CHAPTER 6

OPTIMIZED HIGH-SPEED MULTI-DIGIT BCD ADDERS

A parallel high-speed multi-digit decimal adder is significant in a multi-programming future environment. Particularly the multi-digit decimal adder plays a vital role in the applications that cannot endure the loss of precision that consequences industrial, financial, and internet-based application. Even though, the decimal addition is the basic operation, it is the most essential unit among the decimal arithmetic processes in parallel multiplication to accumulate the partial products and in the division. The research on multi-digit BCD addition in QCA is limited. Hence, this chapter presents how efficiently the multi-digit BCD addition could be performed with higher speed. The main purpose of the proposed research work in this chapter is to present the multi-digit BCD adders to perform the BCD/Decimal addition with less complexity and higher speed.

6.1 Introduction

In this chapter, two multi-digit BCD adders are presented to perform the decimal addition with the optimized area and with lesser delay. The first Multi-Digit BCD Adder (MDBA) is constructed with cascading the SDBA design. The Hybrid Multi-Digit BCD Adder (HMDBA) is proposed with the combination of CFA and CLA-based BCD adders. HMDBA accomplishes the two 4-digit, 8-digit Decimal addition with 36% higher speed compared to the CFA-based BCD adder with minimum overhead in terms of area. Whereas the HMDBA achieves the decrease in the cell count 38% for 4-digit and 29% for 8-digit compared to the CLA-based adder. Also, the HMDBA occupies less area of 23% and 14% with the same speed to perform the decimal addition of two 4-digit, 8-digit numbers respectively. The overall cost of the HMDBA is decreased by 50% compared to the CFA-based BCD adder and a nominal amount of reduction compared to the CLA-based BCD adder.

In the second method, a new design called Enhanced Multi Digit BCD Adder (EMDBA) is proposed by cascading the Enhanced Single-Digit BCD Adders (ESDBA)
for higher order (n-digit) BCD adders. The EMDBA performs two 4-digit BCD addition with 53% faster and for two 8-digit BCD addition 59% faster than the CFA-based BCD adder with the nominal overhead of area. The EMDBA performs two 4-digit BCD addition achieves 24% speed, 33% reduction in the cell count and 23% less area. Similarly, for 8-digit operation, the EMDBA achieves 36% speed, 29% reduction in cell count and 21% less area compared to the existing CLA-based BCD adder design. The overall cost of the proposed EMDBA achieves 45%, 64% less for 4-digit and 8-digit respectively compared to the HMDBA method. The proposed EMDBA multi-digit adder produces significantly less delay of (N-1) + 3.5 clock cycles compared to the N*one digit BCD adder delay required by the conventional BCD adder method. From the obtained results compared to the existing results, it is noticed that as per our knowledge this is the first innovative proposal for multi-digit BCD addition using QCA.

The remaining part of the chapter is organized as follows. In Section 6.2, the proposed design methodology of HMDBA is described in detail with its block diagram and layout. Section 6.3 provides simulation results and discussions of the proposed HMDBA design parameters with the existing designs. In Section 6.4, the EMDBA design methodology is described in detail with its block diagram and layout of 4-digit and 8-digits. Section 6.5 explains the simulation results of EMDBA and its performance parameters comparison with the existing designs. Section 6.6 concludes the proposed designs and its future work.

6.2 Design Methodology of HMDBA

In this section, MDBA is designed as a hybrid structure with the combination of binary adder as CFA Type-II structure and Multiplexers. HMDBA is the first hybrid structure designed to perform the multi-operand Decimal addition with higher speed and less complexity. The proposed HMDBA block diagram is shown in Figure 6.1. The SDBA design is extended, which is explained in detail in Chapter 5, Section 5.2 and cascaded to form the HMDBA design. Commonly, the more time consumption to implement the decimal addition is due it two subsections rather than ordinary binary addition. The subsections, which impact primarily on the decimal addition, are upper 4-bit binary adder to produce the binary sum and later one is lower binary adder to
generate the valid decimal sum. This design is mainly focused on performing the decimal addition by eliminating the delay occurred due to the carry propagation in the final part of the design with the multiplexers instead of a binary adder.

