VLSI Architectures of Carry Skip Adders – A Survey

Bisam Pranaya¹, *C. John Moses²

¹PG Scholar, Department of ECE, Sreyas Institute of Engineering and Technology, Hyderabad
²Asso. Prof, Department of ECE, Sreyas Institute of Engineering and Technology, Hyderabad

ABSTRACT

Adders are key building blocks in the arithmetic and logic units and hence increasing their speed and reducing their power consumption strongly affect the performance of digital circuits. There are many adder families with different delays, power consumptions, and area usages. Carry skip adder (CSKA) is efficient in terms of power consumption and area usage; delay also smaller than the ripple carry adder (RCA). CSKA allows carry to skip over group of n bits. This paper analyses different very large scale integration (VLSI) design characteristics of various CSKAs in terms of area, power and speed. Based on the detailed review, this paper suggests that by using variable size on stages of skip logic, both delay and power can be improved. Further, this work suggests that for high performance implementation, multi-level with variable block size and and-or-invert (AOI) skip logic and multiplexer-based full adder blocks are preferred for reducing the required number of transistors.

Keywords: CMOS, VLSI, ALU, processors, transistors.

1. INTRODUCTION

Adders are used as a basic and important part of an arithmetic and logic operation or arithmetic and logic unit (ALU) [1]. ALUs are widely used in microprocessors, microcontrollers, data processing unit and digital signal processing. Adder is also used for different binary operations like multiplication, subtraction, division, complements (2’ and 1’s) decoding, and encoding [2] and for address generation logic in memory devices [3]. Therefore, adder plays an important role in central processing unit (CPU). Generally, arithmetic and logical operation performs slowly on compared with other processors operations. So the adder performance fixes the optimum frequency of operation of any processor chip. Also any high-speed adder can increase the performance of any chip [4].

The simplest addition operation is the RCA. The hardware architecture of this adder is very simple but the delay in this adder increases linearly by increasing number of bits [5]. Various hardware architectures have been implemented to improve the performance of adder in the past three decades. Different high speed adders are carry look-ahead adder (CLA), carry select adder (CSA) and carry-skip adder (CSKA). The CSKA is efficient as compared with ripple carry adder (RCA) [6].

CSKA is categorized into two types as single carry skip adder and multi-level carry skip adder. CSKA is low-cost adder because x-bit carry skip addition is portioned into m=\sqrt{x} blocks. These blocks require only AND gates and multiplexers. These CSKAs can be implemented by using different complementary metal oxide semiconductor (CMOS) technologies for any application specific operation and these can be fabricated as application specific integrated circuits (ASICs). Further, CSKAs can also be implemented on field programmable gate array (FPGA) for low-power applications by using less number of resources.

CSKAs are favourable to design efficient VLSI circuits by comparing with other adders. Several kinds of CSKAs have been proposed by different researchers from the past three decades such as one-level, two-level and multi-level CSKAs. Some adders use a constant block size and others use variable block size. Generally CSKA with multiple blocks and different stage size is used to improve the performance of the addition operation [7]. Most of the researchers have concentrated on reducing critical path delay by
changing the logic on carry skip. The carry skip logic is an important part of high performance adders which are key elements of modern processors, multimedia processors and digital signal processors [8]. This paper studies the different CSKA designs used to optimize the performance of addition operation using various skip logics.

The remaining part of this paper is structured as follows. Section 2 describes the detailed survey on carry skip adder. Section 3 provides comparative analysis. Section 4 deals with conclusion and future scope.

2. ARCHITECTURES OF CSK ADDERS

This section briefs the different hardware design of CSK adders and their characteristics in terms of area utilization, performance and power consumption to perform different levels (bits) of addition or arithmetic operations.

Guyot et al (1987) [9] built an efficient carry skip adder by using blocks of variable sizes. This is a two level carry skip adder which is implemented on microcomputer using VLSI technologies (2μm gate CMOS). The computing time of the 128 bit adder is approximately 50 ns. The performance of this carry skip adder is higher than other adders like ripple carry adder and carry look-ahead adder.

Chan et al (1992) [7] proposed CSKA using multidimensional dynamic programming to reduce the delay.

Kantabutra (1993a) [11] implemented one level CSKA in 1μm CMOS technology for 64-bit addition. This adder consists of ripple cells (both odd and even ripples), Multiplexer and XOR gate. This implementation using SPICE produces output by 6.23 ns. Further, Kantabutra (1993b) [12] also implemented a two level CSKA using the concept of portioning the bit positions into several blocks and grouping the blocks into different disjoint sections. This adder is designed in 2μm CMOS technology for 60 bit operation and it operates with 12.6 ns critical path delay. This adder outperforms the previous 2 μm CMOS technology 50 bit adder [9].

