A Parallel Decimal Multiplier Using Hybrid Binary Coded Decimal (BCD) Codes

Xiaoping Cui, Weiqiang Liu* and Wenwen Dong

College of Electronic and Information Engineering,

Nanjing University of Aeronautics and Astronautics, Nanjing, China

Fabrizio Lombardi

Department of Electrical and Computer Engineering,

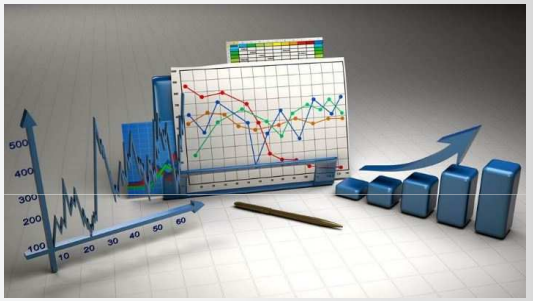
Northeastern University, Boston, USA

OUTLINE

- Motivation
- Review of BCD Representations and Decimal Multiplier
- The Proposed Partial Product Tree
- Evaluation and Comparison
- Conclusions

Why Decimal Arithmetic is Needed?

- Binary arithmetic introduces conversion and rounding errors
- Decimal arithmetic is highly demanded in many applications (financial, commercial and so on) that cannot tolerant errors.
- Decimal specification has been added to the revised IEEE 754-2008 standard.
- **High performance decimal arithmetic circuits are required.**



Introduction

Partial Production Generation

Sign Digit (SD) Radix-10 recoding

Redundant BCD excess-3 (XS-3)

Overloaded decimal digit set (ODDS) code

Double BCD recoding

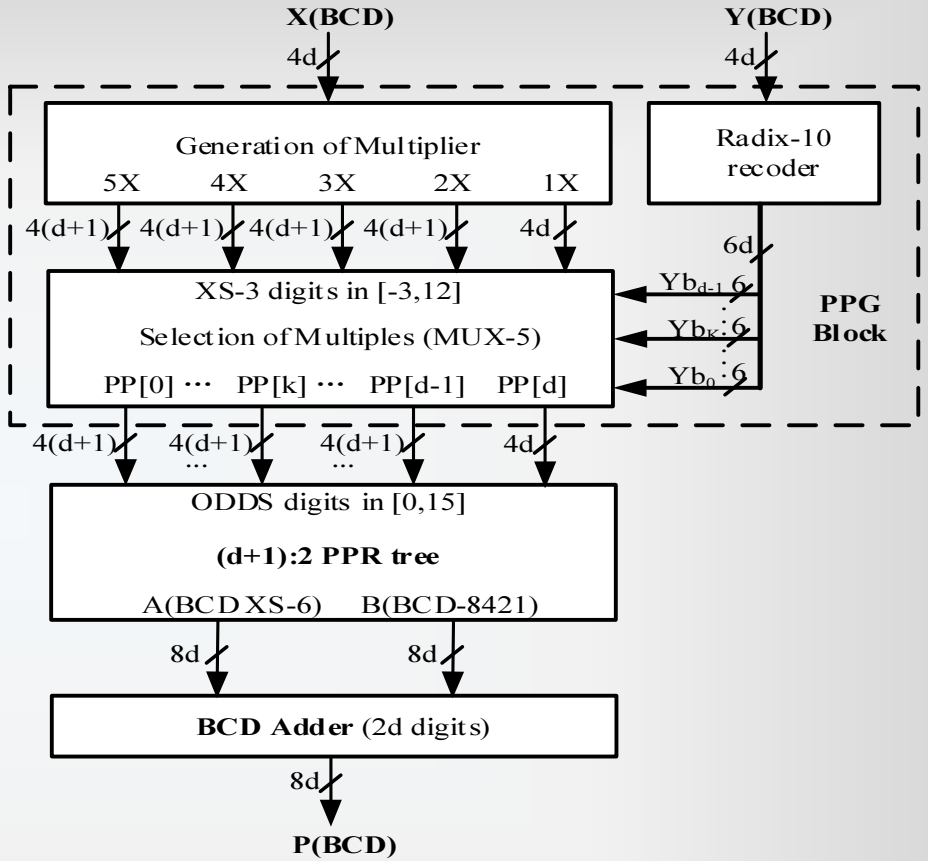
Partial Product Compression

Decimal 3:2 CSA

Binary compressor

Final Decimal Adder

Parallel prefix/carry select adder



[7] A. Vazquez, E. Antelo, and J. Bruguera, "Fast Radix-10 Multiplication Using Redundant BCD codes", *IEEE Transactions on Computers*, vol. 63, no. 8, pp. 1902–1914, Aug. 2014.

