# FPGA Implementation of Low Power Hardware Efficient Flagged Binary Coded Decimal Adder

K.N.Vijeyakumar
Assistant Professor, Ece,
Anna University Of Technology,
Coimbatore, Coimbatore - 641047.

V.Sumathy
Assistant Professor, Ece,
Government College Of Technology,
Thadagam Road, Coimbatore-641013.

A.Dinesh Babu, S.Elango, S.Saravanakumar
Pg Scholars, Ece,
Anna University Of Technology Coimbatore,
Jothipuram, Coimbatore - 641047.

## ABSTRACT
This paper presents a novel architecture for hardware efficient binary represented decimal addition. We extend the two operand ripple carry addition by one with the third input being constant. The addition technique is made fast by generating flag bits appropriate to the constant added. The third constant in case of our proposed design is 6(0110) for converting the outputs exceeding 9 to Binary Coded Decimal (BCD) number. The proposed BCD adder has been designed using VHDL code and synthesized using Altera Quartus II. Experimental results show that the proposed design outperforms the previous researches in terms of power dissipation and area.

## Key Words
Flagged binary adder, Carry look ahead adder, Carry skip adder, Correction circuit, flag bit computation.

## 1. INTRODUCTION AND RELATED WORK
The binary numbering system is, by far, the most common numbering system in use in computer systems today. In days long, however, there were computer systems that were based on the decimal (base 10) numbering system rather than the binary numbering system. Such computer systems were very popular in systems targeted for business/commercial systems. Although systems designers have discovered that binary arithmetic is almost always better than decimal arithmetic for general calculations, the myth still persists that decimal arithmetic is better for money calculations than binary arithmetic. Therefore, many software systems still specify the use of decimal arithmetic in their calculations [16].

BCD representation does offer one big advantage over binary representation: it is fairly trivial to convert between the string representation of a decimal number and its BCD representation. This feature is particularly beneficial when working with fractional values since fixed and floating point binary representations cannot exactly represent many commonly used values between zero and one (e.g., 1/10). Therefore, BCD operations can be efficient when reading from a BCD device, doing a simple arithmetic operation (e.g., a single addition) and then writing the BCD value to some other device. Many architectures and algorithms have been proposed to date for decimal arithmetic. Behrooz shirazi et al., [2] designed an adder for redundant BCD addition. Though the design involves simple conversion of a BCD number to redundant BCD and perform addition in redundant form and again convert the result back to BCD form, it suffers from high delay .In addition the conversion circuitry used adds to the design complexity. Algorithms and adders for BCD addition are presented in [3] by Robert D.Kenny et al., in which the design using speculative addition technique have regular structure and their correction unit is independent of the number of input operands whereas the other design using non-speculative addition have lower delays.

The use of reversible logic for the design of BCD adders were presented in [4][6][17]. The reversible logic adders perform well in terms of power dissipation and logic count, however it is prone to higher delay. In a pioneer work on BCD addition Sreehari Veeramachaneni *et al.*,[7] used multiplexer for the addition of correction bits. The circuit has lower delay compared to the architectures using adders for correction bit addition.

Alp Arslan Bayrakci *et al*.,[8] proposed a BCD adder with efficient carry generation using analyzer circuit. The circuit performs well in terms of delay compared to architectures mentioned in literature and shows better area performance. Anshwal Singh *et al*.,[9] designed a novel architecture for BCD addition and subtraction . The design uses three major blocks viz., PG block, prefix block and the correction block and generates carry without any extra logic thus performing better in terms of area performance compared to the BCD adder in [10].

Sundaresan.C *et al*., [11] in a pioneer work on design of Reduced delay BCD adder used Carry Look Ahead (CLA) adder in the initial stage being followed by carry network and correction logic in the second and third stages. Though the circuit is fast compared to the architecture in [12], the use of CLA adder in initial stage increases area cost. Chetan Kumar *et al*., [12] presented a unified architecture for BCD and binary addition. Though the circuit has lower delay compared to the architectures mentioned in literature the design of post correction circuitry poses problems for multi-bit operands. Osama Al-khaleel *et al*., [13] proposed a correction free BCD adder in which the input operands are split and added in two stages. Stage 1 adds the MSB three bits of a four bit BCD number and the result out of stage 1 is passed to stage 2 and added with the LSB. The latency of the architecture is very less compared to the architectures mentioned in literature.