Strollo & Napoli (1997) [5] implemented a complementary pass transistor (CPL)-based CSKA and simulated the design by using SPICE. This design uses ES2 0.1 μm CMOS technology which utilizes 840 transistors to operate with 4.5 ns delay. The overall area of this design is 0.42 mm² and the resulting power consumption is 0.23 mW for 32-bit addition. This outperforms carry look-ahead adder (CLA) in terms of all parameters such as speed, area and power.

Cha & Swartzlander (2000) [4] presented a design of CSKA for reducing first block delay by using a few additional gates. This design outperforms the conventional CSKA in terms of speed. This architecture uses only 2.6 ns delay to produce 32-bit addition but the conventional CSKA requires 3.0 ns for the same output.

Burgess et al (2001) [13] described a modified carry skip adder (MCSKA) to reduce the propagation delay by using skip multiplexer. The skip multiplexer is used to replace most significant blocks (MSB) by a carry select adder stage. This carry-skip adder increases the speed of operation by comparing with single-level carry-skip adder [11]. For instance, this accelerated CSKA produces 25 gate delays for 64 bit addition but the single-level CSKA uses 30 gate delays to perform the same 64 bit addition operation.

Marienfeld et al (2002) [14] proposed a 64-bit partially duplicated code-disjoint CSKA by using two adders with variable block size. There is no skip logic on first and last block of the adders and there is no propagate signal on adder2. In this design instead of using a single carry in signal of the adder cells, the carry signal is split into two different signal lines. The propagate signals are implemented only once, there are utilized to compute the duplicated sum bits and simultaneously to check the input parity and some internal XOR gates. This method is implemented by using SYNOPSIS. This approach reduces delay, area and power consumption by 3.1%, 33.4% and 15.5 % respectively from the completed duplicated code-disjoint CSKA.

Chirca et al (2004) [15] proposed a high performance 32-bit carry skip adder by dividing the overall addition operation into variable sized blocks. This utilizes carry look-ahead logic to reduce the propagation of carry and to reduce the overall delay. The architecture of this carry skip adder is implemented on 130nm CMOS technology and the performance of this method is compared with different high performance adder. This CSKA consumes 0.786 mw power, further the delay (921 ps) in this CSKA is less than some
other adders like ripple carry adder, carry increment adder.

Schulte et al (2004) [16] proposed a low-power CSKA with fast saturation. This method uses variable sized blocks. This blocks balance the delay of inputs to the carry chain. In general, the adder architecture decreases power consumption by reducing the number of transistors, logic levels, and glitches. This CSKA is implemented by using 130nm CMOS technology. The 40 bit CSKA works with the delay of 1149 ps and consumes 0.928 mW. The same design in 90 nm CMOS technology operates with the delay of 560 ps and consumes 0.335 mW.

Grad & Stine 2005 [17] implemented different logics of CSKA such as and-or-invert (AOI) logic, NOR logic and NAND logic. Among the three logics the NAND logic operates with low-delay as 216 ps. The other two logics NOR and AOI requires 224 ps and 242 ps respectively to produce the output. However, the NAND logic utilizes more number of transistors as compared with the other two logics. The NAND, NOR and AOI uses 936, 888 and 824 transistors respectively.

Amin et al (2006) [18] designed an efficient self-timed (ST) carry skip adder with low area overhead and fast operation. In this design the adder completion detection circuit used based on a simplified realistic process tracking bounded delay model, it achieves less area overhead and fast completion compared with the other self-timed adder by using 0.18 µm CMOS technology. This circuit is more efficient in terms of speed and area and the architecture is implemented by using self-timed carry skip fashion. This adder works with the delay of 0.87ns and with supply voltage 1.8v.

Lin & Radhakrishnan (2008) [19] proposed a delay-efficient 32-bit CSKA using group and propagate generate functions and multiple RCA stages. This adder is designed and implemented by using 0.25 µm CMOS technology and simulated by TSPICE. Both AOI and or-and invert (OAI) circuits are used for carry generation logic and carry skip logics. The 32-bit CSKA consumes 4.2 mW power to operate with the delay of 3.4 ns.

