used until the clock period after it is loaded, the timing of the latch is not critical.

The carry-calculate and adder circuits shown in Figs. 6 and 7, respectively, both use stacked logic trees and single current sources to form the exclusive-OR function basic to their operation. The ORed connection between $\overline{C}$ and the control signal R in both circuits effectively sets the carry to zero at the beginning of each multiplication. When used in the TC cell, the $\overline{D}$ data input to the carry-calculate is open-circuited and the $\overline{D}$ and $\overline{M}$ inputs to the adder are connected to ground.

The final principal element of the bit-cell, the product-save flip-flop shown in Fig. 8, uses one level of logic between the input emitter (Q_1) and the switched current source to effect a sign extension. When used in a normal bit-cell, the second emitter E_2 of Q_1 is returned to ground, and the collectors of Q_2 and Q_3 are connected to the input latch as shown. When used in the two's complementer, E_2 couples the partial product input PP_{in} directly into the flip-flop under control of R and $\overline{M}$. This secondary mode of operation requires the collectors of Q_2 and Q_3 to be connected to E_2 instead of E_3.

The critical delay path in the bit-cell is dominated by the adder delay. This effect results from high parasitic capacitances at nodes shared by two or more collectors and/or emitters. The adder was designed to achieve efficient use of the bias current, but this design also uses a large number of these shared nodes [9]. A detailed analysis of the bit-cell, made with the computer program SPICE [10], indicates that these parasitic capacitances dominate the power-delay tradeoffs in the circuit. In the analysis the delay of the adder followed a near-ideal inverse relationship with the bias current. The following arrangement equalized the adder's contribution of the circuit's delay: adder bias = 1.0 mA; all other principal bias currents (carry-calculate, latch, flip-flops) = 0.33 mA. Adding the level shift and pull down currents to the above results in a total cell current of 3.9 mA. With $V_{EE} = -5.2$, the total cell power (excluding $\overline{D}$ and R registers) is 20 mW.

The complete bit-cell, excluding the $\overline{D}$ and R register storage elements, is shown in Fig. 9. Switches in the schematic merely indicate alterations in the second-level metal necessary only for the two's complementer. In the n-1st bit-cell of an n-bit

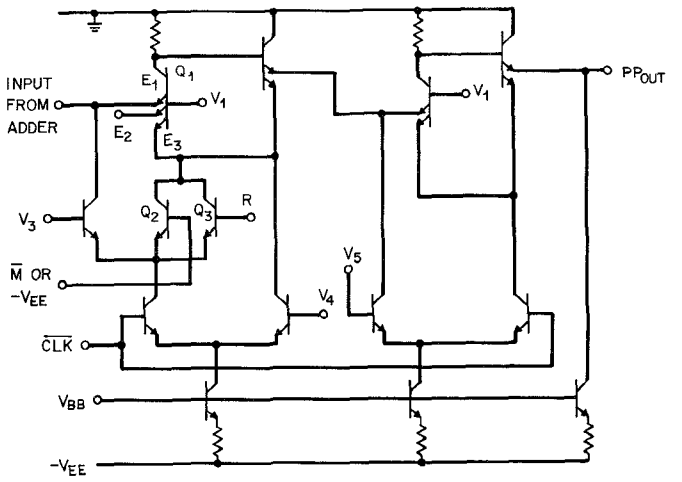


Fig. 8. Partial sum storage element. Base of Q_2 returned to $-V_{EE}$ except when used as two's complementer.