Partial Production Generation

SD Radix-10 Recoding

$$Yb_i = \begin{cases} Y_i & Y_i < 5 \,\&\&\, Y_{i-1} < 5 \\ Y_i + 1 & Y_i < 5 \,\&\&\, Y_{i-1} \geq 5 \\ -(10 - Y_i) & Y_i \geq 5 \,\&\&\, Y_{i-1} < 5 \\ -(10 - Y_i) + 1 & Y_i \geq 5 \,\&\&\, Y_{i-1} \geq 5 \end{cases}$$

- SD Radix-10 recoding scheme

$y_{i,3}y_{i,2}y_{i,1}y_{i,0}$	$y5_i\, y4_i\, y3_i y2_i y1_i$ ($ys_{i-1}=0$)	ys_i	$y5_i\, y4_i\, y3_i y2_i y1_i$ ($ys_{i-1}=1$)	ys_i
0000	00000	0	00001	0
0001	00001	0	00010	0
0010	00010	0	00100	0
0011	00100	0	01000	0
0100	01000	0	10000	0
0101	10000	0	01000	1
0110	01000	1	00100	1
0111	00100	1	00010	1
1000	00010	1	00001	1
1001	00001	1	00000	1

Partial Production Generation

Redundant BCD Codes

$$Z = -s_z \times 10^i + \sum_{i=0}^{d-1} Z_i \times 10^i$$

d: the number of decimal digits

s_z: the sign bit

$Z_i \epsilon [l-e, m-e]$ is the i^{th} digit,

$(0 \le l \le e, 9+e \le m \le 15)$

e: the excess of the representation

(*e*=0, 3 or 6)

ρ the redundancy index $\rho = m - l + 1 - 10$

Different Representations of Z_i

BCD

$Z_i \epsilon [0,9], e = 0,\ l = 0,\ m = 9,\ \rho = 0;$

BCD excess-3

$Z_i \epsilon [0,9], e = 3, l = 3, m = 12,\ \rho = 0;$

BCD excess-6

$Z_i \epsilon [0,9], e = 6, l = 6, m = 15,\ \rho = 0;$

ODDS

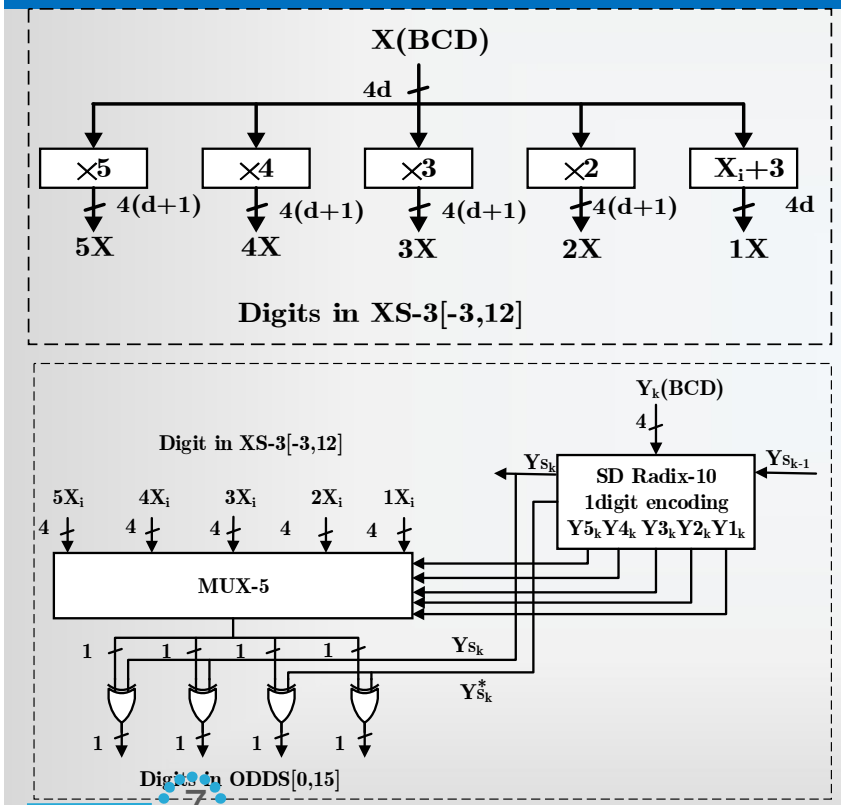
$Z_i \epsilon [0,15], e = 0, l = 0, m = 15,\ \rho = 6;$

