International Journal of Scientific and Technical Advancements
IJSTA
ISSN: 2454-1532

Fast Multiplier for FIR Filters

G. Rajesh Babu[1], C. N. Marimuthu[2]
[1]Assistant Professor, ECE, SVS College of Engineering, Coimbatore
[2]Professor, ECE, Nandha Engineering College, Erode
Email address: rajeshbabu.g@gmail.com

***Abstract*—**RTMCMA (Rounded Truncated Multiple Constant Multiplication and Accumulation) in direct FIR structure is used in FIR filter design. The proposed technique is n/2 number of partial product achieved via two's complement method. The Modified Booth Encoding algorithm is most important algorithm for truncated multiplier and multiplier take important role in high performance system to reduce power, area and delay optimized is very important in high performance system and DSP system. Truncation error is not more than 1 ulp (unit of least position). So there is no need of error compensation circuits and the final output will be precised.

***Keywords*—**FIR; booth multiplier.

I. INTRODUCTION

In Microprocessor and DSP high performance systems uses 70% instructions perform addition and multiplication. There is need of high speed multiplier. Low power design directly reduces the operation time in most of the portable devices. Low power VLSI design has reducing the power consumption of multiplication algorithms with having high-speed structures and appropriate performance.

II. REVIEW OF MEMORY BASED ALGORITHMS

Fixed Width Multipliers

Fixed-width design of signed-digit redundant multipliers that are frequently adopted in high-speed applications. Here consider the fixed-width multiplier design where the Modified Booth Encoded (MBE) partial products are represented by binary signed digits with digit set {+1, 0, +1}. The advantages of the MBE redundant multiplier are the reduced number of partial products and the carry-free partial product addition in the tree structure. Besides, using redundant representations can avoid the sign extension problem. In order to reduce the truncation errors and make the error compensation circuit. The influence of the digits in the minor group can be approximated by a simple combination of the digits in the major group, leading to a very simple compensation circuit.

Fast Parallel Multiplier-Accumulator

The parallel multiplier presents a dependence graph (DG) to visualize and describe the Modified Booth Algorithm (MBA). Booth multiplier, and the accumulator sections to ensure the fastest possible implementation. This multiplier independent of data word size and easy to designing optimum structures with minimal delay. The pipelined parallel MAC design is three times faster than other parallel MAC schemes that are based on the MBA.

Booth Multiplier

Booth multiplication produce the product of two binary numbers like X and Y, which are having m and n number of bits(m and n are equal) using 2's complement representation. Procedure for implementing the booth algorithm.

1. Making booth table: In booth table there are four columns, one column for multiplier, second for previous first LSB of multiplier and other two columns for partial product accumulator (P). From two numbers, choose multiplier (X) and multiplicand (Y). Take 2's complement of multiplicand (Y). Load X value in the table. Load 0 for X-1 value. Load 0 in U and V which will have product of X & Y at the end of the operation.
2. Booth algorithm requires a detailed inspection of the multiplier bits for shifting of the partial product (P). Before shifting, the multiplicand may be added to P, subtracted from the P, or left unchanged based on the following rules:1. {Xi Xi-1} is {0 0} Shift only, {Xi Xi-1} is {1 1} Shift only, {Xi Xi-1} is {0 1} Add Y to U and shift, {Xi Xi-1} is {1 0} Minus Y from U and shift.
3. Concatenate U and V and shift arithmetic right shift which preserves the sign bit of 2's complement number.
4. Circularly right shift X.

Repeat the same steps until nth bit completed.

III. EXISTING SYSTEM

A parallel tree multiplier designing steps are Partial Product (PP) generation, PP reduction, and final carry propagate addition. In deletion operation removes all the avoidable partial product bits which are shown by the light gray dots figure 1. Deletion error should be in the range -0.5 ulp $\leq$ ED$\leq$ 0. From column 3 onwards deletion starts while first two of partial product bits are kept. After the deletion of partial product bits, perform column-by-column reduction method.