To further reduce power and latency in BCD addition we have proposed an adder using flagged binary addition[1] for the correction constant addition. The output of adders of first

stage and flagged computation block are passed through a multiplexer. The control signal for the multiplexer is generated from a control circuit which produces 1 for sum values exceeding 9 and 0 else. The rest of the paper is organized as follows. Section 2 gives a brief description of BCD addition and the architecture of conventional BCD adder. Section 3 discusses about the design of proposed flagged BCD adder. In section 4 the performance of the proposed BCD adder are discussed and compared with the previous approaches. Section 5 gives brief conclusion of the work done.

## 2. OVERVIEW OF BCD ADDITION

In electronic systems, BCD is an encoding for decimal numbers in which each digit is represented by its own binary sequence. It allows easy conversion to digits and results in faster calculations. When BCD numbers are added, each sum digit should be adjusted to skip the six unused codes. For instance, the addition of two decimal digits in BCD, together with a possible carry from a previous least significant pair of digits (assuming maximum value for input digits) viz., 9 + 9 +1 would result in 19 .The equivalent binary sum will be in the range 0 to 19 represented in binary as 0000 to 10011and BCD as 0000 to 1 1001(the first 1 being carry and next four bits being BCD digit sum). For the binary sum equal to or less than 1001 the corresponding BCD digit is correct. However when the binary sum exceeds 1001, the result is invalid BCD digit. The addition of $6(0110)_2$ to the binary sum converts it to the correct digit and also produces carry [14]. Fig.1 shows the block diagram of a 1 digit BCD adder [14] based on the above methodology.

The input digits in binary are $A_3A_2A_1A_0$ and $B_3B_2B_1B_0$. $S_3$' $S_2$'$S_1$'$S_0$'are the outputs of the first stage 4 bit adder, to which correction bits 0110(6) is added at the second stage to produce the BCD number $S_3S_2S_1S_0$ shown in equation (2)-(5)along with carry output $C_N$ shown in equation(1).The carry $C_N$ will be one for digits exceeding 9 or else it will be 0.

$$C_N = C_{OUT} + S_3'S_2' + S_3'S_2' \quad ---(1)$$
$$S_0 = B_0 \quad ---(2)$$
$$S_1 = B_3'B_1 + B_3B_2B_1 \quad ---(3)$$
$$S_2 = B_3'B_2 + B_2B_1 \quad ---(4)$$
$$S_3 = B_3B_2'B_1' \quad ---(5)$$

## 3. PROPOSED FLAGGED BCD ADDER

We have proposed a hardware efficient BCD adder using flagged binary addition [1] technique proposed by Vibhuti Dave et al., . The various blocks of the proposed BCD adder are 4 bit Ripple Carry Adder(RCA), Excess 9 detector, flag bit computation block, flag inversion block and four 2:1 multiplexers whose schematic is shown in figure 3**.** The input *A* $(a_3a_2a_1a_0)$ and $B(b_3b_2b_1b_0)$ are fed to the first stage binary adder**.** The sum output $S(S_3S_2S_1S_0)$ and carry out $C_o$ of this stage is fed to Excess 9 detector shown in figure 3(a). If the sum $S(S_3S_2S_1S_0)$ is less than or equal to 9 the $C_{out}$ of Excess 9 detector will be zero and the sum $S(S_3S_2S_1S_0)$ will be passed out through the multiplexer. If the sum $S(S_3S_2S_1S_0)$ exceeds 9, the $C_{out}$ of Excess 9 detector will be 1 and the sum bits will be passed through the flag bit computation block to generate intermediate carry bits $(d_4d_3d_2d_1)$ shown in equation (6)-(9).

$$d_1 = d_0 \ \& \ s_0 \quad ---(6)$$
$$d_2 = d_1 + s_1 \quad ---(7)$$
$$d_3 = d_2 + s_2 \quad ---(8)$$
$$d_4 = d_3 + s_3 \quad ---(9)$$

