An MOS Integrated Circuit for Digital Filtering and Level Detection

PETER F. ADAMS, JOHN R. HARBRIDGE, AND RODERICK H. MACMILLAN

Abstract—**An LSI circuit for digital signal processing has been designed and manufactured in 5 V n-channel MOS technology. Its main functions are to implement digital filters of the cascaded biquadratic form and to perform level detection operations. The frequency response of the filter is controlled by coefficients supplied from an external memory. The device, known by the acronym FAD (filter and detect), operates from a single-phase clock and can process up to 64 000 samples/s at the maximum permissible clock rate of 2048 kbits/s. Although FAD was designed for one particular requirement, it has sufficient flexibility for use in a variety of applications.**

I. Introduction

TELECOMMUNICATION networks make extensive use of tones to convey information, both for in-band signaling between exchanges and for customer signaling to local exchanges [1], [2]. The imminent introduction [3] of digital telephone exchanges has created a need for a range of digital signaling receivers to detect and decode these tone signals. In the course of developing receivers for various signaling systems [4], it became apparent that there were common functions in the receivers which could usefully be incorporated in a custom designed LSI circuit, bringing about size, power, and cost reductions.

The IC, known by the acronym FAD (filter and detect), has been designed and fabricated by British Telecom Research Laboratories. It contains about 8000 transistors on a 5 mm × 5 mm substrate mounted in a 24 pin dual-in-line package. Power consumption is about 500 mW drawn from a single 5 V power supply. The circuit requires a single-phase clock and very little external circuitry. Its main function is to implement digital filters of the cascaded biquadratic form [5]. In addition, it performs some level detection operations. The filter frequency response is controlled by coefficients supplied from an external memory. Various operational modes are controlled in the same way and by interconnecting pins.

Although developed for a particular application, the IC provides a digital filtering capability which may well find use elsewhere. Compared with other digital signal processing IC's described in the literature [6]-[10], FAD is a relatively conservative device, both in terms of chip area and clock rate. Also other devices are designed to be far more flexible—FAD implements filters based only on second-order section decompositions. However, the price of flexibility is complication: more flexible devices are difficult to use without comprehensive software design aids. FAD, therefore, provides a relatively simple-to-use means of implementing digital filters. In terms of the number of poles and zeros implemented for a given sample rate, FAD is not as efficient as devices [6], [7] which exploit canonical signed digit coding for the coefficients but, in common with other devices using multipliers, dynamically changing the coefficients for adaptive filter applications is possible.

Manuscript received October 20, 1980; revised January 5, 1981.

The authors are with British Telecom Research Laboratories, Martlesham Heath, Suffolk, England.

II. General Description

Digital filtering is the processing of digitally encoded samples of an analog waveform in such a way that, when decoded, the resulting analog waveform is a filtered version of the original waveform. This processing, in its simplest form, consists of the addition of sample amplitudes, the multiplication of sample amplitudes by constants, and the delaying of samples by an integral number of sampling periods. These three elemental processes can be combined in an infinite variety of structures. Although it is possible to design a processor with a flexible structure it would not be very easy to use and might even be inefficient. The philosophy adopted for the FAD IC was to identify the most convenient structure for easy use which, nevertheless, would give sufficient flexibility for a variety of applications. Two functions were identified as most suitable for application to digital signaling receivers: the second-order recursive filter function and the level detection function.

A. Second-Order Recursive Filter Section

The second-order filter section has long been recognized as a convenient building block for higher order digital filters; it gives good coefficient sensitivity and stability properties and its behavior has been extensively investigated for roundoff noise, limit cycle effects, and overflow properties. High-order filters are readily constructed by cascading such sections. There are numerous structures even for the second-order section and the one chosen for FAD is the canonic structure shown in Fig. 1, which uses a minimum number of delays, multipliers, and adders.

The multiplier coefficients A, B, a, and b determine the response of the filter, the latter two defining a pair of complex poles and the former a pair of complex zeros. The relationship of the coefficients to amplitude response is described in [11].

Digital filters exhibit gains of greater than unity for many input signals and, if they are not to overflow their number range, inputs must be scaled by a suitable value [12]; the input scaler—a multiplier limited to multiplication by powers of $\frac{1}{2}$—performs this function.

0018-9200/81/0600-0183$00.75

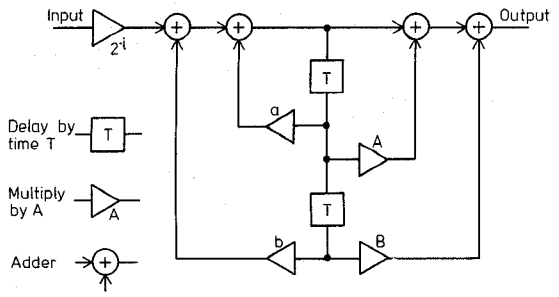

Fig. 1. Basic second-order filter section.

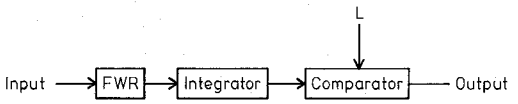

Fig. 2. Basic level detector.