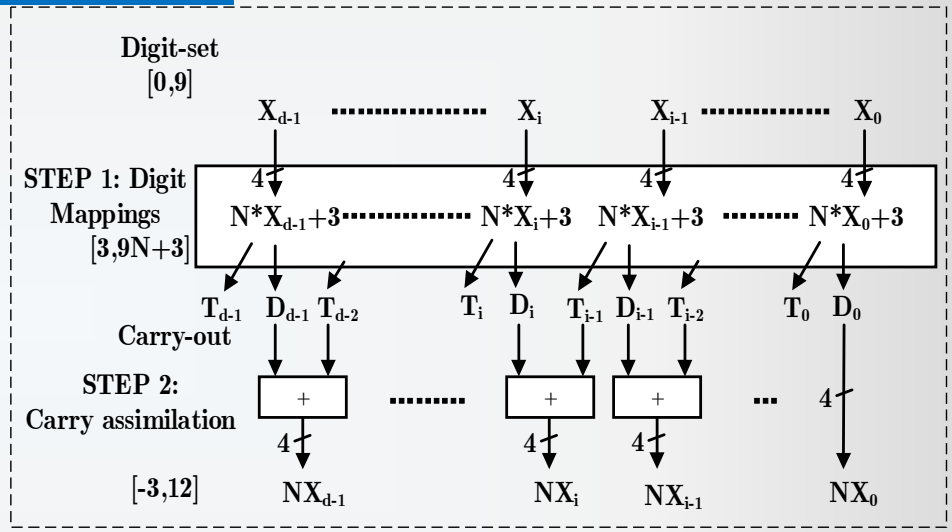
XS-3

$Z_i \epsilon [-3,12], e = 3, l = 0, m = 15,\ \rho = 6.$

Partial Production Generation

XS-3 Recoding (Redundant Odds [0, 15])





Advantage of XS-3 Codes: difficult multiples (such as 3X) can be obtained in a carry-free manner

