

A 230-MHz Half-Bit Level Pipelined Multiplier Using True Single-Phase Clocking

Dinesh Somasekhar and V. Visvanathan

Abstract—An 8 bit \times 8 bit signed two's complement pipelined multiplier megacell implemented in 1.6 μm single poly, double metal N-well CMOS is described. It is capable of throughputs of 230 000 000 multiplications/s at a clock frequency of 230 MHz, with a latency of 12 clock cycles. A half-bit level pipelined architecture, and the use of true single-phase clocked circuitry are the key features of this design. Simulation studies indicate that the multiplier dissipates 540 mW at 230 MHz. The multiplier cell has 5176 transistors, with dimensions of 1.5 mm \times 1.4 mm. This multiplier satisfies the need for very-high throughput multiplier cores required in DSP architectures.

I. INTRODUCTION

A HIGH-SPEED multiplier core forms one of the basic building blocks for digital signal processors. Signal processing applications typically implement iterative algorithms that need a large number of multiply add/accumulate operations. This requires a high throughput from the multiplier block. High throughput rates are best satisfied by pipelining standard multiplier architectures. Recent advances in real-time filter architectures [1], [2] indicate that pipelined multipliers, with a clock frequency much higher than the data rate, can be used to significantly reduce the overall chip area without increasing the power consumption or the I - O bandwidth. The above reasons motivate us to explore the limits of pipelined multipliers implemented in a standard, mature CMOS process.

There is a wide range of possible multiplier architectures [3]. Multiplier cores in general-purpose processors try to minimize the latency of computation. Typically, nonpipelined array or tree architectures are used. The former is preferred, where area and ease of implementation are the main issues, while the latter, e.g., the Wallace tree multiplier [4], the Dadda multiplier [5], and the binary tree multiplier [6] are used when performance from the multiplier block is the key issue. High-performance tree

multipliers typically achieve latency on the order of tens of nanoseconds.

As previously mentioned, digital signal processing applications place greater demands on multiplier cores. What is more important is the rate at which successive multiplications may be delivered from the multiplier core—the throughput—rather than the delay from data entering the multiplier to the time when the product bits finally emerge from it. Highly pipelined versions of array multiplier architectures satisfy this need. The long sequence of successive multiplications needed for these applications ensures that the pipeline operates efficiently. The high throughput comes at the expense of latency, which is typically in units to tens of clock cycles. The structural regularity of carry-save array multipliers, a hardware realization of the shift and add algorithm with carry addition being postponed to a succeeding stage, makes them an attractive candidate for pipelining. This architecture uses an array of AND gates for generating the partial-products, with summing of partial-products being carried out with a full-adder array. A *bit-level pipelined architecture* is one in which pipelining is done by introducing register stages after every full-adder row. As an indication of the throughput offered by these multipliers, see [7] and [8]. The architecture has also been used as the basic block for high-speed multiply accumulate chips [9].

This work describes the design and verification of a CMOS 8 bit \times 8 bit, pipelined signed two's complement carry-save array multiplier, capable of a throughput rate of 230 000 000 multiplications/s. This rate is achieved because of a very fine grained 24 stage half-bit level pipeline. The very-high clock rate makes clock skew the predominant design issue. The current architecture uses the true single-phase clocking scheme proposed by Yuan *et al.* [10], which has the inherent advantage of clock skew problems being restricted to those of a proper distribution of only one clock phase. This scheme has been demonstrated as being very attractive for the design of very-fast added elements [11]. A full 16 bit two's complement product is generated using the modified Baugh-Wooley algorithm [8], [12] to multiply two 8 bit signed two's complement numbers. The work has been previously briefly reported in [13].

A full circuit simulation, although desirable, is not practical for large circuits as in the present case. Multiplier characterization has been done by SPICE simulation

Manuscript received March 25, 1993; revised July 30, 1993. This work was supported in part by the Department of Electronics, Government of India, under the LSI/VLSI Design Center and System and VLSI Design for Digital Television projects, and by the Technical University of Delft-IISe VLSI Collaboration project.

D. Somasekhar was with the Electrical Communication Engineering Department, Indian Institute of Science, Bangalore 560012, India. He is now with Texas Instruments India, Bangalore 560052, India.

V. Visvanathan is with the Computer-Aided Design Laboratory, Super-computer Education and Research Center, Indian Institute of Science, Bangalore 560012, India.

IEEE Log Number 9213085.

for smaller blocks and timing simulations for the overall multiplier. This scheme allows the design to be verified in reasonable time, without resorting to overall circuit simulation [8]. Simulation studies indicate a power dissipation of 540 mW at a clock speed of 230 MHz. The implementation is in a 1.6 μm N-well CMOS single polysilicon double metal process using the NELSIS IC Design System [14]. The multiplier has 5176 transistors in an area of 1.5 mm \times 1.4 mm.

