Implementation of a Self-Resetting CMOS 64-Bit Parallel Adder with Enhanced Testability

Wei Hwang, *Senior Member, IEEE,* George Diedrich Gristede, Pia Sanda, Shao Y. Wang, and David F. Heidel, *Member, IEEE*

***Abstract*—This paper presents a fast, low-power, binary carry-lookahead, 64-bit dynamic parallel adder architecture for high-frequency microprocessors. The adder core is composed of evaluate circuits and feedback reset chains implemented by self-resetting CMOS (SRCMOS) circuits with enhanced testability. A new tool, SRCMOS pulse analyzer (SPA), is developed for checking dynamic circuits for proper operation and performance. The nominal propagation delay and power dissipation of the adder were measured to be 1.5 ns (at 22 °C with $V_{dd} = 2.5$ V) and 300 mW. The adder core size is 1.6 × 0.275 mm^2. The process technology used was the 0.5-μm IBM CMOS5X technology with 0.25-μm effective channel length and five layers of metal. The circuit techniques are easily migratable to multigigahertz microprocessor designs.**

***Index Terms*— Adder, CMOS, CMOS digital integrated circuits, dynamic logic circuits, high-speed integrated circuits, integrated circuit design, microprocessors, self-resetting CMOS (SRCMOS).**

I. Introduction

A NUMBER of adder architectures recently have been implemented [1]–[11]. Most of these designs used conventional static CMOS circuit techniques. As performance targets become more aggressive, dynamic and mixed static-dynamic circuit techniques are increasing in popularity. The dynamic circuit techniques contain a number of basic features.

1) Dynamic circuits perform logic operations by conditionally precharging nodes.
2) All dynamic circuits need clock signals or local timing chains to precharge nodes.
3) The logic operation needs two subcycles to complete (precharge and evaluation).
4) Dynamic circuits have faster switching speed and less area than equivalent static CMOS implementations.
5) Dynamic circuits are more sensitive to layout geometries and noise (charge leakage and sharing, etc.).

Dynamic circuits can be implemented in many different ways, some of which are presented here. A first approach is the conventional domino CMOS circuit, shown in Fig. 1(a). It consists of a pulldown nMOS logic network, a precharge pMOS transistor, an evaluation nMOS transistor, and a static inverter with lever restorer (a half-latch circuit). The logic operation of this circuit can be divided into precharge and evaluation phases, which are controlled by a synchronous global clock signal. The domino circuit has the following major drawbacks.

1) The global clock distribution presents timing, power, and layout problems.
2) The noninverting property of the domino logic style make its usage cumbersome.

A second approach is the conventional NORA CMOS circuit shown in Fig. 1(b). This circuit differs from the domino circuit in that the output inverter of the domino circuit has been replaced by a true single-phase clocked (TSPC) half-latch. The logic operation of this circuit is also divided into precharge and evaluation phases, which are controlled by a synchronous global clock signal. The NORA circuit has the following major drawbacks.

1) The global clock distribution presents timing, power, and layout problems.
2) A slow clock slope can lead to conductance overlap between nMOS and pMOS devices resulting in dc power dissipation.
3) The high output impedance of the TSPC half-latch leads to high sensitivity to noise and leakage.

In this paper, we present a third approach of asynchronous dynamic logic and circuit style. It is a new custom self-resetting CMOS (SRCMOS) circuit technique, as shown in Fig. 1(c). The advantages of SRCMOS circuit technique are:

1) no global clock and associated distribution problems;
2) separate evaluate and reset (precharge) paths;
3) self-reset by local timing chain;
4) fast cycle time and minimum delay;
5) pulse pipelined operation;
6) good testability.

In Section II, discussions of SRCMOS logic circuits and a SRCMOS binary carry-lookahead 8-bit adder architecture are presented. In Section III, the implementation of the dynamic binary carry-lookahead 64-bit parallel adder is described in detail with emphasis on the SRCMOS circuit design techniques. In Section IV, a description of the SRCMOS pulse analyzer (SPA) tool, which automates the design verification of SRCMOS circuits, is given. The fabricated adder and measurement results are presented in Section V. The summary and conclusion of this paper are given in Section VI.

Manuscript received July 1, 1998; revised April 8, 1999

W. Hwang, G. D. Gristede, P. Sanda, and D. F. Heidel are with the IBM T. J. Watson Research Center, Yorktown Heights, NY 10598 USA.

S. Y. Wang is with Columbia University, New York, NY 10027 USA.

Publisher Item Identifier S 0018-9200(99)06166-1.

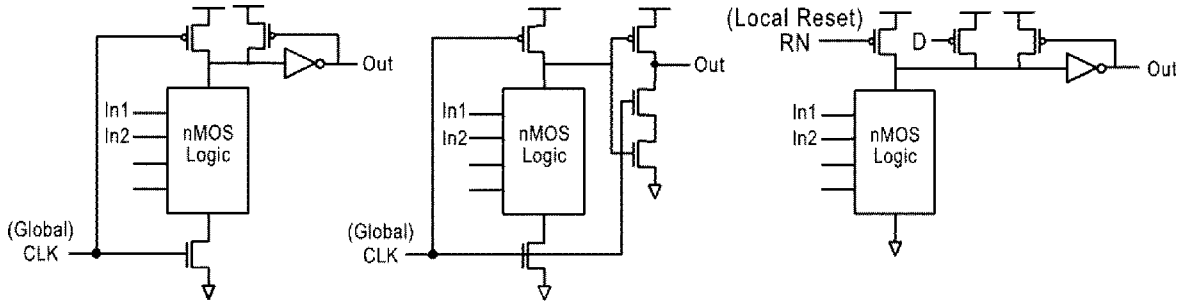

Fig. 1. Various dynamic circuit styles.

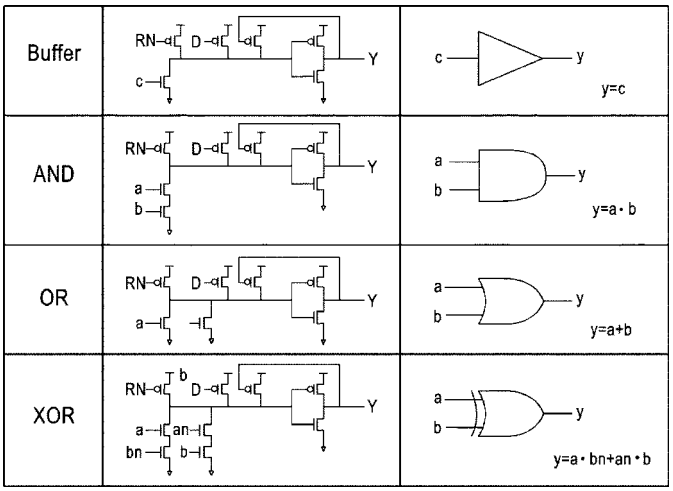

Fig. 2. Basic SRCMOS logic gates.