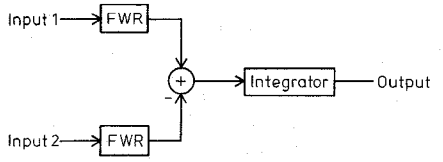

Fig. 3. Basic relative level detection.

The process of multiplication results in the wordlength of sample data increasing; this cannot be permitted in the feedback loops containing a and b because the wordlength would increase ad infinitum. Therefore, the feedback loop has to contain one or more quantizers which restrict the wordlength to the maximum permitted.

Finally, although the input should be scaled to prevent it, overflow can still occur in a practical filter due to abnormal initial values in the delay elements on switch-on, or to noise-induced errors. It is known that overflow can give rise to severe nonlinear oscillations [13]; therefore, an overflow detection and correction circuit is included in the FAD IC to stabilize the filter if overflows occurs.

B. The Detect Function

To detect tones it is necessary to estimate the level of a signal after it has been filtered; this process is called the level detection function. Essentially, it consists of digital full-wave rectification (FWR) followed by integration and comparison with a fixed level, as shown in Fig. 2. In some receivers [4] it is necessary to compare the signals from two filters; this can be done by subtracting the signals, after FWR, and integrating the result (Fig. 3). In general, both functions are required simultaneously, as shown in Fig. 4, where the order of comparison and integration has been reversed for implementation reasons and a multiplier B introduced to enable the proportion of one signal to be adjusted in the relative level detection.

C. Facilities and Options

Based on the two functions described above the FAD IC, which is shown diagramatically in Fig. 5, offers the following facilities and options:

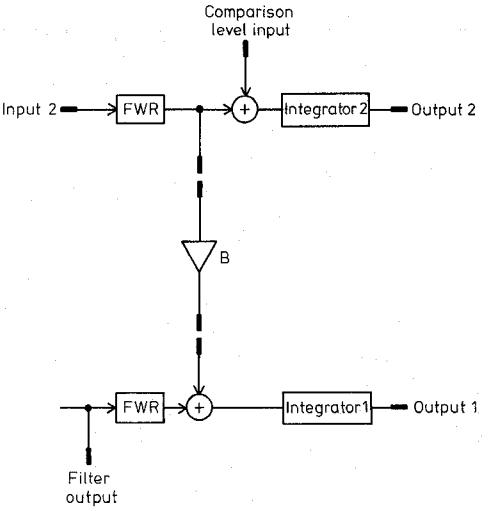

Fig. 4. Relative level detection configuration.

1) A second-order recursive filter section which can be used to realize minimum phase or nonminimum phase networks. Alternatively, if the zeros are confined to the unit circle of the z-plane, one multiplier is released for independent use, for example, in level comparison (Fig. 4).

2) Two full-wave rectification functions, two leaky integrators with programmable leak factors, and two comparators with programmable comparison levels for realizing both level and relative level detection.

3) A maximum sample throughput of 64 000 samples/s and facilities for channel or filter multiplexing with a reduced sample rate; delays for eighth-order multiplex are provided on chip; data and coefficient accuracies are 16 and 13 bits (including sign), respectively.

III. Circuit Description

The design of the filter and detect functions has been strongly influenced by the requirements for efficient integration on to a silicon chip. Serial data transfer was used to minimize the number of pin connections and internal interconnections. A modular logic approach was used in order to simplify the MOS layout design and each of the arithmetic elements shown in Fig. 5 was provided on a dedicated basis to avoid complicated data routing and testing difficulties.

A. Data Wordlength and Coding

A 16 bit fixed-point [14] data wordlength (including sign) gives sufficient accuracy for signaling receiver applications. It is convenient to think of data as being less than unity and this convention is assumed throughout. Arithmetic processing is greatly facilitated by using two's complement coding [14].

B. Arithmetic Elements of the Filter

To achieve high-speed operation combined with serial data transfer a serial-parallel multiplier scheme was chosen. Each multiplier uses a 12 bit two's complement representation of the fractional part of its coefficient. In order to implement poles and zeros anywhere inside or on the unit circle in the z-