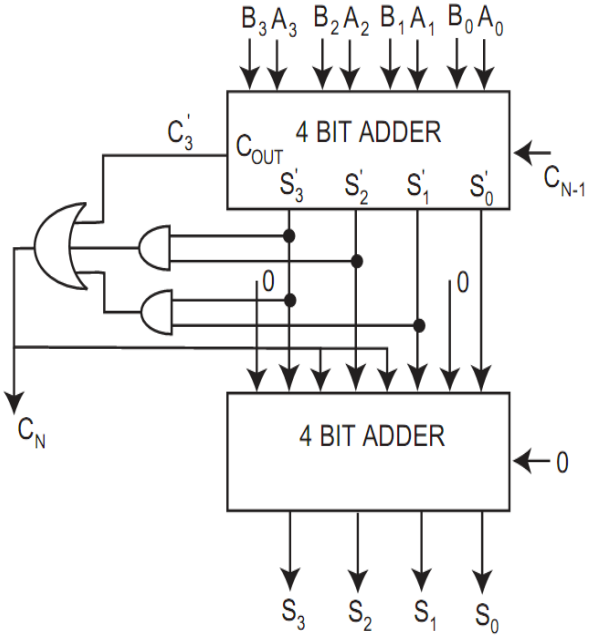



**Fig.1 Block Diagram of BCD Adder**

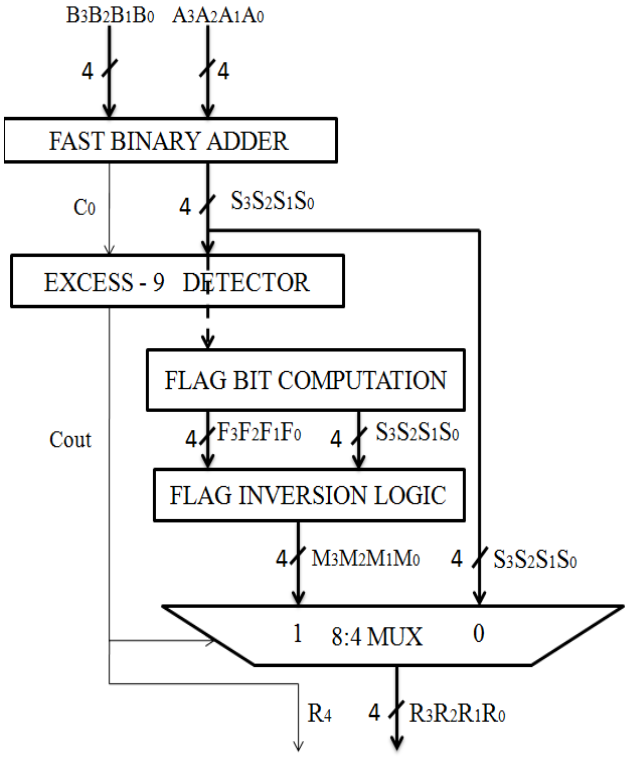
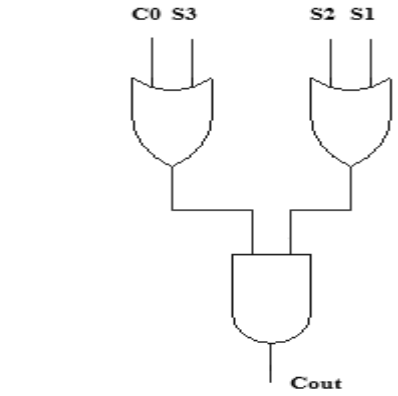
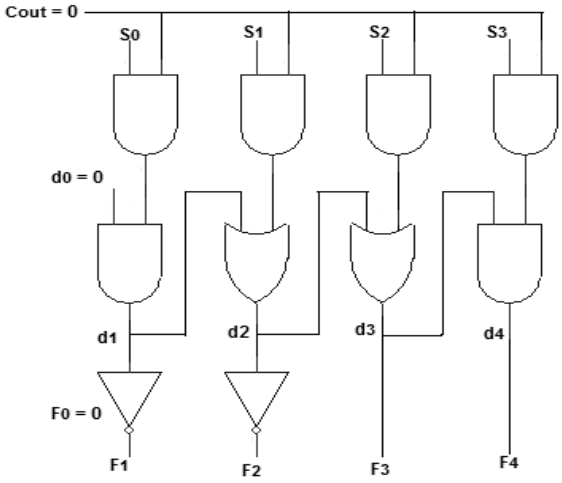