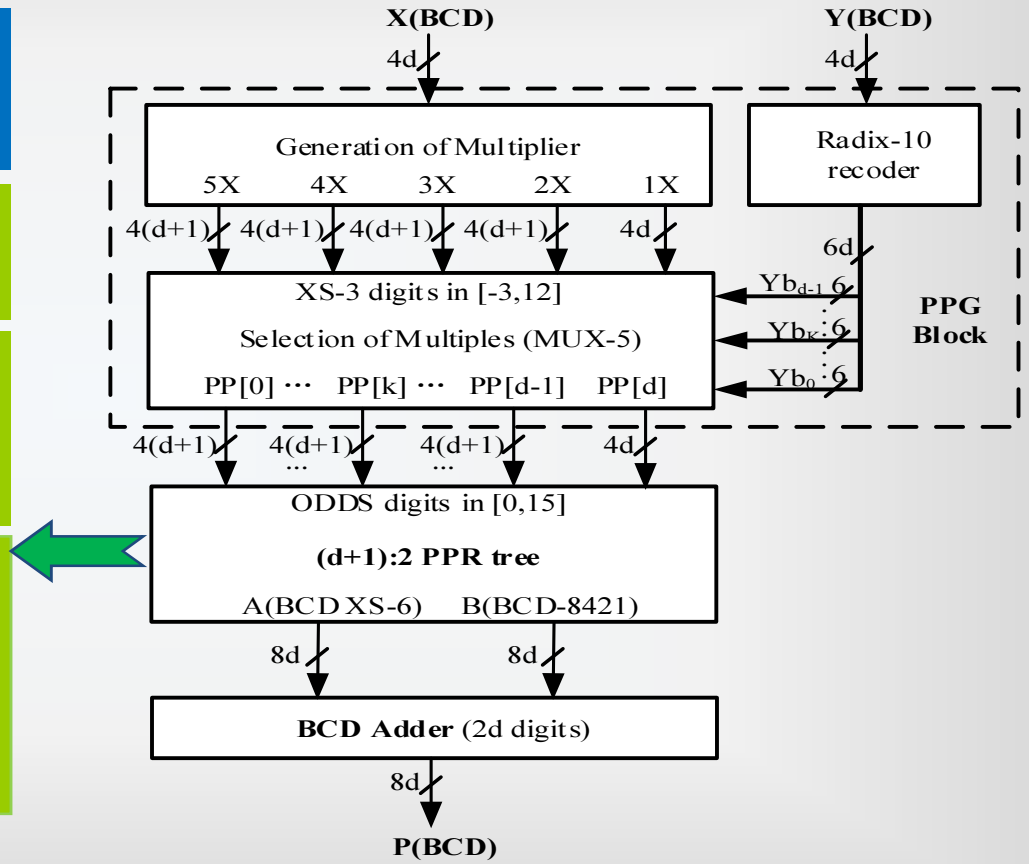
Partial Product Compression

Decimal PP Compression Using ODDS

The (d+1:2) PP Reduction (PPR):
(1) A regular binary CSA tree

(2) A binary counter is used to count carries generated between the digit columns in the binary CSA tree

(3) The ODDS partial products in (1) and (2) are added by the binary CSA tree and the decimal digit 3:2 compressor



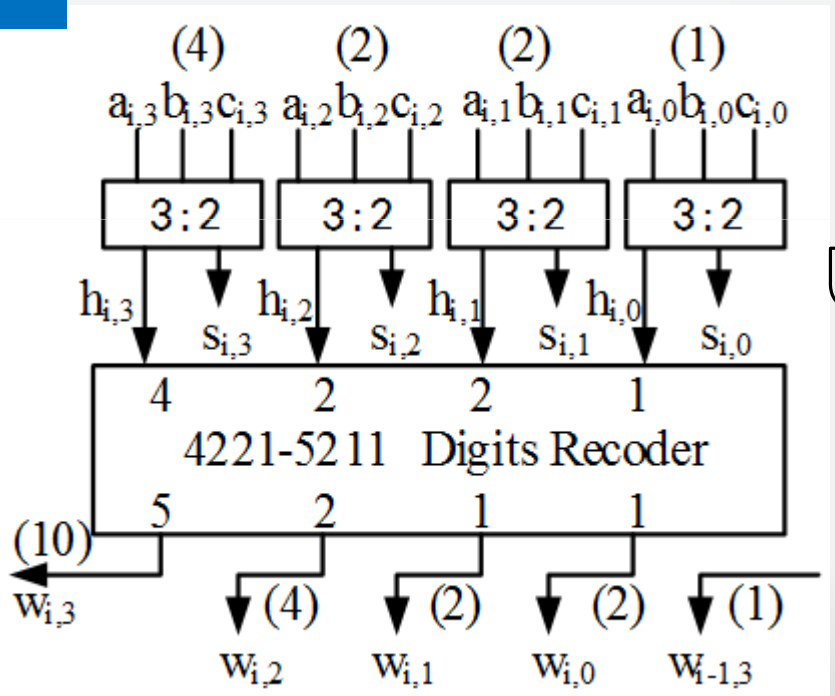
Partial Product (PP) Compression

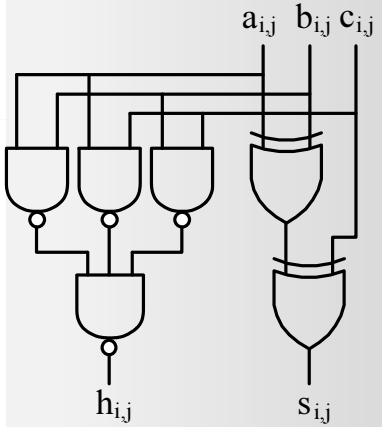
Decimal PP Compression Based on BCD-4221 /5211

