DESIGN OF RADIX 4 FIR FILTER USING BOOTH MULTIPLIER FOR REDUCED POWER CONSUMPTION

Ajin Raj.D.R[1] | Sandeep.P[2] | Varatharaj M[3]

[1](Department of ECE,UG Scholar,Christ the King Engineering College,ajinrajdr@gmail.com)
[2](Department of ECE,Assistant Professor,Christ the King Engineering College,san@gmail.com)
[3](Department of ECE,HOD,Christ the King Engineering College,varatharaj_ms80@rediffmail.com)

Abstract*—Finite impulse response (FIR) filter designs mainly aims on either low area-cost or high speed or reduced power consumption. Finite Impulse Response filters are the most important element in signal processing and communication. FIR filter architecture has multiplier, adder and delay unit. So FIR filter performance is mainly based on multiplier. FIR filter using modified booth multiplier is used to reduce the power consumption and delay time in the circuit. Booth multiplier uses multilevel conditional probability (MLCP) by reducing the simulation time and easily adjusts the accuracy based on mathematical derivations. FIR filter is realized using direct form by which the area can be minimized in the circuit compared to transposed form. FIR filter is designed by using booth multiplier is implemented by HDL coding and synthesis in Xilinx tool.*

Keywords— FIR Filter, Direct form, Booth Multiplier, Multilevel Conditional probability

1. INTRODUCTION

Multiplication is one of the most consuming field arithmetic operations in high performance circuits. Fixed width multiplier takes n number of inputs and n is the number of outputs produced. Combination of fixed multiplier and multiplier Post stump, cuts half of the LSB produces the result after calculating all the products and gives a high degree of accuracy, but it accepts large circuit area .Direct cut Multiplier fixed-width multiplier cuts half LSBs reduce product directly to circuit area, but it produces truncation error. Compensated circuit has developed in this work to get the balance between accuracy and circuit area. In processing, a finite impulse response (FIR) filters whose impulse response (or response to any finite length input) is of finite duration because it settles to zero in finite time. This is in contrast to infinite impulse response (IIR) filters, which may have internal feedback and may continue to respond indefinitely (usually decaying).

We can observe that, with the increase in area, hardware cost of these FIR filters are increasing. This observation leads to design a low area-cost FIR filter with the advantages of reduced power consumption and moderate speed performance. To reduce the hardware cost, the hardware area should be optimised. Fixed-width multiplier is widely used in digital signal processing (DSP) applications, such as fast Fourier transform and discrete cosine transform.

To generate an output with the same width as the input, fixed-width multipliers truncate the half least significant bits (LSBs) in DSP applications. Thus, truncation errors can occur in fixed-width multiplier designs. The fixed-width multiplier with highest accuracy is called a post truncated (P-T) multiplier, which truncates half of the LSBs results after calculating all products. However, a P-T multiplier requires a large circuit area to calculate truncation part products. By contrast, a direct-truncated (D-T) multiplier truncates half of the LSBs products directly to conserve circuit area, but produces a large truncation error. In previous adaptive estimator conditional probability was used to improve the accuracy, it uses single non-zero code, the rounding errors during the MLCP procedure that all non-zero code calculation appreciates for cutting error. The compensated circuit is to respond quickly, creates a closed shape with different bit widths L and column information .Thus accuracy can be adjusted by changing Column information.

Multipliers consume the most amount of area in a FIR filter design. Product of two numbers has twice the original bit width of the multiplied numbers. We can truncate the product bits to the required precision to reduce the area cost. Conventional multipliers are replaced by a modified Booth multiplier here. Modified Booth is twice as fast as Booths algorithm. It produces only half the number of partial products (PPs) when compared with an ordinary binary multiplication. Modified Booth encoding (MBE) scheme is identified as the most efficient Booth encoding and decoding scheme. The truncation error for a modified Booth multiplication is not more than 1 ulp (unit of last place or unit of least precision). So there is no need of error compensation circuits. Previous designs used transposed structure to realize the FIR filter. Transposed structures are good for cross-coefficient sharing. Also, as the filter order is increasing, they will be faster. But, the area of delay elements is larger. So, it is better to use direct form structure for designing a low area-cost FIR filter.