II. TRUE SINGLE PHASE PIPELINED CIRCUITS

Various clocking schemes exist in CMOS for implementing pipeline stages, as for example, two-phase clocking, four-phase clocking, NORA circuitry, and C²MOS [15]. All these schemes rely on multiple clock phases for obtaining race-free clocking. True single-phase clocking [10] uses a single clock phase and still achieves race-free clocking by using the complementary nature of N and P devices in CMOS to simulate the effect of two clock phases. At very-high clock rates where clock distribution is critical, the need for maintaining non-overlap of clock phases in multiple-phase clocking schemes becomes difficult to achieve. In contrast, true single-phase clocking has to ensure proper rise and fall times, and keep skew between communicating blocks within bounds to ensure proper operation.

In a single-phase clock strategy, pipelining is achieved by having alternate P and N blocks, each active on different levels of the single clock. The various ways in which a pipeline can be implemented is described in [11]. The TSPC1/TSPC2 (true single-phase clocked type 1 and type 2) circuits proposed by Yuan and Svensson [11] are used in this work. This scheme fits in well with a finely pipelined design, since it yields very-high-speed circuits. A detailed description of these circuits can be found in [11]. A dynamic clocking strategy is used for these circuits. In this scheme latches are integral to the compute block. In our design, pipelining is carried out within full-adder blocks, so that a full-adder has two pipeline stages: a P half and an N half. The N half evaluates when the clock is high and the P half when the clock is low. In both cases, the half cycle in which evaluation does not occur is used for precharging, and is therefore not wasted. Thus, an n stage pipeline has a latency of $n/2$ clock cycles.

Both TSPC1 and TSPC2 circuits have both the N form and the P form. These circuits are latches with an integral evaluation block consisting of a NMOS/PMOS tree. They have the inherent advantage of dynamic CMOS in offering low capacitive load to previous driving stages. A race-free design is obtained by cascading alternate N and P blocks. One disadvantage of these circuits is their non-inverting nature. This makes the use of separate inverters necessary for evaluating logic that need the complement form of signals.

A comparison of TSPC1 and TSPC2 indicates that for the N form, on the whole, TSPC2 is better. TSPC2 shows lower glitches in the output than TSPC1. Glitches arise in

the output of TSPC1 when the output is high (low) for the N (P) form. Yuan and Svensson [11] describe the cause of these glitches and the reason TSPC2 is better. A further advantage of TSPC2 is an improved rise time. However, the fall time is degraded; but this is not a problem for the N form since the fall time is typically very small. In the current full-adder, the N half is implemented using TSPC2 circuits. For the P type of circuits the opposite is true; the rise time gets degraded for TSPC2, although glitches are lower. It is for this reason that in the P blocks TSPC2 circuitry is not preferred, because fairly large P transistors would be necessary to get a rise time comparable to the N half. This leads to a much greater clock load than TSPC1. The P block uses TSPC1 circuitry with transistors sized to minimize glitching. Blocks in the P half are of a lower complexity than blocks in the N half. This is achieved by using fully complementary CMOS to do part of the P half evaluation.

Single-phase circuits described above are not free from disadvantages, namely:

- a heavy capacitive load on the clock lines
- internal nodes are at a high impedance in the latched state, leading to a degraded noise immunity. This problem can be solved by having pullup/pulldown transistors on these nodes that are qualified by the clock, and
- absence of a path to ground (VCC) for the N (P) form in the latched condition disallows the use of pass-gate-type circuits at the outputs.

III. MULTIPLIER ARCHITECTURE

The array architecture is the architecture of choice for very-high throughput pipelined multipliers because of its structural regularity and semisystolic nature. Most of the signals propagate between local blocks. Various forms of array architectures may be conceived of, depending both on the direction of data flow and the way in which the last stage of the multiplier (the vector merge adder) is implemented.

The current work uses a carry-save architecture, with the LSB partial product being evaluated first. Data flow is solely in the vertical direction. Although this architecture is semisystolic (the multiplier bits are broadcast over an adder row), in comparison to truly systolic architectures that use both vertical and horizontal pipelining, it is more efficient in terms of required silicon area [7]. Because of the nonsystolic nature of the architecture, in addition to the clock distribution problem, delay problems arising due to long data paths for the multiplier bits should also be considered.

