



AN SUPERFLUOUS BINARY MULTIPLIER WITH ALTERED PARTIAL PRODUCT ORIGINATOR

M SANJEEVA, SRIKANTH MYDAPALLY, POREDDY RAJESHWARIREDDY

4.CH.SPANDANA, 5.Dr.SRINIVAS RAO

ASSISTANT PROFESSOR ,ECE

SANJEEVMADDULURI@GMAIL.COM

ELLENKI COLLEGE OF ENGINEERING & TECHNOLOY

ABSTRACT

The requirement of the modern-day computer device is a committed and actually immoderate tempo precise multiplier unit for signed and unsigned numbers. The art work in particular gives with in improving multiplication approach thru using Redundant Binary method. The redundant binary in format of excessive tempo virtual multiplier is beneficial because of excessive modularity and produce unfastened addition. Commonly in excessive radix modified sales space encoding set of rules the partial merchandise are decreased in multiplication technique. But it yields complexity in producing in generation of tough multiples. Consequently booth encoding scheme on the side of redundant binary scheme solves this problem via using booth encoding, RB partial product generator, RB partial product accumulator, RB to NB converter diploma. In this paper executed in Verilog HDL.

Key terms— Redundant Binary Multiplier, immoderate-degree Synthesis, Partial Product Generator, low electricity, low place, cast off, DSP, VLSI.

1. INTRODUCTION

Advent the individual of the art work is introduced to decorate the speed of the multiplier and to dispose of the difficult multiples and to lessen the partial merchandise than previous art work. Through enhancing the multipliers in the ALU processors from past to provide they are giving better effects evaluating to previous upgrades. So, now introducing the

Redundant Binary instance technique in digital multipliers to overcome drawbacks in preceding strategies. Redundant Binary (RB) representations first delivered by way of way of way of Avizienis in 1961 for fast parallel mathematics. This new mathematics end up achieved for instant Multiplication thru takagi and completed via Edamatsu. Multiplication is a most usually used operation in masses of computing structures. Infact multiplication is not something however addition while you remember the fact that, multiplicand gives to itself multiplier range of instances offers the multiplication fee among multiplier and multiplicand. However thinking about the truth that this sort of implementation in truth takes huge hardware property and the circuit operates at certainly low pace. With a view to deal with this such a diffusion of thoughts were supplied to this point for the final three some years. Everyone is aimed in the direction of particular improvement steady with the requirement. One may be aimed inside the course of excessive clock speeds and some other possibly aimed for low power or masses a extremely good deal lots much less area career. Both way very last interest is to offer you an green structure that could address 3 constraints of VLSI pace, location, and strength. Among the ones three speeds is the most effective which calls for specific hobby. If we test carefully multiplication operation includes steps one is producing partial merchandise and together with those partial products. Thus, the charge of a multiplier from time to time relies upon on how rapid generate the partial merchandise and the way speedy we are capable of add them

collectively. If the numbers of partial merchandise to be generated are of loads a lot a lot less than it's far now not straight away method that we've were given executed the fee in producing partial products. Sales region's algorithms are intended for this handiest. To accelerate the addition the numerous partial merchandise we want rapid adder architectures. Because the multipliers have a massive impact at the overall normal overall performance of the complete tool, many excessive regular common ordinary performance algorithms and architectures were proposed.

II PROPOSED MULTIPLIER

In Multiplier and Accumulation form the multiplier can be divided into four operation steps. First step- profits area set of tips, 2nd step – partial product summation, zero.33 step – very last addition, fourth step – accumulation as shown in Fig1. Step 1: Multiplication approach finished among n bits Multiplicand (X) and m bits Multiplier (Y). Step 2: n bits Partial products (P0~Pj) may be generated after multiplying n bits and m bits. Step three: very last addition amongst Partial merchandise Summation (S) bits and produce(C) bits. Step4: Accumulation results takes region among multiplication results(X*Y) and in the end receives Accumulation give up end result (Z).