When the reduction operation is over, perform the truncation, which will further remove the first row of (n-1) bits from column 1 to column (n-1). It will produce the truncation error which is in the range of -0.5 ulp $\leq$ ET$\leq$ 0. So the adjusted truncation error is -0.25 ulp $\leq$ T$\leq$0.25 ulp. here add a bias constant of 0.25 ulp for rounding. All the operations performed finally the PP bits are added to generate final product. Rounding error is in the form of - 0.5 ulp $\leq$ ER$\leq$0.5 ulp. The faithfully rounded truncated multiplier, total error is in range of –ulp< E=(ED+ET+ER) $\leq$ulp.

G. Rajesh Babu and C. N. Marimuthu, "Fast multiplier for FIR filters," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 1, pp. 57-61, 2016.

ISSN: 2454-1532

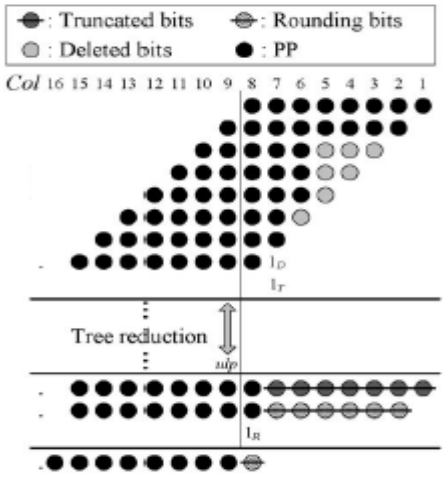

Fig. 1. Deletion, reduction, truncation, and rounding plus final addition.

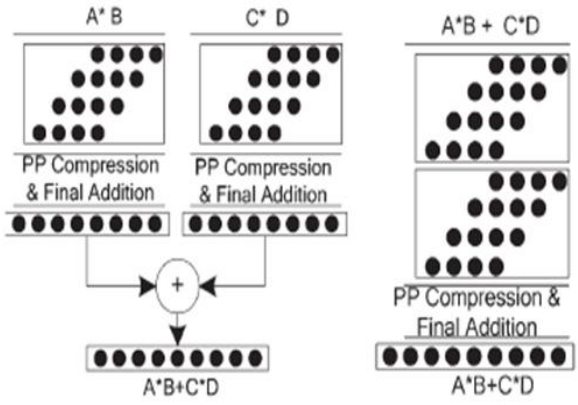

Fig. 2. Multiplication/Accumulation using Individual and Combined PP Compression.