II. Self-Resetting CMOS Logic Circuits and Binary Adder Architecture

Asynchronous SRCMOS logic circuits operate by conditionally discharging dynamic storage nodes to evaluate the desired logic function and then resetting these nodes back to their original charged state by a local feedback timing chain instead of a global clock [12]–[17]. Fig. 2 shows the most basic SRCMOS logic gates (buffer, OR, AND, and XOR). Each gate consists of an n-logic tree, half-latch circuits, reset devices, and a diagnostic (static_evaluate) weak pMOS device. The n-logic tree is a parallel/series network of nMOS devices between ground and the inputs to the output inverters (the dynamic storage nodes). For logic gates with multiple outputs, the n-logic tree can be shared between the inputs to the output inverters. The half-latch circuit consists of a pMOS device with its gate connected to the output of the inverter and its drain connected to the top of the n-logic tree. This device is used to improve the standby noise margin of the circuit. The gate of the reset device is connected to the reset signal RN. This signal RN is generated by reset trigger circuitry that senses the evaluation of the logic gate and generates a reset signal that passes through a local feedback timing chain back to the logic gates that triggered it (discussed in the next section). It is also possible to have an n-channel reset device pulling the inverter output to ground as well with its gate connected to the complement of R (RN). This allows the output inverter(s)

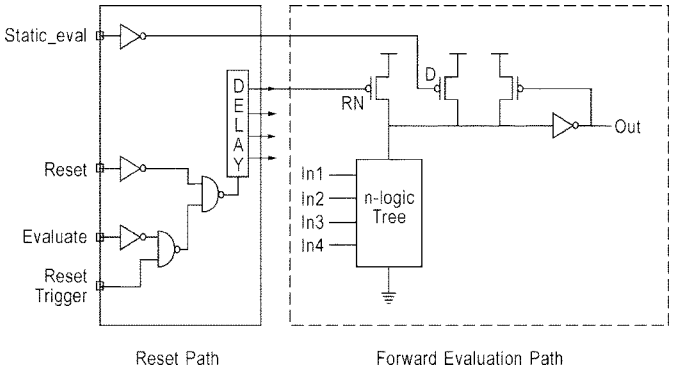

Fig. 3. SRCMOS circuit with test signals.

of the logic gate to be skewed for higher performance. The enhanced testability of the SRCMOS logic gates comes from the addition of a diagnostic device (D) with a gate connected to the global testing Static_eval signal. When Static_eval is active, the SRCMOS gates convert from precharged logic gates to static ratioed logic gates, enabling testability.

The generic self-resetting element has three states of operation: standby, evaluation, and reset. The standby state is achieved when power is first applied to the circuit or after the circuit has reset. The standby state is defined as all nodes in the circuit being in their inactive state. This includes the logic inputs being low, the tops of logic trees being high, and the reset signal R (RN) being high (low). The evaluation state starts when a combination of active high inputs creates a conduction path between the top of an nMOS logic tree node and ground. This forces the node at the top of the logic tree to its active low state and switches the output inverter to create an active high output signal. The active high output causes the reset trigger circuitry to switch to its active low state, which in turn triggers the start of the reset timing chain, (The operation of this reset timing chain will be described in the next section.) The reset state starts with R (RN) going active high (low) and the inputs to the logic tree returning to their inactive low state. Once all the nodes have been returned to their inactive, standby states, the circuit is ready to repeat the process of evaluation and reset.

The test modes have been dealt with up front by applying the SRCMOS diagnostic testing methodology. The typical SRCMOS circuit, which includes the test modes and test signals, is presented in Fig. 3. Two global test signals or

TABLE I
TEST MODES AND SIGNALS

Mode \ Signal	Reset	Evaluate	Static_eval
Normal Functional	Low	Low	Low
Reset	High	Low	Low
Evaluate	Low	High	Low
Static_eval	Low	High	active -High

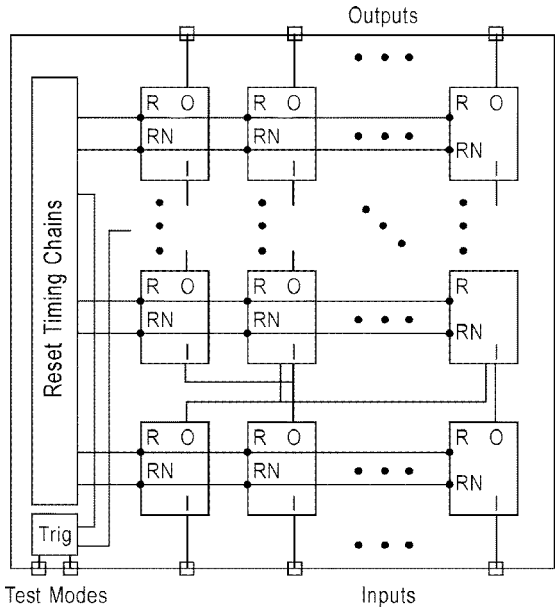

Fig. 4. SRCMOS macro blocks.

modes, "Reset" and "Evaluate," have been defined to force circuits to their reset states or to inhibit resets, thereby keeping circuits in their evaluated states. Also, another global test signal or mode "Static_eval" has been defined, in which all circuits behave as static, instead of pulsed logic. Circuits can then be tested under burn-in conditions in Static_eval mode rather than in high-speed functional mode. Since in Static_eval mode all signals behave as static, so that glitches, other timing faults, and noise eventually die out, noise margin is of no concern, and circuits do not need to be specially engineered for burn-in conditions. The overall test modes and the states of the global test signals are given in Table I, where "Low" stands for ground and "High" stands for $V_{\rm dd}$ [13].

In SRCMOS, there is a natural hierarchy level—the macro level—above the gate level. An SRCMOS macro includes a number of SRCMOS gates or blocks, and a local reset generation as shown in Fig. 4. The reset timing chain generator is triggered by one or more pulsed signals in the macro. A quasi-global approach is used to reset pulse generation and timing control circuits to reduce area overhead. Triggering options including simple ORing of single-rail signals and

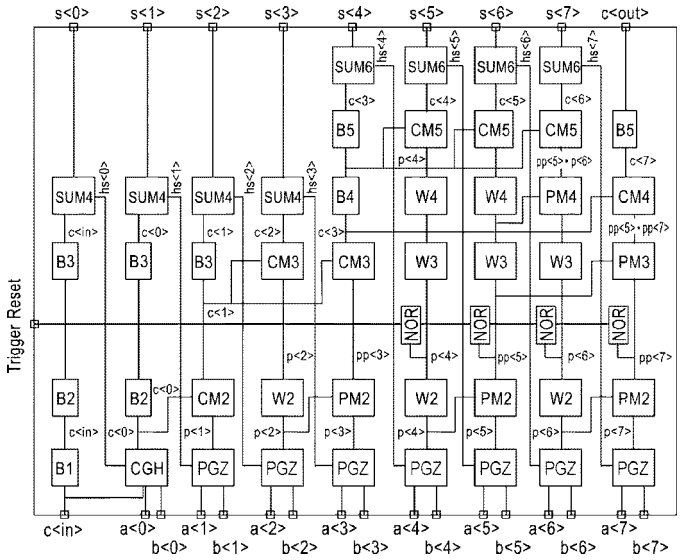

Fig. 5. Block diagram of 8-bit adder.

interlocking of signals from multiple paths. In any case, the reset timing chain generator must be triggered whenever the macro is activated. A delay reset timing chain, started by the trigger circuit, generates the required reset signals. One of these signals resets the trigger, so that the reset signals are self-terminating in normal operation. The reset generator may also include static inputs to force all reset signals on or off for test modes.

For the sake of simplicity, an 8-bit SRCMOS binary carry-lookahead-adder architecture will be introduced first. The block diagram of an 8-bit adder is shown in Fig. 5. All the building blocks described in this example are the basic circuits of any adder size [18]–[20].

The logic architecture for the 8-bit adder consists of an array of SRCMOS blocks that implement a carry lookahead with two-way merging that computes the carry into each bit position. The carry out of the most significant bit (MSB) and final sum of each bit position are needed to create the final result.

The adder consists of several basic building blocks implemented in SRCMOS circuits. The forward evaluation path of the adder core consists of a combination of a propagate/generate/zero generator circuit and a half-sum circuit, which is called a PGZ block, a PGZ buffer circuit (W), a PGZ merge circuit (PM), a carry-generator circuit and half-sum circuit (CGH), a carry-buffer circuit (B), a carry-merge circuit (CM), and a full-sum circuit (SUM). In addition to these basic building blocks, the adder consists of a feedback reset chain and includes as its building block a NOR trigger circuit, test mode circuit, and reset timing chain circuit.

The 8-bit self-resetting parallel binary adder, as shown in Fig. 5, consists of an array of SRCMOS circuits arranged in six rows of evaluation logic and one row of reset logic. The first row of evaluation logic consists of a carry-buffer circuit (B1), three-input carry-generator circuit (CGH), PGZ generator (PGZ) circuits receiving two binary inputs, denoted as a⟨0⟩ to a⟨7⟩ and b⟨0⟩ to b⟨7⟩, and carry input c⟨in⟩. The ⟨i⟩ represents the ith bit position. Here, the bit position 0 is the

least significant bit (LSB) and the bit position 7 is the MSB. The inputs are dual rails, for which both true and complement inputs a⟨i⟩/an⟨i⟩, b⟨i⟩/bn⟨i⟩, and c⟨i⟩/cn⟨i⟩ are provided.