III. SET OF RULES OF THE MULTIPLIER

A. Radix - 4 modified earnings area Encoding (Radix-4) can effectively be done to lessen the form of partial product rows to half in parallel multipliers. This is completed via manner of grouping 3 adjacent multiplier bits (B = bn-1 bn-2...b0) to pick one of the signed multiples as confirmed in table I. The side bits of each agency are overlapped with the two adjoining agencies. The lastbit in every institution is referred as reference bit. The first organisation enterprise is coded through which includes "zero" as reference bit preceding to least huge bit function i.E., (b1, b0, zero).Depending on the ones select indicators, the partial product rows are generated with the aid of manner of choosing one of the combination -2A,-A, 0, A, 2A of the multiplicand (A = an-1an-2...a0). The instances of multiplicand (2A) in desk I is acquired through the use of manner of left moving the multiplicand with the useful resource of manner of 1 bit role and negation

operation is finished through inverting every little littlelittle bit of multiplicand (i.E. One's supplement) and which includes "1" to its least awesome bit function.

The Multiplier uses a Redundant binary instance, this is one of the signed digit instance proposed byAvizienis to perform speedy parallel mathematics. It has a fixedradix-2 and as a digit set, wherein , represents -1. In redundant binary instance, more than a few may be expressed as (2) that is similar to 2's complement example of some of, except that could take " "as price. Normal Binary digits can be coded to form a RB digit as proven in table 2. The coding scheme for RB instance is given as (three) in which, + and - are represented in 2's complement form. This coding scheme may be accomplished to symbolize a RB numberas the summation of NB numbers in a single easy step. This is elaborated through thefollowing expressions. The summation of NB numbers Xand Y may be represented as (four) Implementation of modified Partial Product Generator using income location Encoding set of pointers In Redundant Binary Multipliers worldwide mag of innovative technology quantity.05, hassle No.04, April-2017, Pages: 0700-0705 In two's supplement illustration; -Y may be obtained by way of reversing all bits of Y after which adding "1" to its least massive bit characteristic. This technique is defined as follows: (five) wherein is acquired with the useful resource of the use of reversing all bits of Y. With the useful resource of Substituting above equations summation of NB massive variety may be expressed as (6) permit, xi and yi be the ith digits of X and Y respectively. Moreover we define (7) in which , is the inversion of yi . The time period xi - , will take one of the three values 1, 0, or , in place of using the signed and absolute fee representation, on this paper RB numbers are represented regular with definition as follows (as tabulated in desk II.) (8) as a give up result a RB amount may be form thru using a couple of NB in which NB digits (xi ,yi) integrate to form

be expression above, 1 , i.E.

(nine) thinking about the truth that, from equation above. - 1 is identical to (zero, 1)due to this, through manner of inverting one of the

$\square \square X, Y$
 $\square \square X, Y$
 $\square Y \square \text{equat}$

two partial product rows and inclusive of (0, 1) to its least great bit function, a RB partial product row is generated, this is equal to the summation of NB partial product rows. In 2's complement form sign little bit of diverse is represented through MSB, and therefore prior to RB encoding scheme MSB of every the NB partial product rows are inverted. Both the MBE and RB Partial product generator produces incorrect results. One supply of mistakes is whilst MBE generates terrible multiples of multiplicand. Awful partial product may be most correctly generated with the useful resource of first bit reversing the multiplicand discovered with the aid of way of the addition of "1" at its LSB feature inside the partial product summing tree, thinking about in 's supplement shape negation of a number of goals carry propagation addition. Additionally, some other deliver of errors is RB encoding scheme which calls for the addition of "-1" on the LSB of partial product row to generate the exact summation of NB partial product rows. A unmarried errors-correcting word, together representing the extra bits from each MBE and RB encoding together, gets rid of all the possibilities of occurrence of faulty effects. Desk II: RB Encoding the error correction word has the form (10) wherein, , while it corrects the term from RB Encoding on my own = zero, whilst it corrects the term from every RB encoding and MBE while the horrible multiplicand is Generated because of MBE = zero ,in any other case.

IV FUNDAMENTAL USUAL PERFORMANCE ASSESSMENT

the general fashionable performance of severa 2n-bit RB multipliers using the proposed RBMPPG-2 is classified; the consequences are in comparison with NBBE-2, CRBBE-2 and RBBE-four [14] multipliers which might be the cutting-edge-day and outstanding designs placed within the technical literature. All designs of RB multipliers use the RBFA and RBHA of [7]. An RB-NB converter is needed inside the final diploma of the RB multiplier to transform the summation bring about RB form to a 's complement big variety. It has been proven that the regular-time converter in [7] does not exist [19], [20], [21]. But, there can be a convey-unfastened multiplier that makes use

of redundant adders inside the good buy of partial merchandise through making use of on-the-fly conversion [22] in parallel with the bargain and generates the product without a bring-propagate adder [23], [24]. A hybrid parallel-prefix/carry-choose out out adder [25] is used for the final RB-NB converter. The NBBE-2 multiplier format makes use of the identical encoder and decoder as compressors [26], [27], [28] are used inside the partial product reduce price tree.

