Current Sensing Completion Detection for High Speed and Area Efficient Arithmetic

Balapradeep Gadamsetti[#1], Adit D. Singh[*2]

Department of Electrical and Computer Engineering,
Auburn University, Auburn, AL, 36849
[1]bzg002@auburn.edu, [2]adsingh@auburn.edu

***Abstract*— Providing carry completion signaling in low cost ripple carry adders can allow the control logic to schedule the next addition as soon as an earlier one is complete, thereby achieving the average case, rather than worst case addition delay over a set of computations. Earlier attempts at using current sensing for such carry completion signaling suffered from serious limitations. In this paper we present a new approach for the design of a ripple carry adder with a current sensing capability which observes late settling carry signal nodes in the circuit and indicates when they reach a quiescent state. Simulations show better than 50% speedup, on average, with less than 10% area overhead. To demonstrate a potential application of such an approach, we incorporate our carry completion adder into a Booth multiplier design and study the performance gain over a traditional ripple carry adder based design. Simulation results show that a 32-bit Booth Multiplier using the new completion signaling circuits can outperform a 32-bit Booth Multiplier with ripple carry adder (RCA) by 20-30%, while requiring less than 2% additional silicon area, This is comparable to the gains from the best carry look ahead adder designs at a fraction of the area overhead costs.**

***Keywords*— Current Sensing Completion Detection, Ripple Carry Adder, Booth Multiplier.**

I. INTRODUCTION

While timing in larger designs incorporating conventional Ripple Carry Adders (RCA) must allow for the worst case carry ripple delay for every addition, for many input cases the correct result is in fact available a lot earlier. The incorporation of a computation completion signal into a RCA offers a way for improving the "average case" RCA performance. This provides a mechanism for the RCA to signal to the higher level circuitry controlling it that it has completed the operation. Thus, for example, if 32 repeated additions are to be performed to multiply two 32 bit numbers, using completion signaling to initialize the next addition (before waiting out the worst case delay) can cut down the total multiplication time from 32 worst case addition delays, to 32 "average" case delays. This can achieve performance comparable to that attainable from much more expensive carry look ahead adders. These benefits also hold for the popular Booth's algorithm based multiplication implementations that reduce the number of additions needed by skipping additions for strings 0's and strings of 1's in the multiplier input.

In theory, CMOS designs offer a simple way to determine when all switching activity has ceased and the circuit has reached steady state: there should be only minimal leakage current drawn from the power supply. A number of previous researchers have investigated ways to implement Current-Sensing Completion Detection (CSCD) [1-4] using circuits that monitor the power supply of circuit blocks as shown in Fig-1 and indicate when the outputs stabilize. [2, 4] specifically address the problem of generating completion signals in carry ripple adders. Unfortunately, monitoring the power supply current necessarily involves introducing additional circuitry between the logic gates and the power supply. This can seriously degrade the supply voltage available to the logic, particularly in low power low voltage designs, and thereby significantly impact circuit performance. Furthermore, it remains very challenging to design current monitors with a sufficient dynamic range to provide for the peak switching current demands of a functional block of a hundred or more gates, while still being sensitive enough to reliably indicate when the last gate completes its switching transition. These difficulties have prevented the practical implementation of current sensing based completion detection

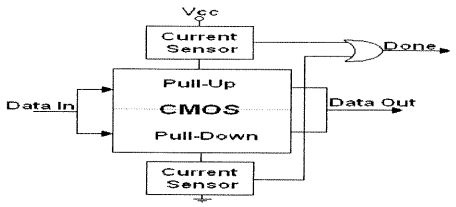

Fig- 1: Current-Sensing Completion Detection (CSCD) circuitry for a CMOS Block

In this paper we present a novel alternative approach for completion detection design that does not monitor the current in the power supply of the functional block. Instead, only a selected set of individual late settling signal nodes in the circuit are observed by connecting simple inverter sensors to them, and monitoring the current drawn by these sensors from the power rail. The overall circuit switching current is not monitored. In this way, functional performance in only minimally impacted (by the small additional fan out loading due to the sensors), while the completion of switching in at the

This research was supported in part by the National Science Foundation through grant 0834620.

observed nodes can be reliably detected by monitoring the current drawn by the sensors. The proposed new design approach thus overcomes key limitations of earlier approaches. Our results in this paper show that new RCA with completion signaling can achieve comparable average case delay to RCA without completion circuitry.

