

A ROBUST SELF-RESETTING CMOS 32-BIT PARALLEL ADDER

Gunok Jung, Venkat Sundarajan and Gerald E. Sobelman

Department of Electrical and Computer Engineering
University of Minnesota
Minneapolis, MN 55455, USA
Phone: (612) 625-8041, Fax: (612) 625-4583
email: sobelman@ece.umn.edu

ABSTRACT

This paper presents new circuit configurations for a more robust and efficient form of self-resetting CMOS (SRCMOS). Prior structures for SRCMOS have very high performance but are difficult to design and are not robust over process, temperature and voltage variations. The new techniques replace delay chains with logical circuits that will create pulses at the correct times, independent of operational and environmental factors. These concepts are illustrated using a 32-bit parallel adder as a design example.

1. INTRODUCTION

Modern digital computing and signal processing systems require fast arithmetic circuits that operate at very high clock frequencies. To address these needs, high-performance dynamic circuit design techniques that eliminate the overheads due to clock skew, intermediate latches and logic imbalance have been developed, including skew-tolerant domino [1], [2] and enhanced precharge contention [3].

Another category of dynamic circuits, called self-resetting CMOS (SRCMOS), represents signals as short-duration pulses rather than as voltage levels. When a set of pulses are sent to the inputs to a logic gate, they must arrive at essentially the same time and they must overlap with one another for a minimum duration. After a logic gate has processed a set of input pulses, a reset signal is activated that restores the logic gate to a state in which it can receive another set of input pulses. (One sometimes hears the term “postcharge” used to describe this reset operation.) The reset operation is timed to occur after the input pulses have returned to zero. Thus, there is no need for an evaluate or “foot” transistor since the pull-down network will be off during the reset operation, and this is one of the factors that leads to high-speed operation. Moreover, since the reset occurs immediately after each gate has evaluated, there is no need for a separate precharge phase. Since short-duration pulses are hard to debug and test, special additional test-mode features are some-

times added for these purposes.

Two types of reset structures have been proposed for use in SRCMOS. In *globally* self-resetting CMOS [4], the reset signal for each stage is generated by a separate timing chain which provides a parallel worst-case delay path. Individual reset signals are obtained at various tap points along this timing chain in such a way that the reset pulse arrives at each stage only after the stage has completed its evaluation. Very careful device sizing based on extensive simulations over process-voltage-temperature corners are required in order to ensure correct operation. Moreover, any extra delay margin that is designed into the timing chain simply reduces the throughput by a corresponding amount.

On the other hand, in *locally* self-resetting CMOS [5], the reset signal for each stage is generated by a mechanism local to that stage. Previous implementations of this technique have been based on single-rail domino stages in which the reset signal is obtained by sending the stage’s own output signal through a short delay chain. Again, this technique requires very careful simulations and device sizing in order to ensure that the reset signals do not arrive too early. As with the other technique, any timing margin that is built in will directly limit the achievable performance.

In this paper, we propose a new technique for constructing high-performance SRCMOS circuits that is easy to apply and which leads to increased robustness. As a design example, we present a high-speed implementation of a 32-bit parallel adder and demonstrate that it operates correctly over a much wider range of environmental conditions compared to existing techniques.

2. ROBUST SELF-GENERATED RESET SIGNAL

In order to make the reset process more robust over process, temperature and voltage variations, we introduce a technique that relies on the evaluation of an individual dual-rail domino circuit. The basic structure of the proposed locally self-resetting CMOS technique is shown in Figure 1.