In the upper part of the HMDBA design, CFA Type-II four-bit binary adder is used to generate the binary sum. The Design of the CFA uses conventional carry propagation techniques, but it is optimized only for QCA circuit layouts. The CFA has three different adder types. The parallel CFA is chosen based on the maximum clock zone length limitations. CFA Type-I is a more compact design but requires more wire channels for input synchronization. CFA Type-II reduces the wire channels so that overall design has a smaller Area [29].

![Block diagram of proposed HMDBA design](image)

**Fig. 6.1** Block diagram of proposed HMDBA design
In addition to the existing conventional correction logic, additional logic circuitry is required to generate the inputs to the multiplexers. The additional circuitry logic is similar to that of SDBA as described in Chapter 5, Section 5.2. The layout of HMDBA to perform 4-digit and 8-digit Decimal addition is shown in Figures 6.2 and 6.3 respectively.

Fig. 6.2 Layout of the proposed HMDBA 4-digit BCD adder

Fig. 6.3 Layout of proposed HMDBA 8-digit BCD adder
6.3 Results and Discussions

All the design parameters used for HMDBA are similar to that of SDBA design explained in Chapter 5, Section 5.3. The simulation results of 4-digit HMDBA design is shown in Figure 6.4.

![Simulation Results](image)

**Fig. 6.4** Simulation results of proposed HMDBA 4-digit BCD adder design
The layout of the 4-digit HMDBA is simulated using Coherence vector engine to obtain the results. In simulation results, the input and output buses are shown as per digit-wise due to the limited graphic capability of the QCADesigner. The first SDBA used to add the least significant decimal digit requires 2.75 clock cycles, out of these two clock cycles are needed to produce the binary sum, correction logic as well as inputs to the multiplexers in the final part of the Decimal Adder. To cascade the correction logic output to the next higher order digit requires a delay of 2 clock cycles. Hence, the subsequent digit data inputs are delayed by two clock cycles for synchronizing all the inputs and outputs. Therefore, the 4-digit HMDBA design produces the significant output after 8.75 clock cycles. From the simulation results it is clearly displayed that the significant output appears after 8.75 clock cycles in clock 2 (clock zone 2), which is marked in Figure 6.4. The performance parameters obtained from the layout and simulation results are shown in Table 6.1.

Table 6.1 shows the performance parameters analysis such as the number of cells (cell count), area, delay (latency) and the number of clock zones of HMDBA with existing QCA based Multi-digit BCD adders. The existing single-digit BCD adders are constructed similar to that of conventional methodology with slight modifications. In the current designs, the final part of the BCD adder uses the 3-bit ripple carry binary adder structure instead of using 4-bit binary adder. As per the correction logic design no adder is required in the least significant bit (LSB) position, therefore in the final part 3-bit adder is enough to produce the valid Decimal sum. Hence, these designs are extended to perform the multi-digit BCD addition requires more area and delay [85], [87] and [88]. Even though the Johnson–Mobius adder is a different technique adopted, the serial JMC adder occupies less area with more delay, whereas the parallel JMC adder requires more hardware therefore again it leads to larger area requirements as well as more delay [86].

Even though the CLA-based BCD design is distinct from the conventional structure, the circuit complexity is more due to the generate and propagate functions, so it occupies more area compared to the SDBA used in the HMDBA. Since the proposed HMDBA design is a combination of CFA-based and CLA-based BCD designs, it occupies less area and achieves higher speed compared to the existing CLA-based BCD design.
Table 6.1: Performance comparison of the HMDBA with the existing designs