The more ECW within the NB multiplier designs is also changed as proposed in [11]. The multiplier designs are defined at gate degree in Verilog HDL and confirmed via Synopsys VCS using randomly generated enter styles; the designs are synthesized with the aid of way of manner of the Synopsys format Compiler using the NanGate 45 nm Open cellular Library. Inside the simulation of each format, a supply voltage of one.25 V and room temperature are assumed. Modern day buffers of a 2X energy are used for every the input energy and the output load. The alternative for common feel structuring is grew to come to be off to save you the tool from converting the shape of the unit cells. The commonplace power consumption is decided the usage of the Synopsys energy Compiler with all over again annotated switching hobby documents generated from 2,500 random enter vectors. Desk 6 summarizes the cast off, vicinity, power and electricity-remove manufactured from the NB and RB multiplier designs; the do away with, area, electricity and PDP metrics are compared one at a time.

Recall the put off first (Fig. 7); in comparison with CRBBE-2, the proposed designs can reduce the dispose of (for instance as a wonderful deal as sixteen.6 percentage for the case of 8_8-bit multiplier; for all times of wordlength, the eliminate is reduced through manner of as a minimum 10 percent. Compared with RBBE-four, the proposed designs can lessen the cast off by way of manner of using as a first rate deal as 24.Eight percent for the case of 32_32-bit and the cast off is reduced with the useful resource of at the least 17 percent for all instances of phrase-duration. The dispose of improvement is completed thru the reduced vital route because of the removal of 1 RBPP accumulation diploma. The do away with of the

proposed RB multipliers is slightly big (approximately five percentage) in evaluation with the extraordinary NB multiplier, i.E., NBBE-2. But, its area and energy are appreciably lower than NBBE-2 for big phrase duration designs (32- and sixty four-bit), as mentioned subsequent. In evaluation with CRBBE-2, the RB multiplier the usage of the proposed RBMPPG-2 has the smallest place for all cases (Fig. Eight). For 8_8-bit and 16_16-bit multipliers, the vicinity of RBBE-four RB multipliers is smaller than that of the proposed RB multipliers because of the reality RBBE-four based absolutely completely designs don't require extra ECW, even as the vicinity is barely multiplied thru the modified partial product within the proposed RB multipliers. As in contrast with NBBE-2 and RBBE-four, the proposed designs can reduce the location with the useful beneficial resource of as much as eleven.Five and 13.0 percent, respectively, for the case of a 64_64-bit multiplier and it's far particularly said for large duration designs, due to this confirming the place widespread performance of the proposed approach. Strength consumptions of NB and RB multipliers are also considered and in assessment (Fig. Nine). The proposed designs can lessen the energy for a 64_64-bit multiplier thru as a whole lot as 30.2, five.Five and forty five.Four percent, respectively, in contrast with NBBE-2, CRBBE-2, and RBBE-4. PDP is a commonly used metric for mixed performance in terms of postpone and electricity intake. In Fig. 10, the RB multipliers the usage of the proposed RBMPPG-2 have the smallest PDP under all times of RB multipliers. As in comparison with CRBBE-2, the proposed designs can lessen the PDP with the aid of over 14 percent for all instances. In assessment with RBBE-4, the proposed designs can lessen the PDP with the aid of using as an lousy lot as 59.6 percent for the case of a 32_32-bit multiplier, and regular times the proposed designs can reduce the PDP through way of over 30 percentage. As a surrender cease end result, the ones effects confirm the proposed.