The second row of evaluation logic consists of buffer and merge circuits, with the buffer circuits being inserted in the evaluation paths to maintain synchronization of pulses propagating in the adder pulse pipelined logic architecture. Here, the second row, from left to right, consists of carry-buffer circuits (B2), respectively connected in series with B1 and CGH circuits. A carry merge (CM2) is connected with CGH and PGZ circuits. A PGZ buffer (W2) circuit and a PGZ merge (PM) circuit are alternately connected to PGZ circuits and also respectively receive input signals, p⟨1⟩ to p⟨7⟩, from PGZ circuits. The PM2 circuit combines the orthogonal input signal coming from the merge tree to the right with the orthogonal input signal coming from the merge tree below to generate the orthogonal output signal pp⟨j⟩, which flows into the final carry block above. It should be noted that p⟨i⟩, for simplicity, represents a tri-rail group of signals of p⟨i⟩/g⟨i⟩/z⟨i⟩ at the ith bit position, and pp⟨j⟩ represents a tri-rail group of signals of pp⟨j⟩/gg⟨j⟩/zz⟨j⟩ at the jth bit position.

The third row of evaluation logic consists of carry buffer circuits (B3), respectively connected to B2 and CM2 circuits. The CM3 and W3 circuits, respectively connected to W2 and PM2 circuits. And the PM3 connected to two PM2 circuits. The CM3⟨2⟩ circuit receives orthogonal input from c⟨1⟩ and p⟨2⟩ to generate the orthogonal carry output signal c⟨2⟩ flow into final sum block above. Similarly, the CM3⟨3⟩ circuit receives input from c⟨1⟩ and pp⟨3⟩ to generate the orthogonal carry output signal c⟨3⟩ flow into the carry buffer block above. The PM3 circuit receives orthogonal tri-rail inputs from pp⟨5⟩ and pp⟨7⟩ and generate receives orthogonal tri-rail outputs pp⟨5⟩pp⟨7⟩.

The fourth row of evaluation logic consists of full-sum (SUM4) circuits B4, W4, PM4, and CM4. The SUM4 circuits are respectively connected to B3, CM3, and PGZ circuits and also respectively receive dual-rail input signals, c⟨in⟩, c⟨0⟩ to c⟨3⟩, and hs⟨0⟩ to hs⟨3⟩, from B3, CM3, and PGZ circuits. The SUM4 outputs are s⟨0⟩ to s⟨3⟩, where s⟨i⟩ represents dual-rail signal s⟨i⟩/sn⟨i⟩. The PM4 circuit combines the orthogonal input signal pp⟨5⟩ coming from the merge tree to the right with the orthogonal input signal p⟨6⟩ coming from the merge tree below to generate the orthogonal output signal pp⟨5⟩p⟨6⟩, which then flow into the CM5 block above. The CM4 circuit receives orthogonal input from c⟨3⟩ and pp⟨5⟩pp⟨7⟩ to generate the orthogonal carry output signal c⟨7⟩ flow into carry buffer B5 block above.

The fifth row of evaluation logic consists of B5 and CM5 circuits. The B5 and CM5 circuits are, respectively, connected to B4, W4, PM4, and CM4. B5 circuits also respectively receive dual-rail input signals, c⟨3⟩ and c⟨7⟩. The CM5 circuits receive orthogonal inputs respectively from c⟨3⟩ and p⟨4⟩, c⟨3⟩, and pp⟨5⟩, and c⟨3⟩ and pp⟨5⟩p⟨6⟩ to generate the orthogonal carry output signals, c⟨4⟩, c⟨5⟩, and c⟨6⟩, flow into final SUM6 blocks above.

The sixth row of evaluation logic consists of full-sum (SUM6) circuits. The SUM6 circuits are, respectively, connected to B5, CM5, and PGZ circuits. That combines the carry bits with half-sum bits to produce the final dual-rail sum outputs, denoted as s⟨4⟩ to s⟨7⟩, while the carry out, denoted c⟨out⟩, is the output from the B5 circuit.

TABLE II
BOOLEAN FUNCTIONS OF BUILDING BLOCKS OF ADDER CORE

Circuit Type	Figure Number	Boolean logic function (n-logic tree implementation)
p/g/z generate (PGZ)	Fig. 6	p<i> = a<i>bn<i> + an<i>b<i> g<i> = a<i>b<i> z<i> = an<i>bn<i> hs<i> = p<i> hsn<i> = g<i> + z<i>
Carry generate (CGH) Circuit	Fig. 7	cy<i> = a<i>b<i> + a<i>c<i> + b<i>c<i> cyn<i> = an<i>b<i> + an<i>cn<i> + bn<i>cn<i>
p/g/z merge (PM) Circuit	Fig. 8	pp<j> = p<i>p<j> gg<j> = g<i>p<j> + g<j> zz<j> = z<i>p<j> + z<j>
Carry merge (CM) Circuit	Fig. 9	cy<i> = p<i>c<i> + g<i> cyn<i> = p<i>cn<i> + z<i>
Full Sum (SUM) Circuit	Fig. 10	s<i> = c<i-1>hsn<i> + cn<i-1>hs<i> sn<i> = c<i-1>hs<i> + cn<i-1>hsn<i>

The self-resetting path portion of the adder core is triggered by the NOR circuit. The NOR circuit is designed to control each input driving NMOS devices and is connected to tri-rail PGZ signals. When one of these inputs is activated, the NOR output will be driven low to trigger the reset timing chain. The adder in Fig. 5 shows a row of NOR circuits with their inputs connected to the second row of the outputs of the evaluation circuits, p⟨4⟩, pp⟨5⟩, p⟨6⟩, and pp⟨7⟩, and all their outputs connected together.

Next, we will present several examples of SRCMOS adder circuits in the transistor-level implementation. All of these circuits are identical with respect to the reset and testing devices but differ in the Boolean logic functions implemented by the n-logic tree. The Boolean logic functions of the basic building blocks of the adder core are summarized in Table II. The reset timing chain is generated from the feedback reset chain block of the adder core. Fig. 6 shows the transistor-level implementation of the propagate, generate, and zero and half-sum combined circuit PGZ. This circuit has inputs a⟨i⟩, an⟨i⟩, b⟨i⟩, and bn⟨i⟩, which are connected to the gates of nMOS devices in a logic tree that implements SRCMOS two-way XOR and AND logic gates to generate the p⟨i⟩, g⟨i⟩, and z⟨i⟩ signals. The p⟨i⟩ signal is buffered to generated the hs⟨i⟩ signal, and the g⟨i⟩ and z⟨i⟩ signals combine in an SRCMOS OR gate to generate the hsn⟨i⟩ signal and develop additional drive. Notice the presence of additional n-channel reset devices controlled by R1 to reset the p⟨i⟩, g⟨i⟩, and z⟨i⟩ outputs to ground. These reset devices allow for the driving inverters to be skewed. The strength of the half-latch is tuned to be one-

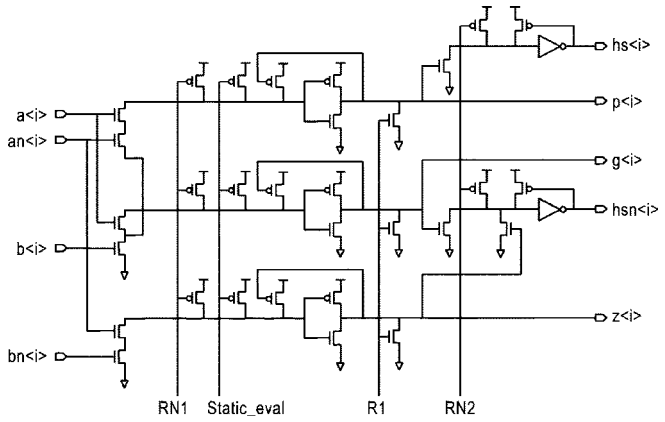

Fig. 6. SRCMOS PGZ and half-sum PGZ circuit.