The multiplier implements the modified Baugh-Wooley algorithm for multiplying signed two's complement numbers. The modified Baugh-Wooley algorithm as applied to an 8 bit \times 8 bit two's complement multiplier is illustrated in Table I. Generation of the partial products can be done most easily by using an AND gate array. Schemes like the Modified Booth Recoding lead to a decrease in the number of partial product terms and consequently, to

A full-adder block has four inputs, the *sum* and *carry* inputs from the previous row, *si* and *ci*, respectively, and the *aj* and *bi* inputs that are used for generating the partial product. It generates the *sum* and the *carry* (*cy*) bits; in addition, it also generates a latched version of the *a* input, *aout*. A complete schematic of the full-adder is shown in Fig. 2. This circuit differs from that in [11] in allowing partial product evaluation in parallel with the sum generation and in minimizing the complexity of the P half.

Evaluation of the *sum* and *carry* is a two step process. In the first step, the P half generates: the *xor* of the *si* and the *ci* inputs, the product term $aj \cdot bi$, *cil*, which is a latched version of the *ci* input, and *ajl*, a latched version of the *aj* input. Evaluation is carried out during the low period of the clock, the outputs being stable on the high period. Either the *xor* or the *xnor* term may be implemented with the same transistor complexity. In the current design, the choice of generating the *xor* term was governed by layout issues. Generation of the *xor* term is carried out by first generating $\overline{si \cdot ci}$ using a fully complementary CMOS gate. The *xor* term is then obtained as $xor = (si + ci) \cdot \overline{si \cdot ci}$.

In the high period, the N half evaluates the *sum* and *carry* terms as follows: $sum = \overline{aj \cdot bi} \cdot xor + aj \cdot bi \cdot \overline{xor}$ and $cy = aj \cdot bi \cdot xor + cil \cdot \overline{xor}$. Inverters are used to generate *xnor* and $\overline{aj \cdot bi}$ from *xor* and $aj \cdot bi$, respectively. A further N latch is used to latch the *ajl* line to produce *aout*.

Note that the computations have been partitioned so that more of it is done in the faster N half. The *xor* and the $aj \cdot bi$ terms in the P half are generated using TSPC1 circuits, while the N half exclusively uses TSPC2 circuits. Because the P half uses TSPC1 circuits, the outputs generated by it show positive spikes. Though the spikes are minimized by proper transistor sizing, they are not totally eliminated. The outputs of the N half show far smaller spikes, since TSPC2 circuits are used. Each full-adder block can drive a load of 0.1 pF at a clock speed of 250 MHz. The load offered at the input *si* and *ci* lines are of the order of 0.1 pF, while the *bi* input has an even lower input load of 0.06 pF. The total load capacitance offered on the *bi* line by one row, comprising seven full-adders and a NAND or AND gate is 0.5 pF. A disadvantage of the true single-phase clocking scheme is the high capacitance load it imposes on the clock lines; each full-adder block adds 0.4 pF to the clock lines.

The transistor count for the full-adder block is 59, with N transistors accounting for 30 of them. This includes the latch for the *a* line, which needs 12 transistors, and the partial product generator, which needs 7 transistors. This yields a transistor count of 40 for the actual adder block. In comparison, an FCC full-adder has a transistor count of 24, while a NORA full-adder requires 25 transistors. The area of the full-adder is $142.8 \mu\text{m} \times 147 \mu\text{m}$.

SPICE simulations (see Fig. 3) indicate that a clock period of 4.0 ns can be easily sustained. At a clock frequency of 250 MHz, the estimated power dissipation using SPICE (following the method proposed by Kang [16])

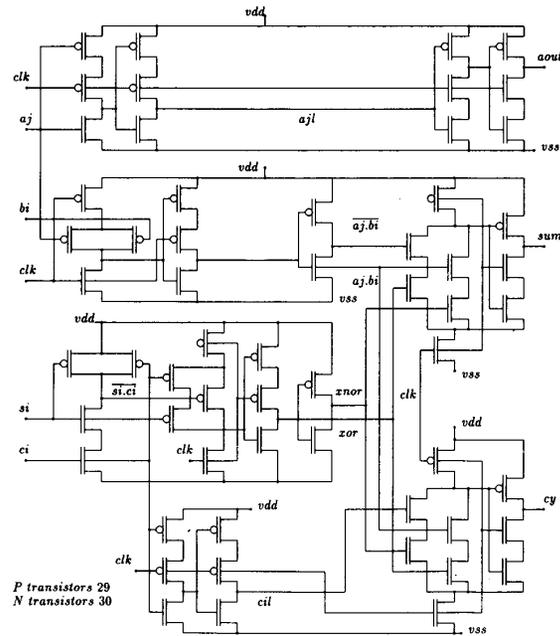


Fig. 2. The full-adder circuit.

is 6 mW when the full-adder block is cycled through all its input combinations. The circuits used are fairly insensitive to the rise and fall times of the clock. SPICE simulations have been done with a triangular clock waveform with 2 ns rise and fall times for a clock frequency of 250 MHz.

