PERFORMANCE IMPROVEMENT AND AREA OPTIMIZATION OF CARRY SPECULATIVE ADDITION USING MODIFIED CARRY GENERATORS

Y PRUDHVI BHASKAR
Department of ECE,
SASI Institute of Technology and Engineering,
Tadepalligudem, Andhra Pradesh, India

S K ALTHEMED ALI
Department of ECE,
SASI Institute of Technology and Engineering,
Tadepalligudem, Andhra Pradesh, India

B V PAVAN KUMAR,
Department of ECE,
SASI Institute of Technology and Engineering,
Tadepalligudem, Andhra Pradesh, India

Abstract

This paper proposes Carry Speculative Addition using Modified Carry generators to reduce critical path delay there by reducing area and power. This paper proposes Modified Carry generators uses less number of gates. A data latching circuit is modified to get continuous data into circuit. In order to generate accurate results, the CSPA using Modified Carry generators(CSPA-M) is implemented with error detection and error recovery circuits to construct a variable latency carry speculative adder(VLCSPA-M).The complete proposed Carry Speculative Addition using Modified Carry generators architecture is implemented using Verilog HDL and the design is simulated using ModelSim and Xilinx ISE 9.2i, Spartan 3 family device XC3S5000 -4FG1156.In this proposed architecture, the area is reduced by 10% and delay is reduced by 15%.

Key words: full adder, error detection, error recovery, speculation, critical path delay, variable latency.

1. INTRODUCTION

In any digital system, adder is the most crucial arithmetic circuit. For traditional adders, the critical path delay and area overhead are very high Ω (log n), Ω (n). Increasingly huge data sets and the need for instant response require the adder to be large and fast. Traditional ripple carry adder (RCA) is not suitable for large adders because of its low speed performance. To reduce the critical path delay and power consumption, approximate designs are used by sacrificing accuracy. Approximation can increase performance or reduce power consumption with a simplified or inaccurate circuit where strict requirements are relaxed. In [5] an accuracy-configurable approximate (ACA) adder for which the accuracy of results is configurable during runtime. Because of its configurability, the ACA adder can adaptively operate in both approximate (inaccurate) mode and accurate mode. The ACA adder achieves approximately 30% power reduction than conventional pipelined adder.

By adopting an emerging concept in VLSI design, error tolerance (ET), a novel error tolerant adder (ETA I, ETA II, ETA III, ETA IV) is proposed in [8], [12], [15], [16]. ETAI is divided into an accurate part and an inaccurate part to achieve approximate results. ETAI achieve tremendous improvements in both power consumption and speed performance. When the inputs are small numbers, the accuracy of the ETAI is poor. ETAII cuts carry propagation to speed up the Addition. The accuracy performance of the adder for small input operands is significantly improved using ETAII, while the accuracy of ETAII for large input operands is degraded than ETAI. The degraded accuracy performance for large input operands may restrict its usage. In ETAIII, the delay reduces and accuracy improves while power consumption improves. In ETAIV, the delay increases and accuracy is very high, while area is high. Variable latency designs may improve the performance of those circuits in which the worst case delay paths are infrequently activated. The basic principle that motivates the implementation of a variable latency resource is that of speeding up the process. Variable latency units exhibit the property that the number of cycles taken to compute their outputs varies depending on the input values.

Speculative technique is an optimization technique based on a prediction mechanism for improving the delay of arithmetic circuits. Speculative adders are built upon the observation that the critical path is rarely activated in traditional adders. In speculative adders, each output depends only on the previous k bits rather than all previous bits. Static window addition (SWA), a novel function speculation technique for the design of low area overhead, high performance variable latency adders is proposed in [10]. A block, called a window, includes several consecutive input bits. Grouping input bits into blocks, the carry chain length can be made comparable to the block size with high probability. Variable Latency addition using SWA based speculative adders is 10% faster than the fastest Design Ware adder with area requirements of 5 to 40% for different adder widths. In [2], a correlation aware speculative addition (CASA), which is a generic lightweight extension to existing speculative adders which intelligently exploits the correlation between the most significant bit of the input operands and the carry in values to improve the correctness of speculative adders. It shows that CASA achieves a significant reduction in error rate with small overhead in timing and area.