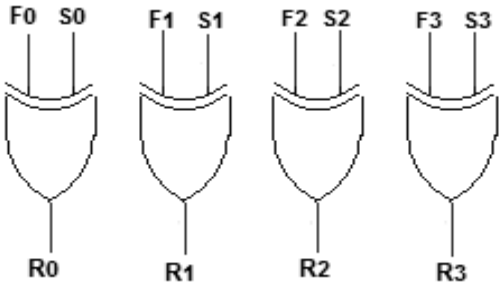

**Fig.2. Block diagram of proposed flagged BCD adder**




**(a)**




**(b)**




**(c)**

**Fig 3. Schematic of (a) Excess-9 Detector, (b)Carry computation and Flag bit computation block (c) Flag inversion logic**

The carry bits*($d_4d_3d_2d_1$)* and sum *$S(S_3S_2S_1S_0)$* are then used by this block to generate flag bits *($F_0$,$F_1$,$F_2$,$F_3$)* shown in equation (10) -(13).

$$F_1 = d_1 \quad ---(10)$$
$$F_2 = d_2 \quad ---(11)$$
$$F_3 = d_3 \quad ---(12)$$
$$F_4 = d_4 \quad ---(13)$$

The flag bits*($F_0$,$F_1$,$F_2$,$F_3$)* and sum *$S(S_3S_2S_1S_0)$* are passed through flag inversion logic shown in fig.3(c) to generate the BCD output $M_3M_2M_1M_0$ for *$S(C_oS_3S_2S_1S_0)$* which exceeds 9. The $M_3M_2M_1M_0$ of the flagged inversion block forms the other input to the multiplexer which is passed out for 1 value of $C_{out}$. The pseudo code for the proposed BCD adder is shown in figure.4.

```
A (A3A2A1A0), B (B3B2B1B0) inputs; Carry in (Cin) = 0; S(S3S2S1S0) First stage adder output,
M (M3M2M1M0) - Flagged binary adder output. R (R3R2R1R0) - BCD addition output, d0=F0=0;
FAs-Full adder sum,  Ac-Full adder Carry.

Step 1:First stage addition  /*Generation of sum of A & B whose value is less than or equal to 9*/

        S0 = FAs (a0,b0,Cin);
        C0 = FAc (a0,b0,Cin);

        For ( i=1 to 3)
            {
            Si = FAs (ai,bi,Ci-1)
            Ci = FAc (ai,bi,Ci-1)
            i=i+1;
            }
            Co= C3;
            S=S3S2S1S0;

Step 2:Generation of Select signal for multiplexer(Cout) using Excess 9 detector

              Cout =(C0 or S3) or (S2  and S1);

Step3: Carry and flag bit  computation

          If Cout=1 then
           {
           d1=d0 and s0;   /*Carry bit generation for addition of correction bias 6(0110)*/
           d2=d1 or s1;
           d3=d2 or s2;
           d4=d3and s3;
  F0 = 0;
           F1=not d1; /*Flag bit generation for addition of correction bias 6(0110)*/
           F2=not d2;
           F3=d3;
           F4=d4;
           }
Step 4 :Generation of Flagged binary output

          If Cout  = 1 then
          {
           M0 = F0 xor  S0;/*Output generation with addition of correction bias 0110*/
           M1 = F1 xor S1;
           M2 = F2 xor S2;
           M3 = F3 xor S3;
          }

Step 5: Final output generation using Multiplexer

 If Cout = 1 then

             R(R4R3R2R1R0) =  '1'M3 M2M1M0;
          else
             R(R4R3R2R1R0) =  '0'S3S2S1S0;
```