For the final row of full-adders, the following modifications are necessary. Since this row requires complemented partial products, the $aj \cdot bi$ and the $\overline{aj \cdot bi}$ lines are exchanged. Further, as this is the last row, there is no need to latch out the *ajl* line. Hence, the N latch generating *aout* is removed.

Signal routing for the full-adder is illustrated in the floorplan for the multiplier shown in Fig. 1. Each full-adder block is designed for abutment. All data flow is in the vertical direction except for the *bi* line, which runs horizontally. Power and clock lines run horizontally in first metal. There are no breaks in these lines since they are routed perpendicular to the data flow direction.

B. Buffered Latch

This block drives the multiplier bit for a row of the full-adder array. Since the full-adder evaluates its inputs on the clock low level, the buffered latch must ensure that the multiplier bit is valid during this phase. The buffered latch consists of a P single-phase latch, an inverter, a comparatively larger N latch, and a large-sized inverter capable of meeting the timing constraints while driving the multiplier line. Proper operation of the multiplier requires $t_{clkh} > t_{dN} + t_{dinv}$, where t_{clkh} , t_{dN} , and t_{dinv} are, respectively, the clock high period, the delay of the N

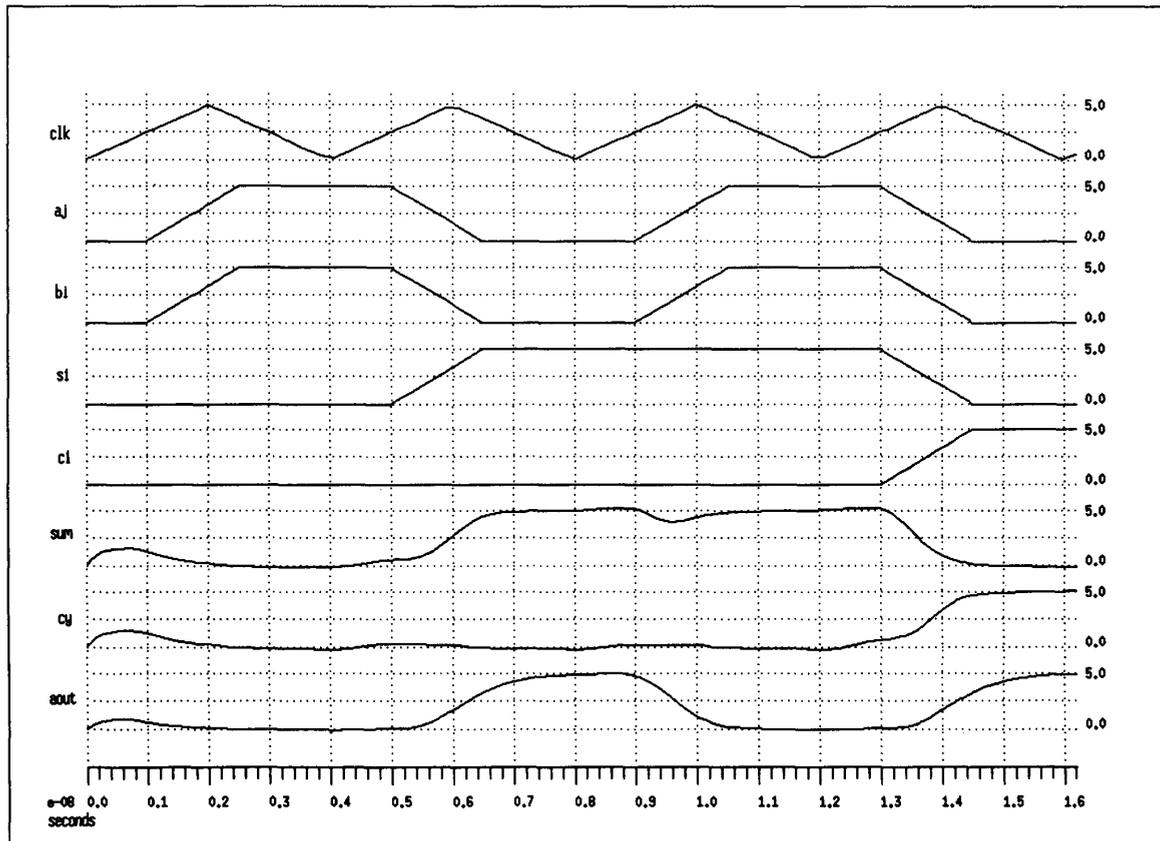


Fig. 3. SPICE waveforms of the full-adder for a triangular clock at 250 MHz.

latch for a high to low output transition, and the delay in switching the inverter from low to high.

Buffer sizing is decided by the above-mentioned timing constraint. The design allows driving the 500 fF multiplier line capacitance, at a clock period of 3.8 ns. In the current design, the multiplier line capacitance has been kept at a minimum, by ensuring that each full-adder connected to the multiplier line contributes a load equal to a single P transistor (see Fig. 2).