**V. RESULTS AND ANALYSIS
RTL SCHEMAIC**

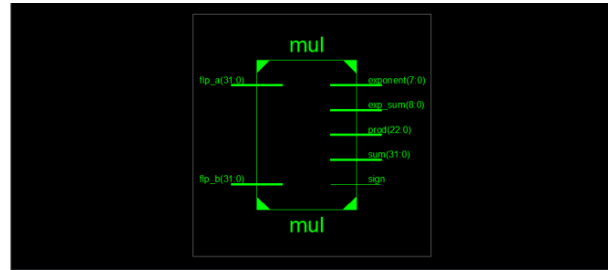


Figure 1: Existing system Schematic diagram

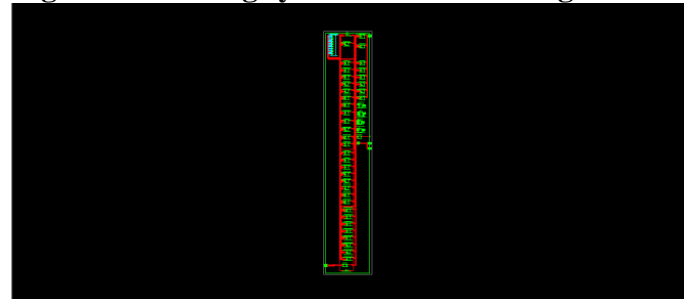


Figure 2: Existing system Schematic RTL diagram

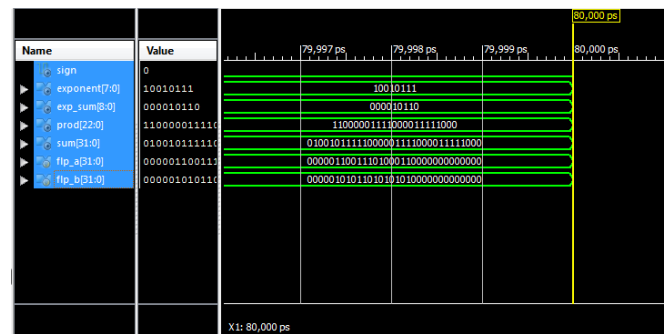


Figure 3: Existing system Output waveform 1

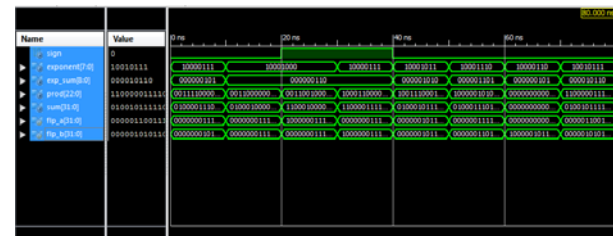


Figure 4: Existing system Output waveform 2

Area Report

Device utilization summary:

Selected Device : 3s100evq100-5

Number of Slices:	74	out of	960	7%
Number of 4 input LUTs:	136	out of	1920	7%
Number of IOs:	137			
Number of bonded IOBs:	137	out of	66	207% (*)
IOB Flip Flops:	1			
Number of MULT18X18SI0s:	4	out of	4	100%

Timing Report

Timing Summary:

Speed Grade: -5

Minimum period: No path found
 Minimum input arrival time before clock: 2.518ns
 Maximum output required time after clock: 4.114ns
 Maximum combinational path delay: 21.652ns

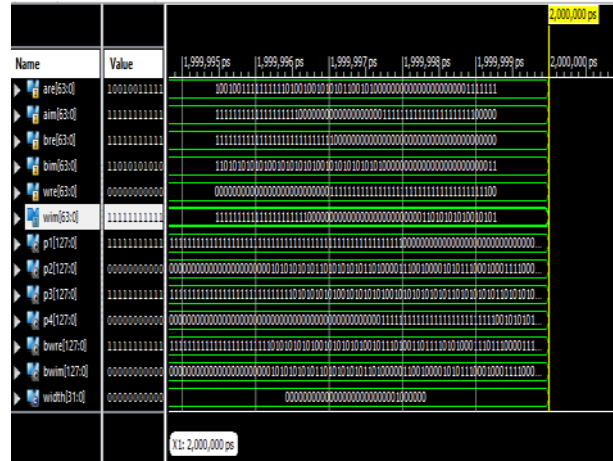


Figure 7: Proposed system Output waveform Area Report

Power Report

Device	On-Chip	Power (W)	Used	Available	Mitigation (%)	Supply	Summary	Total	Dynamic	Quiescent
Family	Virtex5					Source	Voltage	Current (A)	Current (A)	Current (A)
Package	FC484					Signal	1.25	6194	0.152	0.144
Grade	Commercial					IOs	0.024	261	0.152	0.060
Process	Typical					Leakage	2.221	440	0.001	0.001
Speed Grade	3					Total	2.955	1100	0.315	0.315
Environment						Thermal Properties		Effective TJA	Max Ambient	Junction Temp
Ambient Temp (C)	50.0							1.5	81.3	53.7
Use custom TJA?	No					Supply Power (W)		2.454	0.144	2.310
Custom TJA (C/W)	NA									
Reflow (JFM)	220									
Heat Sink	Medium Profile									
Custom TSA (C/W)	NA									
Board Selection	Medium (10x10)									
PCB Board Layers	12 to 15									
Custom PCB (C/W)	NA									
Board Temperature (C)	NA									