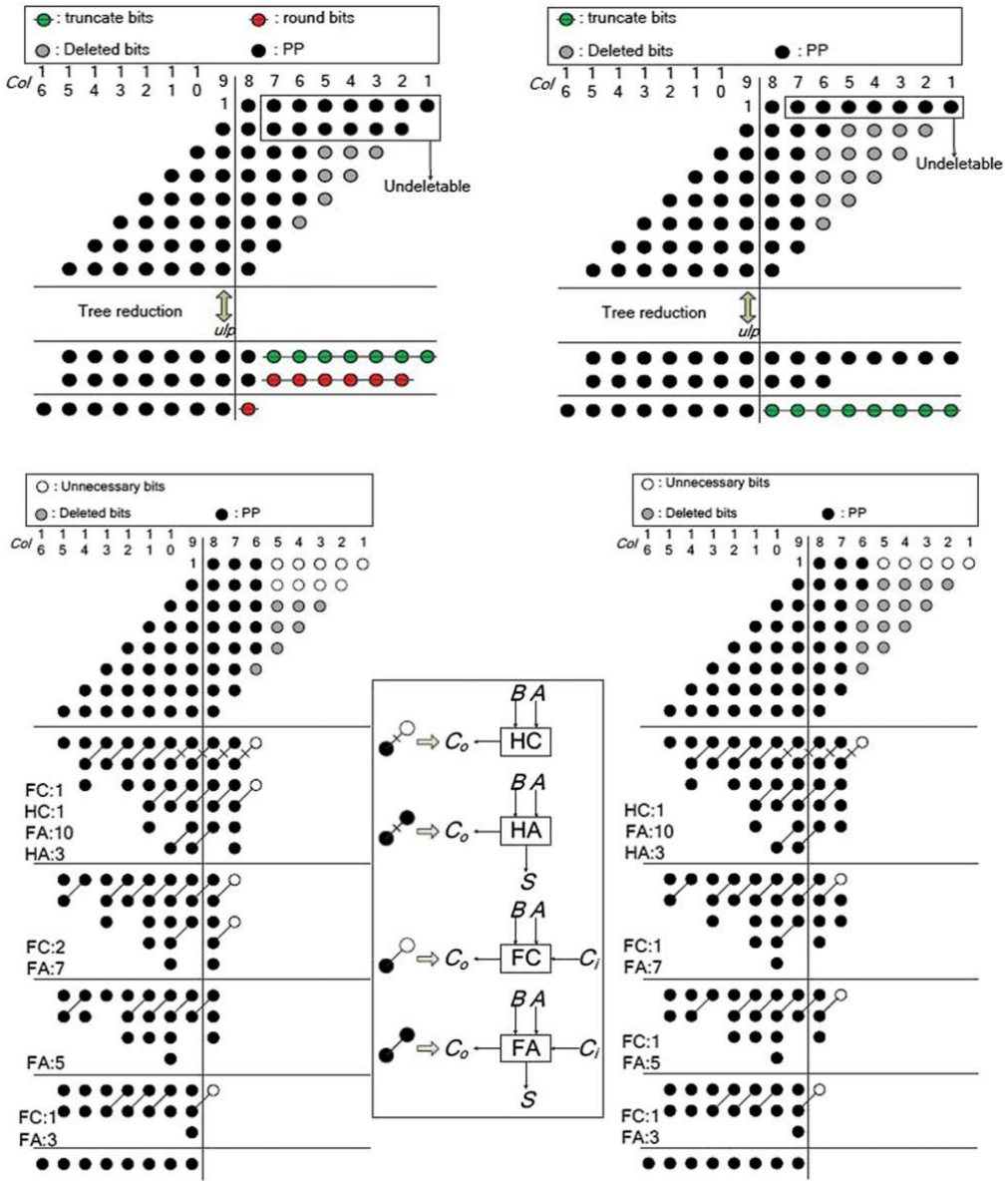

Fig. 3. Existing Truncated Multiplier designs.

G. Rajesh Babu and C. N. Marimuthu, "Fast multiplier for FIR filters," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 1, pp. 57-61, 2016.

Deletion is performed in Stage 1 to remove the PP bits, whereas the magnitude of the total deletion error is no more than 2−P−1. The truncation error is less than 1 ulp, so the accuracy of the final result is improved.

Parallel multipliers are produce product with the 2n number of bits and rounded to n number of bits to avoid growth in word size. Figure 2 shows the difference of individual multiplications and combined multiplication for A × B + C × D.

PP Truncation and Compression:

The MCMA module is realized by combining all the partial products (PPs) where unnecessary PP bits (PPBs) are removed without affecting the final accuracy outputs. Figure 3 shows the bit widths of all the filter coefficients are minimized using non uniform quantization with different word lengths in order to reduce the hardware cost

This architecture of MCMA with truncation (MCMAT) that removes unnecessary PPBs. The white circles in the L-shape block represent the undeletable PPBs. The deletion of the PPBs is represented by black circles. Figure 4 shows the simulation results of existing and proposed method and figure 5 shows Power summary of existing method.

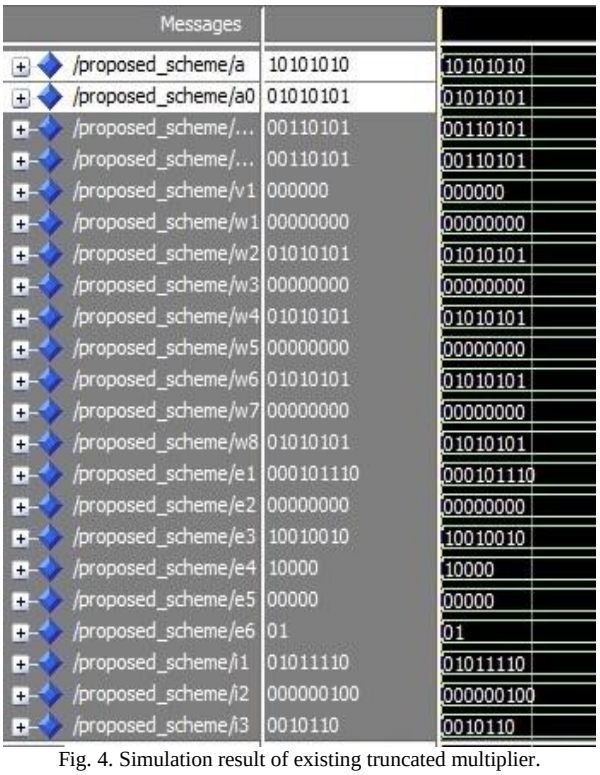

Fig. 4. Simulation result of existing truncated multiplier.