C. Vector Merge Adder

The vector merge adder uses a triangular array of half-adders. This structure is similar to that used in the multipliers in [7] and [8]. In [8] merging two half-adder rows is recommended, which leads to a decrease in latency. This scheme is implemented here (see Fig. 1), which allows an 8-bit sum of two 7-bit numbers (this needs 8 rows of half-adders) to be generated in four clock cycles. Using a triangular half-adder array for the vector merge adder has the advantage of simplicity and regularity of design as compared to designs like a carry lookahead adder or a binary tree adder, which are capable of offering lower latency. In comparison to a pipelined ripple carry adder it offers a lower latency because of two half-adder rows

being merged. An attendant advantage is that fewer deskewing latches are required.

Two rows of the vector merge adder can be combined because the basic delay of a half-adder is half that of a full-adder. A cascade of two half-adders constitutes a basic block (block *ha* in Fig. 1). Each block has one half-adder implemented using P-type TSPC1 circuits and another half-adder, which uses N-type TSPC2 circuits. Blocks on the main diagonal have one half-adder in the P half and a latch in the N half (block *ha2*). Blocks along the left edge of the array (*ha3* and *ha4*) do not need to generate the carry signal, which essentially means that they are XOR gates. The percentage area used by the vector merge adder and the deskewing registers required at the output is 21 percent of the total area of the multiplier.

In this design, the vector merge adder also takes care of the one addition required in the Baugh-Wooley algorithm for signed multiplication. For the 8 bit \times 8 bit case, a 1 has to be added in the 9th bit position and in the 16th bit position. In the 16th bit position, this is done by feeding in a 1 to the top most half-adder at that position (*ha4*). For the 9th bit position, three inputs have to be summed, which implies that a full-adder has to be used. However, this cannot be done because the delay of a full-adder is twice that of a half-adder, which would lead to the top

two half-adder rows not being merged. By noting that an addition of a 1 implies that the *sum* is the EXNOR of the other two inputs and the *carry* is the OR of the inputs, it is possible to design a block very close to a half-adder. This is the scheme that is employed here. This block labeled *fa1* in the floorplan implements $sum = aj \cdot bi + \overline{aj} \cdot \overline{bi}$ and $cy = aj + bi$.

IV. CLOCK AND POWER DISTRIBUTION

A crucial issue involved in high-speed clocked designs is the determination of the clock distribution method to be employed. The clock distribution used is illustrated in Fig. 4. The inverter tree employed is much more attractive than an *H tree* because

- an *H tree* design has to route lines in both the vertical and the horizontal directions. For the two-metal processes used, this makes it difficult to route the clock lines in low resistance metal without using vias
- the capacitance added by the clock lines is much higher for the *H tree* since the total wire length is much greater.

Fig. 4 shows the three-stage clock tree used in the current design. Each row of the multiplier has a common clock line, running horizontally in first metal and driven by a large-sized CMOS inverter. The total capacitive load seen by this inverter is typically 5.3 pF for a full-adder row. The AND gate clock line (*CLK0*), and the vector merge adder clock lines (*CLK8* to *CLK11*) have a relatively smaller capacitance allowing two adjacent rows of the vector-merge adder to be driven by a single clock buffer. A total of 10 inverters are required for driving the final clock lines. These are grouped into sets of three, three, and four and are driven by further clock buffers. The resulting tree architecture has 14 clock buffers.

Extensive SPICE simulations were done to arrive at optimum device sizes for the clock buffers. The width of the P device was selected to be 2.75 times that of the N device. The inverters C1 consist of 8 parallel N devices of width 16 μm and length 1.6 μm and 8 parallel P devices of width 44 μm and length 1.6 μm . Inverters C2 and C3 are smaller, consisting of 6 and 4 parallel transistors, respectively. Rise and fall times as determined by SPICE simulations for C1 driving a load of 5.3 pf are 650 ps. The basic blocks have been simulated using SPICE with much larger rise and falls times of 2.0 ns, thereby indicating that the buffers used are more than adequate. The maximum skew measured between two clock lines was 150 ps, between *CLK0* and *CLK1*. Propagation delay from the near to the far end of a row is reduced by using low resistance first metal (sheet resistance of 45 m Ω), for clock line routing. For the process used, polysilicon lines have an *RC* product that is 1000 times greater than that for metal lines. Calculations for optimum wire width size following the method described in [7], with a buffer dynamic resistance of 280 Ω , gives a width of 3.6 μm for a load of 5 pF. In this design a wire width of 4.0 μm has been used.