<table>
<thead>
<tr>
<th>Adder</th>
<th>Number of cells</th>
<th>Area (μm²)</th>
<th>Delay</th>
<th>Clock Zones</th>
<th>Cost (Area* latency²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-digit BCD Adder[85]</td>
<td>5674</td>
<td>10.22</td>
<td>26</td>
<td>104</td>
<td>6908.72</td>
</tr>
<tr>
<td>8-digit BCD Adder [85]</td>
<td>11622</td>
<td>21.96</td>
<td>52</td>
<td>208</td>
<td>59379.84</td>
</tr>
<tr>
<td>4-digit Serial JMC Adder [86]</td>
<td>1130</td>
<td>1.77</td>
<td>40</td>
<td>160</td>
<td>2832</td>
</tr>
<tr>
<td>8-digit Serial JMC Adder [86]</td>
<td>1130</td>
<td>1.77</td>
<td>80</td>
<td>320</td>
<td>11328</td>
</tr>
<tr>
<td>4-digit Parallel JMC Adder [86]</td>
<td>14470</td>
<td>17.05</td>
<td>23</td>
<td>92</td>
<td>9019.45</td>
</tr>
<tr>
<td>8-digit Parallel JMC Adder [86]</td>
<td>29165</td>
<td>36.1</td>
<td>46</td>
<td>184</td>
<td>76387.6</td>
</tr>
<tr>
<td>4-digit Efficient BCD Adder [87]</td>
<td>7942</td>
<td>23.95</td>
<td>29</td>
<td>116</td>
<td>20141.95</td>
</tr>
<tr>
<td>8-digit Efficient BCD Adder [87]</td>
<td>16214</td>
<td>49.32</td>
<td>61</td>
<td>244</td>
<td>183519.72</td>
</tr>
<tr>
<td>4-digit CFA-based BCD Adder [88]</td>
<td>3842</td>
<td>6.39</td>
<td>13.75</td>
<td>55</td>
<td>1208</td>
</tr>
<tr>
<td>8-digit CFA-based BCD Adder [88]</td>
<td>7798</td>
<td>14.32</td>
<td>25.75</td>
<td>103</td>
<td>9495</td>
</tr>
<tr>
<td>4-digit CLA-based BCD Adder [88]</td>
<td>7592</td>
<td>8.4</td>
<td>8.5</td>
<td>34</td>
<td>606.9</td>
</tr>
<tr>
<td>8-digit CLA-based BCD Adder [88]</td>
<td>15424</td>
<td>19.91</td>
<td>16.5</td>
<td>66</td>
<td>5420.49</td>
</tr>
<tr>
<td>Proposed HMDBA 4-digit BCD Adder</td>
<td>4690</td>
<td>6.42</td>
<td>8.75</td>
<td>35</td>
<td>491.53</td>
</tr>
<tr>
<td>Proposed HMDBA 8-digit BCD Adder</td>
<td>10944</td>
<td>17.04</td>
<td>16.75</td>
<td>67</td>
<td>4780.785</td>
</tr>
</tbody>
</table>

It is observed from the results shown in Table 6.1 that, a significant improvement in speed is achieved compared to the CFA-based BCD adder. In addition, it is noted that the implementation of this method achieves higher speed with a nominal amount of area increment compared to the CFA-based BCD adder. From the Table 6.1,
it is also observed that the HMDBA occupies 6.42 $\mu m^2$ only with 4690 cells to perform the two 4-digit decimal addition compared to the 8.4 $\mu m^2$ and 7592 cells in CLA-based BCD adder with similar speed. The CLA-based BCD adder takes 2.5 clock cycles for single-digit BCD addition itself when it is cascaded to perform the multi-digit operation it requires more wire channels to propagate the carry to the upper BCD adder part. Similarly, the simulation results obtained with the HMDBA for 8-digit operation also achieved less area with competing speed compared to the CLA-based BCD adder. Furthermore, the overall cost of the HMDBA is less compared to the all the existing designs.