Islam et al (2009) [20] provided a CSKA using fault tolerant reversible logic. This CSKA uses 40 two input Ex-OR gates, 80 two input AND gates, 12 NOT gates, 11 constant inputs and 16 outputs. This utilizes less number of gates as compared with the related work (Bruce et al 2002) [21]. Another reversible logic-based CSKA is designed by Pang et al 2012 [22] for 16-bit addition. A 4-bit reversible CSKA consists of four adder blocks, one compare block and one multiplexer. By this method, a 16-bit addition utilizes 13 slices to operate with 90.3 ns. By comparing with ripple carry adder (RCA) this CSKA is fast but utilizes more number of slices.

Uma et al (2012) [23] proposed different adder topologies. The adder structures are implemented by using 0.12 µm CMOS technology. The CSKA architecture utilizes 388 gates and consumes 0.603 mW power to operate with the delay of 3.022 ns.

Kim & Kim (2012) [24] synthesized a context-aware CSKA to reduce the delay on skipping the carry. This method uses different components like propagate and generate network, multiplexer, carry-skip logic and RCA. This architecture is tested by SYNOPSIS for TSMC 40 nm CMOS technology. This CSKA outperforms the conventional CSKA presented by Chan et al 1992 [7] by increasing the speed of operation by 16%.

Kim et al (2013) [25] implemented an approximate adder using carry skip logic to reduce power consumption. The architecture of this adder is designed by precise carry prediction scheme and implemented in 90 nm CMOS technology using Verilog HDL and HSPICE. This CSKA utilizes 466 µm² area and 0.600 mW power to work on 356 ps delay for providing 16-bit addition.

Shirakol et al (2014) [26] proposed a 16-bit carry skip adder by reducing power consumption. This work verifies different kind of CSKAs like two-block CSKA, four-block CSKA and eight-block CSKA. Among these three kinds of adders, the two-block CSKA consumes less power as 0.33mW and the delay of the adder is high as compared with other two types. The eight-block CSKA consumes more power but its delay is low as compared with other types of adder. However, the 8-block CSKA utilizes less number of slices as 21 slices and 36 LUTs. The 2-block and 4-block CSKA uses 25 slices, 43 LUTS and 27 slices and 47 LUTs respectively.
Kaur & Sood (2015) [2] implemented an eight bit CSKA using RCAs and logic gates. The circuit of carry skip is constructed by using AND gates an OR gate. This CSKA implementation consumes 0.603 mW power and utilizes 388 gates to operate with 3.622 ns delay.

Bahadori et al (2016) [1] designed a CSKA structure and compare the higher speed and lower energy consumption with the conventional one. Instead of using multiplexer logic this structure uses AND-OR invert (AOI) and OR-AND invert (OAI) compound gates for skip logic. This design use the variable latency extension, which lowers the power consumption without considering the speed. This extension used the modified parallel structure for increasing the slack time. This adder is useful to improve the energy by 38%, and to reduce the delay by 44% from conventional CSKA. This is implemented by using 45 nm static CMOS and simulated by HSPICE. This CSKA utilizes 1332 transistors and the occupied area is 241.5 µm².

Mahesh & Sravanthi (2016) [27] designed a high speed 32-bit carry skip adder using concatenation and incrimination (CI) logic. This design utilizes AND-OR-invert and OR-AND invert compound gates. This method of CI-CSK shows a higher-speed compared with the other adders. Further, the CSK is realized with both fixed size and variable stage size, the speed improvement is achieved by using variable size blocks, parallel prefix adders in nuclear stage. This implementation works with the delay of 19.04 ps and it is low by comparing with similar adders.

Nagaraj et al (2017) [28] analysed the characteristics of CSKAs using different technologies like CMOS (complementary Metal Oxide Semiconductor), CPL (Complementary Pass Transistor Logic) and DPL (Double Pass Transistor Logic) with HSPICE – 180 nm technology. By comparing the three different methods, the CSKA using DPL utilizes less number of transistors as 1664 for 32 bit addition and it operates with less delay as 0.1171ns. But the power consumption is higher than the other two methods. The DPL-based 32 bit addition consumes 64.7941 nanowatts, but CMOS and CPL – based adder consumes 58.4378 nanowatts and 2.1402 milliwatts respectively.

Nayak et al (2017) [29] implemented a CSKA using pass transistor logic (PTL). This implementation uses PTL to enhance the parameters like area, power and delay. This is implemented by using Cadence virtuoso 61.4 software at 90 m technology. By comparing with other conventional CSK, this implementation provides optimized result on number of transistors is 472 and power consumption is 351.2 mW but the conventional CSK uses 472 transistors and consumes 4941 mw power.