Device utilization summary:

Selected Device : 6vhx255tff1155-3

Slice Logic Utilization:
 Number of Slice LUTs: 7536 out of 158400 4%
 Number used as Logic: 7536 out of 158400 4%

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used: 7536
 Number with an unused Flip Flop: 7536 out of 7536 100%
 Number with an unused LUT: 0 out of 7536 0%
 Number of fully used LUT-FF pairs: 0 out of 7536 0%
 Number of unique control sets: 0

IO Utilization:
 Number of IOs: 256
 Number of bonded IOBs: 256 out of 440 58%

PROPOSED SYSTEM

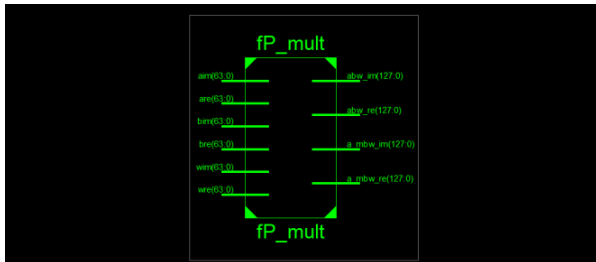


Figure 5: Proposed system Schematic diagram

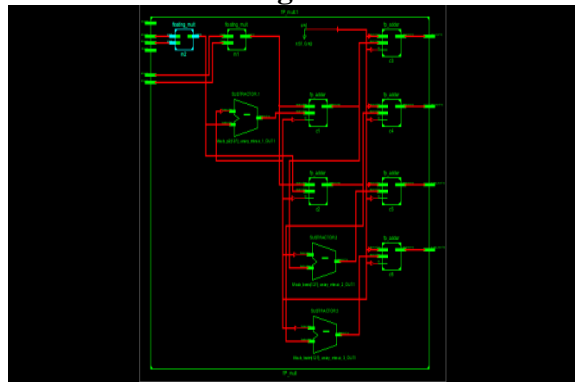


Figure 6: Proposed system Schematic RTL diagram

Timing Report

Timing Summary:

Speed Grade: -3

Minimum period: No path found
 Minimum input arrival time before clock: No path found
 Maximum output required time after clock: No path found
 Maximum combinational path delay: 38.588ns