II. The New Design Implementation

The proposed method for implementing the current sensor involves using a sense-inverter such as the one shown in Fig-6. A CMOS inverter draws current from the power supply as long as its input is midrange between a high and a low voltage, such that both the P and N transistors see a gate source voltage above their respective threshold voltages. The supply current does not flow once the input reaches within a threshold of either the high or low supply voltages. Thus monitoring the supply current provides a means to determine if the input has stabilized close (within a threshold voltage) of a high or low.

Incorporating a sense-inverter current-sensing implementation into a standard 32-bit RCA involves adding a minimally-sized inverter to each of the 32 carry signals as shown in Fig-6. Recall that standard cells in typical cell libraries generally use transistors that are sized up in width by a 5-10X factor for N transistors and 10-20X (or more) for P transistors for performance. Using a minimally-sized inverter in this application ensures that the sense-inverter does not significantly load the carry signals in comparison to the normal load on the lines, minimizing any performance impact.

Observe in Fig-6 that only the power supply current in the sense-inverters is monitored. Furthermore the sense inverters have no load at the output, other than parasitic capacitances associated with the small minimum sized transistors. Thus a single current monitor connected on the ground rail is sufficient to provide a reliable switching completion signal. (In the case of large capacitive load on outputs, it is possible that the N transistor turn off, while the P transistor is still on and charging the output capacitance for significantly longer. This is the reason for the double sensors, one for each supply rail, in classical CSCD designs.) Importantly, the entire current-sensing circuitry is only associated with the sense inverters, which are not in the functional path. The power supply of the functional logic is not monitored. Thus performance impact on functional performance is minimal. Selecting only a limited number of circuit nodes to be monitored also reduces the peak and dynamic range in the current observed at the current sensor. Even so, observe in Fig-5 that the peak current when all the full adders simultaneously generate carries can be 32 times the largest current seen in an inverter.

A. Sensor Operation

The sensor consists of two inverters connected in a latch configuration. The sources of the two P transistors (T5 and T6) are connected to power through another P transistor (T3), which serves as a switch. The measured voltages are connected to the sensor through two N transistors (T1 and T2), which serve as pass transistors. The gates of T1, T2, T3 and T4 are connected to 'accum', 'precharge' and 'eval' signals generated from control logic having clock and delayed clock signal as inputs and the state of transistors T1, T2, T3 and T4 are shown in Fig-2. During the precharge phase, T4 is on which maintains the inverter inputs at same voltage. In accumulation phase, T1 and T2 are on. This allows the charge from the measured voltages to build up on the inverter inputs, but the output is not affected and T2 are turned off, which latches the voltages into the inverter inputs. The power to the inverters is also turned on at this time because T3 is turned on. When the inverters are powered up, the inverter with the higher input voltage dominates, and its output is pulled low. The outputs from both inverters are connected to another set of inverters Inv3 and Inv4, which serve to buffer the sensor output from the rest of the circuit. If the current from the current generator is greater than the reference current, then the node that was connected to the current generator (output of Inv1) is pulled high. This node is connected to one of the output inverters (Inv3) and pulls its output low. This creates the active-low signal that indicates carry propagation.

It should be noted that the current sensor is triggered before the evaluation period and the output of the RCA is valid only when the Addcomp signal is high, if not, the next cycle is tested for the addition completion.