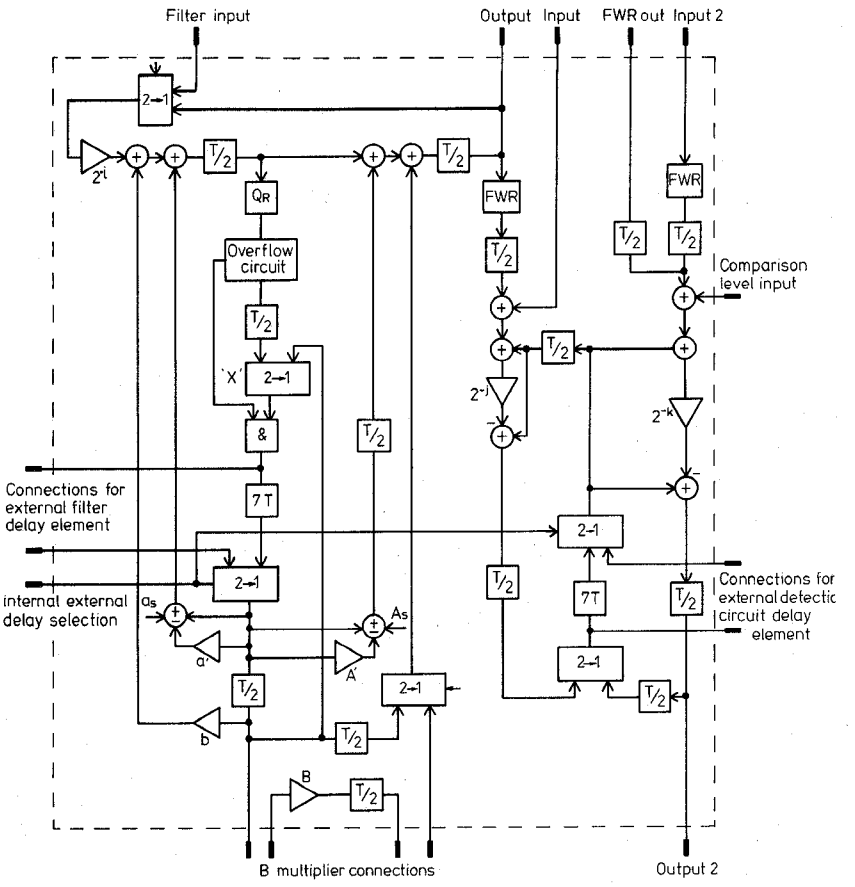

Fig. 5. Complete block diagram of the FAD circuit.

plane [5], A, a, B, and b are bounded by

$$2 \geqslant A \geqslant -2, \quad 2 \geqslant a \geqslant -2, \quad 1 \geqslant B \geqslant -1, \quad 1 \geqslant b \geqslant -1.$$

Thus, an extra nonfractional bit is required to represent the A and a coefficients. It is convenient to constrain the multipliers to handle fractional coefficients only; this can be effected for the A multiplier by separating the coefficients into the form $(-1)^{A_S} + A'$ where A' is a fractional number. The bit A_S is then used to control an adder/subtractor following the multiplier. Similarly, the a coefficient is separated into a_s and a'.

The chosen multiplier (Section IV-E) requires at least $(N + M - 1)$ clock pulses to complete a multiplication where N is the data wordlength and M the coefficient wordlength (both including sign), i.e., 28 clock pulses for the chosen values of M and N. To obtain a convenient clock rate and to have some clock pulses for other purposes a device computation cycle of 32 bits was chosen.

The input scaler performs multiplication by 2^{-i} by delaying the input data by $(13 - i)$ bits where $13 \geqslant i \geqslant 0$. The value of i is controlled by 4 bits, encoded in binary form, i.e., $i = 0$ is 0000, $i = 9$ is 1001, etc.

C. Channel and Filter Multiplexing

The second-order filter shown in Fig. 1 is very easily multiplexed by increasing the delay function in steps of T (where T is the sampling period) and time-sharing the arithmetic elements. The limit on this process is the maximum clock rate of the FAD IC. With a 32 bit computation cycle the clock rate f_{cl} is given by

$$f_{cl} = 32 \times f_s \times Y$$

where Y is the number of times the filter is multiplexed and f_s is the sampling rate $(=1/T)$. For $f_s = 8000$ samples/s (the sampling rate used in the signaling receivers) and a maximum permissible clock rate of 2048 kbits/s, Y is constrained to be less than or equal to 8.

By presenting an input sample during every 32 bit computation cycle 8 separate second-order filters can be implemented. As the inputs can be independent of each other the filter is then said to be channel multiplexed.

In order to build higher order filters, the output of the second-order filter may be fed back to the input via a data selector since there are 32 bits of delay between input and output. Thus, by controlling the data selector the second-order filter sections can be cascaded. This arrangement can give any combination of filter and channel multiplexing.

D. Delay Implementation

Each 32 bit computation cycle contains a 16 bit data word and a block of 16 bits generated by the processing elements but not used for subsequent calculations. The latter bits can, therefore, be discarded to economize on delay storage. It is arranged that the inputs to the two delay elements (Fig. 1) have their 16 bit blocks of valid data out of phase; by using the 2-to-1 line selector (marked X, Fig. 5) the two data streams are interleaved and fed through one $(Y - 1)\,T$ delay. If $Y = 1$, then the

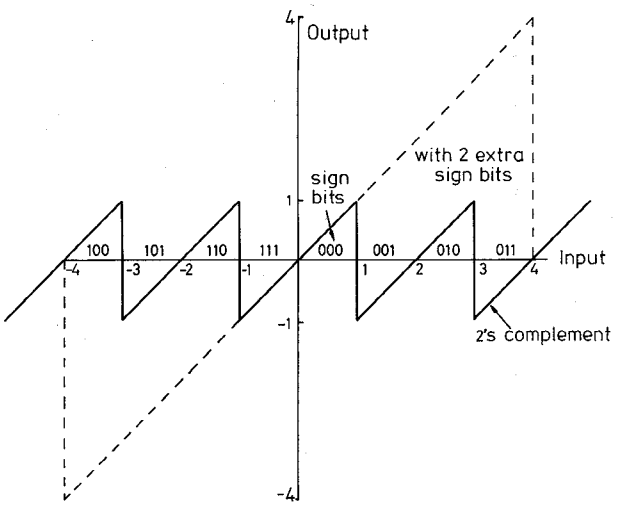

Fig. 6. Overflow characteristics.