$$A_i + B_i + C_i = \sum_{j=0}^{3} (a_{i,j} + b_{i,j} + c_{i,j}) r_j$$

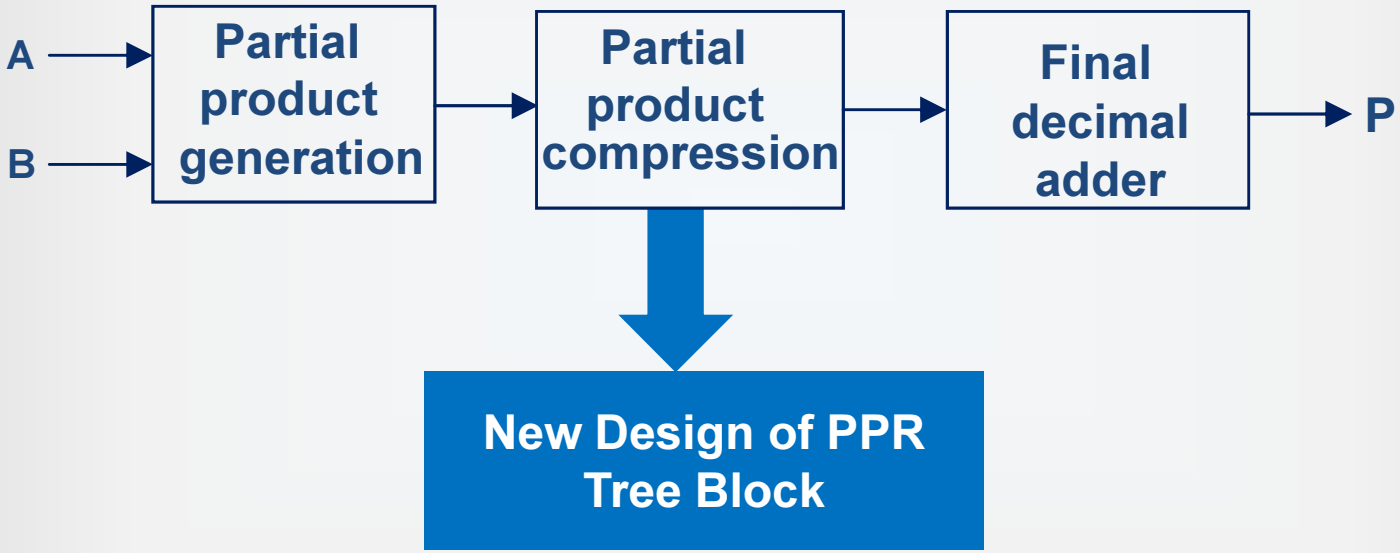
$$= \sum_{j=0}^{3} s_{i,j} r_j + 2 \times \sum_{j=0}^{3} h_{i,j} r_j = S_i + 2 \times H_i$$