www.ijsret.org
Speculative carry select addition (SCSA) for the design of low error rate speculative adders and low area overhead, high performance, reliable variable latency adder is presented in [6]. The key idea is to segment the chain of propagate signals in addition into blocks of the same size. Specifically, the input bits of addends are segmented into blocks and the carry bits between blocks are selectively truncated to 0. All outputs of a block, instead of each output, are speculated together, which mitigates the area overhead problem. SCSA based speculative addition is 10% faster than the Design Ware adder with up to 43% area reduction. In [1], proposes a carry speculative adder, in which traditional full adder is modified into two separate carry generator and sum generator with an additional logic gate, which increase the speed with increase in power consumption. On simulation of carry speculative adder, it achieves a 26.59% delay reduction, a 14.06% area reduction and a 19.03% power consumption reduction compared to the corresponding values for an existing speculative carry-select adder.

II. EXISTING TECHNIQUE

2.1 CARRY SPECULATIVE ADDER

CARRY SPECULATIVE ADDER (CSPA) is used to reduce the critical path delay of the circuit which is based on carry speculation. The block diagram of the carry speculative adder (CSPA) is shown below in figure 1. The n-bit CSPA is divided into several small blocks adders that are operated independently and Carry Predictor Circuits. The size of each block adder is x-bit, except the leftmost block adder. Therefore, There are ‘m’ independent block adders and ‘(m − 1)’ carry predictor circuits in a CSA, where m= (n/x).

2.1.1 CARRY PREDICTOR

A carry predictor is used to predict the carryout bit of the corresponding block adder. To predict the carry-out bit of the corresponding block adder, carry predictor circuits uses the input bits that are near to MSB which reduces area and power consumption with minimal loss of accuracy. The probability of affecting the carryout bit is low when the inputs bits are near to LSB are used. Hence, a low error rate can be maintained, if the carry predictor circuit only uses the input bits near the MSB to predict the carryout bit, and the area overhead of the carry predictor circuits can be reduced. Since the probability of a carryout bit of the block adder depends on the k previous bit positions is 1/2^k because the probability of propagate signal P_i (a_i XOR b_i) having a value of 1 is 1/2 in each bit position.

2.1.2 BLOCK ADDER

Figure 2 represents the block diagram of the block adder. For an “x” bit block adder, there are two “x” bit carry generators, an “x” bit multiplexer and “x” bit sum generator.

2.1.2.1 CARRY GENERATOR AND SUM GENERATOR

In Traditional Full Adder, The carry bit is produced after three gate delays as shown below figure 3. A Modified full adder (MFA) is used in block adder of the CSPA to separate carry generator and sum generator as shown in figure 4. The MFA uses an additional logic gate compared to TFA, to reduce the delay of the carry bit which produces after two gate delays with higher power consumption. CSPA is implemented using two carry generators and a sum generator in block adders to reduce power consumption.
2.1.2.2 OPERATION OF BLOCK ADDER

An x-bit internal carry generator contains “x” 1-bit carry generators. The input bit $C_{out}^{(i-1)}$ is the carry out bit of the $(i-1)\text{th}$ block adder. This bit is used to choose one of the outputs from internal carry generators and the carry in bit of the sum generator. The sum generator produces the partial sum bits using the output from the internal carry generator that comes through multibit multiplexer and the input bit $C_{out}^{(i-1)}$.

The internal carry generator is used to produce internal carry signals that can be used in the sum generator. The output of the previous carry predictor circuit can be used as the selector of the multi bit multiplexer to select one of the internal carry signals. The sum generator is used to calculate the partial sum bits of the block adder.

2.1.2.3 OPERATION OF CSPA

When an input pattern is arrived, the internal carry generators and the carry predictor circuits operate parallelly. The internal carry generator produces corresponding internal carry signals with respect to carry in signal “1” and “0”. The carry predictor produces the predicted carryout bits of the block adders. The predicted carryout bit $(i-1)\text{th}$ is given to the $i\text{th}$ block adder and is used as the select signal to the multibit multiplexer and the carry-in bit of the sum generator. Based on the multibit multiplexer select signal, the corresponding internal carry signals are selected and are given as inputs to the sum generator to produce partial sum of the $i\text{th}$ block adder. The inputs to sum generator comes from the x bit input patterns, [x-2:0] bits from the multiplexor and (i-1)\text{th} carry predictor circuit. Since the carry predictor circuit uses input bit bear to MSB, The results generated from the CSPA are almost correct.