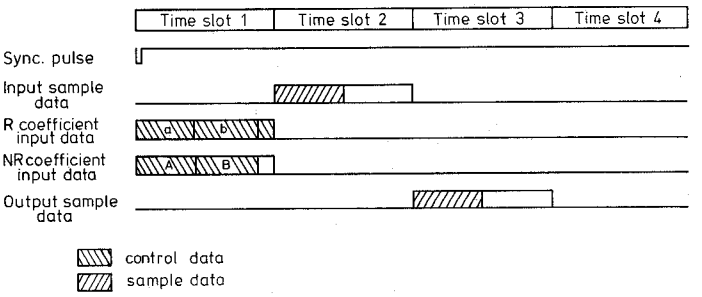

Fig. 7. Filter timing diagram for time-slot 2 input sample data.

processing elements provide the correct delay; for $Y = 8$ a 224 bit shift register is provided on the chip; for other orders of multiplex an external delay of 32 $(Y - 1)$ bits is required. The delay elements in the level detection functions are provided in the same way.

E. Quantization

Mention has already been made of the need to include a quantizer (Q_R) in the feedback loop of the filter as indicated in Fig. 5. Quantization can be performed in a number of different ways, i.e., truncation, rounding, but in order that the errors introduced shall have zero mean and the smallest possible variance FAD uses rounding. Rounding is achieved by adding 2^{-16} to the output of the previous adder and truncating the result. Truncation of the two's complement numbers results in errors between -2^{-15} and 0, but the preaddition of 2^{-16} means that the error is between -2^{-16} and $+2^{-16}$, i.e., the data are rounded.

F. Overflow Protection

The two's complement code has a maximum possible value of $1 - 2^{-N+1}$ and, because it is an asymmetric code, its most negative value is -1. As addition takes place in the digital filter it is possible for the maximum positive or negative values to be exceeded and then overflow is said to have occurred. Fig. 6 indicates the overflow characteristic of the two's complement code. Clearly, overflow introduces a severe nonlinearity into the filter resulting in a gross error. If such an error occurs in the feedforward section of the filter, although the output is incorrect, the error lasts for one sample time only. However, if the overflow occurs in the feedback section of the filter the effect is more severe. At best the overflow will cause an error which will die away over a period of time related to the time constant of the filter; at worst, the filter will oscillate ad infinitum. Two's complement overflow must, therefore, be prevented.

In normal operation overflow is obviated by the correct scaling of the filter input, but overflow may still occur as explained in Section II-A. To prevent the filter oscillating, the FAD IC contains an overflow circuit which stabilizes the filter if overflow occurs; the position of the circuit is indicated in Fig. 5. Overflow is detected in the following manner. Two extra sign bits are allocated to each of the inputs to the adders so that the effective number range of the adders is now between -4 and 4. As there are only four inputs to the adders it can be seen that the extra number range is never exceeded and the adders have the characteristic shown in Fig. 6. The overflow detector simply examines the three output sign bits and, if they are nonidentical, indicates that overflow has occurred. On detection of an overflow the reset procedure commences. In order to maintain the flexibility of the FAD IC with regard to variable order of multiplex, it is necessary that the resetting occurs on the data emerging from the 2-to-1 line selector. A stable overflow strategy is obtained by gating the output of the data selector to zero for Y complete computation cycles. The reset commences when data for channel one are emerging from the selector and finishes after the last data for channel Y have emerged.

G. Coefficient Loading

The multipliers in the filter require their coefficients in parallel but this is not compatible with serial data transfer, and hence a low number of input pins to the IC. This is overcome by entering the coefficient data into serial to parallel converters and then capturing it on latches. Coefficient data are entered in one computation cycle and then used during the next (Fig. 7). Three input pins are required to be able to enter all the coefficients and other control data for both the filter and detect functions.

H. Digital Full-Wave Rectification (FWR)

FWR is the process of forcing all data to have the same sign. In the FAD IC data are forced to be positive by inverting all the bits of negative words. This is not strictly accurate because, to multiply the 2's complement number by -1, 2^{-15} should be added to the word after bit inversion. However, the error is not significant.