- Decimal 3:2 CSA

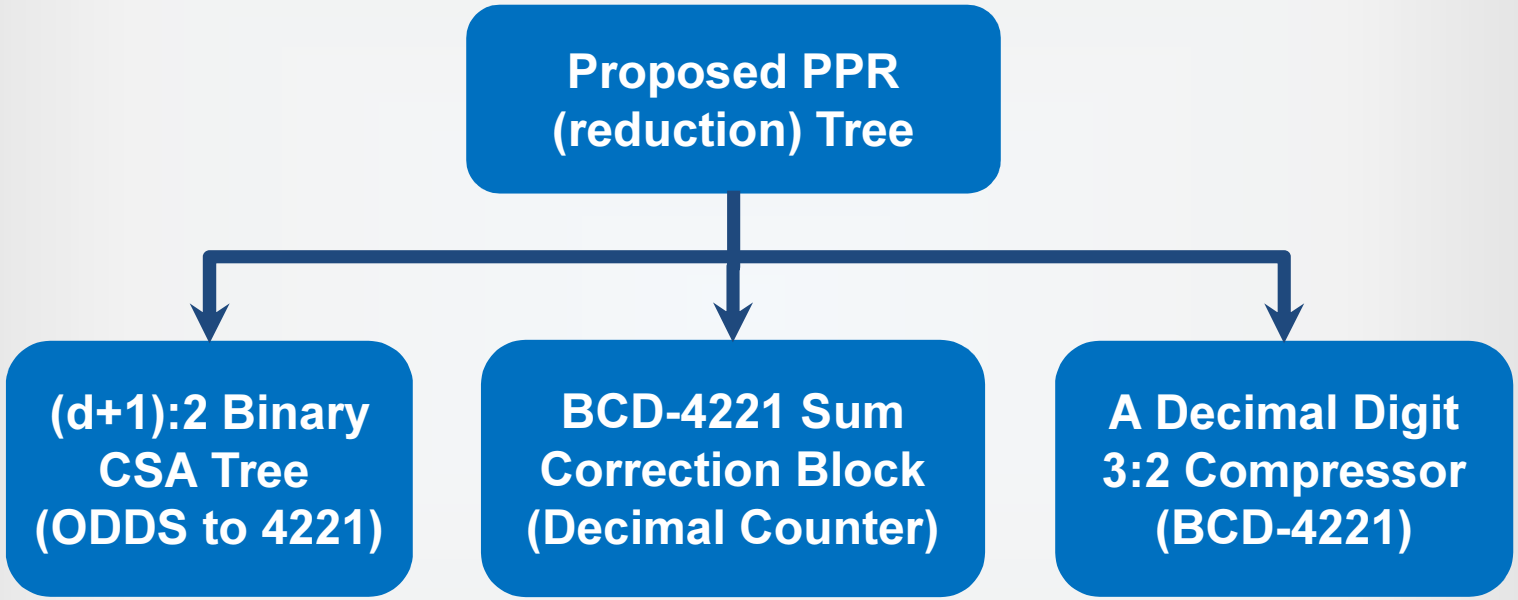




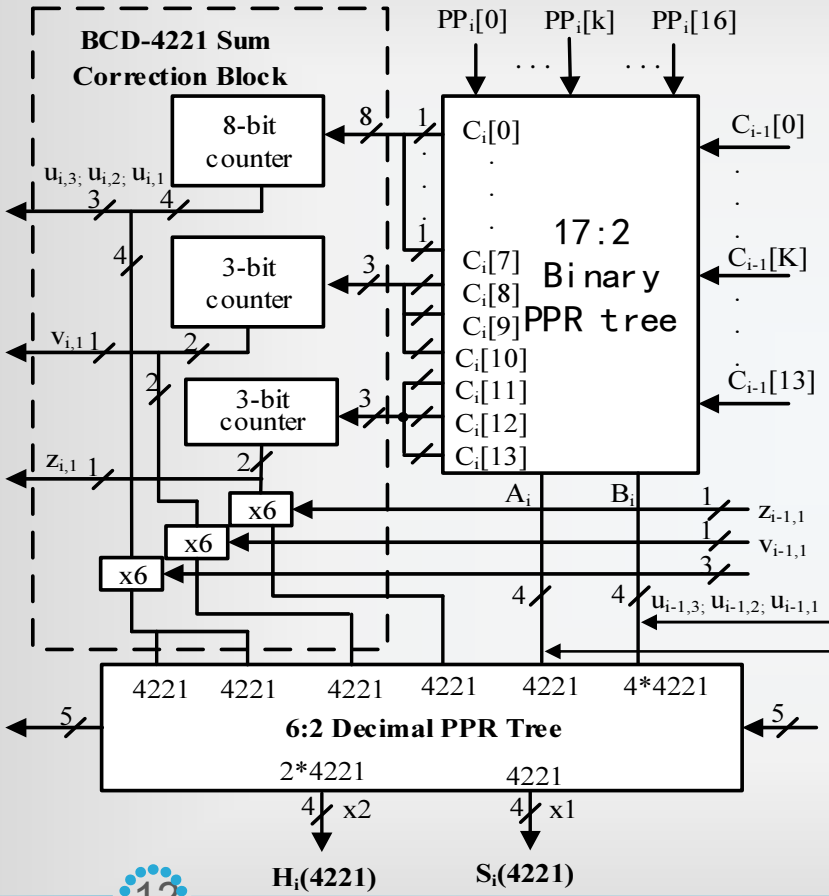
Proposed Design



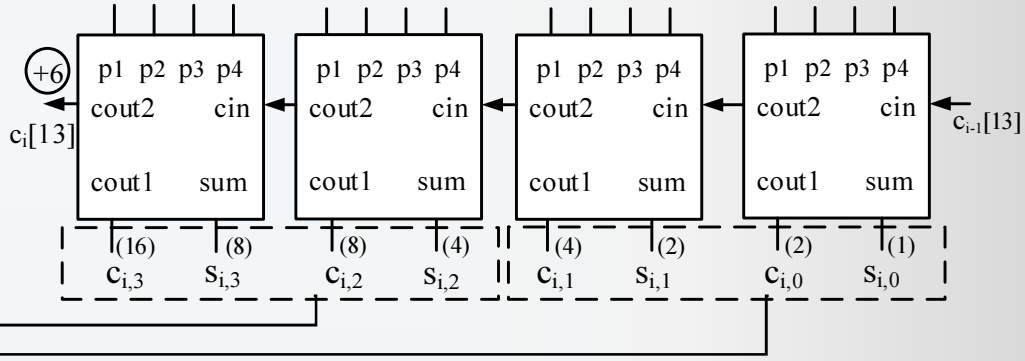
Proposed Design: A New PPR Tree