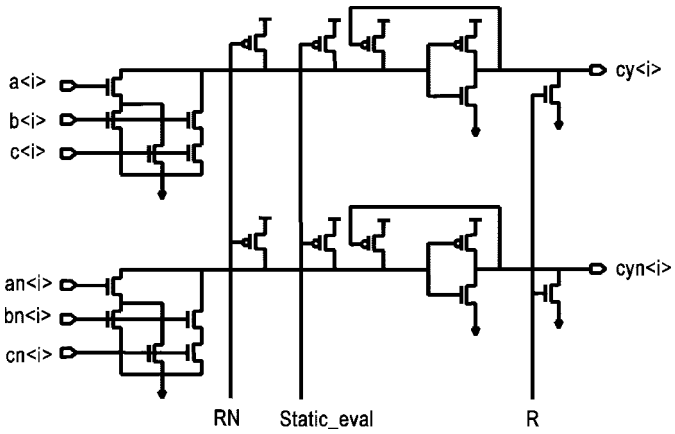

Fig. 7. SRCMOS carry-generate circuit.

tenth the strength of the top device(s) in the nMOS logic tree. The gate of the diagnostic pMOS device is connected to the global testing Static_eval signal. When Static_eval is active, the SRCMOS gates convert from precharged gates to ratioed static logic gates.

The transistor implementation of the carry-generate circuit is in Fig. 7. In the top half of the circuit, the three inputs c⟨in⟩ (carry in), a⟨i⟩, and b⟨i⟩ (operand in) combine in the n-channel logic tree to generate the cy⟨i⟩ (carry out) signal. The bottom half of the circuit generates the cyn⟨i⟩ (carry out complement) signal in the same way from cn⟨in⟩ (carry in complement), an⟨i⟩ and bn⟨i⟩ (operand in complements).

Fig. 8 shows the transistor implementation of the PGZ merge circuit. This circuit combines the orthogonal p⟨i⟩/g⟨i⟩/z⟨i⟩ input tri-rail signals from the merge tree to the right with the orthogonal p⟨j⟩/g⟨j⟩/z⟨j⟩ input tri-rail signals coming from the merge tree below to generate the group orthogonal pp⟨j⟩/gg⟨j⟩/zz⟨j⟩ output signals, which flow into the final carry block above. The output pp⟨j⟩ branch is generated from a two-way SRCMOS AND gate. The output's gg⟨j⟩ and zz⟨j⟩ branches are generated from three-way SRCMOS area of interest (AOI) gates, respectively.

Fig. 9 shows the transistor implementation of the carry merge circuit to generate true and complement carry outputs, cy⟨i⟩ and cyn⟨i⟩. The true carry output cy⟨i⟩ branch consists of a three-way SRCMOS AOI gate, which receives three inputs, p⟨i⟩, c⟨i⟩, and g⟨i⟩. Similarly, the complement carry output

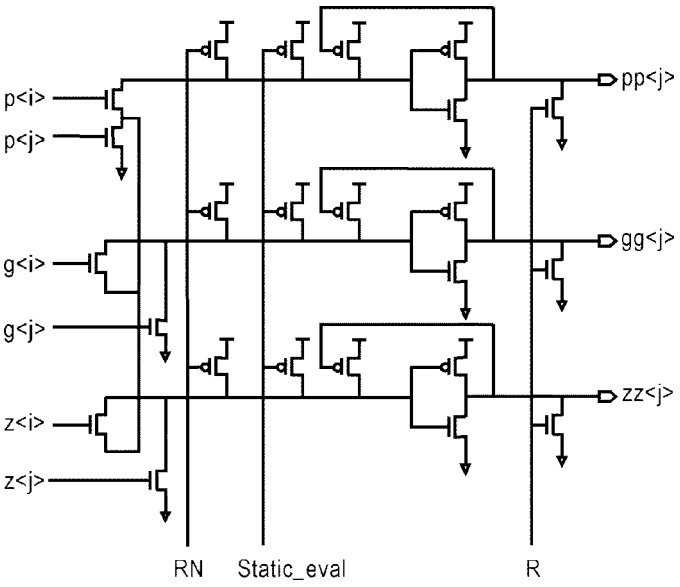

Fig. 8. SRCMOS PGZ merge circuit.

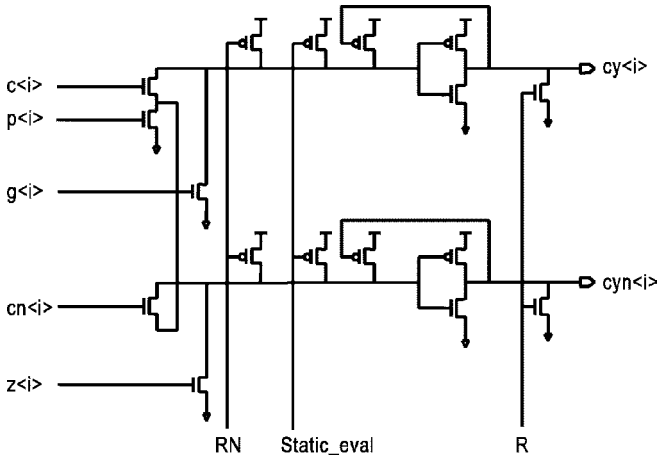

Fig. 9. SRCMOS carry-merge circuit.

cyn⟨i⟩ branch consists of a three-way SRCMOS AOI gate, which receives three inputs, p⟨i⟩, cn⟨i⟩, and z⟨i⟩.

Fig. 10 shows a circuit schematic of the dual-rail full-sum circuit. Either SUM4 or SUM6 circuit represents the last rows 4 or 6 of the adder. The binary inputs are c⟨i − 1⟩, cyn⟨i − 1⟩, hs⟨i⟩ and hsn⟨i⟩ and the sum outputs are s⟨i⟩ and sb⟨i⟩.

III. Dynamic 64-Bit Parallel Adder Implementation

The 64-bit adder logic architecture is a carry-lookahead architecture with 2-bit group building blocks and thus requires six rows of merge logic to calculate the carry out of bit 0 (the MSB). The adder utilizes a pulse pipelined circuit architecture, which is implemented with SRCMOS circuits. Pulse pipelining is like wave pipelining except that a fully formed pulse data is propagated through the circuitry instead of just a single transition. A fast cycle time and minimum delay for each macro are achieved by using fast forward amplification of the leading edge of the pulse input signals followed by quick self-resetting of all dynamic nodes back to their standby state. The

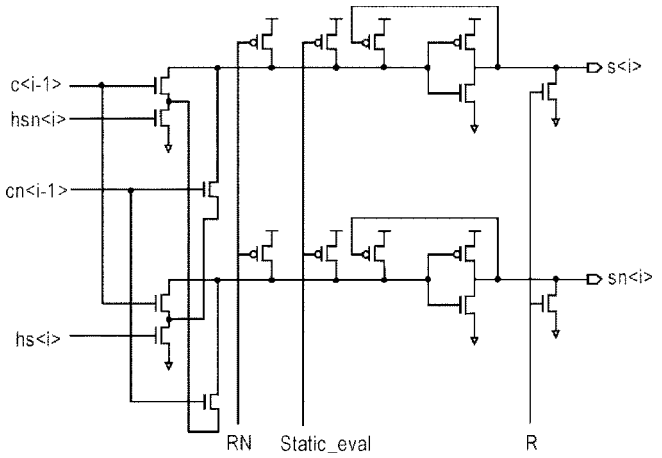

Fig. 10. SRCMOS dual-rail full-sum circuit.

forward amplification uses large devices in the critical path along with smaller devices for holding the standby state. The self-reset phase uses large devices, which subsequently turn off before a new cycle begins. When the circuit cycle time is too long, the reset circuits can be broken up into additional self-resetting pipeline macros [12]–[17]. Thus, the unique SRCMOS circuits make possible pulse-pipelined operation and good testability.