2. FIR (FINITE IMPULSE RESPONSE) DIGITAL FILTERS

Digital filters are rapidly replacing classic analog filters.Programmable DSP with MAC can be used to

implement digital filters.For high-bandwidth signal processing applications, FPGA technology can provide multiple MACs to achieve the desired thoughput.FIR Filter with Transposed Structure - A variation of the direct FIR model is called the transposed FIR filter. It can be constructed from the direct form FIR filter by exchanging the input and output inverting the direction of signal flow and Substituting an adder by a fork, and vice versa

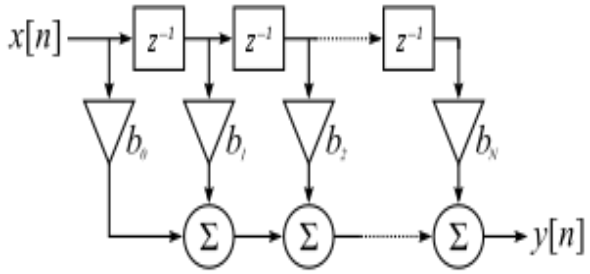

Fig 1. Direct form Structure

The FIR filter consists of three main components: A D-FF to implement a simple delay, A Multiplier to implement the coefficients, An Adder to sum the nodes at the end of each tap.

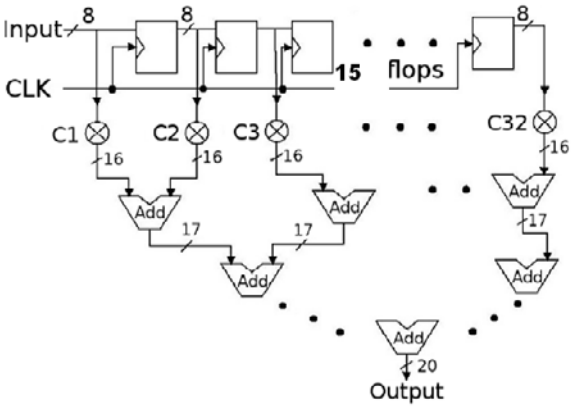

Fig 2: FIR Filter Design

Comparison of the two forms of the FIR filter:
The direct form FIR filter needs extra pipeline registers between the adders to reduce the delay of the adder tree and to achieve high throughput. The FIR filter with transposed structure has registers between the adders and can achieve high throughput without adding any extra pipeline registers. Filters are widely used in Digital communication, audio and video processing.

3. BOOTH MULTIPLIER

A systems performance is determined by the performance of the multiplier because the multiplier is generally the slowest element in the system. So, a modified Booth multiplier is suggested since it saves more area and it is faster than other conventional multipliers. The proposed new low area-cost FIR filter using a modified Booth multiplier is shown in Fig. 2. A direct form filter is such that at each clock cycle a new data sample and the corresponding filter coefficient can be applied to the multiplier‟s inputs

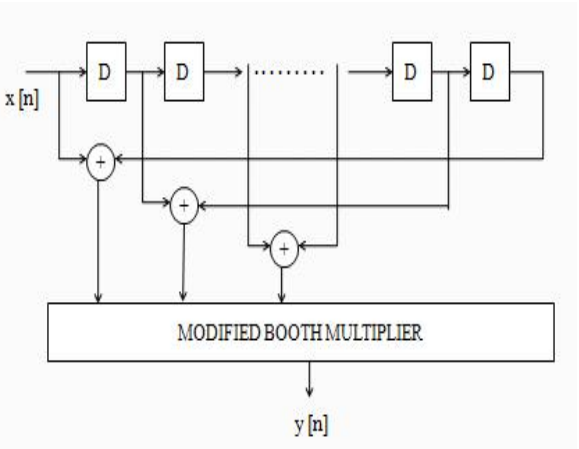

Fig 3 FIR filter Design using booth multiplier

x [n] is given as the input signal. D-FFs are used as the delay elements. Modified Booth multiplier block is provided for multiplying the input signal with the set of filter coefficients corresponding to the selected filter order. Then, modified Booth multiplier block will provide the output signal y [n].