**Fig 4. Pseudo code for proposed flagged BCD adder**

## 4. RESULTS AND DISCUSSION

The proposed flagged BCD adder is described using structural VHDL to produce gate level net list and synthesized using Altera Quartus II. We evaluated the proposed BCD adder design using Carry Skip Adder(CSA) [18] and Carry Select Adder(CSLA)[17] for the first stage addition. Conventional BCD adder [14], Correction free BCD adder [13], CSA BCD adder [15] are used for comparison. The area, delay and total power dissipation results are shown in table 1. From the reports, it is seen that our proposed flagged BCD adder design using CSK adder has lower logic cell count compared to all other architectures used for comparison, thanks to the low area CSK adder, flagged bit computation logic and flag inversion logic [1] which realizes constant(0110) addition with fewer gates. The delay of the proposed flagged BCD adder designs is less when compared with all other BCD adder designs used for comparison. This is due to the long carry generation($C_{out}$) path

**Table 1.Comparison of area, delay and power dissipation of proposed BCD Adder and state-of the art designs**.

| BCD Adder | Area (No. of Logic Elements) | Delay (ns) | Total thermal power dissipation (mW) | Core dynamic thermal power dissipation (mW) | I/O thermal power dissipation (mW) |
|---|---|---|---|---|---|
| Conventional [14] | 24 | 18.415 | 172 | 2.82 | 89.05 |
| Correction free [13] | 58 | 16.326 | 173.06 | 3.59 | 89.34 |
| Carry skip [15] | 32 | 21.123 | 174.27 | 2.57 | 91.56 |
| Proposed-Using CSLA | 25 | 15.650 | 175.39 | 2.55 | 80.14 |
| Proposed-Using CSA | 21 | 16.494 | 169.22 | 2.27 | 86.83 |

which is 2 OR and 1 OR excess of 4 xor delays in CSA based BCD adder[15] and Conventional BCD adder [14] respectively. The Area-Delay-Product (ADP) of the proposed design fair better compared to the previous designs as seen from figure 5. In addition our proposed BCD adders using carry skip addition for the initial stage demonstrates better power dissipation performance compared to the state-of the art designs

.

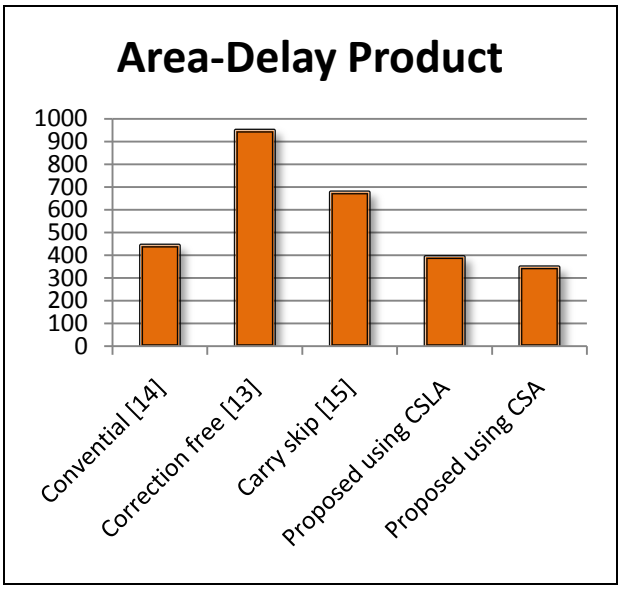



**Fig 5. Graphical representation of Area-Delay-Product**

## 5. CONCLUSION

We have proposed a new BCD adder design by using the concept of high speed addition for three operands (two input operand and a constant) by generating flag bits. Extensive comparison using synthesis results shows that the proposed flagged BCD adder outperformed all other previous designs in terms of power dissipation and area. The potential benefits of reduced logic cell count of our proposed flagged BCD adder can be realized in fair ADP performance. This suggests the suitability of our proposed flagged BCD adder for portable VLSI implementation. Moreover, the proposed BCD adder can be easily extended to multi digit addition.

## 6. REFERENCES

[1] Vibhuti Dave, Erdal Oruklu and Jafar Saniie,"Constant addition with flagged binary adder arhitectures", Integration the VLSI journal Vol.43 pp. 258–267 2010.