The block diagram of the carry-lookahead 64-bit adder is shown in Fig. 11, which consists of two different blocks. One is the forward evaluation path block (labeled as adder core) and the other is the reset-generation path block (RS1 and RS2, respectively). The forward evaluation path consists of many sets of resettable dynamic nodes. These node sets are reset by a block of reset pulse generation circuits, which generate a sequence of reset signals. The detailed design of the evaluation and reset path blocks is presented as follows.

The 64-bit-adder forward evaluation block is partitioned into eight rows of evaluation logic (rows 1–8) and two rows of reset-trigger logic (RR and RR8 shared with row 8), as shown in the block diagram of Fig. 11. The eight rows of evaluation consist of a first row of PGZ circuits, six rows of merge/carry logic (PM, CM, or buffer blocks) used in bit position $\langle 43\text{-}0\rangle$, five rows of merge/carry logic used in bit position $\langle 54\text{-}44\rangle$, four rows of merge/carry logic used in bit position $\langle 59\text{-}55\rangle$, two rows of merge/carry logic used in bit position $\langle 63\text{-}60\rangle$, and the final row of sum circuits. It should be noted that bit position $\langle 0\rangle$ is the MSB, and bit position $\langle 63\rangle$ is the LSB. The RR row of reset-trigger logic consists of identical three-input NOR circuits (RR) used in bit position $\langle 59\text{-}16\rangle$ and dotted together to form a larger NOR to generate the reset-triggering DONE signal. Row RR8 of reset NOR's is integrated into the SUM8 circuit block used in bit positions $\langle 43\text{-}0\rangle$ They are also dotted together to form a larger NOR to generate the reset-triggering DONE8 signal. The other major block for the adder is the reset-generation path block. This block consists of two subblocks, RS1 and RS2, which are triggered by the DONE and DONE8 signals, respectively. The dynamic outputs of the reset blocks are the reset pulses R1–RN7 and R8–RNMUX, respectively, which reset the dynamic node sets in the forward evaluation path.

The first seven rows of the adder, with the exception of the last row, are reset with the timing chain generated from the RS1 block. A simplified illustration of the circuit showing the interaction between the forward and reset paths is shown in Fig. 12. The uppermost chain of circuits represents the forward circuitry performing the logic. Only four of the seven rows of circuitry are shown here due to length limitations. For simplicity, the pulldown trees are represented by a single transistor. An input pulse is shown to arrive into the input. This represents two inputs from the RA and RB registers. Below the forward path is the timing chain of inverters forming the reset path. The trigger devices that pull down the "done" line (labeled "done" signal) are part of the forward path evaluation but are shown in the reset loop to illustrate the transistors which control the reset of the dynamic nodes, reset of the reset circuitry and the width of the reset pulse. The output pulse width of row 7 grows but is less than 1500 ps. This allows for good pulse coalescence, but it is desirable to have a shorter pulse width at the macro boundaries. Thus, separate reset circuity (RS2) is used to reset the SUM8 blocks.

Fig. 13 is a circuit schematic of the dual-rail full-sum (SUM8) circuit. This SUM8 circuit represents the last row (row 8) of the adder. The binary inputs are cy$\langle$i$\rangle$, cyn$\langle$i$\rangle$, hs$\langle$i$\rangle$, and hsn$\langle$id$\rangle$; and the sum output is s$\langle$i$\rangle$. The circuit also implements an XOR function of the binary inputs. However, this circuit is different from Fig. 10 in that SUM8 has a flip-flop that holds the signal arriving at input hs$\langle$i$\rangle$ from the PGZ circuit until it is ready to be evaluated with the cy$\langle$i$\rangle$ carry input. The trigger reset circuit is formed by a two-way NOR circuit, which receives signals as its inputs from the true and complement sum output signals. The output of NOR generates a DONE8 signal to trigger a reset timing chain (RS2 circuit), as shown in Fig. 14. Also, the reset signals RNFF, RN8, and R8 received by the SUM8 circuit are generated by the RS2 circuit in Fig. 14.

Fig. 12 also shows how the "Power-On/Reset" and "Evaluate" functions are added to the basic circuit. The Evaluate signal inhibits the reset. This block is defined as an "Inhibit Reset" block. If the reset is not inhibited, the reset signal is passed to the inverter circuit as well as returned over a feedback path as a "reset-reset" signal to the reset device. This feedback signal can be inhibited by an active Power-On/Reset so that the reset state can be maintained for testing purposes. Otherwise, the reset trigger is reset to its standby state to await another evaluation. The Power-On/Reset is implemented by being ANDed with the complement of the "reset" signal to hold the reset. This block is defined as a "Hold Reset State" block.

In the actual RS1 layout design, the inverters in the reset chain are arranged vertically so that their outputs are close to the reset lines they control. The layout is designed so that the inverters supplying signals—R1, RN1, R2, RN2, R3, and RN3—to rows 1, 2, and 3 are placed to the left of the corresponding rows. Rows 4 and 5 have their corresponding inverter signals—R4, RN4, R5, and RN5—placed above the first tier of SUM's. Rows 6 and 7 have their inverters—R6, RN6, R7, and RN7-above the second tier, and the inverters driving row 8 are above the third tier. While the SUM4 and

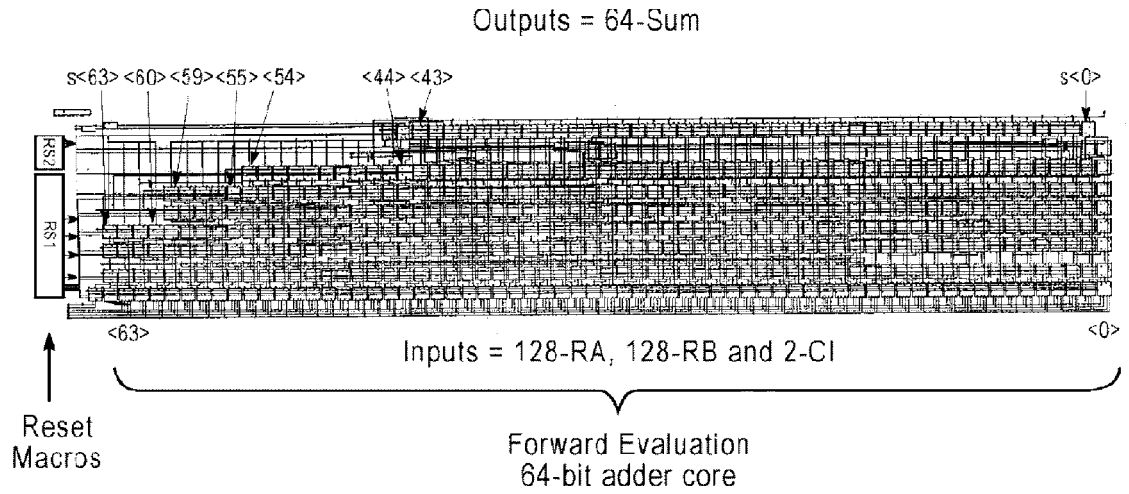

Fig. 11. Block diagram of 64-bit adder.