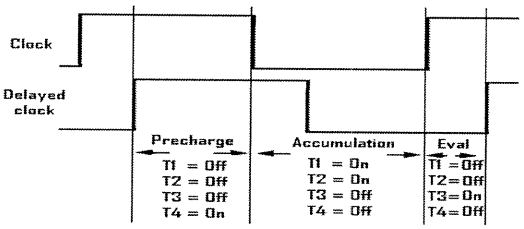

Fig- 2: Control signals

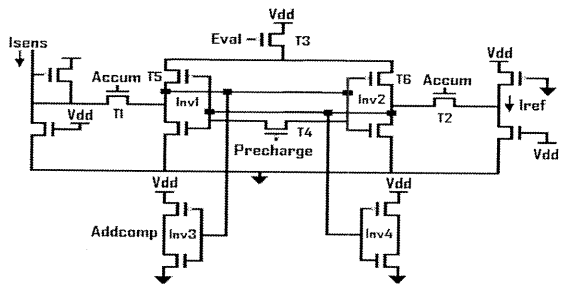

Fig- 3: Current sensor circuitry

In practice, the supply current cannot obviously be more than about 20X the detection threshold current. The reduced voltage across the inverters automatically limits the current drawn from the power supply. Note that while this serious degradation in available voltage across the sense-inverters is acceptable in our design because these sense-gates are not in the functional path, the performance impact of such supply

degradation would be unacceptable if the current sensors were in the functional power supply, as in traditional CSCD designs. This is a major reason why past CSCD designs have not been very practical. In fact, design of the current sensor circuits to meet this demanding requirement of handling a very wide range of supply currents without excessively degrading functional performance was the key challenge in earlier designs.

A classical RCA design uses a string of full-adders, consisting of 2 gate delays each, that are interconnected with the (ripple) carry signals. In an experiment running 100000 random input vectors through such a 32 bit ripple carry adder, the average delay is observed to be 12-14 gate delays associated with an average 6-7 bit longest carry chain length. (Other equal size or shorter carry chains can also be simultaneously active for the same computation.) Compared to the worst case 32-bit carry chain length, which is 64 gate delays, the average case delay reflects a dramatic 5-6X performance improvement. This suggests that a RCA with computation completion signaling can potentially incorporate the low power and area of RCAs with average performance comparable to that of CLAs. However, there is an additional overhead associated with the circuitry producing the completion signal; the challenge is to keep this as low as possible.

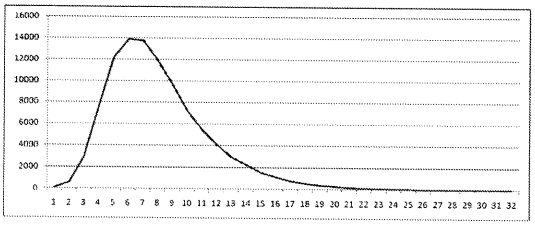

Fig- 4: Carry Propagation graph for two 32 bit additions for 100000 random vectors.

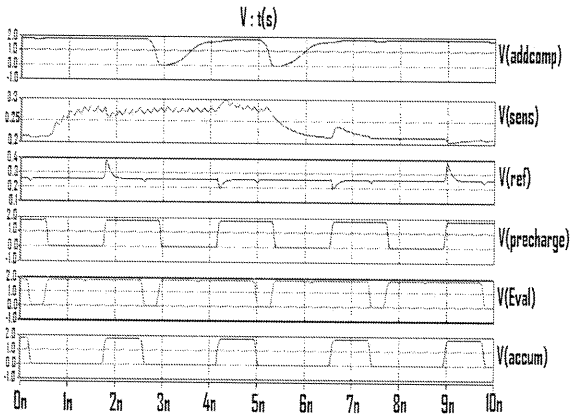

Fig- 5: Voltage waveforms at different nodes of the current sensor circuitry for a 32-bit addition.