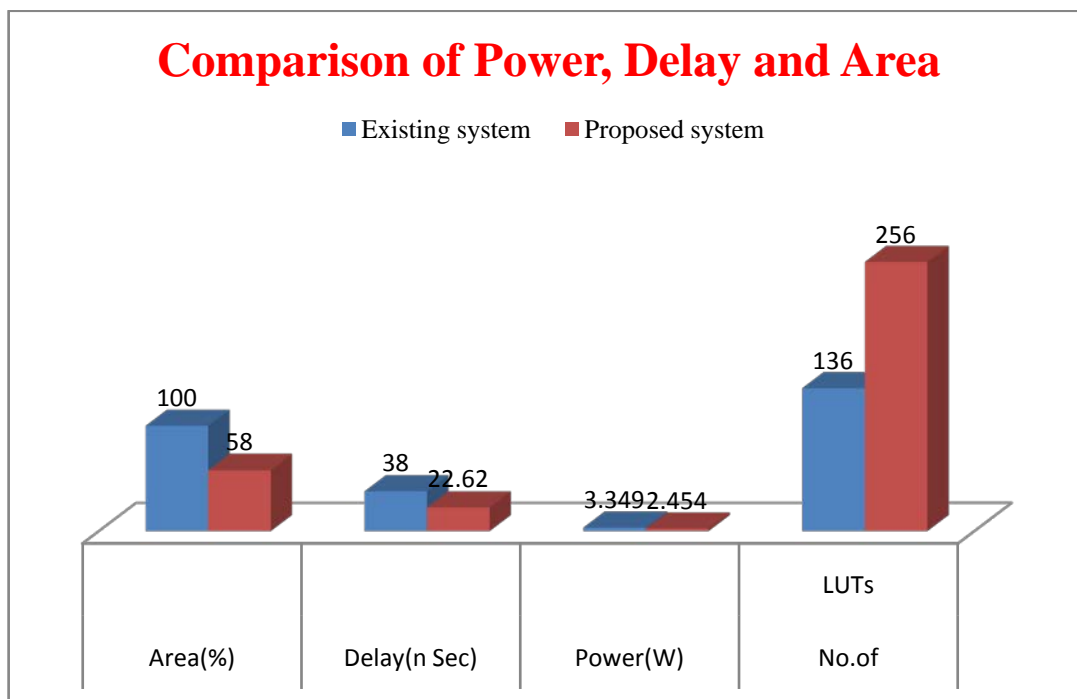
Power Report

Device	On-Chip	Power (W)	Used	Available	Mitigation (%)	Supply	Summary	Total	Dynamic	Quiescent
Family	Virtex5					Source	Voltage	Current (A)	Current (A)	Current (A)
Package	FC1155					Signal	1.25	6194	0.152	0.144
Grade	Commercial					IOs	0.024	261	0.152	0.060
Process	Typical					Leakage	2.221	440	0.001	0.001
Speed Grade	3					Total	2.955	1100	0.315	0.315
Environment						Thermal Properties		Effective TJA	Max Ambient	Junction Temp
Ambient Temp (C)	50.0							1.5	81.3	53.7
Use custom TJA?	No					Supply Power (W)		2.454	0.144	2.310
Custom TJA (C/W)	NA									
Reflow (JFM)	220									
Heat Sink	Medium Profile									
Custom TSA (C/W)	NA									
Board Selection	Medium (10x10)									
PCB Board Layers	12 to 15									
Custom PCB (C/W)	NA									
Board Temperature (C)	NA									

Comparison Table:1

S.No	Area(%)	Delay(n Sec)	Power(W)	No. of LUTs
Existing system	100	38	3.349	136
Proposed system	58	22.62	2.454	256

Comparison of Power, Delay and Area



VI CONCLUSION

A cutting-edge day changed RBPP generator has been proposed in this paper; this format gets rid of the more ECW this is introduced through the usage of previous designs. Consequently, a RBPP accumulation diploma is stored due to the removal of ECW. The new RB partial product era technique can be carried out to any 2n-bit RB multipliers to lessen the form of RBPP rows from N=four p 1 to N=four. Simulation consequences have mounted that the general overall performance of RB MBE multipliers the use of the proposed RBMPPG-2 is advanced significantly in terms of eliminates and area. The proposed designs acquire huge discounts in place and strength consumption whilst the phrase duration is at the least 32 bits. The PDP can be decreased thru as much as fifty nine percentages the use of the proposed RB multipliers at the equal time compared with gift RB multipliers. Therefore, the proposed RBPP era method is a very useful method at the same time as designing place and PDP inexperienced strength-of- RB MBE multipliers.

REFERENCES

- [1] X. Cui, W. Liu, X. Chen, E. E. Swartzlander and F. Lombardi, "A Modified Partial Product Generator for Redundant Binary Multipliers," in IEEE Transactions on Computers, vol. 65, no. 4, pp. 1165-1171, 1 April 2016, doi: 10.1109/TC.2015.2441711.
- [2] N. Takagi, H. Yasuura, and S. Yajima, "High-speed VLSI multiplication algorithm with a redundant binary addition tree," IEEE Trans. Comput., vol. C-34, no. 9, pp. 789-796, Sep. 1985.
- [3] Y. Harata, Y. Nakamura, H. Nagase, M. Takigawa, and N. Takagi, "A high speed multiplier using a redundant binary adder tree," IEEE J. Solid-State Circuits, vol. SC-22, no. 1, pp. 28-34, Feb. 1987.
- [4] H. Edamatsu, T. Taniguchi, T. Nishiyama, and S. Kuninobu, "A 33 MFLOPS floating point processor using redundant binary representation," in Proc. IEEE Int. Solid-State Circuits Conf., 1988, pp. 152-153.
- [5] H. Makino, Y. Nakase, and H. Shinohara, "A 8.8-ns 54x54-bit multiplier using new