Vishalbharat et al (2017) [30] proposed a 4-bit CSKA using NMOS pass transistor logic to reduce the delay and it is implemented by using cadence RTL front end design tool for 180 nm technology. This architecture consumes 289.2 µW with the delay of 78.84 ps. Both the power consumption and delay are reduced from the CMOS PTL technology for the similar operation [29].

Kumaran et al (2017) [31] implemented a CSKA using multiplexer-based full adder (MBFA) for reducing area utilization. This architecture is designed by using SPICE and which uses only 36 transistor but CSKA using conventional full adder utilizes 56 transistors. Further, it operates on 10.43 ns delay, however the CSKA using conventional full adder requires 11.16 ns delay to produce its output.

Arora and Niranjan (2017) [32] provided a 16 bit high speed and variable stage carry skip adder. This adder is simulated using 90 nm CMOS technology in cadence virtuoso. This architecture consists of seven stages, the first and last stages are contains 1 bit each, and it is increasing steadily till the middle stage. This design reduces power consumption by 8% and reduces the delay by 61.75% from the conventional CSKA. This architecture uses variable stage skip logic to reduce the delay. This CSKA consumes 7.07 mW power with the delay of 55.2 ps.

Manasa & Swapmna (2017) [33] designed a high speed and energy efficient carry skip adder, by using AND-OR-invert and OR-AND-invert compound gates for the skip logic. In this design both fixed size and variable stage size styles are used, the speed enhancement is achieved by the modifying the structure through concatenation and incrimination (CAI) techniques. The total delay of this carry skip adder is 15.61ns.

3. COMPARATIVE ANALYSIS

This section compares different types of CSKAs based on different CMOS
technologies, different VLSI parameters like area, delay and power. Table A1 describes different adders and their characteristics. Figure B1 and Figure B2 show the comparison of different CSKAs in terms of number of transistors used and critical path delay respectively.

The optimum VLSI design of CSKA is implemented by [9] in 2 µm CMOS technology by using bimodal of blocks for skip logic. The same CMOS technology is also used to design CSKA by using uni-model block size for skip logic [12]. By comparing both [9] and [12], the [12] is best in terms of delay due to variable block size. Both [9] and [12] are two-level CSKAs. By comparing 90 nm CMOS CSKAs [25], [29] and [32], the approximate CSKA [25] consumes very less power as 0.600 mW by reducing the complexity of the circuits and by using approximation methods. Based on Figure B1, the multiplexer-based full adder logic requires less number of transistors. Based on Figure B2, the 90 nm CMOS technology requires less delay to produce output. Further it shows that the pass transistor logic [32] also outperforms other skip logics. From Table 1, it is noticed that the dual pass transistor logic [28] consumes very less power but it operates in very slow.

4. CONCLUSION

Based on the survey, this work observes that the conventional carry skip adder uses more number of ripple carry adder blocks, more number of gates so increase the area, power and delay. Further, by using the single level carry skip adder blocks the delay is also increased. That is, to increase the speed of operation, multiple level carry skip logic and variable size blocks can be used. The chip area can be reduced by using low complexity full adders like multiplexer-based full adders. The AOI logic is also used to increase the speed of operation. The combination of AOI and OAI logic can also be used to reduce the area and delay. In technological wise, 90 nm CMOS technology outperforms other CMOS technologies. The overall performance of the carry skip adder can be improved by selecting optimum number of stages with optimal size on stages for the skip logic. Further, by combining pass transistor logic, reversible logic with AOI and OAI compound gates high performance additions may be derived for efficient processors.