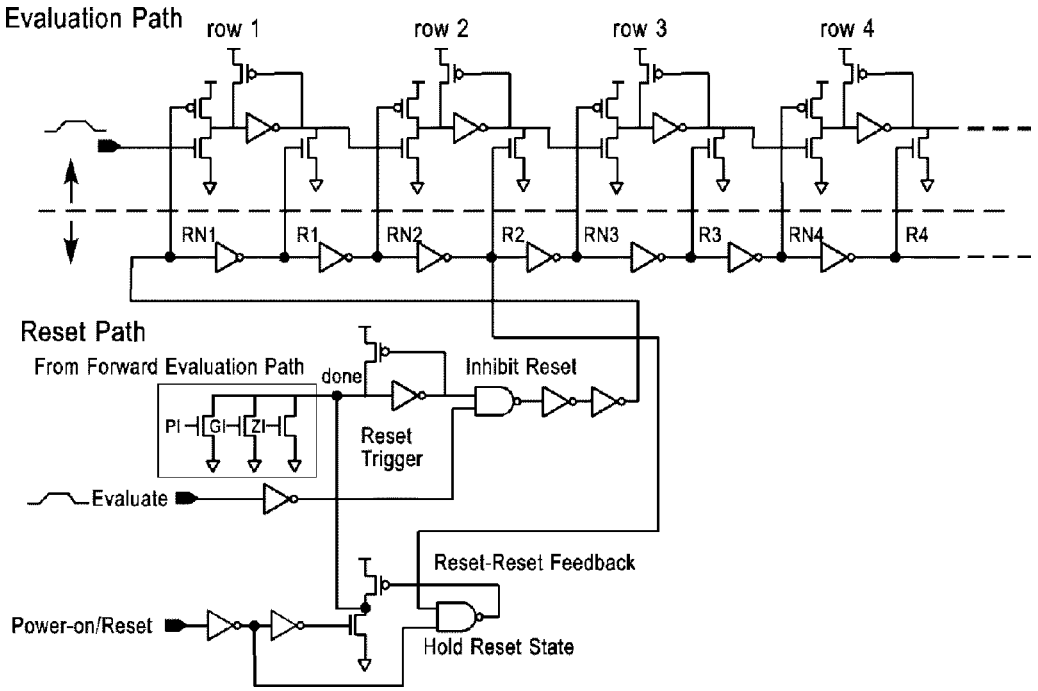

Fig. 12. Simplified illustration of forward evaluation and reset paths of RS1 block.

SUM6 reset signals are somewhat complex, they are still taken from the RS1 circuit, and the devices are sized accordingly.

The reset timing for the SUM8 block is different. The SUM8 blocks in row 8 are reset with a separate reset circuit, RS2, as shown in Fig. 14. The actual RS2 reset schematic is almost the same as Fig. 12. This reset schematic also consists of a reset trigger, inhibit reset block, reset-reset feedback path, hold reset state block, and inverter timing chains RNFF, RN8, R8, RN9, RNMUX, and RMUX. Again, the reset chains are arranged vertically so that their outputs are close to the reset lines they control.

IV. SRCMOS Pulse Analyzer—A CAD Tool

In general, circuit designers are required as part of their design methodology verification process to spend considerable time checking transistor sizes and analyzing simulation waveforms manually. Many challenging problems can occur in the design of custom SRCMOS circuits. Such problems include:

- inadequate pulse-width-to-delay ratios;
- conduction "collisions" between reset devices and n-logic trees;
- inadequate circuit cycle times for pipelined operation;
- charge sharing, coupling, leakage, power supply, and substrate bounce noise.

To verify the design and "safe" operation of SRCMOS circuits, a new computer-aided design (CAD) tool, SPA, has been developed. This tool combines a detailed analysis of circuit topology with circuit simulation data (using either SPICE or AS/X circuit simulator) to automate many of the design methodology verification checks required for the successful design of SRCMOS circuits. The program checks for inade-

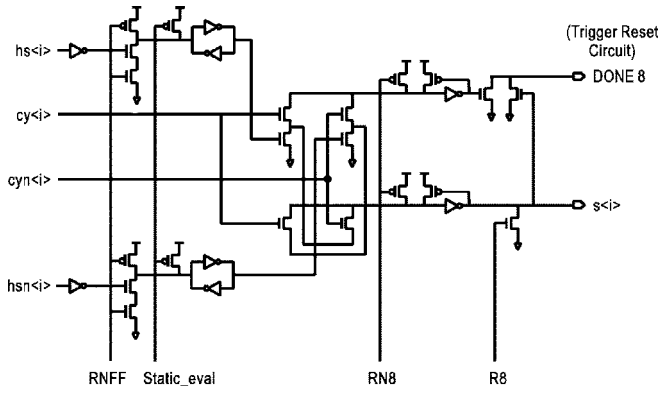

Fig. 13. SRCMOS full-sum circuit at row 8 (SUM8).

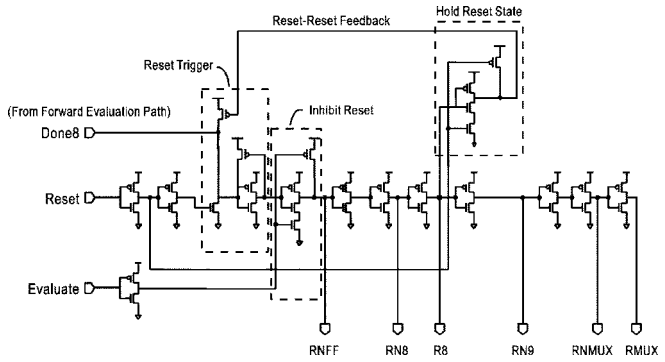

Fig. 14. Circuit schematic diagram of the reset timing chain of macro RS2.

TABLE III
TECHNOLOGY INFORMATION

Feature Size (μm)	0.50
Channel Length L_{eff} (μm)	0.25+0.08
Junction Depth X_j (μm)	0.18
Gate Oxide (nm)	7
V_{DD} (V)	2.5
V_{Tn} / V_{Tp} (V)	0.5/-0.55
MC Width/Pitch (μm)	0.5/1.0
M1 Width/Pitch (μm)	0.6/1.2
M2, M3, M4 Width/Pitch (μm)	0.9/1.8
M5 Width/Pitch (μm)	2.4/4.8

quate pulse-width-to-delay ratios, collisions, inadequate circuit cycle times, and all of the sources of noise listed above. The various parameters involved in the checks (e.g., pulse-width-to-delay ratios) can be set by the user. As an example, a minimum pulse-width-to-delay ratio of five and a maximum circuit cycle time of 2.3 ns were used for the 64-bit adder circuit.

V. EXPERIMENTAL RESULTS

An experimental carry-lookahead 64-bit parallel SRCMOS adder macro, designed for a 2.5-V, 0.5-μm CMOS technology with 0.25-μm effective channel length and five layers of metal [21], has been successfully fabricated and tested. The technology features are summarized in Table III. The full

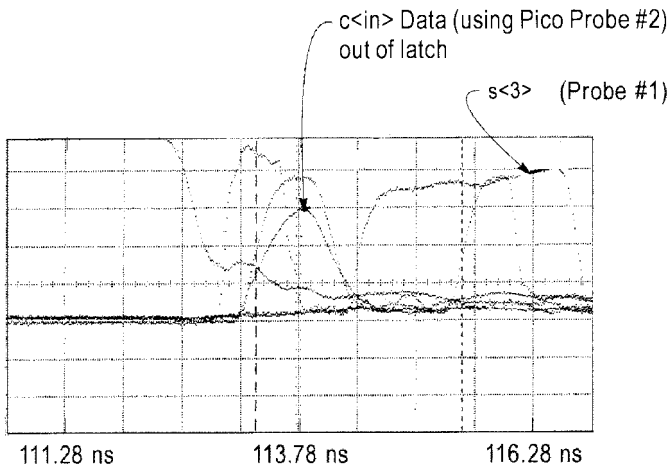

Fig. 15. Measured waveforms for input carry bit and selected outputs.