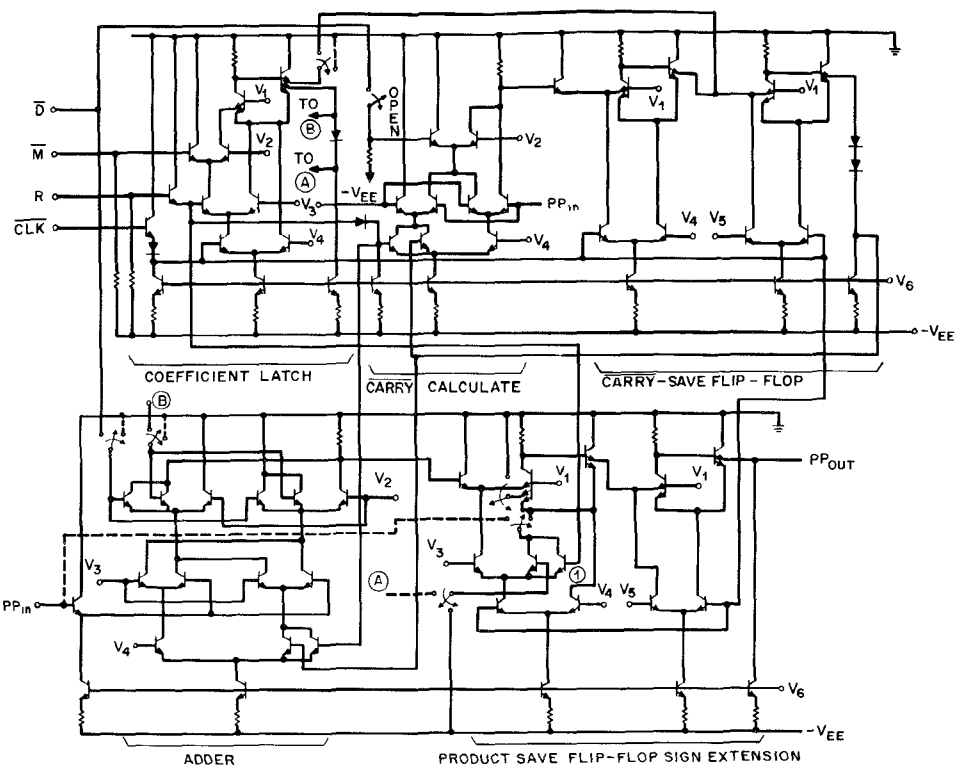


Fig. 9. Complete bit-cell. All solid line connections pertain to normal bit-cells. All dashed line connections pertain to two's complement cells. Switches indicate alternate second layer metal patterns. In the final cell 1 is tied to $-V_{EE}$.

multiplier, the node labeled 1 in the product-save flip-flop is tied to $-V_{EE}$. The threshold voltages V_1-V_4 are derived from an on-chip negative one-half V_{BE} power supply, referenced to ground, and obey the relationship given in (1).

$$V_n = -(n - \tfrac{1}{2})\, V_{BE}, \quad n = 1, 2, 3, 4. \tag{1}$$

V_5 is 0.1 V lower than V_4 to insure proper data transfer in the flip-flops.

Two basic building blocks emerge from this multiplier configuration. Construction of a multiplier begins with the normal bit-cell, four of which are grouped together as an example in a normal configuration outlined in Fig. 10(a). Multipliers of arbitrary complexity can be constructed by cascading these units of four cells. Or, within limits imposed by chip size, multipliers dedicated to coefficients with a fixed number of bits can be realized on a single chip by making repeated use

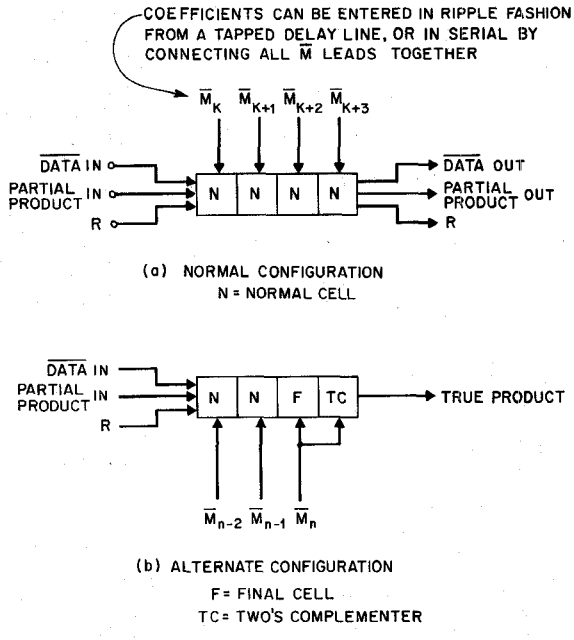


Fig. 10. The two modes of operation of the four-bit TC/SM multiplier.