[2] Behrooz Shirazi, David Y.Y.Yun , and Chang N.Zhang,"Vlsi Designs for Redundant Binary-Coded Decimal Addition", IEEE pp.52-56 1988.

[3] Robert D. Kenney and Michael J. Schulte, High-Speed Multi-operand Decimal Adders", IEEE Transactions On Computers, Vol.54, No.8, pp.953-963, August 2005.

[4] Hafiz Mohammed Hasan Babu and Ahsan Raja Chowdhury, "Design of a Reversible Binary Coded Decimal Adder by Using Reversible 4-bit Parallel Adder", IEEE, VLSID 2005.

[5] Himanshu Thapliyal and Nagarajan Ranganathan, "A New Reversible Design of BCD Adder" EDAA 2011.

[6] Ashis Kumer Biswas, Md. Mahmudul Hasan, Moshaddek Hasan, Ahsan Raja Chowdhury and Hafi Md. Hasan Babu, "A Novel Approach to Design BCD Adder an carry Skip BCD Adder" 21st International Conference on VLSI Design pp. 566-571 2008.

[7] Sreehari Veeramachaneni, M.Kirthi Krishna, Lingamneni Avinash, Sreekanth Reddy P,M.B.Srinivas, "Novel, High-Speed 16-Digit BCD Adders Conforming to IEEE 754r Format" IEEE Computer Society Annual Symposium on VLSI (ISVLSI'07), 2007.

[8] Alp Arslan Bayrakci and Ahmet Akkas, "Reduced Delay BCD Adder", IEEE pp.266-271, 2007.

[9] Anshul Singh, Aman Gupta, Sreehari Veeramachaneni and M.B. Srinivas, " A High Performance Unified BCD and Binary Adder / Subtractor*", IEEE Computer Society Annual Symposium on VLSI, pp.*211-216 2009.

[10] Sreehari Veeramachaneni.M, Kirthi Krishna.V, Prateek G.S, Subroto.S, Bharat, M.B.Srinivas, "A Novel Carry-Look Ahead Approach to a Unified BCD and Binary Adder/Subtractor", 21st International Conference on VLSI Design 2008, pages 547-552, January 2008.

[11] SundaresanC, Chaitanya CVS, PR Venkateswaran, Somashekara Bhat and Mohan Kumar J, "Modified Reduced Delay BCD Adder" IEEE4th International Conference on Biomedical Engineering and Informatics (BMEI) 2011, pages 2148-2151.

[12] Chetan Kumar V, Sai Phaneendra P, Syed Ershad Ahmed, Sreehari Veeramachaneni, N, M.B Srinivas,"A Unified Architecture for BCD and Binary Adder/Subtractor" , *14th Euromicro Conference on Digital System Design pp.426-429 2011.*

[13] Osama Al-Khaleel, Mohammad Al-Khaleel, Zakaria Al-Qudah,Christos A. Papachristou and Khaldoon Mhaidat, and Francis G. Wolff," Fast Binary/Decimal Adder / Subtractor with a Novel Correction-Free BCD Addition" *IEEE* pp.455-459 2011.

[14] Moris Mano - Digital design 3rd edition Prentice Hall Puclication 2001.

[15] Himanshu Thapliyal, Saurabh Kotiyal and M.B Srinivas, "Novel BCD Adders and their Reversible Logic Implementation for IEEE 754r Format", Proceedings of the 19th International Conference on VLSI Design (VLSID'06) 2006.

[16] http:// webster.cs.ucr.edu / AoA / Windows / HTML / Advanced Arithmetica6.html – Chapter 4 More data representation – subsection 4.3 - Basics of Binary Coded Decimal (BCD) Representation

[17] B. Ramkumar and Harish M Kittur "Low-Power and Area-Efficient Carry Select Adder", IEEE transactions on Very Large Scale Integration (VLSI) systems, Vol. 20, No. 2, pp.371-375, February 2012.

[18] Min Cha Earl.E.Swartzlander, Jr. "Modified carry-skip adder for reducing first block delay", proceedings of 43rd Midwest symposium on circuits and systems, August 2000.