I. The Digital Integrator

The integrators implemented in the FAD IC are unity gain, leaky integrators with leak factors of $1-2^{-p}$ where p is an integer in the range 1 to 8 inclusively; Fig. 8 shows a leaky integrator. The variable leak factor is used to control the rise time

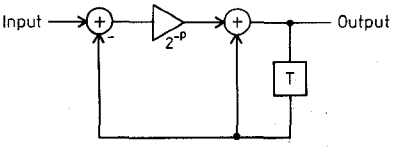

Fig. 8. Leaky integrator.

of the integrator. The two integrators in the FAD IC have independent leak factors.

J. Relative Level Detection

A relative detection capability is easily provided from the previously described elements. However, to provide a dedicated multiplier for this purpose would be very wasteful in chip area. Fortunately, the relative detection capability is normally used when the preceding filter has its complex zeros located on the unit circle, i.e., the multiplier coefficient B is unity. This factor is exploited in the FAD IC by arranging the B multiplier to be used either in the filter or not as required. As Fig. 5 indicates, the B multiplier is completely independent of the filter and detect functions and the appropriate external connections must be made to determine whether it is used in the filter or relative detection circuit.

IV. Circuit Implementation

A. Technology

The FAD integrated circuit was designed using an in-house 5 V Si gate NMOS process with depletion implants and 5 μm minimum line widths. This process requires a substrate bias which it was decided to generate on-chip. The bias generator used comprises a Schmitt-trigger oscillator followed by a charge pump designed to produce a bias voltage of approximately −2.5 V.

B. MOS Circuit Design

The FAD logic is implemented using static and two-phase ratioed circuitry. With 5 V logic levels being produced from gates by the use of depletion load transistors, and the generation of 5 V clock levels, it was possible to use transfer transistors to perform many logic functions without incurring voltage drops greater than one threshold voltage. Using transfer transistors in this way considerably reduces gate complexity and hence propagation delays and power dissipation. For example, transfer devices were used in the input scaler (Fig. 9), which is required to multiply incoming data by powers of one half. The data is entered LSB first into a 13 stage shift register, the output being selected from the register input or from any one of the stages. If the output is taken from the 13th register stage it is deemed to be multiplied by unity, but for each stage bypassed by the selector the data will appear at the output one clock pulse earlier and so be shifted by one bit towards the LSB, which is the same as dividing it by 2. The scaling factor is supplied as a 4 bit code which has to be decoded and used to operate a 14:1 line selector. These functions are performed in the array of transfer devices which considerably simplifies the logic that would be required to perform separate decoding and gating operations, and allows minimum dimension transistors to be used as they do not carry any dc current.

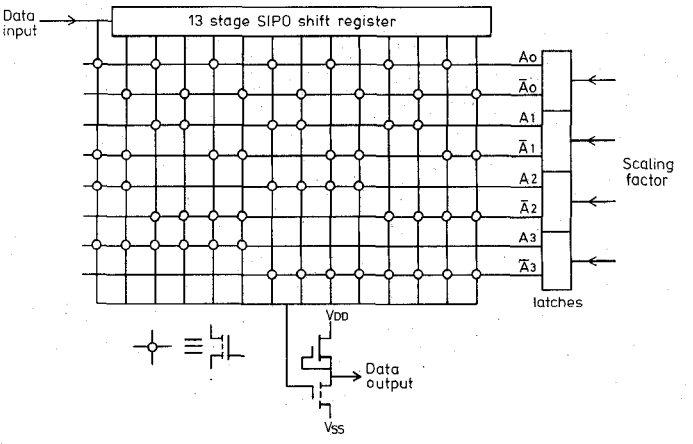

Fig. 9. Input scaler.

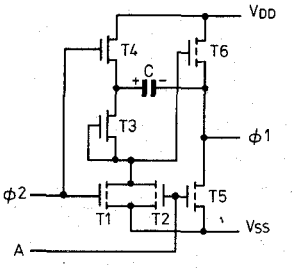

Fig. 10. Push-pull stage for one phase of the clock generator.

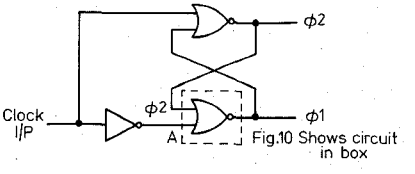

Fig. 11. Two-phase clock generator.

C. Clock Generation

The two-phase clocks are produced on-chip from a master clock and are required to be of 5 V amplitude. Because of the large capacitance loading on the clock generators, it was impracticable to use standard depletion load drivers, as this would have resulted in excessive power dissipation. A bootstrapped push-pull driver was therefore designed (Fig. 10), and two such drivers were cross-coupled to form the two-phase clock generator (Fig. 11). The operation of one driver, for $\phi 1$, is as follows: when A is high, $\phi 1$ is low and the feedback voltage $\phi 2$ is high turning T_1 and T_4 on. The capacitor is charged via T_4 and T_5, the dimensions of the depletion transistors T_3 and T_4 being designed to give a small voltage drop across T_4. When the clock input changes, "A" goes low turning off T_2 and T_5, but not