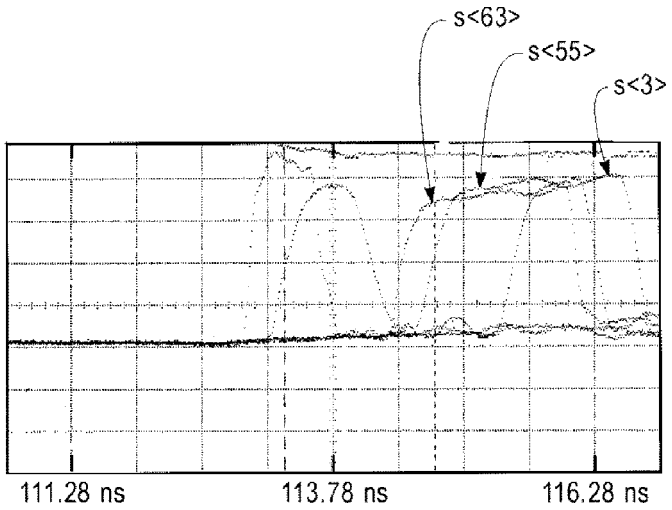

Fig. 16. Measured waveforms for input carry bit and selected outputs.

adder core occupies an area of 1.6 $\times$ 0.275 mm^2. Adder chips were measured on one wafer using a checkerboard pattern with a carry_in bit. The minimum operating cycle time was measured around 2.19 ns. These cycle-time results were obtained with $V_{\rm dd}$ set to 2.5 V; however, the actual on-chip voltage swing was approximately 2.3 V while operating at these speeds. The power dissipation was measured at 299 mW. The nominal propagation delay of the adder was measured to be 1.5 ns (at 22 °C with $V_{\rm dd} = 2.5$ V). Fig. 15 shows the picoprobe waveforms for the input carry bit pulse and the output pulses, s$\langle 63 \rangle$. Fig. 16 shows the picoprobe waveforms for the input carry bit pulse and the output pulses, s$\langle 63 \rangle$, s$\langle 55 \rangle$, and s$\langle 3 \rangle$. The delay from input latch_out to s$\langle 63 \rangle$, s$\langle 55 \rangle$, and s$\langle 3 \rangle$ are 0.81, 1.1, and 1.5 ns, respectively. Fig. 17 shows the experimental waveforms for the power supply $V_{\rm dd}$ and ground line "droop" in the adder. It is shown that the power and ground bouncing noises are kept minimum. The overall 64-bit SRCMOS adder characteristics are summarized in Table IV. The adder macro is being migrated to advanced 0.25-μm CMOS technology with 0.12-μm effective channel length and

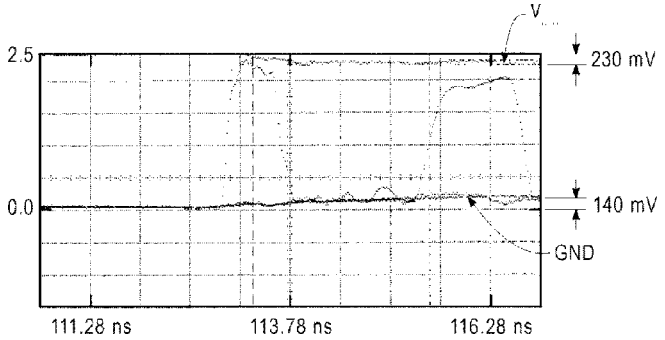

Fig. 17. Measured waveforms for power supply and ground lines.

TABLE IV
OVERALL CHARACTERISTICS OF 64-BIT SRCMOS ADDER

- Size : 1.6 mm X 0.275 mm
- Inputs : 128-RA, 128-RB, 2-CI
- Outputs : 64-SUM
- Delay
  - Latch_out to s<63> = 0.810 ns
  - Latch_out to s<55> = 1.100 ns
  - Latch_out to s<44> = 1.330 ns
  - Latch_out to s<3> = 1.500 ns
- Minimum Cycle Time : 2.25 ns (@ 2.5V)
- Power : 300 mW (@ 3.3 ns)

six layers of metal and is expected to have subnanosecond performance [22].

VI. CONCLUSIONS

A 64-bit dynamic adder has been successfully designed and implemented using self-resetting CMOS circuit techniques. It demonstrates a fast cycle time, minimum delay time, high-speed arithmetic (addition and subtraction) operation with less power dissipation, better area tradeoff, and improved testability. The timing, noise margins, and power can be easily adjusted by using the SPA tool in conjunction with a circuit simulator. Built-in testing circuits are incorporated within the resetting circuitry. The adder can be easily migrated to more advanced technologies. The circuit techniques and methodology presented here are extremely useful for the future of high-frequency microprocessor design.

ACKNOWLEDGMENT

The authors would like to thank T. Chappell, B. Chappell, M. Beaks, B. Fleischer, and T. Nguyen for their first-version adder design and contributions; V. Narayanan, S. Basavaiah, B. Fleischer, P. Cook, P. Emma, and L. Terman for their helpful discussions and comments; and J. Paraszczak, C. Anderson, and M. Rosenfield for their management support.

REFERENCES

[1] C. F. Webb, C. J. Anderson, L. Sigal, K. L. Shepard, J. S. Liptay, J. D. Warnock, B. Curran, B. W. Krumm, M. D. Mayo, P. J. Camporese, E. M. Schwarz, M. S. Farrell, P. J. Restle, R. M. Averill, T. J. Siegel, W. V. Huott, Y. H. Chan, B. Wile, P. G. Emma, D. K. Beece, C. T. Chuang, and C. Price, "A 400 MHz S/390 microprocessor," in *ISSCC Dig. Tech. Papers*, Feb. 1997, pp. 168–169.

[2] P. E. Gronowski, P. J. Bannon, M. S. Bertone, R. P. Blake-Campos, G. A. Bouchard, W. J. Bowhill, D. A. Carlson, R. W. Casteline, D. R. Donchin, R. M. Fromm, M. K. Gowan, A. K. Jain, B. J. Loughlin, S. Mehta, J. E. Meyer, R. O. Mueller, A. Olsesin, T. N. Pham, R. P. Preston, and P. I. Rubinfeld, "A 433 MHz 64b quad-issue RISC microprocessor," in *ISSCC Dig. Tech. Papers*, Feb. 1996, pp. 222–223.

[3] J. Mori, Y. Kondo, and N. Ikumi, "High speed adder circuit using dummy carry method," in *1997 Symp. VLSI Circuits Dig. Tech. Papers*, June 1997, pp. 39–40.

[4] G. A. Ruiz, "Evaluation of three 32-bit CMOS adders in DCVS logic for self-timed circuits," *IEEE J. Solid-State Circuits*, vol. 33, pp. 604–613, Apr. 1998.

[5] N. Weste and K. Eshraghin, Principles of CMOS VLSI Design, 2nd ed. Reading, MA: Addison-Wesley, 1993.

[6] A. Bellaouar and M. I. Elmasry, *Low-Power Digital VLSI Design: Circuits and Systems.* Norwell, MA: Kluwer Academic, 1995.

[7] H. Morinaka, H. Makino, Y. Nakase, H. Suzuki, K. Mashiko, and T. Sumi, "A 2.6 ns 64-b fast and small CMOS adder," *IEICE Trans. Electron.*, vol. E79-C, no. 4, pp. 530–537, Apr. 1996.

[8] S. Naffziger, "A sub-nanosecond 0.5 μm 64b adder design," in *ISSCC Dig. Tech. Papers*, Feb. 1996, pp. 362–363.

[9] D. Dobberpuhl, R. Witek, R. Allmon, R. Anglin, S. Britton, L. Chao, R. Conrad, D. Dever, B. Gieseke, G. Hoeppner, J. Kowaleski, K. Kuchler, M. Ladd, M. Leary, L. Madden, E. McLellan, D. Meyer, J. Montanro, D. Priore, V. Rajagopalan, S. Samudrala, and S. Santhanam, "A 200 MHz 64b dual-issue CMOS microprocessors," in *ISSCC Dig. Tech. Papers*, 1992, pp. 106–107.

[10] K. Udea, N. Sasaki, H. Sato, and K. Mashiko, "A 64-bit carry look ahead using pass transistor BiCMOS gates," *IEEE Journal of Solid-State Circuits*, vol. 31, pp. 810–817, 1996.

[11] D. Harris and M. A. Horowitz, "Skew-tolerant domino circuits," in *ISSCC Dig. Tech. Papers*, 1997, pp. 422–423.

[12] T. I. Chappell, B. A. Chappell, S. E. Schuster, J. W. Allen, S. P. Klepner, R. V. Joshi, and R. L. Franch, "A 2-ns cycle, 3.8 ns access 512-Kb CMOS ECL SRAM with a fully pipelined architecture," *IEEE J. Solid-State Circuits*, vol. 26, pp. 1577–1585, Nov. 1991.