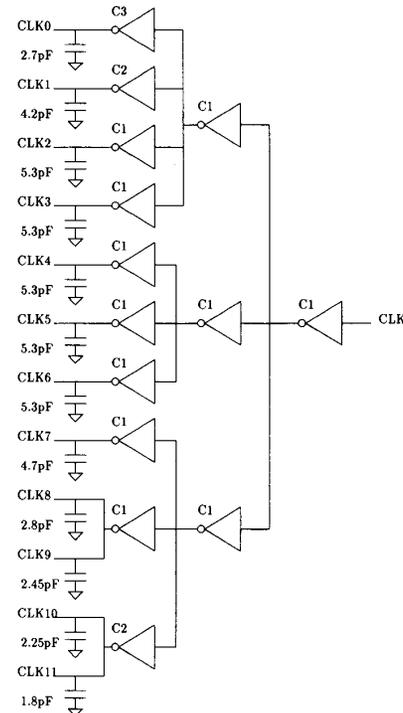


Fig. 4. Clock distribution tree.

Power dissipation in the clock drivers forms a large fraction of the total power consumed by the multiplier. In addition, charging and discharging the fairly large clock line capacitance causes large spike currents to be drawn from the power rails. SPICE simulations indicate that a typical inverter (C1) draws an average current of 7.2 mA, i.e., dissipates 36 mW of power while driving a 5.3 pF load at 250 MHz. Peak currents during switching are 30 mA for an inverter and 200 mA for the entire clock tree.

A. Output Buffering and Power Net Routing

Since the multiplier is intended to be a megacell in a larger chip, outputs are not designed to handle off-chip loads. Final outputs are buffered by a chain of two inverters so that a load of 0.5 pf can be driven at 230 MHz. Ishibe *et al.* [17] describe circuit methods that may be employed to enable driving off-chip loads at high speeds.

Power net routing is done in first metal throughout. Power lines, clock lines, and multiplier lines driven by the buffered latch are routed horizontally through the full-adder rows. Multiplicand and product lines are routed vertically in second metal. Power routing for the vector merge adder is carried out in a manner similar to that of the full-adder array, with power and clock lines routed horizontally in the first metal and data lines routed vertically in second metal. At the cell edges, lines from different rows are connected together. Power nets are routed using techniques described in [18]. Power routing for the clock system is kept distinct from the power nets of the