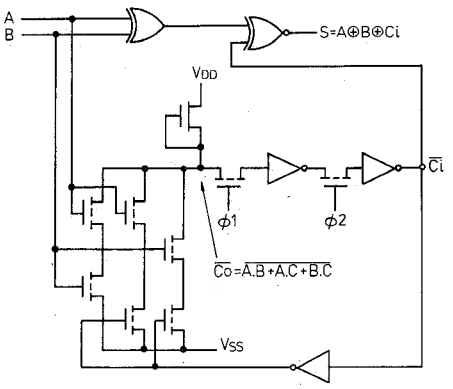

Fig. 12. Adder circuit.

until $\phi 2$ goes low can T_1 turn off, so preventing excessive clock overlap. When T_1 turns off the capacitor positive plate is connected to T_6 gate via T_3. As T_6 starts to turn on, and the voltage $\phi 1$ starts to rise, the gate-to-source voltage of T_6 will be maintained at the voltage across C, so keeping T_6 turned on and allowing the final level of $\phi 1$ to rise to the V_{DD} supply voltage of 5 V. As the voltage on the capacitor positive plate starts to increase, the source of T_4 becomes positive with respect to its gate and T_4 then turns off preventing it from discharging the capacitor. The capacitor C has a value of 10 pF and is formed by a depletion transistor with its drain and source connected to form the negative plate and the gate as the positive plate.

D. Adder Design

The adders and adder/subtractors used in the circuit are of a serial carry-save design. The adder (Fig. 12) is realized using two exclusive-OR/NOR circuits to produce the sum output, and combinational logic with a shift register stage to generate and store the carry for addition to the next MSB. Subtraction is performed by the use of an adder circuit and adding the two's complement of the subtrahend. The two's complement is produced by the use of an exclusive-OR gate in the subtrahend input which can be used to selectively invert the data by the application of an add/subtract control signal. This method is only approximate, as the true two's complement is obtained by inverting the data and then adding 1, but this error can be tolerated in FAD where subtraction is performed on the multiplier output data which are later rounded and truncated.

E. The Multiplier

The heart of the FAD circuit is the four serial-parallel multipliers, which each multiply a 16 bit serial number by a 13 bit parallel coefficient. The multiplier design used is an implementation of Booth's algorithm [15].

Each multiplier contains an input stage followed by 13 stages of an essentially standard cell (Fig. 13) to perform the logical operations required to add or subtract the coefficient (CF) from the partial product (S_i), or do nothing to the partial product, depending on the current (Y_o) and previous (Y_{-1}) values of the serial input. Latches then store the sum and carry products generated, the sum for passing to the next stage of the the multiplier (S_o), and the carry for adding to the next bit of the partial product as it is clocked LSB first through the multiplier.

The multiplier requires the multiplier coefficient bits in parallel, although in practice these data have to be applied to the FAD serially to reduce the number of input pins. Hence, the coefficients are entered into a serial-in parallel-out shift register incorporated into the multiplier stages and are then transferred to the coefficient stores at the commencement of each multiplication cycle by CLEAR which also resets the sum and carry stores. The layout for the multiplier cell occupies a chip area of 0.1 mm^2 and has a power consumption of 2 mW; this compares well with previously described circuits [16].

Booth's algorithm requires that the value of the last (sign) bit of the serial input stream is held as the input for the remainder of the multiplication cycle. A "sign extend" buffer has, therefore, to be incorporated at the front of each multiplier to transmit the serial data through to the multiplier for the first 16 clock pulses of each cycle and then hold the last bit for the remaining 16 clock pulses.

V. Conclusion

Initial samples of FAD (Fig. 14) were produced at British Telecom Research Laboratories and are currently being used in tone signaling receivers in System X local exchanges [3]. Devices are now being produced by commercial semiconductor manufacturers using the original mask production data.

Although designed for a specific application, efforts have been made to make the IC as versatile as possible. At the same time, with the nonspecialist engineer in mind, the circuit has been made easy to use. The result, it is hoped, is an IC which will find use beyond its primary application.

Acknowledgment

The authors would like to thank the Director of Research of British Telecom (a part of the Post Office) for permission to publish this paper.