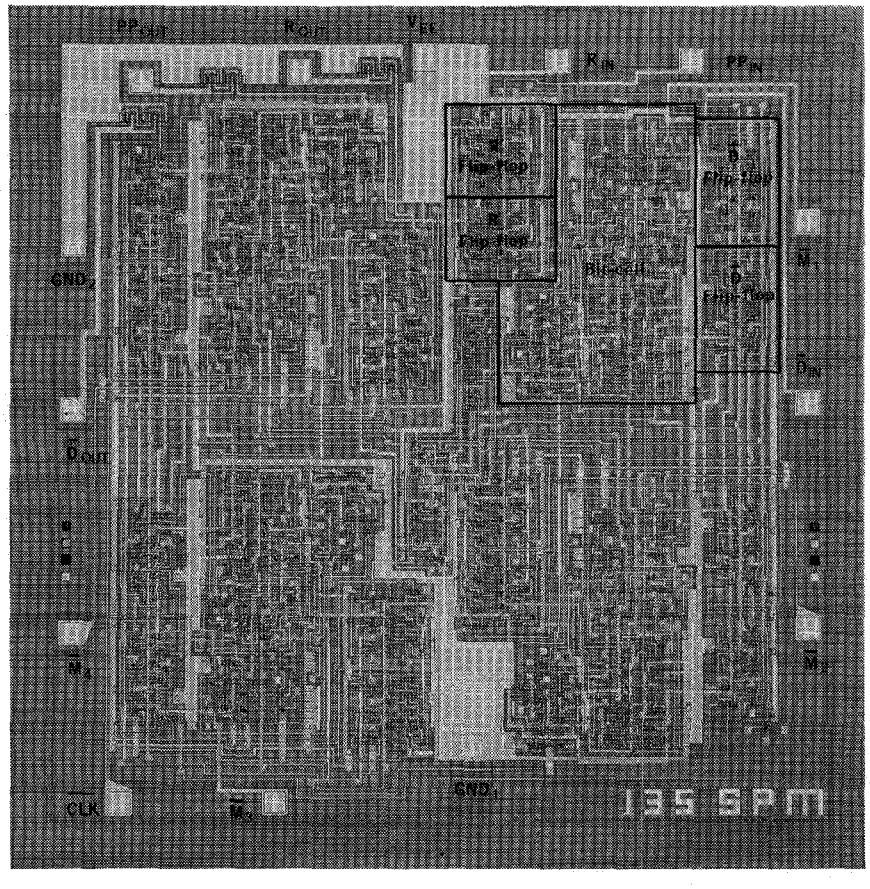


Fig. 11. Photomicrograph of four-bit TC/SM multiplier. Chip size is 120 × 120 mil^2.

of the basic normal cell. The last two cells of a multiplier require the alternate configuration shown in Fig. 10(b). The building blocks shown in Fig. 3 form the final cell (F) and the two's complementer (TC) of Fig. 10(b). When realized with bit-cells grouped in this way, multipliers with coefficients up to $4N$ bits in length require one alternate configuration and N-1 normal configurations.

The photomicrograph in Fig. 11 shows a complete four-bit

version of the TC/SM multiplier. The L-shaped bit-cells are each surrounded by two bits of data register and two bits of control register. The on-chip negative $\frac{1}{2} V_{BE}$ reference, consisting of about 15 transistors and about 15 resistors, occupies an L-shaped portion of the center of the chip. Area consumption within each quadrant of the multiplier is not dominated by metal interconnect owing to the liberal use of upper level metal (lightly shaded in the photo). Open spacing in the wiring channels between quadrants of the chip results from a simple replication of the bit-cell/shift register combination. This approach led, of course, to a somewhat larger chip than could have been achieved with a hand-packed layout. Nonetheless, all four bits of multiplier and 16 bits of shift register are contained on a 120 mil^2 chip that requires only 14 pins to make off-chip connections. The four-bit TC/SM multiplier performs the functional equivalent of about 250 gates, contains about 650 emitters, and uses more than 200 via holes between metal interconnect patterns.