- redundant binary architecture,” in Proc. Int. Conf. Comput. Des., 1993, pp. 202–205.
- [6] H. Makino, Y. Nakase, H. Suzuki, H. Morinaka, H. Shinohara, and K. Makino, “An 8.8-ns 54_54-bit multiplier with high speed redundant binary architecture,” *IEEE J. Solid-State Circuits*, vol. 31, no. 6, pp. 773–783, Jun. 1996.
- [7] Y. Kim, B. Song, J. Grosspietsch, and S. Gillig, “A carry-free 54b_54b multiplier using equivalent bit conversion algorithm,” *IEEE J. Solid-State Circuits*, vol. 36, no. 10, pp. 1538–1545, Oct. 2001.
- [8] Y. He and C. Chang, “A power-delay efficient hybrid carry-lookahead carry-select based redundant binary to two’s complement converter,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 1, pp. 336–346, Feb. 2008.
- [9] G. Wang and M. Tull, “A new redundant binary number to 2’s-complement number converter,” in Proc. Region 5 Conf.: Annu. Tech. Leadership Workshop, 2004, pp. 141–143.
- [10] W. Yeh and C. Jen, “High-speed booth encoded parallel multiplier design,” *IEEE Trans. Comput.*, vol. 49, no. 7, pp. 692–701, Jul. 2000.
- [11] S. Kuang, J. Wang, and C. Guo, “Modified Booth multiplier with a regular partial product array,” *IEEE Trans. Circuits Syst. II*, vol. 56, no. 5, pp. 404–408, May 2009.
- [12] J. Kang and J. Gaudiot, “A simple high-speed multiplier design,” *IEEE Trans. Comput.*, vol. 55, no. 10, pp. 1253–1258, Oct. 2006.
- [13] F. Lamberti, N. Andrikos, E. Antelo, and P. Montuschi, “Reducing the computation time in (short bit-width) two’s complement multipliers,” *IEEE Trans. Comput.*, vol. 60, no. 2, pp. 148–156, Feb. 2011.
- [14] Y. He and C. Chang, “A new redundant binary booth encoding for fast n -bit multiplier design,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 56, no. 6, pp. 1192–1199, Jun. 2009.
- [15] Y. He, C. Chang, J. Gu, and H. Fahmy, “A novel covalent redundant binary booth encoder,” in Proc. IEEE Int. Symp. Circuits Syst., 2005, vol. 1, pp. 69–72.
- [16] N. Besli and R. Deshmukh, “A novel redundant binary signed-digit (RBSD) Booth’s encoding,” in Proc. IEEE Southeast Conf., 2002, pp. 426–431.
- [17] A. Booth, “A signed binary multiplication technique,” *Quart. J. Mech. Appl. Math.*, vol. 4, pp. 236–240, 1951.
- [18] O. MacSorley, “High-speed arithmetic in binary computers,” *IRE Proc.*, vol. 49, pp. 67–91, 1961.
- [19] Y. Kim, B. Song, J. Grosspietsch, and S. Gillig, “Correction to ‘a carry-free 54b_54b multiplier using equivalent bit conversion algorithm’,” *IEEE J. Solid-State Circuits*, vol. 38, no. 1, p. 159, Jan. 2003.
- [20] W. Rulling, “A remark on carry-free binary multiplication,” *IEEE J. Solid-State Circuits*, vol. 38, no. 1, pp. 159–160, Jan. 2003.
- [21] M. Ercegovac and T. Lang, “Comments on ‘a carry-free 54b_54b multiplier using equivalent bit conversion algorithm’,” *IEEE J. Solid-State Circuits*, vol. 38, no. 1, pp. 160–161, Jan. 2003.
- [22] M. Ercegovac and T. Lang, “On-the-fly conversion of redundant to conventional representations,” *IEEE Trans. Comput.*, vol. C-36, no. 7, pp. 895–897, Jul. 1987.
- [23] M. Ercegovac and T. Lang, “Fast multiplication without carry-propagate addition,” *IEEE Trans. Comput.*, vol. C-39, no. 11, pp. 1385–1390, Nov. 1990.
- [24] L. Ciminiera and P. Montuschi, “Carry-save multiplication schemes without final addition,” *IEEE Trans. Comput.*, vol. 45, no. 9, pp. 1050–1055, Sep. 1996.
- [25] G. Dimitrakopoulos and D. Nikolos, “High-speed parallel-prefix VLSI Ling adders,” *IEEE Trans. Comput.*, vol. 54, no. 2, pp. 225–231, Feb. 2005.
- [26] S. Hsiao, M. Jiang, and J. Yeh, “Design of high-speed low-power 3–2 counter and 4–2 compressor for fast multipliers,” *Electron. Lett.*, vol. 34, pp. 341–343, Feb. 1998.