III. Ripple Carry Adder

The potential from Ripple Carry Adders with completion signaling is an average case computation time comparable to that of Carry Look Ahead Adders. The new carry completion signaling design discussed in section II was implemented in a 32-bit RCA to produce a modified RCA with a completion-detection capability as shown in Fig-6. This modified RCA has a strong delay dependency on carry chain length. Typically the clock period is selected based on the carry propagation graph as shown in Fig--4.

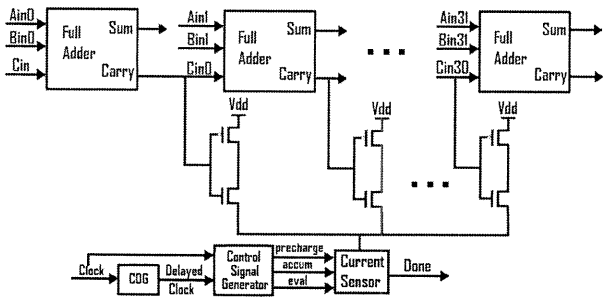

Fig- 6: Ripple Carry Adder with sensor circuitry

A. Simulation Results

The initial SPICE simulations performed were to explore the correlation between delay for the RCA with and without the sensor circuitry. Clock Delay Generator (CDG) for the design with the sensor is set to 0.4ns, i.e. the evaluation time of the current sensor. The design of the 32-bit modified RCA used 1193 transistors in its design which includes the CSCD circuitry. In contrast, the 32-bit RCA without sensor circuitry used 1090 transistors. This suggests a 9-10% area overhead.

Clk1 = 32 stage delay without sense invertors
(4.9ns = worst case delay)
Clk2 = 4.9ns + 0.2ns* +0.4ns**
* Delay added due to sense invertors on carry signals
** Delay added for evaluation over one complete addition
Timed saved = (T1-T2)/T1
Where, T1 = Total time with Clock set to clk1
T2 = Total time with Clock set to clk2/n.
n = 2, 3, 4.

TABLE- I

SIMULATION RESULTS FOR A SET OF 100000 REPEATED RANDOM VECTORS ADDITION OPERATIONS WITH RIPPLE CARRY ADDER

Clock(ns)	Clk2/2 = 2.75	Clk2/3 = 1.84	Clk2/4 = 1.375
Time Saving	40.51%	54.54%	54.72%

Note that the results tabulated in Table-I are from SPICE simulations where circuit parameters are extracted from layout. The technology used was TSMC 180nm process at 1.8V.

IV. BOOTH MULTIPLIER

The potential time saving from Ripple Carry Adders (RCA) with completion signaling is used in our Radix-4 Booth Multiplier design as shown below.

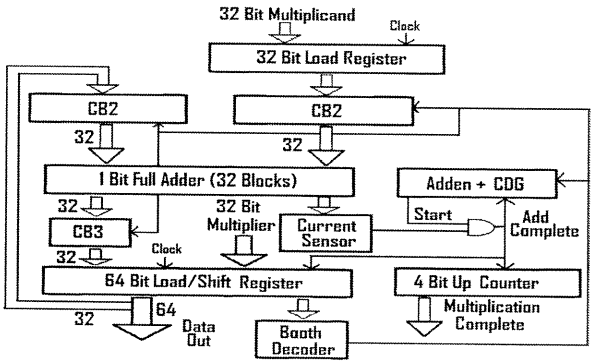

Fig- 7: Booth multiplier with Ripple Carry Adder and Sensor circuitry.

A. Simulation Results

For comparison, we performed simulation for 1000 random input vectors in three different runs. In each run of simulation a performance improvement between 20-30% was observed while keeping the overall area overhead less than 2% of the silicon area (Table-II). In the first implementation a standard 32-bit Booth Multiplier was designed using 32 Ripple Carry Adder with Clock signal connected to Clk1 and second the Ripple Carry Adder with Current sensor circuitry with CDG set to a delay of 0.4ns i.e. response time of the Current sensor with Clock signal connected to Clk2.