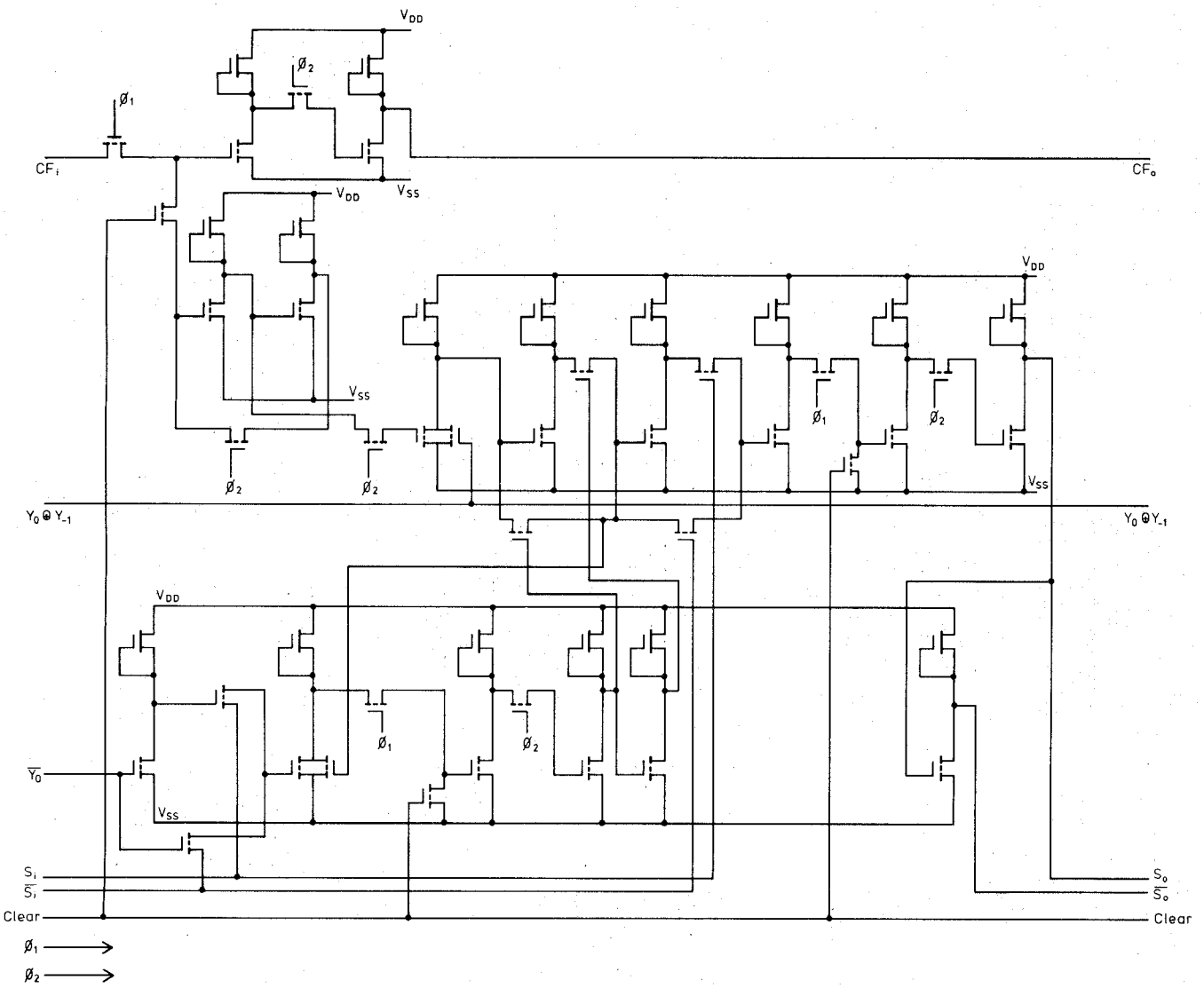

Fig. 13. Multiplier cell circuit.

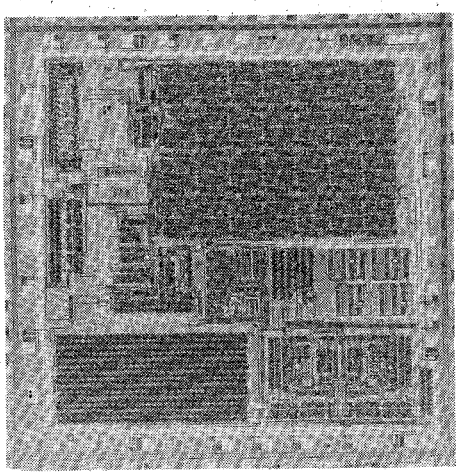

Fig. 14. FAD circuit microphotograph.

References

[1] C. B. Miller and W. J. Murray, "Transit-trunk network signaling systems," *Post Office Elec. Eng. J.*, vol. 63, pp. 43–48, 91–95, 159–163, 1970.

[2] B. R. Horsefield, "Fast signaling in the UK telephone network," *Post Office Elec. Eng. J.*, vol. 63, pp. 242–252, Jan. 1971.

[3] J. Martin, "System X," *Post Office Elec. Eng. J.*, vol. 71, pp. 221–224, Jan. 1979.

[4] E. J. Welsh and P. F. Adams, "A new digital 1 VF signaling unit," *Post Office Elec. Eng. J.*, vol. 72, pp. 108–116, July 1979.

[5] B. Gold and C. M. Rader, *Digital Processing of Signals.* New York, NY: McGraw-Hill, 1969.

[6] L. A. Schmidt, "Designing programmable digital filters for LSI implementation," *Hewlett-Packard J.*, vol. 29, no. 13, pp. 15–23, 1975.

[7] M. Townsend, M. E. Hoff, Jr., and R. E. Holm, "An NMOS microprocessor for analog signal processing," *IEEE Trans. Comput.*, vol. C-29, pp. 97–102, Feb. 1980.