The existing truncated multiplication simulation result is obtained by using ModelSim. The input parameters are forced for signed multiplication is X=10101010; Y=01010101; and product result is, Z=00110101; by using rounded truncated multiplication scheme.

Power summary:	I(mA)	P(mW)
Total estimated power consumption:		65
Vccint 1.80V:	32	58
Vcco33 3.30V:	2	7
Inputs:	7	13
Logic:	8	14
Outputs:		
Vcco33	0	0
Signals:	2	4
Quiescent Vccint 1.80V:	15	27
Quiescent Vcco33 3.30V:	2	7

Fig. 5. Power summary of existing method.

Drawback of the truncated multiplier method has more area, delay and consumes more power.

IV. PROPOSED SYSYTEM - MODIFIED BOOTH MULTIPLIER

Modified Booth (MB) is a prevalent form used in multiplication. It is a redundant signed-digit radix-4 en-coding technique. Figure 6 shows block Diagram of Modified Booth Multiplier.

Drawback of the truncated multiplier method has more area, delay and consumes more power.

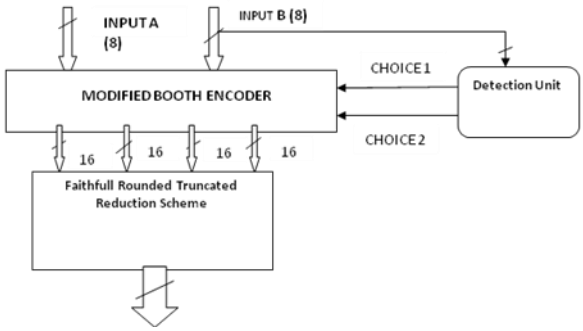

Fig. 6. Block diagram of modified booth multiplier.

Its main advantage is that it reduces by half the number of partial products in multiplication comparing to any other radix-2 representation. Let us consider the multiplication of 2's complement numbers and with each number consisting of n=2k bits. Figure 7 shows the Add end Generation for Modified Booth Multiplier.

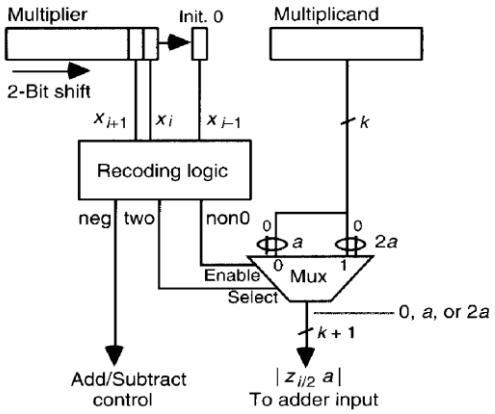

Fig. 7. Add end generation for modified booth multiplier.

G. Rajesh Babu and C. N. Marimuthu, "Fast multiplier for FIR filters," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 1, pp. 57-61, 2016.

One of the solutions of realizing high speed multipliers is to enhance parallelism which helps to decrease the number of subsequent calculation stages. The original version of the Booth algorithm (Radix-2) had two drawbacks.

They are:

The number of add subtract operations and the number of shift operations becomes variable and becomes inconvenient in designing parallel multipliers.

$$Y = \langle y_{n-1} y_{n-2} \cdots y_1 y_0 \rangle_{2's} = -y_{2k-1} \cdot 2^{2k-1} + \sum_{i=0}^{2k-2} y_i \cdot 2^i$$

$$= \langle \mathbf{y}_{k-1}^{MB} \mathbf{y}_{k-2}^{MB} \cdots \mathbf{y}_1^{MB} \mathbf{y}_0^{MB} \rangle_{MB} = \sum_{j=0}^{k-1} \mathbf{y}_j^{MB} \cdot 2^{2j}$$

$$\mathbf{y}_j^{MB} = -2y_{2j+1} + y_{2j} + y_{2j-1}.$$

The algorithm becomes inefficient when there are isolated 1's.