[13] T. I. Chappell, R. A. Haring, T. K. Jabes, E. Seewan, M. P. Beakes, B. A. Chappell, and B. M. Fleischer, "High performance self-resetting circuits with enhanced testability," IBM Res. Rep. RC20321, Jan. 1996.

[14] R. A. Haring, M. S. Milshtein, T. I. Chappell, S. H. Dhong, and B. A. Chappell, "Self-resetting logic register and incrementer," in *1996 Symp. VLSI Circuits Dig. Tech. Papers*, June 1996, pp. 18–19.

[15] V. Narayanan, B. M. Fleischer, and B. A. Chappell, "Static timing analysis for self resetting circuits," in *Proc. Int. Conf. Computer-Aided Design (ICCAD)*, Nov. 1996, pp. 119–126.

[16] W. Hwang, W. Henkels, and R. V. Joshi, "A pulse-to-static conversion latch with a self-timed control circuit ," in *Proc. IEEE Int. Conf. Computer Design: VLSI in Computers and Processors*, Oct. 1997, pp. 712–717.

[17] W. Hwang, R. V. Joshi, and W. Henkels, "A 500 MHz 32-word $\times$ 64-bit 8-port self-resetting CMOS register file," *IEEE J. Solid-State Circuits*, vol. 34, pp. 56–67, Jan. 1999.

[18] W. Hwang, G. D. Gristede, P. N. Sanda, S. Y. Wang, and D. F. Heidel, "Design and implementation of high performance dynamic 64-bit parallel adder with enhanced testability," in *Proc. IEEE 1998 Custom Integrated Circuit Conf. (CICC)*, May 1998, pp. 519–522.

[19] R. Brent and H. T. Kung, "A regular layout for parallel adders" *IEEE Trans. Comput.*, vol. C-31, pp. 260–264, Mar. 1982.

[20] J. L. Hennessy and D. A. Patterson, *Computer Architecture: A Quantitative Approach*, 2nd ed. Los Altos, CA: Morgan Kaufmann, 1996.

[21] C. W. Koburger, W, F. Clark, J. W. Adkisson, E. Adler, P. E. Bakeman, A. S. Bergendahl, A. B. Botula, W. Chang, B. Davari, J. H. Givens, H. H. Hansen, S, J, Holmes, D. V. Horak, C. H. Lam, J. B. Lasky, S. E. Luce, R. W. Mann, G. L. Miles, J. S. Nakos, E. J. Nowak, G. Shahidi, Y. Taur, F. R. White, and M. R. Wordeman, "A half-micron CMOS logic generation," *IBM J. Res. Develop.*, vol. 39, no. 1/2, pp. 215–227, Jan./Mar. 1995.

[22] J. Silberman, N. Aoki, D. Boerstler, J. L. Burns, S. Dhong, A. Essbaum, U. Ghoshal, D. Heidel, P. Hosfstee, K. T. Lee, D. Meltzer, H. Ngo, K. Nowka, S. Posluszny, O. Takahashi, I. Vo, and B. Zoric, "A 1.0-GHz single-issue 64-bit PowerPC integer processor," *IEEE J. Solid-State Circuits*, vol. 33, pp. 1600–1608, Nov. 1998.

Wei Hwang (S'68–M'69–SM'90) received the B.Sc. degree from National Cheng-Kung University, Taiwan, R.O.C., the M.Sc. degree from National Chiao-Tung University, Taiwan, and the M.Sc. and Ph.D. degrees from the University of Manitoba, Winnipeg, Man., Canada.

He is currently a Research Staff Member at the IBM T. J. Watson Research Center, Yorktown Heights, NY, and an Adjunct Professor of electrical engineering at Columbia University, New York. Prior to that, he was an Associate Professor of electrical engineering at Columbia University and an Assistant Professor of electrical engineering at Concordia University, Montreal, P.Q., Canada, where his research focused on organic electroluminescent materials and devices and polysilicon photovoltaic devices. At IBM, he has contributed to several areas of microelectronics, DRAM, and microprocessor design and technology. His current interests include low-power/high-speed CMOS logic and memory circuits, advanced microprocessor and memory architectures, and high-performance embedded system integration design. He has received 40 U.S. patents and has published about 100 technical papers in the areas of semiconductor technologies, materials, devices, and VLSI logic and memory circuits. He is coauthor of *Electrical Transport in Solids—With Particular Reference to Organic Semiconductors* [Oxford, U.K.: Pergamon, 1981; and USSR: Mir Publisher, 1984 (in Russian)].

Dr. Hwang is a member of the New York Academy of Science, American Physical Society, Sigma Xi, and Phi Tau Phi Society. He has received the IBM Fifteenth Plateau Invention Achievement Award and three IBM Research Division Technical Awards. He was recognized as one of IBM's top inventors in 1991 and 1994 and earned the IBM Research Division honorary title of "Master Inventor" in 1997. He was President of the Chinese American Academic and Professional Society (CAAPS) in 1986, Chairman of the Board of Directors of CAAPS from 1988 to 1990, and a member of the Board of Directors of CAAPS during 1986–1991 and 1994–1999. He was member of the Governing Board and Treasurer of the Chinese Language Computer Society (1993–1998). He received the 1985 Outstanding CAAPS Service Award, the 1992 Courvoisier Leadership Award, and the 1995 CAAPS 20th Anniversary Special Service Award.

George Diedrich Gristede was born in Mt. Kisco, NY, in 1962. He received the B.A. degree in professional option engineering from Columbia College, New York, in 1984. He received the B.S., M.S., M.Phil., and Ph.D. degrees from Columbia University, New York, in 1985, 1988, 1990, and 1992, respectively, all in electrical engineering.

His Ph.D. dissertation dealt with the derivation of a new mathematical formulation for analyzing the convergence properties of relaxation-based simulation methods. In 1992, he joined the IBM T. J. Watson Research Center, Yorktown Heights, NY, as a Research Staff Member, where he has done extensive work in the areas of high-performance circuit design and the development of a new generation of custom CAD tools to support such designs. His research interests include high-performance circuit design, automated circuit checking, and optimization.

Dr. Gristede is a member of Tau Beta Pi and Eta Kappa Nu.

Pia Sanda received the Ph.D. degree in physics from Cornell University, Ithaca, NY, in surface Raman scattering.

She is currently a Manager in the VLSI Design Area, IBM T. J. Watson Research Center, Yorktown Heights, NY. She began her career at IBM in imaging science. In silicon technology, she designed and built 0.1-μm channel length CMOS FET's using phase shift lithography and contributed to the device and cell design for the 256-Mb DRAM. She has been engaged in designing high-performance circuits for microprocessors and has recently explored new avenues for test and improved semiconductor manufacturability, such as the new PICA measurement technique.

Shao Y. Wang received the B.S. degree in electronic engineering from Fu-Jen University, Taiwan, R.O.C., in 1991. She received the M.S. degree in electrical engineering from Columbia University, New York, in 1993, where she currently is pursuing the Ph.D. degree.

From 1994 to 1996, she was with the IBM T. J. Watson Research Center, Yorktown Heights, NY, where she was involved in the design of high-performance dynamic circuits used in advanced microprocessors. In 1996, she joined NeoParadigm Labs, San Jose, CA, where she worked on low-power SRAM design and several ASIC products in the area of multimedia and image scanning. Her research interests include high-performance circuit design, low-power circuit techniques used in large logic arrays, and VLSI implementation for bio-sequence search incorporated with fast search algorithm.

David F. Heidel (M'85) received the B.S. degree in physics from Miami University, Oxford, OH, in 1974 and the M.S. and Ph.D. degrees in low-temperature physics from Ohio State University, Columbus, in 1976 and 1980, respectively.

In 1980, he joined IBM's T. J. Watson Research Center, Yorktown Heights, NY, working on Josephson technology. Since 1984, he has been working on the design and testing of high-speed semiconductor devices and circuits. He is presently Manager of the High Speed Analog/Digital Test Group within IBM's Research Division.