On Fast Carry Select Adders

M. Shamanna and S. Whitaker
NASA Space Engineering Research Center for VLSI System Design
University of Idaho
Moscow, Idaho 83843

Abstract –This paper presents an architecture for a high-speed carry select adder with very long bit lengths utilizing a conflict-free bypass scheme. The proposed scheme has almost half the number of transistors and is faster than a conventional carry select adder. A comparative study is also made between the proposed adder and a Manchester carry chain adder which shows that the proposed scheme has the same transistor count, without suffering any performance degradation, compared to the Manchester carry chain adder.

1   Introduction

The trend in Very Large Scale Integrated (VLSI) circuits is toward ever increasing complexity at faster clock rates. This necessitates the use of high-performance arithmetic circuits. The computing performance of many systems is limited by the propagation delay through the arithmetic processing units. Very long bit length adders are becoming the norm. For example, in order to conform to IEEE standards dictates that a double-precision floating-point parallel multiplier needs at least a 108-bit adder[1].

Recently, a 112-b transmission gate adder has been proposed [1] using a Manchester scheme with a conflict-free bypass circuit. The authors concluded that their adder was 20% faster with less than the half the number of transistors compared to a conventional carry select adder. This paper presents a novel carry select adder architecture with the same transistor count and propagation delay as Sato's adder[1].

A discussion on conflict-free bypass circuits has also been presented in [1]. One of the bypass circuits discussed therein suffers from a potential failure. This issue has been addressed in Section 2 of this paper. The design of the proposed adder is presented in Section 3. Section 4, includes a comparative study of the proposed adder with the Manchester adder and the conclusions are presented in the last section.

2   Conflict-Free Bypass Circuits

Manchester adders have been primarily implemented in dynamic and domino logic families. It was shown in [2] that the incorporation of a bypass circuit decreases the carry propagation delay time in a Manchester adder. This circuit was also implemented in dynamic logic. The same bypass circuit design seems to have been adopted in [3], but transmission gates have been used in order to ensure that the circuit remains static. This is illustrated in Figure 1.

It has been stated by Sato et al[1] that *"this circuit solves the power consumption problem. However there are some transition phases in which the bypass circuit does not provide any*
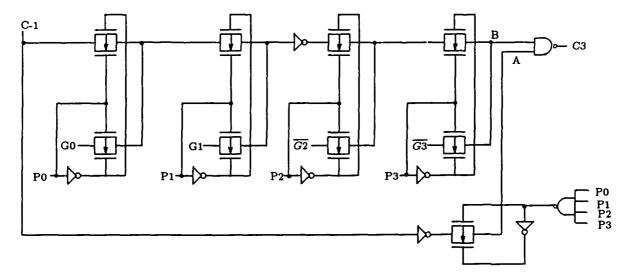
Figure 1: Manchester adder bypass circuit.

performance improvement". The authors further explain that, when the C-1 signal changes from low to high, the signal passes through the bypass circuit to node A and the signal C3 immediately responds. However when C-1 signal changes from high to low, the transition of C3 does not occur until the ripple carry signal reaches the NAND gate at B thereby yielding no performance enhancement.

In addition, under certain conditions the circuit may fail to function as desired. Referring to Figure 1, consider the case when C-1=1 and $P0 \ldots P3 = 1 \ldots 1$. Then A=0 and C3=1. For the next addition, if $P0 \ldots P3 \neq 1 \ldots 1$, the transmission gate which outputs A is disabled and so node A is left storing the logic level in the form of charge on the input capacitance of C3. Over time, if the condition $P0 \ldots P3 \neq 1 \ldots 1$ continues, this charge is affected by leakages through the reversed biased junctions associated with the transmission gate. Leakage paths exist to both VDD and VSS. The dominant path is determined by both layout and processing parameters. If the node A leaks to a logic 0 then C-3 would be erroneously evaluated as a logic 1 irrespective of the value of B. If node A settles midway between VDD and VSS, then there is excessive power consumption due to the dc path set up by turning on both the p-channel and n-channel transistors in C3 driven by A. The circuit would work as desired only if the node A remains at a logic 1. This is clearly undesirable since the length of time $P0 \ldots P3 \neq 1 \ldots 1$ is data dependent during operation leading to possible failure.