4. MODIFIED BOOTH MULTIPLIER

Modified Radix-4 Booth‟s Algorithm is made use of for fast multiplication. The salient feature of this algorithm is only n/2 clock cycles are needed for n-bit multiplication as compared to n clock cycles in Booth‟s algorithm. This type of multiplier operates faster than an array multiplier for longer operands because its computation time is proportional to the logarithm of the word length of operands. Booth multiplication is a technique that allows for smaller, faster multiplication circuits, by recoding the numbers that are multiplied.
Modified Booth multiplier consists of Booth algorithm, including Booth encoder and Booth decoder, Wallace tree compressor (WTC) and carry look-ahead adder (CLA). Architecture of the modified Booth multiplier is shown in Fig. 2. Multiplicand X and multiplier Y are the external inputs for Booth algorithm. Usually, a multiplication includes a generation of the PPs, addition of the generated PPs until the last two rows are remained and then computing the final multiplication result by adding the last two rows.

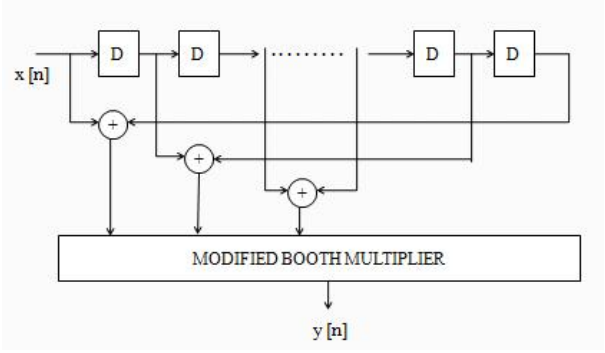

Fig. 4 Modified booth multiplier

The Booth Radix-4 algorithm reduces the number of partial products by half while keeping the circuit's complexity down to a minimum.The Radix-4 Booth Recoding is simply a multiplexor that selects the correct shift-and-add operation based on the groupings of bits found in the product register. The product register holds the multiplier. The multiplicand and the two's complement of the multiplicand are added based on the recoding value.
The Booth Radix-4 multiplier can be scaled from 4 bits up in even values such as 6, 8, 10... The user is limited by the logic density and speed of the PLD. Larger word widths require larger circuits with longer propagation delays.

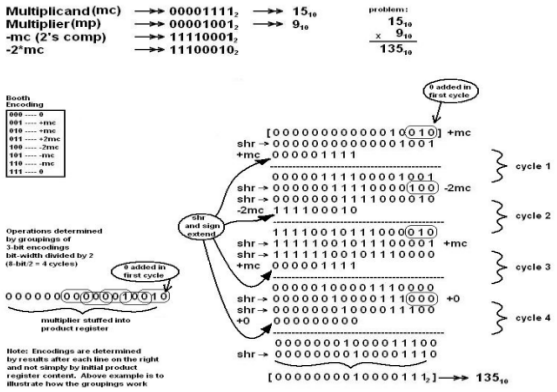

*Fig. 5 Radix 4 algorithm for 8*8 bit*

The proposed modified Radix-4 Booth encoder (MBE) is deployed in the first step of partial product generation. The general idea of radix-4 Booth encoder is that it only takes every second column of the multiplier term and multiplies with 1, 2, or 0, instead of shifting and adding for every column and multiplying by 1 or 0 in the traditional way.