Expansion of the TC/SM multiplier, using this same bit-cell/shift register combination, to larger scales of integration is quite consistent with state-of-the-art technology. It can be estimated that a $12 \times N$ TC/SM multiplier would occupy a chip with about 173 mil^2 active area, would consume about 450 mW, and would require a 20-pin package (for parallel coefficient entry; 14 pins for all-serial).

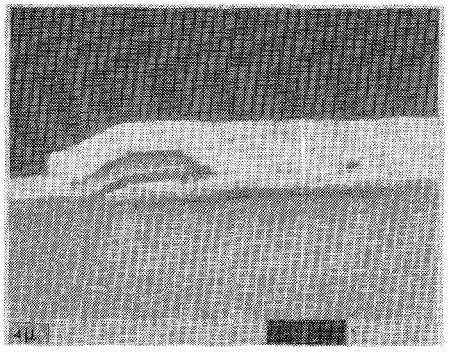
V. Fabrication Technology

One aspect of this multiplier design concerns a choice of technology capable of offering the maximum performance for a given level of power. Computer simulations showed that a data rate of 60 Mbits/s could be achieved with a cell power of 22 mW using a thin-basewidth (0.3 μm nominal), thin-epi (4 μm nominal) standard buried-collector process. However, the size and complexity of the circuit require a very low density of defects, especially those resulting from emitter-collector short or "pipes." Since the severity of emitter-collector shorts increases with decreasing basewidth [11], increasing the basewidth to a nominal 0.6 μm and the epi thickness to a nominal 5 μm resulted in a significant yield improvement. Although these modifications caused a speed reduction of about 25 percent, 30 percent of the multipliers functioned correctly at wafer-probe test on a typical wafer.

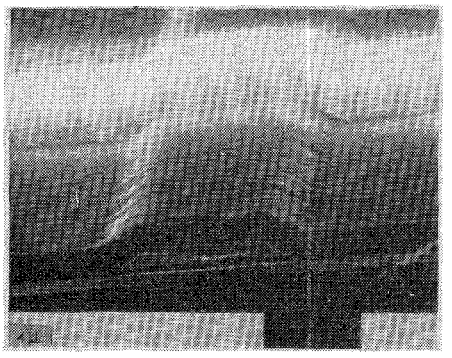
Minimum-geometry devices in the multiplier employed a 14 μm line pitch and a 4 μm minimum feature size (re-registered emitter windows). The emitter windows were taper-etched to provide smooth continuity of the contacting metal. In addition, a new two-level metallization process, the details of which are shown in Fig. 12, was used [12]. Both photographs in Fig. 12 show a smooth, continuous second-level metal profile that contributed significantly to the high yields and performance of the multiplier.

VI. TC/SM Multiplier Performance

The dynamic performance of the four-bit TC/SM multiplier was ascertained by comparing its serial output with that of a high-speed emulator made with commercial emitter-coupled logic. All possible combinations of a four-bit coefficient word



(a)



(b)

Fig. 12. (a) Second-to-first level metal connection through dielectric "via hole" showing tapered edge. (b) Conformal dielectric coverage over tapered first level metal and continuous coating in second level crossover. Black bars are 4 μm long.

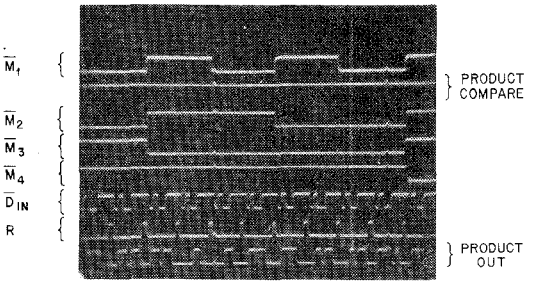


Fig. 13. Logic analyzer pulse pattern showing serial input data, coefficients $\bar{M}_1$-$\bar{M}_4$, and product out. Small blips are produced by the logic analyzer.

and an eight-bit data word were applied to the multiplier and its emulator. The two outputs were then subtracted with an exclusive-OR circuit to observe differences in the serial products. The photograph in Fig. 13 shows one sequence of the test pulses and the multiplier output as depicted on a high-speed logic analyzer. The integrated multiplier was found to operate up to 44 Mbits/s at room temperature using this test method. This result agreed very well with predictions from computer simulations at a power level of 20 mW/bit-cell and 4.2 mW/flip-flop. The overall circuit performance fit well within the comparison parameters for multipliers proposed by Baugh and Wooley [13], ($\beta \simeq 4$, $\sigma \simeq 4$).