2.3 ERROR DETECTION

In CSPA, the addition operation is based on speculation which may produce accurate or inaccurate results. An error detection circuit is used to check whether produced results are accurate or inaccurate.

The block diagram of error detection circuit is shown in figure.5. This Error detection circuit also determines whether an error is occurred or not occurred. The operation of error detection is as follows. When the $C_{out}^{i}$ signal and $C_{out}^{i*}$ signal of each block adder are given to the error detection circuit to check whether these two signals are the equal or not. For this, an exclusive or (XOR) operation on $C_{out}^{i}$ and $C_{out}^{i*}$ is performed where $C_{out}^{i}$ is the carryout bit of the $i\text{th}$ block adder that is produced from the multiplexor and $C_{out}^{i*}$ is the predicted carryout bit produced by the $i\text{th}$ carry predictor circuit. The carry out bit of the $i\text{th}$ block adder is correct, If $C_{out}^{i}$ and $C_{out}^{i*}$ are the same. Otherwise, the predicted $C_{out}^{i*}$ signal is not correct. The error signal is obtained from the output of OR operation performed on the output of the XOR gates. The error signal is “1” when error detection circuit determines an Error otherwise error signal is “0”.

---

Fig.3. Block diagram of full adder

Fig.4.a Block diagram of Sum generator

Fig.4.b Block diagram of Carry generator

Fig.5. Block diagram of error detection
2.4 ERROR RECOVERY

The Error Recovery Circuit corrects the incorrect partial sum bits of the block adders according to the ERR_block signal. The block diagram of Error Recovery Circuit is shown in figure 6.

Sum is the partial sum bits of the ith block adder and ERR_block is the outputs of the error detection circuit. If ERR_block [i] is 1, it means that the carry out bit of ith carry predictor circuit and the partial sum bits of the (i+1)th block adder are incorrect. If ERR_block [i] is 0, it means that the (i + 1)th block adder generates correct partial sum bits. In error recovery circuits, Partial sum bits perform exclusive or operation with Error_block signal to obtain correct partial sum.

2.5 VARIABLE LATENCY CARRY SPECULATIVE ADDER

A variable latency design can reduce circuit timing waste when critical path delay is used as execution period. In variable latency design, two clock cycles are used. Figure 7 shows the CSPA with error detection and recovery circuits. When an input pattern is arrived, VLCSPA gives the result of the CSPA (i.e., SUM*) in a cycle. The error detection circuit gives Error_block signals and error signal. The Error_block signals that are generated by the error detection circuit indicate which block adder generated inaccurate results and the accurate results. The error signal indicates if an error is occurred or not.

If the error signal ER indicates “0” and the VALID signal is indicates “1” when the results are accurate. Then, the results calculated by the CSPA are correct and are used as the output. If the ER signal indicates “1” and the VALID signal is indicates “0” when the results are inaccurate, then the results calculated by the CSPA are not correct and the correct results are recovered from Error Recovery Circuit in one more cycle and result is given as SUMREC. If an error is occurred, the input registers are disabled and no new input is loaded in the circuit. The average latency of the VLCSPA is close to that of the CSP A since the error rate of the CSPA is low.

III. PROPOSED TECHNIQUE

3.1 CARRY SPECULATIVE ADDER USING MODIFIED CARRY GENERATORS

In CSPA, the block is implemented with modified full adder which separates Sum and Carry generators with an extra logic which leads to increase in power consumption and area increase. To reduce the area particularly, the carry generator are replaced with two separate carry generators for Cin=1 and Cin=0 to generate carry’s. The block diagrams of modified carry generators are shown in figure 8.

For Cin=0, the four gated 1bit carry generator is replaced with logic gate “AND”. Similarly, For Cin=1, the four gated 1bit carry generator is replaced with logic
gate “OR”. Thus in block adders, instead of two carry generators, we use two types of modified carry generator which provide one gate delay with reduced area. These carry generators generate carry’s simultaneously without using $C_{in}$ bit.