5. MULTI LEVEL CONDITIONAL PROBABILITY

MLCP process creates a closed shape with different bit width L and columns Information w, so that the compensated circuit can be found quickly, and the accuracy can by set column information w.The accuracy is high in MLCP method compare to conditional probability adaptive conditional probability estimator .Although MLCP method has a higher complexity to estimate the truncation Error when compared with adaptive conditional probability estimator.

5.1 FIXED WIDTH BOOTH MULTIPLIER

Modified Booth encoding is popular to reduce the number of partial products. The modified Booth encoder maps three concatenated inputs y 2i+1, y2i , and y 2i-1, which are tabulated in Table I,where P{y'i} stands for the probability of y'i . After encoding, there are Q =L/2 rows in the partial product array with an even width L. The corresponding partial products into y'I represented in input xi are tabulated in Table II, where the last column n stands for the sign of each partial product.

Fixed width Stand encryption algorithm is are mostly used in multiplier designs, to reduce the number of the partial products. The partial product array in a cabin Multiplier for induction of the column Information w, where w is the Included Number of true product columns in the compensated circuit.

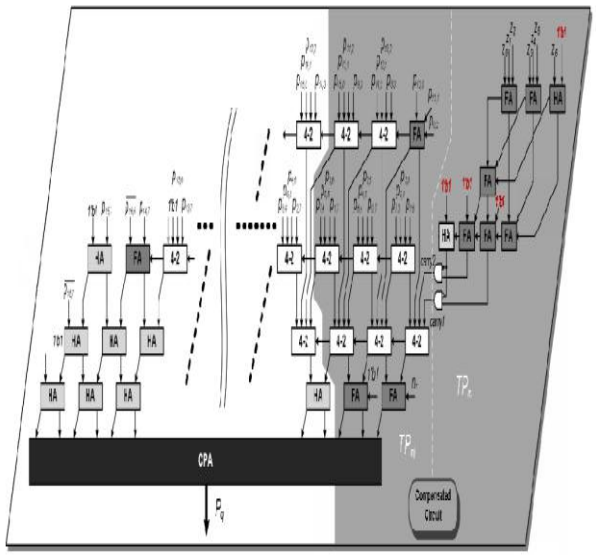

Fig.6 Proposed MLCP with booth multiplier

6. RESULTS AND DISCUSSION

The proposed system provides a comparison of accuracy power consumption and speed performance.

6.1 *ACCURACY*

In the architecture described above, we reduce the number of partial product compared. But it's the output width is the same as that of the input.

6.2 *POWER AND DELAY*

Hence by reducing the number of partial product so the circuit consumes low power and great improvement in speed performance