To remedy this problem, the circuit needs to be modified as shown in Figure 2, so that when $P0 \ldots P3 \neq 1 \ldots 1$, the P type transistor turns ON, forcing node A to a logic 1. It is noted that although this modification guarantees the circuit to function as desired but fails to provide any performance improvement. The transition of C3 still must wait for the ripple carry signal to reach the final NAND gate, when C-1 goes from high to low with $P0 \ldots P3 = 1 \ldots 1$.

Another solution which yields performance improvement without posing any circuit problems at the expense of extra hardware is shown in Figure 3 [1]. The three transmission gates controlled by T1, T2 and T3 perform an OR operation. The signals T1, T2 and T3 are given
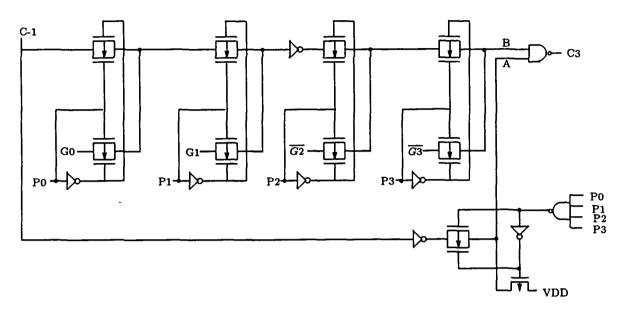
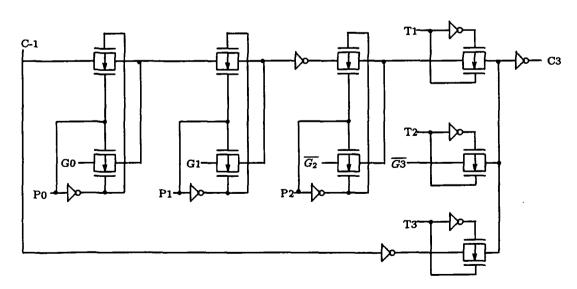Figure 2: Improved bypass circuit.

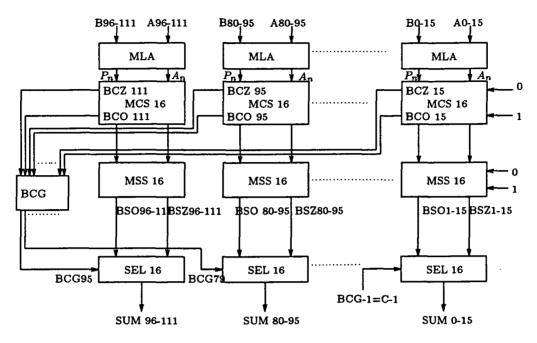Figure 3: Performance enhanced bypass circuit.

Figure 4: Block diagram for a 112-bit adder.

by :

$$T1 = P0' * P1' * P2' * P3$$
$$T2 = P3'$$
$$T3 = P0 * P1 * P2 * P3.$$

The principle used is one of the underlying principles of the BTS pass design methodology [4]. This scheme not only resolves signal conflict but also reduces power consumption.

3   Architecture

This section discusses the architectural details of the proposed Carry Select Adder. In order to provide an effective comparison between the proposed scheme and the Manchester carry adder scheme suggested in [1], we will follow the same design style and Module organization as in [1]. The proposed carry select adder is different from the conventional carry select adder since the carry path has been optimized for minimum delay. As a result of the optimization, the generate signal is no longer used in our scheme. Also, the adder cells, which usually have two transistor delays between carry in and carry out, have been redesigned to have only one transistor delay.