To compare this performance with that achieved using other

TABLE I
COMPARISON OF SEVERAL APPROACHES TO SERIAL MULTIPLICATION OF 24-BIT DATA BY A 12-BIT COEFFICIENT

Multiplier	Size	Number Packages Required N_p	Data Rate f (Mbit/s)	Power per Package P (mW)	Scaler K[b]	Figure of Merit F
TC/SM	$4 \times N$	3	44	140	1	4.4
TC/TC [3]	$4 \times N$	3	20	150[a]	1	1.8[a]
CMOS/SOS [4]	$24 \times N$	0.5	10 (@ 5 V)	20	1.5	28
	$24 \times N$	0.5	18 (@ 15 V)	420	1.5	2.4
TTL [2]	$8 \times N$	1.5	25	455	1.5	1.02
NMOS [14]	$12 \times N$	1	7	245	1	1.2

[a]Does not include data, coefficient, or R registers.
[b]K = output data length relative to 24-bit data.

$$F = \frac{f}{24} \cdot \frac{1}{PN_p} \cdot \frac{1}{K} \cdot 10^{-6} = \text{million multiplications per second per watt.}$$

approaches, a figure of merit has been chosen that focuses on a typical application. The figure of merit indicates the speed/power ratio for serial multiplication using 24-bit data words and 12-bit coefficients. These word lengths are representative of those used in the digital processing of voice signals. Table I summarizes the results. It can be seen that the TC/SM multiplier exhibits superior performance in the high-speed range (greater than 20 Mbits/s). Only the CMOS/SOS approach offers a better overall figure of merit, but at less than one-quarter the speed available from the TC/SM multiplier.

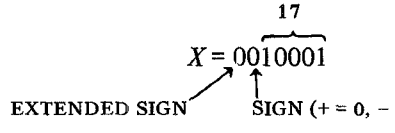
As mentioned in the Introduction, parallel-array multipliers can also be used in digital filter applications. However, comparisons between the parallel and serial approaches are difficult. Parallel multipliers are not intended for serial data, and the additional serial/parallel conversion hardware must be operated at extremely high rates. Serial multiplication using parallel multipliers has therefore been properly omitted from the comparison in Table I. Nevertheless, for an all-parallel digital filter organization it is worth noting a figure of merit similar to that obtained above for a 12 X 24 multiplication. Using two 12 X 12 multipliers to produce a 24-bit partial product every 150 ns [15] yields a figure of merit of about 0.4. The difference between this and the performance of the TC/SM multiplier suggests that, even with the additional serial/parallel conversion circuitry necessary to interface to a parallel data format, the TC/SM multiplier warrants consideration as an attractive alternative.

VII. CONCLUSION

An approach to the realization of high-speed, serial, pipelined digital mulitpliers has been described. The improvements in performance afforded by this approach result from the combination of two key ingredients: 1) a high-speed circuit technology with an excellent power-delay product; and 2) the manipulation of the TC/SM multiplication algorithm to reduce speed constraints on elements within the circuit. This multiplier will add significantly to the ease of implementation, lower chip count, and improved performance of high-speed digital filters. Further improvements should result from a thinner epitaxial bipolar process with fewer defects and from an improved interface to highly capacitive input connections.

APPENDIX
TWO'S COMPLEMENT ALGORITHM WITH MODIFIED BIT-CELL

Consider the example of the two's complement of a number X, TC(X). Let $X = 17$.



TC(X) = $2^{n+2} - X$, for an n-bit number with extended sign. Hence, TC(17) = (10000000) - (0010001) = - 17.

The same subtraction can be performed by adding to X a number Y generated by the following rule.