In both implementations, the new values are loaded at active clock edge but, with sensor circuitry, values are loaded only when add completion signal is high at active clock edge.

TABLE- II

SIMULATION RESULTS FOR A SET OF 1000 REPEATED RANDOM VECTORS MULTIPLICATION OPERATIONS WITH BOOTH MULTIPLIER WITH RIPPLE CARRY ADDER AND NEW SENSOR CIRCUITRY.

Runs	Run 1	Run 2	Run 3
Time Saving	29%	21%	24%

Note that the results tabulated in Table-II are from SPICE simulations where circuit parameters are extracted from layout. The technology used was TSMC 180nm process at 1.8V.

V. DISCUSSION

In the analysis so far we have ignored leakage power by working with an older technology. Our Booth Multiplier RCA design, including the completion sensing circuitry has 7299 transistors, as compared to 7196 transistors for the RCA without current sensor. The measured time calculations for 1000 random inputs show a performance improvement of 20-30% achieved with an area overhead less than 2%.

Also, observe from Table-I that the best performance is seen when clock is set to average case RCA delay is about 1.375ns this corresponds to a 8-9 longest carry chain length on average. At first sight this number appears inconsistent with the 6-7 long average case carry chain length for 32-bit adders found by our preliminary simulations Fig-4. Recall however that our design incorporates the current sensor that limits the earliest completion signal by adding additional overhead of 0.6ns. Careful designs along with reduction of the CDG delay can potentially further improve the average case delay of our design by 10-20%.

VI. CONCLUSION

Providing carry completion signaling in low cost ripple carry adders can allow the control logic to schedule the next addition as soon as an earlier one is complete, thereby achieving the average case, rather than worst case addition delay over a set of computations. Earlier attempts at using current sensing for such carry completion signaling suffered from serious limitations. In this paper we present a new approach for the design of a ripple carry adder with a current sensing capability which observes late settling carry signal nodes in the circuit and indicates when they reach a quiescent state. Simulations show better than 50% speedup, on average, with less than 10% area overhead. To demonstrate a potential application of such an approach, we incorporate our carry completion adder into a Booth multiplier design and study the performance gain over a traditional ripple carry adder based design. Simulation results show that a 32-bit Booth Multiplier using the new completion signaling circuits can outperform a 32-bit Booth Multiplier with ripple carry adder (RCA) by 20-30%, while requiring less than 2% additional silicon area, This is comparable to the gains from the best carry look ahead adder designs at a fraction of the area overhead costs.

Future work is focused on investigating other applications of the completion signaling approach presented here.

REFERENCES

[1] F. Cheng, S. Unger, and M. Theobald, "Self-Timed Carry Lookahead Adders". IEEE Transactions on Computers, vol. 49, no. 7, July 2000.

[2] M. E. Dean, D. I. Dill, and M. Horowitz, "Self-timed logic using current-sensing completion detection (CSCD)". Journal of CLSI Signal Processing, vol. 7, no. 1-2, pp. 7-16, February 1994.

[3] E. Grass and S. Jones, "Asynchronous circuits based on multiple localized current-sensing completion detection". Proceedings: Second Working Conference on Asynchronous Design Methodologies. pp. viii+223, 170-177, London, UK, May 1995.

[4] O. A. Izosimov, I. I. Shagurin, and V. V. Tsylyov, "Physical approach to CMOS Self-Timing". Electronic Letters, vol. 26, pp. 1835-1836, 1990.

[5] C. G. Knight, A. D. Singh, and V. P. Nelson, "An IDDQ Sensor for Concurrent Timing Error Detection". IEEE Journal of Solid-State Circuits, vol. 33, no. 10, October 1998.

[6] H. Lampinen, O. Vain, "Circuit design for current-sensing completion detection". ISCAS '98: Proceedings of the 1998 IEEE International Symposium on Circuits and Systems. Monterey, CA, vol. 2, pp. 185-188,June1998