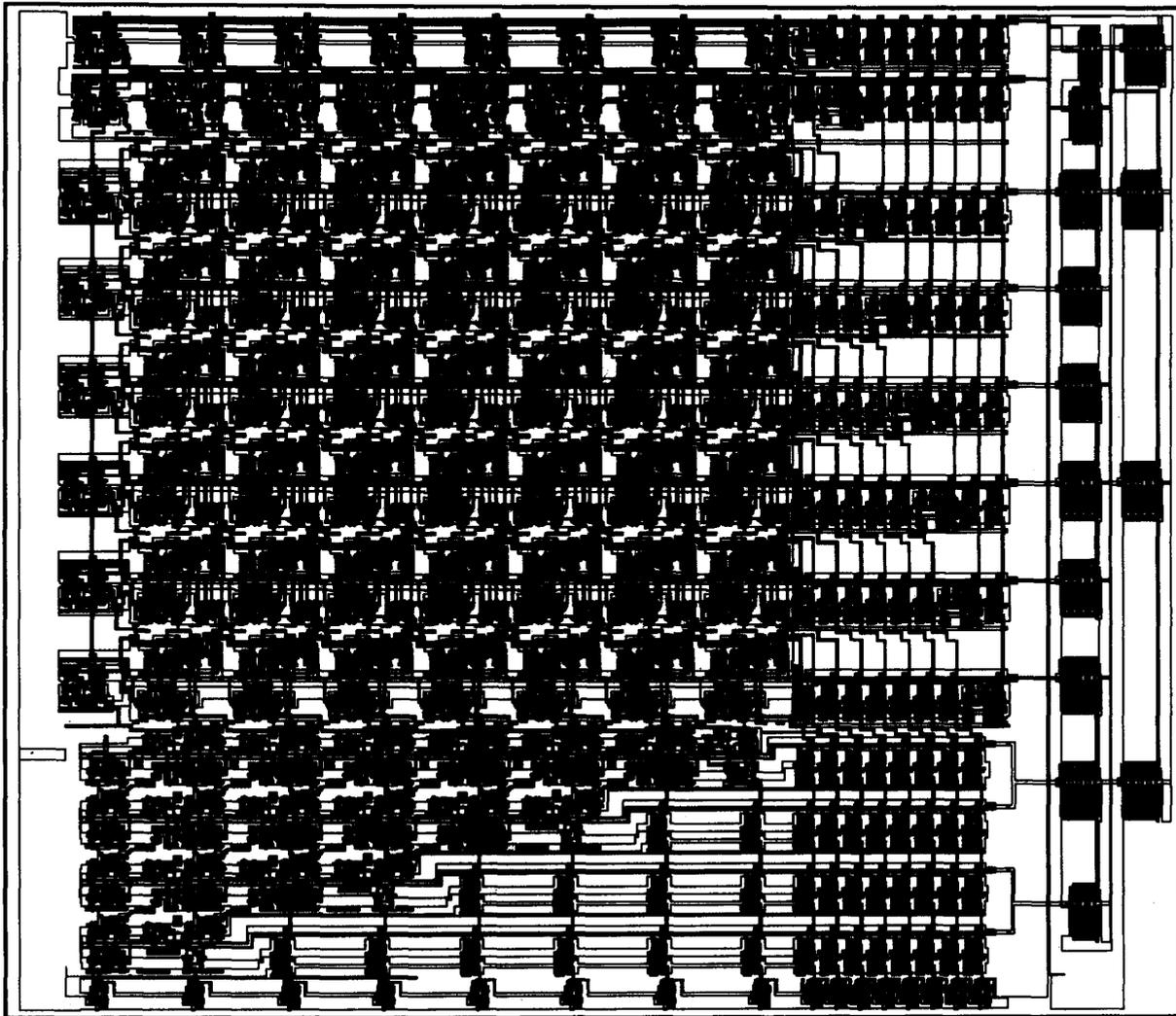


Fig. 5. Multiplier layout.

core full-adder array, because of the very large peak currents drawn by the clock tree.

V. RESULTS

SPICE simulation results for the full-adder block shown in Fig. 3 confirms proper basic block operation at 250 MHz. The full-adder operates properly even with large clock rise and fall times. SPICE results shown are with a triangular clock waveform. The robustness of the full-adder design with respect to high temperature is indicated by its capability to sustain clock rates of 180 MHz at 125°C. Full-adder power dissipation as obtained from SPICE indicates a value of 6 mW. All the blocks and the multiplier as a whole were extracted with full parasitic resistances and capacitances using the NELSIS hierarchical circuit extractor SPACE.

Overall timing simulations, using the NELSIS timing simulator SLS, confirm the correct operation of the mul-

TABLE II
COMPARISON OF PIPELINED MULTIPLIERS

	Noll 8 × 8 multiplier [7]	Hatamian 8 × 8 multiplier [8]	Lu 12 bit Multiply accumulate [9]	This Work 8 × 8 Signed Multiplier
Clock Frequency	330 MHz	75 MHz	140 MHz	230 MHz†
Latency	18 cycles	16 cycles	13 cycles	12 cycles
Transistor count	5480	--	10616	5176
Dimensions (mm × mm)	1.5 × 0.4	--	2.5 × 3.7	1.5 × 1.4
Power dissipation	1.5 Watts	0.25 Watts	1.5 Watts	0.54 Watts
Technology	NMOS 1μm 3V supply	CMOS 2.5μm	CMOS 1μm	CMOS 1.6μm

† Clock frequency obtained by timing simulations.

tiplier at 230 MHz. Detailed simulation results, not included in this paper, can be found in [19]. Multiplier power dissipation, as estimated by SLS, is 540 mW at 230 MHz. We expect a higher dissipation, since the simula-

tion estimate is not very accurate. The layout of the multiplier is shown in Fig. 5.

VI. CONCLUSIONS

The work shows the feasibility of implementing very-high-speed clocked systems, by using single-phase clocked circuitry. To the knowledge of the authors this is the fastest reported multiplier in a mature standard CMOS technology. Single-phase clocked circuits used in this work are no doubt more complex than standard logic styles. However, the performance obtained by such circuits is superior to that obtained by circuits using multiple clock phases. In addition, single-phase clocking allow pipelining at much finer levels than currently employed. A comparison of the performance obtained by this multiplier with respect to other high performance multipliers is shown in Table II.

For multipliers having longer word lengths, the semi-systolic architecture used here may not be suitable. This is because the length of the multiplier line (and hence the capacitance) is a direct function of the multiplicand word length. A truly systolic architecture would be preferable in such cases. For low latency multipliers, tree architectures would be more suitable. The advantages of using single-phase clocking would be appreciable for longer word length multipliers, where clock distribution over a larger area is required.

ACKNOWLEDGMENT

The authors wish to thank T. Krishna for his design effort [20], which laid the foundation for this work, and S. Balakrishnan for his help throughout the course of this work.