1) Preset the least significant bit (LSB) of Y to zero and add this to the LSB of X, i.e., pass LSB of X unaltered.

2) When the first nonzero bit of X is encountered, proceeding from LSB to MSB, set the *next* and all succeeding bits of Y to 1.

3) Continue to *ADD* Y to X, *discarding* the normal carry bit that would be generated in each column, until the addition of the two numbers is complete.

Therefore,

$$\begin{array}{ll} 0010001 & (X) \\ \underline{+1111110} & (Y) \\ 1101111 = \text{TC}(X). & \end{array}$$

The rule is generally applicable to any number X. This rule is particularly useful for the bit-cell shown in Fig. 9. There are two possible modes of operation.

1) If $\overline{M} = 1$ (positive coefficient), always pass the incoming partial product unaltered. A careful examination of the input to the triple-emitter transistor in the product-save flip-flop shows that this rule is followed.

2) If $\overline{M} = 0$ (negative coefficient), set $\overline{D} = 0$ in the carry-calculate circuit. At the beginning of a word $R = 1$, forcing $\overline{C} =$

1 ($C = 0$). $\overline{C}$ then remains 1 as long as $PP_{in} = 0$, even after R returns to 0. When the first $PP_{in} = 1$ bit is encountered, C is set to zero at the next clock time. With $\overline{M}$ and $\overline{D}$ connected to ground in the adder circuit, the complement of PP_{in} is generated at the adder output. $\overline{C}$ remains equal to zero by the nature of the feedback from the carry-save flip-flop to the carry circuit. The effect of this is to discard carry bits that would normally be generated as each pair of numbers in a column is added. Since $\overline{M} = 0$ the output from the adder gets loaded into the product-save flip-flop, and the correct TC number is passed to the output of the multiplier.

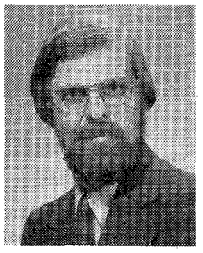
This algorithm is implemented with the interconnect pattern indicated by the dashed lines in Fig. 9.

Acknowledgment

The authors gratefully acknowlege the contributions of J. H. Condon to this work. Without his insights into both the arithmetic and circuit design, the results achieved would not have been possible. The authors are also appreciative of numberous discussions with S. L. Freeny, J. W. Grace, R. B. Kieburtz, and S. K. Tewksbury.

References

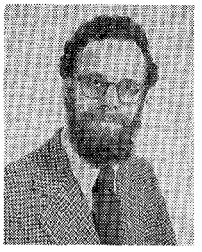
[1] G. W. McIver, R. W. Miller, and T. G. O'Shaughnessy, "A Monolithic 16 × 16 multiplier," in *1974 ISSCC Dig. Tech. Papers*, vol. 17, pp. 54–55.
[2] J. R. Mick and J. Springer, "Single-chip multiplier expands digital role in signal processing," *Electronics*, vol. 49, pp. 103–108, May 13, 1976.
[3] J. Kane, "A low-power, bipolar, two's complement serial pipeline multiplier chip," *IEEE J. Solid-State Circuits*, vol. SC-11, pp. 669–678, Oct. 1976.
[4] D. Hampel, K. E. McGuire and K. J. Prost, "CMOS/SOS serial-parallel multiplier," *IEEE J. Solid-State Circuits*, vol. SC-10, pp. 307–313, Oct. 1975.
[5] S. L. Freeny, "Special-purpose hardware for digital filtering," *Proc. IEEE*, vol. 63, pp. 633–648, Apr. 1975.
[6] L. B. Jackson, J. F. Kaiser, and H. S. McDonald, "An approach to the implementation of digital filters," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 413–421, Sept. 1968.
[7] Z. E. Skokan, "Emitter function logic-logic family for LSI," *IEEE J. Solid-State Circuits*, vol. SC-8, pp. 356–362, Oct. 1973.
[8] R. F. Lyon, "Two's complement pipeline multipliers," *IEEE Trans. Commun.*, vol. COM-12, pp. 418–425, Apr. 1976.
[9] C. R. Baugh and B. A. Wooley, "A high-speed, low-power bipolar one-bit full adder," to be published.
[10] L. W. Nagel and D. O. Pederson, "Simulation program with Integrated circuit emphasis," in *Proc. 16th Mid-West Symp. on Circuit Theory*, Waterloo, Canada, Apr. 12, 1973.
[11] S. Dash, "Gettering and dislocations," in *Semiconductor Silicon 1973*, Int. Symp. on Silicon Materials, Science, Technol., 1973, Chicago. Princeton, NJ: Electrochem. Soc. 1973.
[12] D. B. Fraser and A. J. Tretola, "A two-level metallization process," to be published.
[13] C. R. Baugh and B. A. Wooley, "Digital multiplier power estimates," *IEEE Trans. Commun.*, vol. COM-23, pp. 1287–1288, Nov. 1975.
[14] J. R. Verjans, "A serial-parallel mutliplier using the NENDEP Technology," *IEEE J. Solid-State Circuits*, vol. SC-12, pp. 323–325, June 1977.
[15] "LSI multipliers," TRW, Redondo Beach, CA, TRW LSI Products Data Sheets, Mar. 1977.