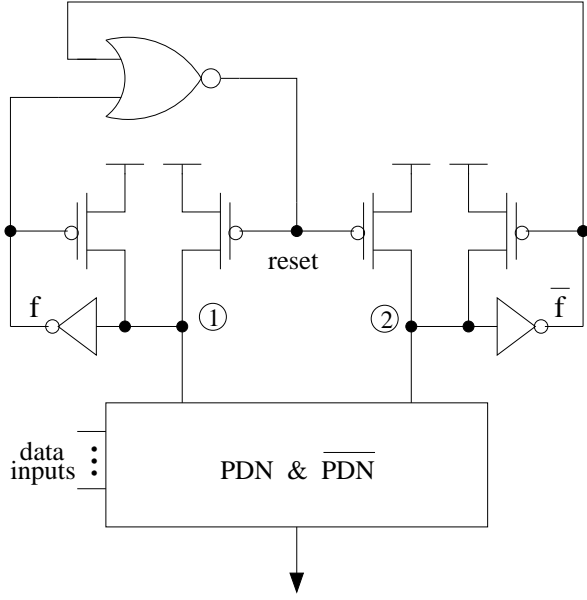


Fig. 1. The proposed reset mechanism.

In the circuit, the locally generated reset signal is created by the output of a static CMOS NOR gate. The operation is as follows: Using an input initialization pattern, either node 1 or node 2 will be forced low through the pull-down network (PDN) and its complement. (As in standard dual-rail domino circuits, PDN and its complement may be implemented using either individual complementary networks or as a merged pull-down structure.) Thus, one input to the NOR goes high, which forces the the output of the NOR gate (i.e., the reset signal) low. Note that by this time, the short-duration pulses corresponding to the input signals would have returned low, turning the pull-down nets off. When the two PMOS precharge transistors turn on, nodes 1 and 2 are precharged high. This causes both function outputs to go low, so that the reset signal goes high, thereby turning the PMOS precharge transistors off. The circuit is now ready to accept another set of input pulses. The above cycle is repeated for each set of arriving input data.

From the above discussion, one can see that the NOR gate enforces that the desired sequence of operations will occur at the earliest allowable times. Furthermore, the mechanism will operate over a wide range of process, voltage and temperature conditions since a logical structure controls the sequencing rather than a timing chain. Normally, the sum of the delays around the loop should be longer than the width of the input pulses, so that fighting between the precharge transistors and the pull-down network will be avoided. However, the circuit would still operate correctly if this were not the case, albeit at a higher power dissipation. Optionally, one can insert a non-inverting buffer (i.e., an even number of inverters) following the NOR gate to pro-

vide a little additional delay if this were found to be an issue for a particular stage of logic.

This dual-rail form of locally self-resetting CMOS should be distinguished from some related circuit configurations. In the self-timed pipeline or ring structure [6], dual-rail domino outputs are combined in a NOR gate to create a request/acknowledge signal. However, this signal is used to trigger the reset of a *previous* stage as part of a handshaking protocol between *different* stages. Similarly, in the self-resetting CMOS design given in [7], dual-rail outputs are sent through a NOR gate and inverter chain to provide the reset signal for a *previous* stage. In that approach, timing chains are still required to propagate the reset signal to subsequent stages. In contrast to these previous methodologies, the locally self-resetting mechanism presented here operates *internally* within each single stage of logic.

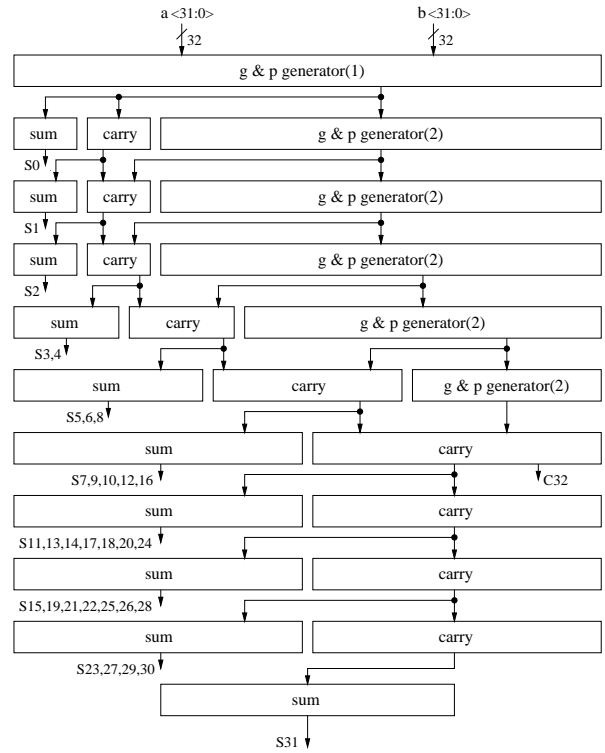


Fig. 2. Block diagram of the self-resetting 32-bit CLA adder.