REFERENCES

- [1] V. Visvanathan, N. Mohanty, and S. Ramanathan, "An area-efficient systolic architecture for real-time VLSI finite response filters," in *Proc. 6th Int. Conf. VLSI Design*, Bombay, India, Jan. 1993, pp. 166-171.
- [2] N. Mohanty, S. Ramanathan, and V. Visvanathan, "A new systolic architecture for real time VLSI infinite impulse response filters," in *Signal Processing and Communications*. New Delhi, India: Tata McGraw-Hill, 1993, pp. 190-194.
- [3] E. E. Swartzlander, Jr. Ed., *Computer Design Development: Principal Papers*. Roselle Park, NJ: Hayden, 1976.
- [4] C. S. Wallace, "A suggestion for a fast multiplier," *IEEE Trans. Electron. Comput.*, vol. 13, pp. 14-17, Feb. 1964.
- [5] P. R. Capello and K. Steiglitz, "A VLSI layout for a pipelined Dadda multiplier," *ACM Trans. Comput. Syst.*, vol. 1, no. 2, pp. 157-174, May 1983.
- [6] P. J. Song and G. De Micheli, "Circuit and architecture trade-offs for high-speed multiplication," *IEEE J. Solid-State Circuits*, vol. 26, pp. 1184-1198, Sept. 1991.

- [7] T. G. Noll, D. Schmitt-Landsiedel, H. Klar, and G. Enders, "A pipelined 330-MHz Multiplier," *IEEE J. Solid-State Circuits*, vol. 21, pp. 411-416, June 1986.
- [8] M. Hatamian, and G. Cash, "A 70-MHz 8-bit \times 8-bit parallel pipelined multiplier in 2.5 μ m CMOS," *IEEE J. Solid-State Circuits*, vol. 21, pp. 505-513, Aug. 1986.
- [9] F. Lu and H. Samuelli, "A 140-MHz CMOS bit-level pipelined multiplier-accumulator using a new dynamic full-adder cell design," in *Proc. 1990 Symp. VLSI Circuits*, 1990, pp. 123-124.
- [10] Y. Jiren, I. Karlsson, and G. Svensson, "A true single-phase-clock dynamic CMOS circuit technique," *IEEE J. Solid-State Circuits*, vol. 22, pp. 899-901, Oct. 1987.
- [11] Y. Jiren and C. Svensson, "High-speed CMOS circuit technique," *IEEE J. Solid-State Circuits*, vol. 24, pp. 62-70, 1989.
- [12] C. R. Baugh and B. A. Wooley, "A two's complement parallel array multiplier algorithm," *IEEE Trans. Comput.*, vol. C-22, pp. 1045-1047, Dec. 1973.
- [13] D. Somasekhar and V. Visvanathan, "A 230 MHz half bit level pipelined multiplier using true single phase clocking," in *Proc. 6th Int. Conf. VLSI Design*, Bombay, India, Jan. 1993, pp. 347-350.
- [14] P. Van der Wolf *et al.*, "An introduction to the NELSIS IC design system," Delft Univ. Technol., Delft, The Netherlands, Aug. 1990.
- [15] N. Weste and K. Eshragian, *Principles of CMOS VLSI Design. A Systems Perspective*. Reading, MA: Addison-Wesley, 1985.
- [16] S. M. Kang, "Accurate simulation of power dissipation in VLSI circuits," *IEEE J. Solid-State Circuits*, vol. 21, pp. 889-891, Oct. 1986.
- [17] M. Ishibe, S. Otake, J. Takeda, S. Tanaka, Y. Toyoshima, S. Takatsuka, and S. Shimizu, "High-speed CMOS I/O buffer circuits," *IEEE J. Solid-State Circuits*, vol. 27, Apr. 1992.
- [18] W. S. Song and L. A. Glasser, "Power Distribution Techniques for VLSI Circuits," *IEEE J. Solid-State Circuits*, vol. 21, Feb. 1986.
- [19] D. Somasekhar, "Asynchronous and synchronous pipelined multipliers," M.E. thesis, Dept. Electr. Commun. Engineering, Indian Instit. Sci., Bangalore, 1992.
- [20] T. Krishna, "Bit systolic design for high speed VLSI architectures," M.E. thesis Dept. Electr. Commun. Eng., Indian Instit. Sci., Bangalore, 1991.



Dinesh Somasekhar received the B.E. degree in electronics from the Maharaja Sayajirao University, Baroda, India, and the M.E. degree in electrical communication engineering from the Indian Institute of Science, Bangalore, India.

He is currently with Texas Instruments India, Bangalore.



V. Visvanathan received the B. Tech. degree in electrical engineering from the Indian Institute of Technology, New Delhi, India, the M.S.E.E. degree from the University of Notre Dame, Notre Dame, IN, and the Ph.D. degree in electrical engineering and computer sciences from the University of California, Berkeley.

He is currently a faculty member with the Division of Electrical Sciences, Indian Institute of Science, Bangalore, India. Previously, he was a distinguished member of technical staff at AT&T Bell Laboratories, Murray Hill, NJ. His research interests include analog design automation and VLSI signal processing.