Gary L. Baldwin (S'66–M'70–SM'75) was born in El Centro, CA, on October 12, 1943. He received the B.S., M.S., and Ph.D. degrees in electrical engineering in 1966, 1967, and 1970, respectively, all from the University of California, Berkeley.

He was employed by the IBM Corporation, San Jose, CA, on a work-study program from 1963 to 1966. In the summer of 1968 he was involved in linear integrated-circuit design with the Advanced Circuits Research Group, Signetics Corporation, Sunnyvale, CA. He was an Acting Assistant Professor of Electrical Engineering at the University of California, Berkeley, during 1969 and 1970. He has been a member of the Technical Staff in the Signal Processing Research Department, Bell Laboratories, Holmdel, NJ, since 1970.

Dr. Baldwin is a member of Eta Kappa Nu and Sigma Xi.

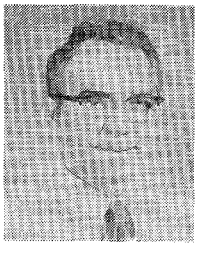


Bernard L. Morris was born in Brooklyn, NY, in 1942. He received the Sc.B. degree in physics from the Massachusetts Institute of Technology, Cambridge, in 1964, and the Ph.D. degree in physics from Brown University, Providence, RI, in 1969.

He worked at the NASA Electronics Research Center, Cambridge, MA from 1969 to 1970. In 1970 he joined Bell Laboratories, Allentown, PA. His work has included characterization of thin epitaxial layers, effects of material defects on junction yield, and diffusion processes. His present work involves bipolar process development, with emphasis on high-speed structures for digital systems. He is the author of ten technical publications.

David B. Fraser was born in Winnipeg, Man., Canada, on August 8, 1932. He received the B.Sc. (Hon) from the University of Manitoba in 1954 and the M.A. and Ph.D. degrees in solid state and low temperature physics from the University of Toronto, Toronto, Ont., Canada, in 1955 and 1958, respectively.

He was on the Physics Faculty of the University of Waterloo during 1958–1959 and joined Bell Laboratories, Murray Hill, NJ, in 1959. Early work at Bell Laboratories centered on physical acoustics, e.g., studies of anelastic properties of crystalline quartz, the Jahn–Teller effect in solids, and acoustic properties of vitreous silica. He is currently working on problems of metallization, both depositon and etching, as well as other aspects of silicon integrated circuit processing.



Angelo R. Tretola (S'67–M'68) was born in Newark, NJ, on December 16, 1930. He received the B.S.E.E. degree from Newark College of Engineering, Newark, NJ, in 1968.

After serving in the U.S. Navy, he was employed by Western Electric from 1953 to 1956, where he worked on custom exchange equipment. He joined Bell Laboratories, Murray Hill, NJ, in 1956 and began working in a diode development group. He has authored and coauthored several articles on Si and GaAs defect studies and IMPATT diodes. His present work is mainly concerned with Si IC process development.