6.4 EMDBA Design Approach

The main objective of this EMDBA design is to minimize the delay overhead occurred due to the carry propagation in the multi-digit BCD adder. The EMDBA is a parallel multi-digit BCD adder designed with new correction logic to accomplish the decimal addition in a faster manner. Figure 6.5 shows the proposed design with input carry to the lower 4-bit adder instead of giving to upper 4-bit adder, which overcomes the carry propagation overhead from one digit to the next digit. All the n-number of BCD digits are added concurrently to produce the binary sum, without waiting for input carry propagation in the upper part of the design.

In the proposed design a 4-bit area-delay efficient binary adder is used, and the carry chain in the binary adder is optimized using the generate (G) and propagate (P) functions [125]. Propagate signal is computed as $P_i = M(A_i, B_i, 1)$ and generate signal as $G_i = M(A_i, B_i, 0)$. The carry computation to the next bit in the area-delay efficient binary adder is computed by $C_{i+1} = M(P_i, G_i, C_i)$. Further, it is designed by considering the $C_{in} = \text{‘0’}$ (carry-in) in the LSB position and without the requirement of the propagation of $P_0$, thus the carry propagation path is simplified. Therefore, it utilizes lesser-cascaded majority gates to propagate the carry signal. Besides this, the area-delay efficient adder uses basic logic and the minimum number of clock cycles in the layout. Also, the proposed methodology uses the carry-in = ‘0’ in the upper part. Hence, the area-delay efficient binary adder suits well to the proposed correction logic. In addition
to the new correction logic, the best-optimized type of CFA Type-II binary adder [23] is used to diminish the delay in the final part of the BCD adder to produce the final Decimal sum.

To design of EMDBA shown in Figure 6.5 is implemented by extending the ESDBA correction logic, which is explained in detail in Chapter 5, Section 5.4. The extension of ESDBA logic expressions are shown in equations from 6.1 to 6.4.

\[
X_0 = M[M(M(S_3, M(S_2, S_1, 1), 0), C_4, 1), M(M(S_3, S_0, 0), C_{in}, 0), 1)] \quad (6.1)
\]

Similarly, the correction logic for second digit decimal adder is shown in Equation (6.2).

\[
X_1 = M[M(M(S_7, M(S_5, S_6, 1), 0), C_8, 1), M(M(S_4, S_7, 0), X_0, 0), 1] \quad (6.2)
\]

The correction logic for third-digit decimal adder is shown in Equation (6.3).

\[
X_2 = M[M(M(S_{11}, M(S_9, S_{10}, 1), 0), C_{12}, 1), M(M(S_8, S_{11}, 0), X_1, 0), 1] \quad (6.3)
\]

The correction logic for Fourth-digit decimal adder is shown in Equation (6.4).

\[
X_3 = M[M(M(S_{15}, M(S_{14}, S_{13}, 1), 0), C_{16}, 1), M(M(S_{15}, S_{12}, 0), X_2, 0), 1] \quad (6.4)
\]

Similarly, the same correction logic is used for N-digit BCD adder.

Figure 6.5 shows the implementation of the correction logic part up to N-1 digits for constructing the N-digit BCD adder.

In the proposed design, the input carry is taken as zero to compute the binary sum of an upper 4-bit binary adder. The correction logic output from the previously available digit is given as carry input to the lower 4-bit RCA to the successive stages for anticipating the delay overhead of input carry.

By cascading the carry out of one digit to the next digit, the carry out is delayed by only one clock cycle with the implementation of correction logic. Hence, when increasing the number of digits, one clock cycle is increased to get the correct BCD sum in the output compared to the current single digit BCD adder (ESDBA). The single-digit BCD adder in the first stage takes 3\(1/2\) clock cycles to perform the addition of lower significant decimal part. The subsequent stages take one excess clock cycle, which is needed for correction.
6.5 Results and Discussions

The EMDBA simulation results are obtained with the help of the Coherence vector simulation method. Default settings (Euler Method and Randomize Simulation Order option) have been chosen for the simulation of the proposed BCD adder. The layouts of proposed 4-digit BCD and 8-digit BCD adders are shown in Figures 6.6 and 6.7 respectively.