7. SIMULATION RESULTS

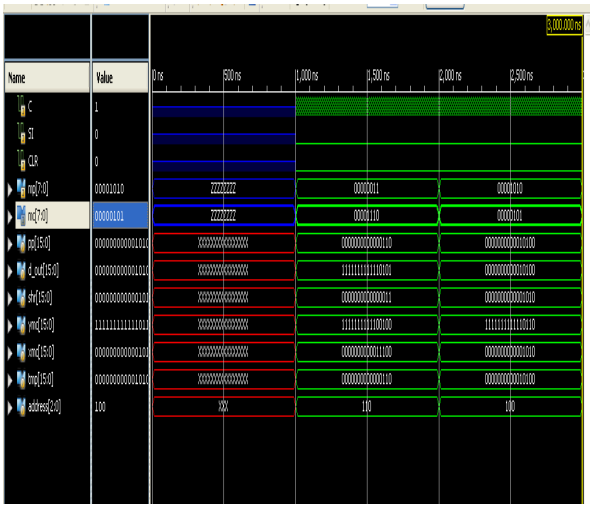

Fig 7 Radix 4 Algorithm for partial product generation

Fig 8 : Booth multiplication for Radix 4 Algorithm

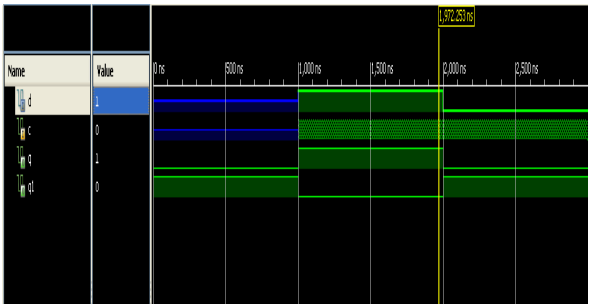

Fig 9 D flip flop design in FIR Filter

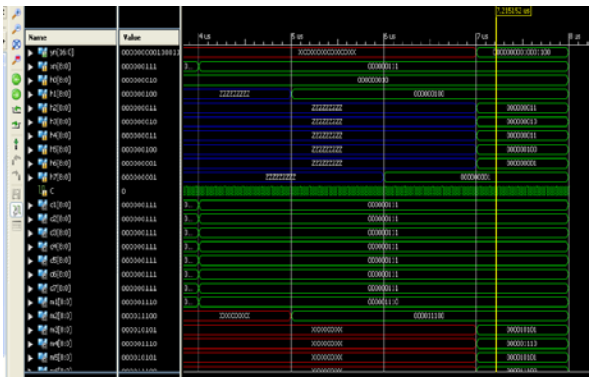

Fig .10 FIR Filter using Radix 4 Algorithm

```
Data Path: xn<0> to yn<8>
                                        Gate     Net
  Cell:in->out      fanout   Delay     Delay  Logical Name (Net Name)
  ----------------------------------------  ------------
   IBUF:I->O              2    1.106   0.380  xn_0_IBUF (xn_0_IBUF)
   MULT18X18SIO:A0->P1    3    3.251   0.603  u0/Mmult_result_mult0001 (m1<1>)
   LUT2:I0->O             0    0.612   0.000  Madd_a2C1 (Madd_a2C)
   MUXCY:DI->O            1    0.773   0.000  Madd_a2_Madd_cy<2> (Madd_a2_Madd_cy<2>)
   MUXCY:CI->O            1    0.052   0.000  Madd_a2_Madd_cy<3> (Madd_a2_Madd_cy<3>)
   MUXCY:CI->O            1    0.052   0.000  Madd_a2_Madd_cy<4> (Madd_a2_Madd_cy<4>)
   XORCY:CI->O            1    0.699   0.509  Madd_a2_Madd_xor<5> (Madd_a4R4)
   LUT1:I0->O             1    0.612   0.000  Madd_a4_Madd_cy<5>_rt (Madd_a4_Madd_cy<5>_rt)
   MUXCY:S->O             1    0.404   0.000  Madd_a4_Madd_cy<5> (Madd_a4_Madd_cy<5>)
   XORCY:CI->O            1    0.699   0.509  Madd_a4_Madd_xor<6> (Madd_a6R5)
   LUT1:I0->O             1    0.612   0.000  Madd_a6_Madd_cy<6>_rt (Madd_a6_Madd_cy<6>_rt)
   MUXCY:S->O             1    0.404   0.000  Madd_a6_Madd_cy<6> (Madd_a6_Madd_cy<6>)
   XORCY:CI->O            1    0.699   0.509  Madd_a6_Madd_xor<7> (Madd_yn_addsub0000_lut<7>)
   LUT1:I0->O             1    0.612   0.000  Madd_yn_addsub0000_cy<7>_rt (Madd_yn_addsub000C
   MUXCY:S->O             1    0.404   0.000  Madd_yn_addsub0000_cy<7> (Madd_yn_addsub0000_cy
   XORCY:CI->O            1    0.699   0.357  Madd_yn_addsub0000_xor<8> (yn_8_OBUF)
   OBUF:I->O                   3.169          yn_8_OBUF (yn<8>)
  ----------------------------------------
  Total                      17.725ns (14.858ns logic, 2.867ns route)
                                      (83.8% logic, 16.2% route)
```