These problems are overcome by using modified Radix4 Booth algorithm which scan strings of three bits with the algorithm given below:

1. Extend the sign bit 1 position if necessary to ensure that n is even.
2. Append a 0 to the right of the LSB of the multiplier.
3. According to the value of each vector, each Partial Product will he 0, +y , -y, +2y or -2y. The negative values of y are made by taking the 2's complement. The multiplication of y is done by shifting y by one bit to the left. Thus, in any case, in designing a n-bit parallel multipliers, only n/2 partial products are generated.

The modified Booth's algorithm starts by appending a zero to the right of X2 (multiplier LSB). Table I shows triplets are taken beginning at position X-1 and continuing to the MSB with one bit overlapping between adjacent triplets. If the number of bits in X (excluding x-1) is odd, the sign (MSB) is extended one position to ensure that the last triplet contains 3 bits.

TABLE I. Modified Booth Algorithm

X_i	X_{i-4}	X_{i-2}	Operation	Comments
0	0	0	+0	String of zeros
0	1	0	+A	A single 1
1	0	0	-2A	Beginning of 1's
1	1	0	-A	Beginning of 1's
0	0	1	+A	End of 1's
0	1	1	+2A	End of 1's
1	0	1	-A	A single 0
1	1	1	+0	String of zeros

The low-cost implementations of FIR filters based on the direct structure in figure with faithfully rounded truncated multipliers. The MB-RTMCMA module is realized by accumulating all the partial products (PPs) where unnecessary PP bits (PPBs) are removed without affecting the final precision of the outputs.

The bit widths of all the filter coefficients are minimized using non-uniform quantization with unequal word lengths in order to reduce the hardware cost while still satisfying the specification of the frequency response. Figure 8 and 9 shows the simulation result of FIR filter and proposed truncated multiplier respectively.

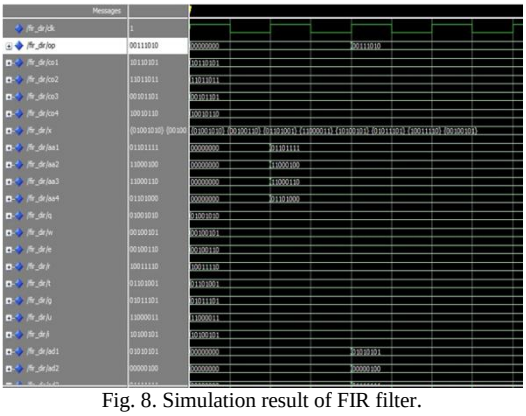

Fig. 8. Simulation result of FIR filter.

Fig. 9. Simulation result of proposed truncated multiplier.

V. Experimental Results and Comparison of Truncated Multipliers

Figure 10 shows the comparison of gate count, power and delay in bar chart. Table II gives the detailed description of proposed method is better than existing method.

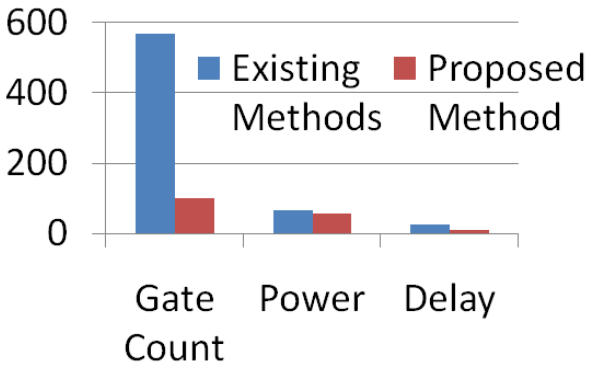

Fig. 10. Bar chart of existing method and proposed method.

TABLE II. Comparison.

S. No.	Description	Existing method	Proposed method
1	Gate delay	26.51ns	2.63ns
2	Gate count	570	100
3	Power	65mw	55mw

G. Rajesh Babu and C. N. Marimuthu, "Fast multiplier for FIR filters," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 1, pp. 57-61, 2016.