REFERENCES


12


**APPENDIX A** [Large Tables can be placed here in single column format with the caption Table A1, A2 etc.,]

Table A1: Comparison of area, power and delay of different carry skip adders

<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Type of CSKA/ Technology used</th>
<th>Area/Gate count/ slices/ LUTs/Transistors</th>
<th>Delay</th>
<th>Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>128-bit; 2 µm CMOS Guyot et al (1987) [9]</td>
<td>N/A</td>
<td>50 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>2</td>
<td>1 µm CMOS 64-bit Kantabutra 1993a [11]</td>
<td>N/A</td>
<td>6.23 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>3</td>
<td>2 µm CMOS 60-bit Kantabutra 1993a [12]</td>
<td>N/A</td>
<td>12.6 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>4</td>
<td>CPLCSKA 32-bit Strollo &amp; Napoli 1997 [5] ES2 0.1 µm CMOS</td>
<td>Transistors: 840 Area: 0.24 mm²</td>
<td>4.5 ns</td>
<td>0.23 mW</td>
</tr>
<tr>
<td>5</td>
<td>Modified CSKA – 32 bit; Cha &amp; Swartzlander 2000 [4]</td>
<td>Gates: 318</td>
<td>2.6 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>6</td>
<td>32-bit/130nm CMOS (Chirca et al 2004) [15]</td>
<td>N/A</td>
<td>N/A</td>
<td>0.786 mW</td>
</tr>
<tr>
<td>7</td>
<td>40-bit 130nm CMOS 40-bit 90 nm CMOS (Schulte et al 2004) [16]</td>
<td>N/A</td>
<td>1149 ps 560 ps</td>
<td>0.928 mW 0.335 mW</td>
</tr>
<tr>
<td>8</td>
<td>Static CMOS 16-bit CSKA Grad &amp; Stine 2005 [17] NAND Logic NOR Logic AOI Logic</td>
<td>Transistors: 936 888 824</td>
<td>216 ps 224 ps 242 ps</td>
<td>N/A</td>
</tr>
<tr>
<td>9</td>
<td>32-bit ST; 0.18 µm CMOS (Amin 2006) [18]</td>
<td>No. of transistors 1181</td>
<td>0.87 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>10</td>
<td>0.25 µm CMOS 32-bit CSKA (Lin &amp; Radhakrishnan 2008) [19]</td>
<td>N/A</td>
<td>3.4 ns</td>
<td>4.2 mW</td>
</tr>
<tr>
<td>11</td>
<td>Islam et al 2009 [20]</td>
<td>Gate count- 14</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>12</td>
<td>Reversible CSKA 16-bit (Pang et al 2012)[22]</td>
<td>13 slices</td>
<td>90.3 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>13</td>
<td>0.12 µm CMOS Uma et al 2012 [23]</td>
<td>Gates: 338 Area: 3486 µm²</td>
<td>3.022 ns</td>
<td>0.603 mW</td>
</tr>
<tr>
<td>14</td>
<td>90 nm CMOS Approximate CSKA Kim et al 2013 [25]</td>
<td>466 µm²</td>
<td>359 ps</td>
<td>0.600 mW</td>
</tr>
<tr>
<td>15</td>
<td>8 Block CSKA (Shirakol et al 2014) [26]</td>
<td>36 LUTs</td>
<td>27.09 ns</td>
<td>0.37 mW</td>
</tr>
<tr>
<td>16</td>
<td>Kaur &amp; Sood 2015 [2]</td>
<td>Gate count – 388</td>
<td>3.62 ns</td>
<td>0.60 mW</td>
</tr>
<tr>
<td>17</td>
<td>CIL 32-bit (Mahesh &amp; Sravanthi (2016) [27]</td>
<td>N/A</td>
<td>19.096 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>18</td>
<td>High-speed CSKA (Bahadori et al 2016) [1]</td>
<td>Transistors: 1332 Area: 241.5 µm²</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>19</td>
<td>DPL-based (Nagaraj et al 2017) [28] 32 bit adder</td>
<td>1664 Transistors</td>
<td>0.11715 ns</td>
<td>64.7941 nW</td>
</tr>
<tr>
<td>20</td>
<td>PTL 90 nm CMOS (Nayak 2017) [29]</td>
<td>Transistors: 472; Area: 27 µm²</td>
<td>136.8 ns</td>
<td>351.2 mW</td>
</tr>
<tr>
<td>21</td>
<td>CAI Manasa &amp; Swapmna (2017) [33]</td>
<td>N/A</td>
<td>15.61 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>22</td>
<td>CSKA using MBFA Kumaran et al 2017 [31]</td>
<td>Transistors: 36 Area: 938 µm²</td>
<td>10.43 ns</td>
<td>N/A</td>
</tr>
<tr>
<td>23</td>
<td>NMOS PTL 4-bit (Vishalbharat 2017) [30]</td>
<td>N/A</td>
<td>78.84 ps</td>
<td>289.2 µW</td>
</tr>
<tr>
<td>24</td>
<td>90 nm CMOS 16-bit CSKA (Arora &amp; Niranjan 2017) [32]</td>
<td>N/A</td>
<td>35.2 ps</td>
<td>7.07 mW</td>
</tr>
</tbody>
</table>
APPENDIX B [ Large Figures can be placed here in single column format with the caption 
Figure B1, B2 etc., ]

Figure B1: Comparison of CSKAS using number of transistors

Figure B2: Comparison of CSKAS using critical path delay (ps)