3. IMPLEMENTATION AND COMPARISON

The architecture for the 32-bit carry look-ahead (CLA) parallel adder that we consider consists of the set of basic blocks shown in Figure 2. For comparison purposes, this architecture has been implemented using two types of SRCMOS circuits, namely the globally self-resetting style and the proposed locally self-resetting technique. All input, output and

Name of blocks	Boolean logic equation
g & p generator(1)	$g_i = a_i b_i$ $\bar{g}_i = \bar{a}_i + \bar{b}_i$ $p_i = a_i \bar{b}_i + \bar{a}_i b_i$ $\bar{p}_i = a_i b_i + \bar{a}_i \bar{b}_i$
g & p generator(2)	$gg = g_j + p_j g_i$ $\bar{gg} = \bar{g}_j (\bar{p}_j + \bar{g}_i)$ $pp = p_j p_i$ $\bar{pp} = \bar{p}_j + \bar{p}_i$
carry	$c_j = g_i + p_i c_i$ $\bar{c}_j = \bar{g}_i (\bar{p}_i + \bar{c}_i)$
sum	$sum_i = c_i \bar{p}_i + \bar{c}_i p_i$ $\bar{sum}_i = c_i p_i + \bar{c}_i \bar{p}_i$

Table 1: Functions of basic blocks of SRCMOS adder.

internal signals are formed as dual-rail true/complement pairs.

The Boolean functions for the basic blocks are listed in Table 1. In the first stage, first-level generate and propagate circuits are employed. At the next stage, higher-level generate and propagate circuits are used to merge these signals, resulting in a reduction in the number of signals by half. The sum generation at each bit position is completed by an XOR operation on the corresponding carry and propagate signals.

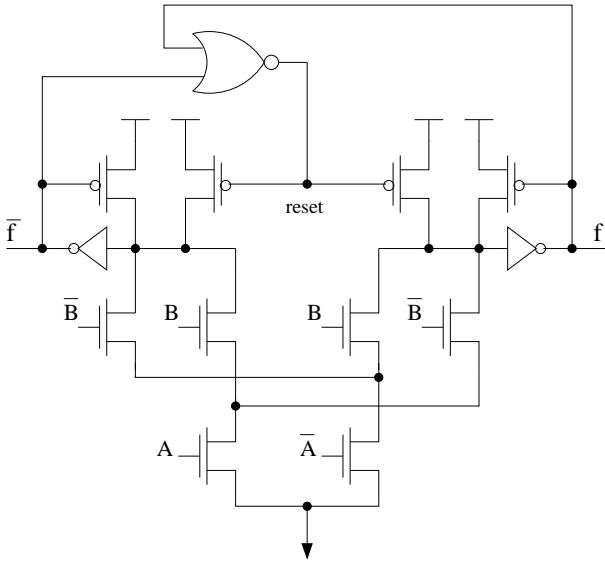


Fig. 3. Implementation of the final XOR (sum) function.

As an example, the implementation of the final XOR (sum) function in the proposed circuit style is shown in Figure 3. Transistor sizing has been done in accordance with

standard techniques, including the use of HI-skew inverters to decrease the evaluation time [8]. In addition, we have used secondary PMOS precharge transistors (not shown) connected to high-capacitance internal nodes in the pull-down network to eliminate charge sharing and weak PMOS keepers for increased robustness against noise.

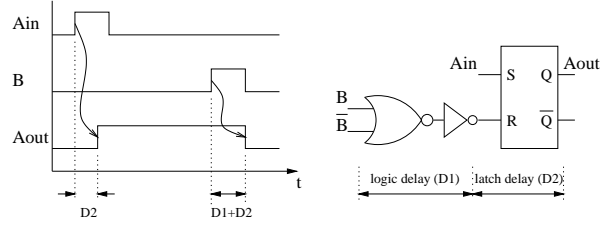


Fig. 4. Pulse stretcher using an SR latch and NOR gate.