Fig. 6.5 Block diagram of proposed EMDBA N-digit BCD Adder
From the simulation results, the latency (delay) achieved for the proposed 4-digit BCD adder is 6.5 clock cycles, which is shown in Figure 6.8. The first significant
output appeared in the second clock (clock zone 1) tick after 6.5 clock delays and highlighted in Figure 6.8.

**Fig. 6.8** Simulation results of proposed EMDBA 4-digit BCD adder design

The proposed EMDBA design is compared with the other existing adder designs, and the comparisons are based on precise circuit layouts and simulations. The performance results obtained with the proposed multi-digit design is compared with the existing single-digit designs such as the number of cells, area, delay and number of clock zones. The comparison results shown are achieved with the cell size of 18 nm x 18 nm. The comparison of the proposed BCD adder with other existing BCD adder methods with the number of cells consumption is shown in Figure 6.9.
From Figure 6.9 it is observed that the number of cells of EMDBA method achieved with 4-digit BCD adder is 5069 and with 8-digit BCD adder is 10899 competing to the HMDBA design for BCD addition. The number of cells required by EMDBA is less compared to the CLA-based BCD adder with the reduced complexity in the correction logic as well as improving the regularity in the upper part of the parallel binary addition. However, the number of cells required by EMDBA is slightly higher than the CFA-based BCD adder because of generate and propagate functions to perform the parallel binary addition for N number of digits without propagation of the input carry in the upper part of the BCD addition from LSDD to MSDD. Also, the final part of the design optimizes by using the CFA Type-II adder to make fewer wire channels, which results in the lesser area.

**Fig. 6.9** Comparison of EMDBA performance in terms of number of cells with existing methods

![Comparison of EMDBA performance in terms of number of cells with existing methods](image)
The proposed EMDBA is compared with the existing methods with respect to the area ($\mu m^2$) consumption is shown in Figure 6.10.

![Comparison of EMDBA performance in terms of area ($\mu m^2$) with existing methods](image)

**Fig. 6.10** Comparison of EMDBA performance in terms of area ($\mu m^2$) with existing methods

From Figure 6.10 it is observed that the area ($\mu m^2$) of EMDBA method achieved with 4-digit BCD adder is 6.44 and with 8-digit BCD adder is 15.82 for BCD addition and the EMDBA method occupies the less space compared to the CLA-based and achieves high speed compared to the other existing methods. The comparison of the proposed BCD adder with other existing BCD adder methods with respect to the clock zones consumption is shown in Figure 6.11. From Figure 6.11 it is observed that the clock zones of EMDBA method achieved with 4-digit BCD adder are 26 and with 8-digit BCD adder are 42 for performing the BCD addition. Also noticed that the EMDBA method takes comparatively fewer clock zones compared to the other existing methods.
The comparison of the proposed BCD adder with other existing BCD adder methods on the delay consumption is shown in Figure 6.12. From Figure 6.12 it is observed that the delay (clock cycles) of EMDBA method achieved with 4-digit BCD adder is 6.5 and with 8-digit BCD adder is 10.5 for BCD addition. Also noticed that the EMDBA method takes comparatively less delay compared to the other existing methods. This less delay is achieved with the new correction logic and by cascading the single digit BCD adder in an optimized manner to anticipate the ripple carry propagation. It is noticed from the obtained results that the proposed 8-digit BCD adder utilized the same number of clocks required for conventional 4-digit BCD adder. Hence, the proposed design is much faster than the existing designs.
The comparison of the proposed EMDBA with other existing BCD adder methods regarding overall cost function evaluated by using CMOS cost function [124] is shown in Figure 6.13. Figure 6.13 clearly showed as the proposed design is consuming less cost compared to the existing designs. The lowest overall cost is attained because of higher speed and an optimized area. Furthermore, the overall cost of the proposed EMDBA achieves 45%, 64% less for 4-digit and 8-digit HMDBA design individually and if the number digits increase, it will reduce the overall cost further. From the obtained results and discussions, it is found that the proposed EMDBA is the optimized design to perform the multi-digit decimal addition compared to the existing designs. According to the QCA cost function discussed in Chapter 3, Section 3.3, and Chapter 5, Section 5.5 the QCA cost function is evaluated for multi-digit BCD adders.
The QCA cost function is evaluated concerning both area \((P = 2)\) and speed \(P = 4\). The comparison of QCA cost function of multi-digit BCD adders for different length of digits is shown in Figure 6.14 with \(P = 2\) reflecting the complexity of the design in terms of majority gates and number of crossovers. As seen from the Figure 6.14, it is observed that the HMDBA design and EMDBA design achieves the best performance compared to the existing designs. Figure 6.15 displays the QCA cost function regarding speed. The QCA cost function increases with increase in the number of bits in the earlier designs. However, the increase in the cost function is less in EMDBA with the increase of number of bits that can be clearly identified from the Figure 6.15. The QCA cost function of EMDBA design is reduced by 50% to 75% if the number of digits is increased due to the novel correction logic and regularity of the structure.