Fig .11 Delay report of FIR Filter

```
2.  Summary
2.1.  On-Chip Power Summary
-------------------------------------------------------------------------------
|                          On-Chip Power Summary                              |
-------------------------------------------------------------------------------
|       On-Chip       | Power (mW) |  Used  | Available | Utilization (%) |
-------------------------------------------------------------------------------
| Clocks              |       0.00 |      0 |       --- |             --- |
| Logic               |       0.00 |     33 |      1920 |               2 |
| Signals             |       0.00 |     92 |       --- |             --- |
| IOs                 |       0.00 |     44 |        66 |              67 |
| MULTs               |       0.00 |      2 |         4 |              50 |
| Quiescent           |      33.59 |        |           |                 |
| Total               |      33.59 |        |           |                 |
-------------------------------------------------------------------------------
```

Fig. 12 Power report of FIR Filter

$$X=-X_{L-1}.2^{L-1}+\sum_{i=0}^{L-2} X_i.2^i$$

$$Y=-Y_{L-1}.2^{L-1}+\sum_{i=0}^{L-2} Y_i.2^i$$

$$SP=X \times Y$$

TABLE I :

COMPARISION TABLE

ALGORITHM	NO OF BITS	POWER (MW)	DELAY (ns)
BOOTH MULTIPLIER USING MAT LAB	8	50.21	20
BOOTH MULTIPLIER USING FIR FILTER DESIGN	8	33.59	17.725

8. CONCLUSION

This paper proposes an adaptive conditional-probability estimator (ACPE) in fixed-width Booth multipliers. Without heuristic method, the ACPE is derived from conditional probability theory, which can be easily applied to large length Booth multipliers for achieving higher accuracy performance. The column information is also induced in ACPE to adjust the accuracy with respect to system requirements. For the 2-D DCT implementation with the ACPE Booth multipliers, an excellent performance is exhibited in terms of PSNR and area cost. As a result, the proposed ACPE provides a flexible and high accuracy compensation circuit applied to fixed-width Booth multipliers. A highly area-efficient FIR filter using modified Booth encoding scheme is designed based on the direct form realization. FIR filters are also designed using MCMAT and using an older version truncated multiplier for comparison. The results show that the modified Booth multiplier based FIR filter leads to reduced power consumption and reduced delay time.

REFERENCES

[1]. Yuan Ho Chen, "An accuracy adjustment fixed width booth multiplier using multilevel conditional probability," in IEEE Transactions On Very Large Scale Integration (Vlsi) Systems, Vol. 23, No. 1, January 2015.
[2] J. Han and M. Orshansky, "Approximate computing: An emerging paradigm for energy-efficient design," in ETS, 2013, pp. 1–6.
[3] K.-J. Cho, K.-C. Lee, J.-G. Chung, and K. K. Parhi, "Design of lowerror fixed-width modified booth multiplier," IEEE Transactions on VLSI Systems, vol. 12, no. 5, pp. 522–531, 2004.
[4] J.-P. Wang, S.-R. Kuang, and S.-C. Liang, "High-accuracy fixedwidth modified booth multipliers for lossy applications," IEEE Transactions on VLSI Systems, vol. 19, no. 1, pp. 52–60, 2011.
[5] C.-Y. Li, Y.-H. Chen, T.-Y. Chang, and J.-N. Chen, "A probabilistic estimation bias circuit for fixed-width booth multiplier and its dct applications," IEEE Transactions on Circuits and Systems II, vol. 58, no. 4, pp. 215–219, 2011.
[6] Y.-H. Chen, C.-Y. Li, and T.-Y. Chang, "Area-effective and powerefficient fixed-width booth multipliers using generalized probabilistic estimation bias," IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol. 1, no. 3, pp. 277–288, 2011.
[7] Y.-H. Chen and T.-Y. Chang, "A high-accuracy adaptive conditional-probability estimator for fixed-width booth multipliers," IEEE Transactions on Circuits and Systems I, vol. 59, no. 3, pp. 594–603, 2012.