A PPR Tree FOR 16*16-digit MULTIPLIER

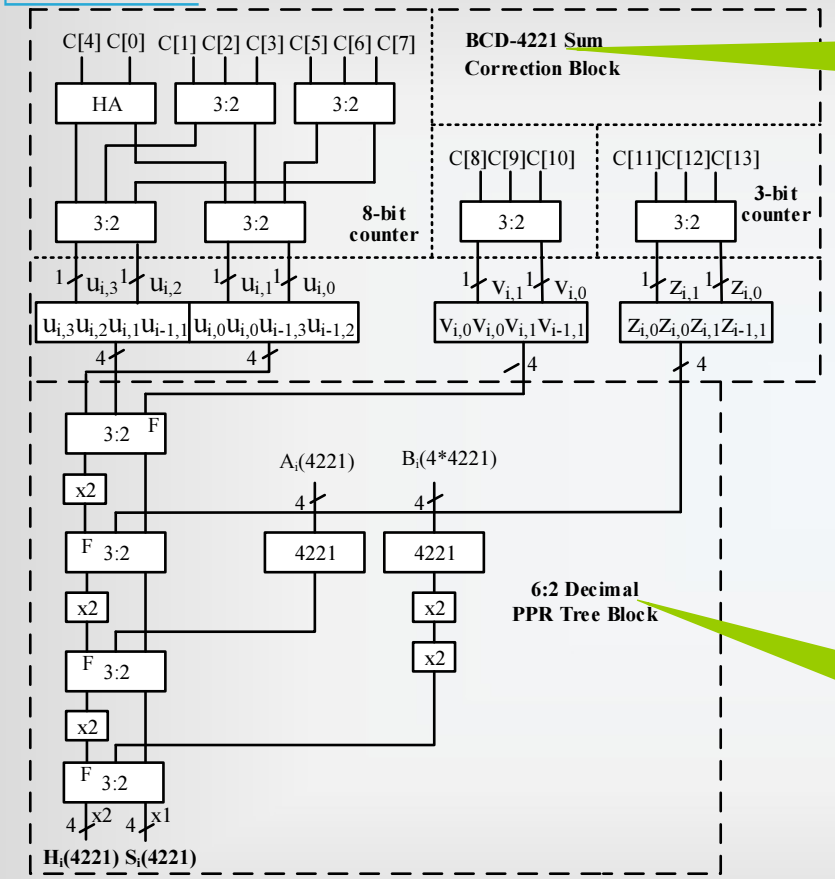


- The No. of PP rows in the 1st, 2nd, 3rd and 4th stages are 17, 9, 6 and 4, respectively.