The 112-bit adder is divided into seven 16-b adder blocks. Figure 4 shows the block diagram of the adder. Figure 5 shows a 1-b circuit diagram of MS-16; It comprises of MLA, MCS16, MSS16 and SEL16 blocks (SEL 16 block is not shown in Figure 5). Transmission gates are employed to reduce area, power consumption and to reduce delay times. The MLA consists of an XOR circuit producing the carry propagate signal (P_n). MCS16 produces two carry-out signals $BC\dot{Z}_n$ and BCO_n associated with carry-in = 0 and 1 respectively. MSS16
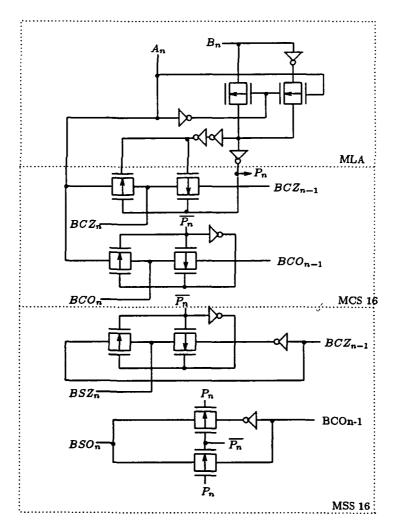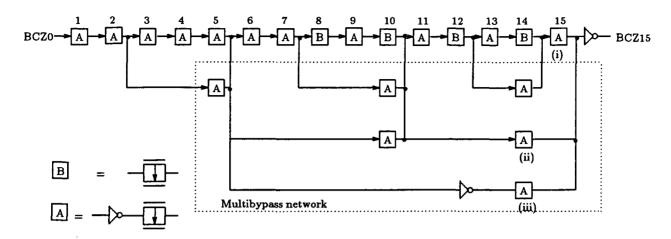
Figure 5: MS-16 circuit diagram.

Figure 6: 16-bit multiple bypass circuit.

produces two sum signals BSO_n and BSZ_n corresponding to carry-in= 1 and 0 respectively. SEL16 selects either BSO_n or BSZ_n depending on the value of BCG_m. The SEL16 is a 16-b, two to one, Multiplexor constructed entirely from transmission gates. This is not shown in the Figure 5 for the sake of brevity. The critical paths for the MCS16 block are formed by the paths through which the signals BCO_n and BCZ_n propagate. These critical paths have sixteen transmission gates in a chain. In order to linearize the quadratic nature of the relationship between the length of the chain and the propagation delay, inverters are placed at suitable points. The placement of the inverters depends on the process parameters and is a fit candidate for global optimization. However, to achieve the high performance expected, the use of bypass circuits in the critical paths become a necessity. Figure 6, shows the 16-b multiple bypass circuitry generating the BCZ15 signal. This is similar to the bypass circuit in [1]. It is noted that the speed could be increased further by optimal placement of inverters. However for the sake of comparison, we shall restrict ourselves to the same inverter positioning as the scheme in [1]. It is seen that the propagation delay of BCZ15 consists of four or less transmission gates (excluding the inverter delays). The by-pass circuitry shown in Figure 6, provides conflict-free by-pass performance.

Similar circuitry ensures that a delay of less than four transmission gates for BCO15 signal. In Figure 6, the signal BCZ15 has three paths which are wired-OR (Input of the final inverter). The three transmission gates are controlled by the following three signals:

(i) $T2' * P15$

(ii) $T1' * T2' * P15$

(iii) $T1 * T2 * P15$

where $T1 = P6 * P7 * P8 * P9 * P10$
                   $T2 = P11 * P12 * P13 * P14$

Only one of the transmission gates is enabled at a time enabling conflict-free performance.