One of the major complications involved in implementing this architecture in SRCMOS is that the input pulses to many of the sum and carry cells will be arriving at substantially different times. This is due to the fact that these blocks combine signals from various levels of logic together. A standard design technique would be to delay the slower signal(s) into a logic stage by passing them through a chain of inverters. However, this creates two significant problems: First, precise delay matching is required, and this is difficult to establish and maintain over the required range of environmental and operational conditions. Second, the large number of signals and the large variations in arrival times would necessitate the use of a very large number of buffers. A further study of this issue revealed that the number of buffers required for this architecture would have amounted to 45% of the total number of transistors, with a resulting large increase in power and area. In order to ameliorate this problem, the novel interface circuit of Figure 4 was used instead of a buffer. The figure illustrates a case in which a pulse A_{in} , which arrives early, must be combined in a stage of SRCMOS logic with another signal B , which arrives late. The interface circuit, composed of a NOR gate, inverter and SR latch, is used as a “pulse stretcher” so that A_{out} can be properly combined with B at the inputs to an SRCMOS logic gate. Note that this design improvement has been applied in *both* of the SRCMOS implementations of the adder considered here.

The simulation results for the two types of self-resetting circuits are shown in Table 2. We used the 0.25 micron CMOS process of TSMC that is available from MOSIS. Model parameters for “fast” or “slow” processes were not available, so we approximated these by decreasing or increasing the magnitude of the threshold voltage value that is specified in the nominal process parameters, respectively. The table shows that the proposed locally self-resetting technique is much more robust than the globally self-resetting technique, since it operates correctly over a much wider

Various process corners for 32-bit CLA adder	Circuit outputs for the proposed locally self-resetting design (delay/pw)	Circuit outputs for the globally self-resetting design (delay/pw)
Temp:25°C, Vdd:3.3v, Vthn:0.4v Vthp:-0.58v	GOOD (1.52ns/0.38ns)	GOOD (1.49ns/0.38ns)
Temp:0°C, Vdd:3.3v, Vthn:0.4v Vthp:-0.58v	GOOD (1.42ns/0.36ns)	GOOD (1.39ns/0.38ns)
Temp:25°C, Vdd:3.5v, Vthn:0.4v Vthp:-0.58v	GOOD (1.48ns/0.38ns)	GOOD (1.44ns/0.39ns)
Temp:25°C, Vdd:3.3v, Vthn:0.2v Vthp:-0.38v	GOOD (1.35ns/0.37ns)	GOOD (1.33ns/0.39ns)
Temp:0°C, Vdd:3.5v, Vthn:0.2v Vthp:-0.38v	GOOD (1.25ns/0.36ns)	GOOD (1.23ns/0.39ns)
Temp:100°C, Vdd:3.3v, Vthn:0.4v Vthp:-0.58v	GOOD (1.84ns/0.38ns)	GOOD (1.80ns/0.25ns)
Temp:25°C, Vdd:3.0v, Vthn:0.4v Vthp:-0.58v	GOOD (1.60ns/0.37ns)	GOOD (1.56ns/0.35ns)
Temp:25°C, Vdd:3.3v, Vthn:0.8v Vthp:-0.98v	GOOD (2.01ns/0.37ns)	WRONG (1.96ns/0.17ns)
Temp:100°C, Vdd:3.0v, Vthn:0.4v Vthp:-0.58v	GOOD (1.98ns/0.38ns)	WRONG (1.93ns/0.19ns)
Temp:100°C, Vdd:3.3v, Vthn:0.8v Vthp:-0.98v	GOOD (2.40ns/0.42ns)	WRONG (—/0ns)
Temp:25°C, Vdd:3.0v, Vthn:0.8v Vthp:-0.98v	GOOD (2.21ns/0.39ns)	WRONG (2.15ns/0.08ns)
Temp:100°C, Vdd:3.0v, Vthn:0.8v Vthp:-0.98v	GOOD (2.69ns/0.50ns)	WRONG (—/0ns)