- 4-bit binary 4:2 compressor in last compression stage

THE PROPOSED PPR TREE

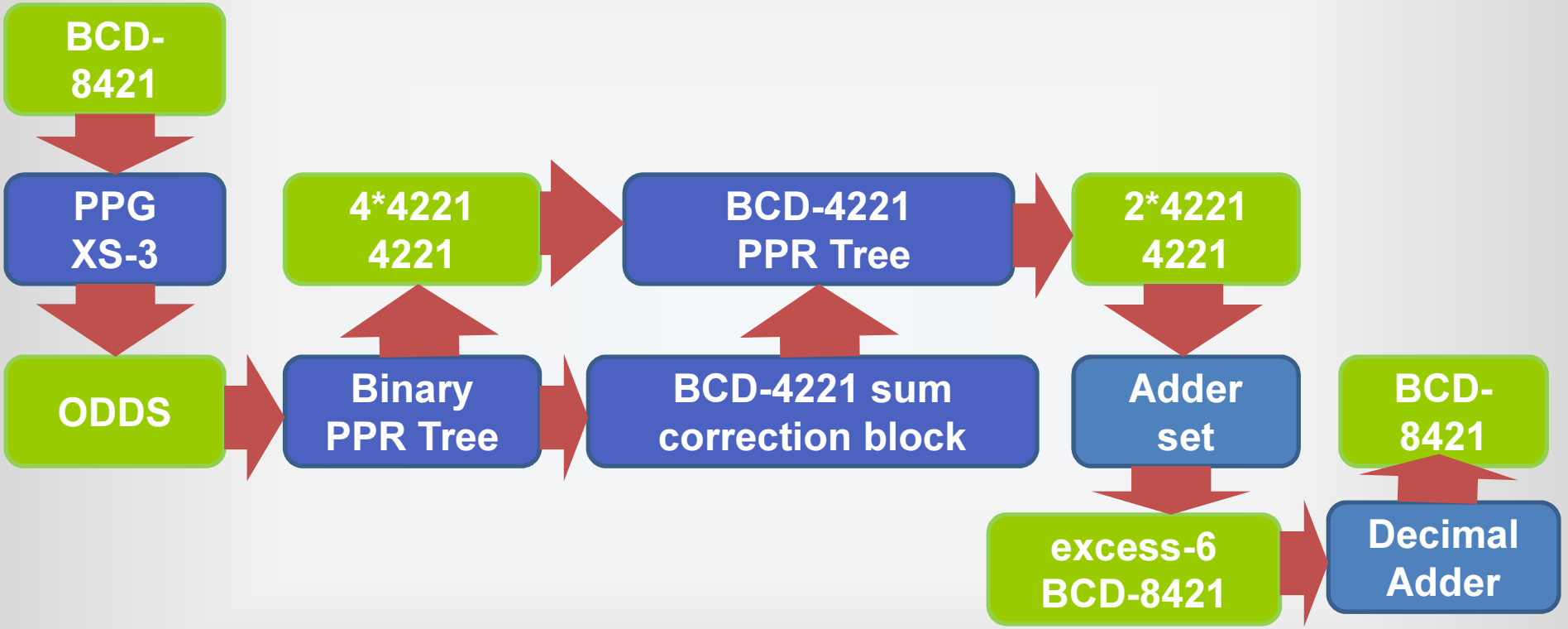


BCD-4221 8-bit and 3-bit counter correction

- The 8-bit BCD-4221 counter is faster than a binary counter (only two 3:2 CSA delay).
- 3-bit counters are used to generate a BCD-4221 decimal correction digit by using only one 3:2 compressor.
- To balance the paths in the decimal 6:2 PPR tree and reduce the critical path.

6:2 decimal PPR tree block

Using Hybrid (Multiple) BCD Codes



Advantages of the proposed PPR tree

1. A BCD-4221 counter is faster than a binary counter (a 8-bit counter has two 3:2 CSA stages, and 3-bit counter has one 3:2 CSA stage.)

2. A non-fixed size BCD-4221 counter correction block is used to balance the paths and reduce the critical path delay of decimal 6:2 PPR tree.

3. The final two PP rows are generated using a decimal PPR tree based on BCD-4221 that is easy to be converted to BCD-8421.

Evaluation

Area and Delay (LE-Based Model) for the Proposed 16×16-digit Multipliers.

Block	Delay #FO4	Area #NAND2
PPG Stage	10.2	14900
PPR Tree	25.3	14306
Adder Setup	3.2	1050
Decimal Adder	11.5	2400
Total	50.2	32656

Evaluation

Area and Delay (LE-Based) Comparision for Different BCD Multiplier Designs.

Design	Delay #FO4	Area #NAND2
Non-Redundant [13]	58.3	35750
Redundant [7]	51.4	30600
Proposed	50.2 Compared with [13] -13.89% Compared with [7] -2.23%	32656 Compared with [13] -8.65% Compared with [7] +6.05%

[7] A. Vazquez, E. Antelo, and J. Bruguera, "Fast Radix-10 Multiplication Using Redundant BCD codes", *IEEE Transactions on Computers*, vol. 63, no. 8, pp. 1902–1914, Aug. 2014.
[13] A. Vazquez, E. Antelo and P. Montuschi, "Improved Design of High-Performance Parallel Decimal Multipliers", *IEEE Transactions on Computers*, vol. 59, no. 5, pp. 679–693, May 2010.

Evaluation

Area and Delay Comparison Using NanGate 45nm open cell library

Design	Delay (ns)	Ratio	Area(μm^2)	Ratio
Proposed	3.21	1	43053.5	1
Non-Redundant [13]	3.66	1.14	48326.1	1.12

■ The proposed design reduces the delay by 12.30% and the area by 10.9% compared with [13].

■ [7] reduces the delay by 10.75% and the area by 11.1% compared with [13] (no direct comparison as some parts of PPR circuit of [7] are not provided in detail).

Conclusion

- Design of parallel decimal multiplier is studied
- A parallel decimal multiplier based on a new PPR tree is proposed by using:
  - ✓ A BCD-4221 sum correction block with non-fixed size counters,
  - ✓ A decimal PPR tree based on BCD-4221 decimal digit 3:2 compressor.
- The proposed parallel decimal multiplier is faster than previous best designs.

Thank you!

Questions?