VI. CONCLUSION

RTMCMA leads to the smallest area cost, delay and power consumption. Partial products are generated directly with the help of booth encoding table. Further booth encoding table is converted into combination circuits which results into n/2+1 PP rows ,further direct 2's complement technique are implement to reduce n/2+1rows to n/2 ,that shows the results of gate count used for Modified Booth Rounded truncated multiplier and is less than Rounded truncated multiplier. Similarly power analysis results show that proposed Modified Booth Rounded Truncated Multiplier consume less power than existing Rounded Truncated Multiplier.

REFERENCES

[1] M. M. Peiro, E. I. Boemo, and L. Wanhammar, "Design of high-speed multiplierless filters using a nonrecursive signed COMMAON subexpression algorithm," *IEEE Transactions on Circuits and Systems II, Analog and Digital Signal Processing*, vol. 49, no. 3, pp. 196–203, 2002.

[2] F. Xu, C. H. Chang, and C. C. Jong, "Design of low-complexity FIR filters based on signed-powers-of-two coefficients with reusable common subexpressions," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 10, pp. 1898–1907, 2007.

[3] Y. J. Yu and Y. C. Lim, "Design of linear phase FIR filters in subexpression space using mixed integer linear programming," *IEEE Transactions on Circuits and Systems I, Regular Papers*, vol. 54, no. 10, pp. 2330–2338, 2007.

[4] P. K. Meher, "New approach to look-up-table design and memory-based realization of FIR digital filter," *IEEE Transactions on Circuits and Systems I, Regular Papers*, vol. 57, no. 3, pp. 592–603, 2010.

[5] P. K. Meher, S. Candrasekaran, and A. Amira, "FPGA realization of FIR filters by efficient and flexible systolization using distributed arithmetic," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3009–3017, 2008.

[6] S. Hwang, G. Han, S. Kang, and J.-S. Kim, "New distributed arithmetic algorithm for low-power FIR filter implementation," *IEEE Signal Processing Letters*, vol. 11, no. 5, pp. 463–466, 2004.

[7] C.-H. Chang, J. Chen, and A. P. Vinod, "Information theoretic approach to complexity reduction of FIR filter design," *IEEE Transactions on Circuits and Systems I, Regular Papers*, vol. 55, no. 8, pp. 2310–2321, 2008.

[8] H.-J. Ko and S.-F. Hsiao, "Design and application of faithfully rounded and truncated multipliers with combined deletion, reduction, truncation, and rounding," *IEEE Transactions on Circuits and Systems II, Express Briefs*, vol. 58, no. 5, pp. 304–308, 2011.

[9] J. M. Jou, S. R.Kuang, and R. D. Chen, "Design of low-error fixed-width multipliers for DSP applications," *IEEE Transactions Circuits and Systems II, Analog and Digital Signal Processing*, vol. 46, no. 6, pp. 836–842, 1999.

[10] L. Van, S. Wang, and W. Feng, "Design of the lower error fixed-width multiplier and its application," *IEEE Transactions Circuits and Systems II, Analog and Digital Signal Processing*, vol. 47, no. 10, pp. 1112–1118, 2000.

[11] Y. Takahashi and M. Yokoyama, "New cost-effective VLSI implementation of multiplierless FIR _lter using common subexpression elimination," *in Proceedings ISCAS 2005*, Kobe, Japan, pp. 845.848, 2005.

[12] Y. C. Lim and S. R. Parker, "FIR filter design over a discrete power-of-two coefficient space," *IEEE Transactions on Acoustics Speech and Signal Processing*, vol. ASSP-31, no. 6, pp. 583–591, 1983.

[13] Y. C. Lim and S. R. Parker, "Discrete coefficient FIR digital filter design based upon an LMS criteria," *IEEE Transactions on Circuits and Systems*, vol. CAS-30, no. 10, pp. 723–739, 1983.

[14] Y. C. Lim et al., "Signed power-of-two term allocation scheme for the design of digital filters," *IEEE Transactions on Circuits and Systems II, Analog and Digital Signal Processing*, vol. 46, no. 5, pp. 577–584, 1999.

G. Rajesh Babu and C. N. Marimuthu, "Fast multiplier for FIR filters," *International Journal of Scientific and Technical Advancements*, Volume 2, Issue 1, pp. 57-61, 2016.