Table 2: Output results for various process corners.

range of process, voltage and temperature conditions. The table entries also give the total delay for the critical path and the pulse width (pw) of the sum outputs. The input pulses have a width of 0.5 ns, so an output pulse is considered to be unacceptable if its width has been degraded to less than half of this size, i.e. if it is narrower than 0.25 ns. For those cases in which both circuits operate correctly, the delay for the locally self-resetting implementation is within 3% of the delay for the globally self-resetting design. Also, the average power dissipation was found to be approximately 10% higher for the locally self-resetting design. (Since the pulse stretcher technique has been applied in both designs, this power difference is due to the differences in the reset mechanism.)

The waveforms for a few representative sum outputs are shown in Figure 5. Note that these sum outputs are produced at different stages, which is why they appear at different points in time.

4. CONCLUSIONS

We have introduced two design improvements to increase the robustness and efficiency of SRCMOS circuits. The first technique uses a logical structure to properly sequence the reset and evaluate modes of an SRCMOS logic stage, without having to rely on a timing chain. This simplifies the design and also allows the circuits to operate over a much wider range of process, voltage and temperature conditions. The second improvement is the use of a pulse stretcher so that input pulses of widely different arrival times can be properly combined at a given logic stage. This eliminates the need for buffer chains, saving considerable power and area. Both of these techniques have been utilized in the

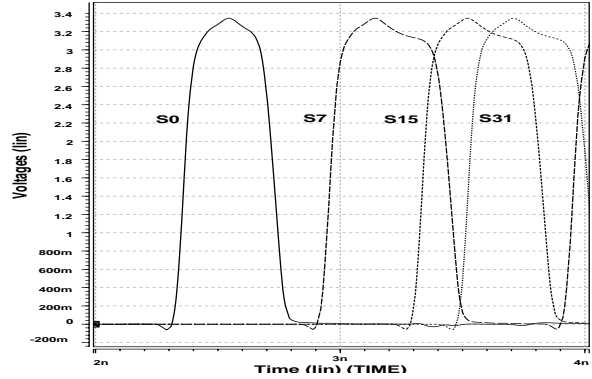


Fig. 5. Waveforms for selected sum outputs.

design of a 32-bit parallel adder, and the improvement in robustness has been confirmed through a set of circuit simulations.

5. REFERENCES

- [1] D. Harris and M. A. Horowitz, "Skew-Tolerant Domino Circuits," *IEEE Journal of Solid-State Circuits*, Vol. 32, No. 11, pp. 1702-1711, November, 1997.
- [2] G. Jung, V. Perepelitsa and Gerald E. Sobelman, "Time Borrowing in High-Speed Functional Units Using Skew-Tolerant Domino Circuits," *IEEE Int'l Symp. on Circuits and Systems*, pp. V-641 - V-644, 2000.
- [3] G. Jung and Gerald E. Sobelman, "High-Speed Adder Design Using Time Borrowing and Early Carry Propagation," *IEEE Int'l. ASIC/SOC Conf.*, pp. 43-47, 2000.
- [4] W. Hwang et al, "Implementation of a Self-Resetting CMOS 64-Bit Parallel Adder with Enhanced Testability," *IEEE Journal of Solid-State Circuits*, Vol. 34, No. 8, pp. 1108-1117, August 1999.
- [5] A. E. Dooply and K. Y. Yun, "Optimal Clocking and Enhanced Testability for High-Performance Self-Resetting Domino Pipelines," *20th Conference on Advanced Research in VLSI*, pp. 200-214, 1999.
- [6] T. Williams, "Self-Timed Rings and Their Application to Division," Ph.D. thesis, Stanford University, Technical Report CSL-TR-91-482, 1991.
- [7] D. H. Allen et al, "Custom Circuit Design as a Driver of Microprocessor Performance," *IBM J. Research and Development*, Vol. 44, No. 6, pp. 799-822, November, 2000.
- [8] I. Sutherland, B. Sproull and D. Harris, *Logical Effort: Designing Fast CMOS Circuits*, Morgan Kaufmann, 1999.