[8] J. Boddie, G. T. Daryanani, I. I. Eldumiati, R. N. Gadenz, J. S. Thompson, and S. M. Walters, "A digital signal processor for telecommunications applications," presented at IEEE Int. Solid-State Circuits Conf., 1980.

[9] D. F. Millet, "A signal processor fast enough to do real-time voice band," *Electron. Des.*, vol. 4, pp. 86–89, Feb. 1980.

[10] R. W. Blasco, "V-MOS chip joins microprocessor to handle signals in real time," *Electronics*, pp. 131-138, Aug. 30, 1979.
[11] E. R. Broad and P. F. Adams, "Digital filters: A template method of design," *Radio and Electronic Engineer*, vol. 42, no. 11, pp. 489-495, Nov. 1972.
[12] A. V. Oppenheim and C. J. Weinstein, "Effects of finite register length in digital filtering and fast Fourier transform," *Proc. IEEE*, vol. 60, no. 8, pp. 957-976, Aug. 1972.
[13] P. M. Ebert *et al.*, "Overflow oscillations in digital filters," *Bell Syst. Tech. J.*, vol. 48, pp. 2999-3020, Nov. 1969.
[14] Y. Chu, *Digital Computer Design Fundamentals*. New York, NY: McGraw-Hill, 1962.
[15] J. R. Mick and J. Springer, "Single-chip multiplier expands digital role in signal processing," *Electronics*, vol. 49, no. 10, pp. 103-108, May 13, 1976.
[16] H. J. De Man, C. J. Vandebulcke, and M. M. Van Cappellen, "High-speed NMOS circuits for ROM accumulator and multiplier type digital filters," *IEEE J. Solid-State Circuits*, vol. SC-13, pp. 565-572, Oct. 1978.

Peter F. Adams was born in London, England, in 1946. He joined the British Post Office in 1966 as a Student Apprentice. In 1967 he went to Southampton University, and received the B.Sc. degree in electronic engineering in 1970.

Since 1970 he has been with the Post Office Research Department, Dollis Hill, London, now British Telecom Research Laboratories, Martlesham Heath, Suffolk, England, working on the application of digital filtering techniques in telecommunications, specializing in signaling systems. Currently, he is Head of the Speech-Band Data Transmission Group investigating echo cancellation techniques and signal processing for high-speed modems.

Mr. Adams is a member of the Institution of Electrical Engineers.

John R. Harbridge was born in Edgware, England on June 20, 1949 and joined British Telecom Research Laboratories, Martlesham, Suffolk, England, in 1962. He worked on the construction and then design of life test equipment for reed switches before becoming involved in 1971 in the application of microelectronics to telephone switching systems, including working on the development of early microprocessor systems. In 1978 he joined the microelectronics division where he is concerned with the design of custom LSI circuits.

Roderick H. Macmillan was born in Faversham, England, on February 19, 1947. He joined British Telecom (formerly Post Office Telecommunications) in 1965 and transferred to their Research Laboratories, Martlesham, Suffolk, England, in 1970. Until 1978 he worked in the Microwave Propagation Section, where he was mainly concerned with the design of data handling systems for satellite propagation measurements. He is currently in the Digital Signal Processing Group, where he is concerned with the design and development of signaling receivers using custom LSI techniques.

1.5 V 1 K-CMOS-RAM with Only 8 Pins

GÜNTHER MEUSBURGER

***Abstract*—A new kind of a 256-bit × 4-bit static RAM working on a supply voltage down to 1.2 V will be described. A serial interface for the address and the data with a 4-bit bus reduces the pincount of the RAM to only 8. Special design techniques to reach the design goal—very low power at a reasonable circuit speed—are discussed in detail. The device is fabricated in a low power silicon gate CMOS process. An operating power of 500 μW/MHz and a standby power of less than 1 μW at 1.5 V supply voltage was achieved. With this serial interface a cycle time of 1 μs at 1.5 V was measured.**

I. Introduction

As microcomputers have become more versatile, users have been able to add intelligence to their products in ways that were completely unthinkable just a few years ago. This is partly due to computing power, but factors like chip size, power consumption, and ease of interfacing can be even more important [1].

A new microcomputer developed to serve these new applications features a keyboard interface and a direct drive for 8 digits of LCD [2]. This increased interfacing power does not mean doing without memory, but requires concurrent design of peripherals that can use available pins most efficiently.

A suitable memory interface uses two control signals and four data address lines—the same number of pins as for intelligent peripherals. This resolves the tradeoff between speed and size. The resulting 8-pin memory must interface directly to the microcomputer to minimize chip count. It must also operate at 1.2 V to 1.8 V with negligible power consumption for battery powered applications. This paper shows how these design goals were met.

II. Circuit Description

Fig. 1 is a block diagram of the new memory device. It is composed of the well-known blocks cell array, bit and word

Manuscript received October 20, 1980; revised December 22, 1980. This work has been supported by the technological program of the Department of Research and Technology, Federal Republic of Germany.

The author is with Eurosil, GmbH, Munich, Germany.

0018-9200/81/0600-0190$00.75