**Fig. 6.13** Overall cost (Area x Latency\(^2\)) of EMDBA design

The QCA cost function is evaluated concerning both area \((P = 2)\) and speed \(P = 4\). The comparison of QCA cost function of multi-digit BCD adders for different length of digits is shown in Figure 6.14 with \(P = 2\) reflecting the complexity of the design in terms of majority gates and number of crossovers. As seen from the Figure 6.14, it is observed that the HMDBA design and EMDBA design achieves the best performance compared to the existing designs. Figure 6.15 displays the QCA cost function regarding speed. The QCA cost function increases with increase in the number of bits in the earlier designs. However, the increase in the cost function is less in EMDBA with the increase of number of bits that can be clearly identified from the Figure 6.15. The QCA cost function of EMDBA design is reduced by 50% to 75% if the number of digits is increased due to the novel correction logic and regularity of the structure.
A new model of BCD adder is implemented in combination with CFA Type-II binary adder and multiplexer is presented to perform the multi-digit BCD addition using QCA. The comparison of the HMDBA with the various existing methods clearly showed that the HMDBA offers better trade-off with the speed and area between CFA
and CLA-based BCD adders. The proposed HMDBA design is significantly faster compared to CFA-based BCD adder and utilized less number of cells compared to the CLA-based BCD adder. From the obtained results it is noticed that N-digit proposed HMDBA design has a delay of (N-1)*2 + 2.75 clock cycles (2.75 clock cycles is the delay of the first BCD adder) compared to the conventional design methods expected delay of N*One digit BCD adder. Moreover, the overall cost of the HMDBA is reduced by 50% compared to the CFA-based BCD adder. Hence, the proposed design is an optimized design for multi-digit BCD operations, however, still there is a possibility to improve the speed of the multi-digit BCD addition by optimizing the input carry path. Therefore, a new concept of carry-propagate decimal addition algorithm is introduced in the EMDBA. A new correction logic is proposed to design multi-digit BCD adder and implemented using QCA.

The EMDBA offers the best trade-off among all the existing BCD adders. The ESDBA and EMDBA adders are designed based on a new carry-propagate algorithm for Decimal addition. The multi-digit BCD adder designed with ESDBA is the first innovative design to do the Multi-digit decimal operation with higher speed and less complexity. Decimal adders based on the new carry-propagate algorithm performs the operations in a faster manner with less complexity. The EMDBA has a delay of (N-1) + 3.5 clock cycles (3.5 clock cycles is the delay of the first BCD adder) compared to the conventional design methods predicted delay of N*One digit BCD adder. The EMDBA proves how effectively the multi-digit operation can be performed by reducing the delay caused by the carry propagation. Hence, the EMDBA is well suitable for real-time applications in future nano computing processors.