![Fig. 8.c block diagram of block adder with Modified carry generators](image)

3.2 VARIABLE LATENCY CARRY SPECULATIVE ADDER USING MODIFIED CARRY GENERATORS

In variable latency carry speculative adder, Carry Speculative Adder, error detection circuit, error recovery circuit, data latching circuit and multibit multiplexor are used as shown in figure 9. When an input pattern is arrived, VLCSPA gives the result of the CSPA (i.e., SUM) in a cycle. The error detection circuit gives Error_block signals and error signal. The error recovery circuit recovers the results when an error occurs based on the Error_block, the results from error recovery circuit and CSPA are given to multibit multiplexor and the error is used as multiplexor select signal. When the ER signal is “1” which selects recovers signal from error recovery circuit. When the ER signal is “0” which selects results from CSPA.

![Figure.9 CSPA modified implementation with Variable latency](image)

When an error is occurred, the input registers are disabled and no new input is latched in the circuit. This data latching which consists of not gate is replaced with Xor gate. In the latching circuit, an exclusive or operation is performed between error signal and complement of error signal. The valid signal enabling latch new data into input registers after the old data is recovered.

IV. EXPERIMENTAL RESULTS

Carry Speculative Addition using Modified Carry generators architecture is implemented in Verilog HDL and the design is simulated using ModelSim and Xilinx ISE 9.2i, Spartan 3 family device XC3S5000-4FG1156. This adder is implemented for four widths. In each case, the maximum combinational path delay and number of gates required is calculated. Table I and Table II show the results of CSPA adders for four widths. From the simulation results the performance of CSPA with modified carry speculative generators is improved by 15%. While area is decrease by 10%.

<table>
<thead>
<tr>
<th>ADDER WIDTH</th>
<th>DELAY(ns)</th>
<th>AREA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CSPA</td>
<td>CSPAM</td>
</tr>
<tr>
<td>32</td>
<td>15.877</td>
<td>13.878</td>
</tr>
<tr>
<td>64</td>
<td>16.230</td>
<td>13.899</td>
</tr>
<tr>
<td>128</td>
<td>16.515</td>
<td>13.940</td>
</tr>
<tr>
<td>256</td>
<td>16.733</td>
<td>14.061</td>
</tr>
</tbody>
</table>

Table II: Number of Gates of CSPA and CSPAM

<table>
<thead>
<tr>
<th>ADDER WIDTH</th>
<th>CSPA</th>
<th>CSPAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>531</td>
<td>423</td>
</tr>
<tr>
<td>64</td>
<td>1005</td>
<td>888</td>
</tr>
<tr>
<td>128</td>
<td>1950</td>
<td>1824</td>
</tr>
<tr>
<td>256</td>
<td>3846</td>
<td>3711</td>
</tr>
</tbody>
</table>

V. CONCLUSION

In this paper, two types of carry generators are used for input carry one and carry zero. Implementation of separate design, the carry out bit of the adder is produced in only one gate delay thereby decreasing the area overhead. A data latching
circuit is also implemented in variable latency design to use continuous input to adder without area overhead. This type of adders used in DSP applications, image processing and video compression applications involves large number of addition.

REFERENCES
[1] Ing-Chao Lin, Senior Member, IEEE, Yi-Ming Yang, and Cheng-Chian Lin, “High-Performance Low-Power Carry Speculative Addition With Variable Latency”, IEEE transactions on very large scale integration (vlsi) systems, 2014

AUTHOR PROFILE:
Y Prudhvi Bhaskar pursuing his M.Tech in Very Large Scale Integration from Sasi Institute of Technology and Engineering, West Godavari District, Tadepalligudem, Andhra Pradesh 534101.
Email: prudhvi460@gmail.com.

S K Ahemed Ali Working as Assistant Professor, from Electronics and communication Engineering in Sasi Institute of Technology and Engineering, West Godavari District, Tadepalligudem, Andhra Pradesh 534101.
Email: ahmedali@sasi.ac.in.

B V Pavan Kumar Working as Assistant Professor, from Electronics and communication Engineering in Sasi Institute of Technology and Engineering, West Godavari District, Tadepalligudem, Andhra Pradesh 534101.
Email: bvpavan@sasi.ac.in.