The BCG block provides the BCG signals for the terminal multiplexors (SEL16) in the 16-b MS16 slices. These BCG signals act as the control signals for the MUX'es to select the proper sum outputs. Depending on whether BCG=1 or 0, BSO1–15 or BSZ1–15 is output as
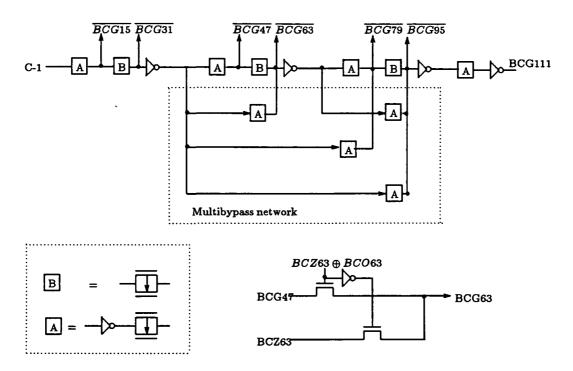
Figure 7: BC63 and BCG111 logic.

SUM0–15 respectively. The logic generating the BC63 signal is shown in Figure 7. Similar logic generates the rest of the BCG signals. The BCG block also uses a multi-bypass circuit as shown in Figure 7 to reduce the propagation delay.

4   Comparison

The transistor count of the one bit slices of the proposed adder and Sato's adder will be compared and it is shown that both have the same transistor count and propagation delays. Since the only comparison vehicle is Sato's adder, it will be referred to as the "existing adder".

The existing adder consists of seven 16 bit MS16 slices and a BCG block. Each MS16 slice, comprises of MLA, MB16 and a ML16 block. The proposed adder follows the same module organization and consists of seven 16 bit MSS16 slices and one BCG block. Each MSS16 slice is made up of MLA, MCS16, MSS16 and SEL16 subblocks.

The proposed MLA is the same as the MLA block in the existing adder. The MB16 of the existing adder has three transmission gates and one P transistor. The proposed MCS16 block has four transmission gates. The ML16 of the existing adder also has four transmission gates plus one NAND gate (four transistors). The MSS16 and SEL16 blocks have four and two transmission gates respectively. Running a global BCG'_m through the SEL16 block saves an inverter (two transistors). Accounting for the extra transistor used in our MCS-16 block, we are still left with one transistor less than the existing adder configuration.

Sato et al. concluded that their adder has half the number of transistors as that of a conventional carry select adder and is about 20% faster than the same. The proposed carry

select adder design has the same advantage in the number of transistors. Additionally, the critical paths of the proposed adder have the same number of the transistors as that of the existing adder. Hence, it should be possible for the proposed adder to be at least as fast as Sato's scheme. Since in the existing adder, the critical signal BC_m must pass through a NAND gate and an inverter before the final transmission gate is enabled to output SUM while in the proposed scheme, a similar signal BCG_m is fed directly to the final transmission gate the proposed scheme should have a speed advantage. The delay of both the signals BCG_m and BC_m are the same because both of them are produced by similar circuitry.

5   Conclusion

In summary, we have presented a new design for a carry select adder with conflict-free bypass circuit with optimal hardware. An existing bypass circuit was shown to suffer from potential failure and was modified to ensure satisfactory performance. A novel 1-b adder design was also proposed with an optimized carry path. Incorporating this design in the long bit adder along with the bypass circuit results in a saving of about 50% in transistor count and also increases the speed by about 20% compared to the conventional carry select adder designs.

References

[1] T. Sato et al., "An 8.5 ns 112-b Transmission gate adder with a conflict free bypass circuit," *IEEE Journal of Solid State Circuits*, Vol. 27, April 1992, pp. 657–659.

[2] P. K. Chan et al., "Analysis and Design of CMOS Manchester adders with variable carry skip," *IEEE Transactions on Computers*, Vol. 39, Aug. 1990, pp. 983–992.

[3] T. Kudo and T. Tokumaru, "Design of 32 bit high speed adder," *IEICE 70th Anniv. Nat. Conv. Rec.*, No. 284, 1987, p. 2–88.

[4] G. Peterson and G. Maki, "Binary Tree Structure Logic Circuits: Design and fault detection," *Proc. IEEE Int. Conf. Comp. Design: VLSI in Computers*, Oct. 1984, pp. 139–144.