International Journal of Scientific & Engineering Research, Volume 3, Issue 9, September-2012
ISSN 2229-5518

Advanced Approach To Look-up Table Design For Memory Based Realization of FIR Digital Filter

K.SRIKARSH VARDHAN REDDY

Abstract— In look-up-table(LUT)-multiplier-based approaches for an efficient memory-based implementation of finite impulse response (FIR) filter is present where the memory elements store all the possible values of products of the filter coefficients. But in this paper, memory locations in look-up table where reduced to 4 from 8 could be also an area-efficient alternative to DA-based design of FIR filter with the same throughput of implementation. We present a modified transposed form FIR filter, where a single segmented memory-core with only one pair of decoders are used to minimize the combinational area. The proposed LUT-based FIR filter is found to involve nearly half the memory-space and (1/N)times the complexity of decoders and input-registers, at the cost of marginal increase in the width of the adders, and additional ~(4N x W) AND-OR-INVERT gates and ~(2N x W) NOR gates.

Index Terms— DSP,FIR filter, LUT based multiplier,Xilinx synthesis tool.

1 INTRODUCTION

Finite impulse response (FIR) digital filter is widely used as basic tool in various signal processing and image processing applications . In which Multipliers are key components of high per formance FIR filters. A system's per formance is generally determined by the performance of the multiplier because the multiplier is generally the slowest element in the system. Furthermore, it is generally the most area consuming. Hence, optimizing area of the multiplier is a major design issue. Memory-based multipliers are well-suited for designing the FIR digital filters, which involve multiplication with a fixed set of coefficients, and consumes less area than non memory based multipliers like Wallace and array multiplier. In the "memory-based structures" or "memory-based systems" for those systems where memory elements like RAM or ROM is used either as a part or whole of an arithmetic unit .Memorybased structures are more regular compared with the multiplyaccumulate structures; and have many other advantages, of greater potential for high-throughput and reduced-latency implementation, and are expected to have less area comparedto the conventional multiplier . There are two basic variants of memor y-based techniques for digital FIR filters for DSP applications. One of them is based on distributed arithmetic (DA) for inner product computation and the other is based on the computation of multiplication by look-up-table (LUT) with inner-product-length LUT-multiplier-based implementation where the memory-size is reduced to nearly half of the conventional approach.

- *K.Srikarsh Vardhan Reddy is currently pursuing masters degree program in Vlsi System Design inJawahar Lal Nehru Technological University Hyderabad ,India, PH- 9440617208. E-mail:srikarsh31988@gmail.com*

2. DISTRIBUTED ARITHMETIC MEMORY BASED MULTIPLICATION

DA is basically (but not necessarily) a bit-serial computational operation that forms an inner (dot) product (multiply and accumulation) of a pair of vectors in a single direct step. In the DA-based approach, an LUT is used to store all possible values of inner-products of a fixed N - point vector with any possible N-point bit-vector. If the inner-products are implemented in a straight-forward way, the memory-size of DA based implementation increases exponentially with the inner-product-length. Attempts have been made to reduce the memory-space in DA-based architectures for reducing the memory-size of DA-based implementation of FIR filter. But, it is observed that the reduction of memory-size achieved by such decomposition is accompanied by increase in latency as well as the number of adders and latches. The above Fig. 1 shows the Distributed arithmetic multiplier.

3. Look up Table based Memory Multiplication

In the LUT-multiplier-based approach, multiplications of input values with a fixed-coefficient are performed by an LUT consisting of all possible pre-computed product values corresponding to all possible values of input multiplicand.If the inner products are implemented the memory-size of LUT multiplier implementation increases exponentially with the word length of input values, while that of the DA based approach increases exponentially with the inner-product length. In this paper ,we are presenting the LUT-multiplier based implementation where it is having several advantages than conventional non memory based multiplications like array and Wallace multiplications generally used in digital FIR filter implementation.Let A be a fixed coefficient and X be an input word to be multiplied with A. If we assumeX to be an unsigned binary number of word-length L, there can be 2L possible values of X, and accordingly,there can be 2 possible values of product C=AX .The proposed LUT based multiplier method needs only 2L L/2 words to the odd multiplies of A only stores in the LUT.One of the possible product words is

zero,while all the rest 2/2 -1 are even multiples of A which could be derived by left-shift operations of one of the odd multiples of A.Table I for L=4.At eight memory locations, eight odd multiples A X (2i+1) are stored as P L for i=0,1,2…,7.The even multiples 2A,4A,8A are derived from shift operations of A.Similarly, 6A,12A are derived by left shifting 3A, while 10A and 14A are derived by left shifting 5A,7A respectively.The address=(0000) corresponding to (A.X)=0,which can be obtained by resetting the LUT output.For an input multiplicand of word size L similarly,only 2L I/2 odd multiple values need to be stored in memory-core of LUT, while the other 2L/2-1 non-zero values could be derived by left-shift operations of the stored values. Based on above , the proposed LUT based multiplier for input L=4 word size is shown in fig a.It consists of memory array of eight words of (W+4) –bit width and 2 to 4 line address decoder along with a nor cell , a barrrael shifter, a 4 to 2 bit encoder to map the 4 bit input operand to 2 bit LUT- address and a control circuit for generating the control-word(s0,s1,f0) for the barrel-shifter with adder, and the RESET signal for the NOR-cell. The 4 to 2 bit input encoder receives a four-bit input word(x3,x2,x1,x0) and maps that onto the three-bit address word (d1,d0) according to the logical relations shown in equation (1) and equation (2).

The decoder takes the 3-bit address from the input encoder, and generates 8 word-select signals, to select the referenced word from the memory-array. The output of the memory-array is either AX or its sub-multiple in bit-inverted form depending on the value of X. From Table I, we find that the LUT output is required to be shifted through location to left when input X is one of {(0010),(0110),(1010),(1011),(1110),(1111)}.Two left-shifts are required if X is either (0100),(1100),(1101).Only when the input word X=(1000),(1001)three shifts are required. The number of shifts required to be performed on the output of the LUT and the control-bits s0, s1 and for different values of X are shown Table I.The f0 is used to add 0100 after shifting as mentioned in the table. The control circuit [shown in Fig. 3] accordingly generates the control-bits given by equation (3) and equation (4)

A logarithmic barrel-shifter for W=L=4 consists of two stages of 2-to-1 line bit-level multiplexors with inverted output, where each of the two stages involves (W+4) number of 2-input AND-OR-INVERT (AOI) gates. The control-bits (s0, s0l), (s1,s1l) and are fed to the AOI gates of stage-1 ,stage-2 of the barrel-shifter, respectively. Since each stage of the AOI gates perform inverted multiplexing, after two stages of inverted multiplexing, outputs with desired number of shifts are produced by the barrel-shifter in un-inverted form.

The input X = (0 0 0 0) corresponds to multiplication by X = 0 which results in the product value A.X = 0 Therefore, when the input operand word X = (0 0 0 0), the output of the LUT is required to be reset. The reset function is not implemented by a NOR-cell consisting of (W+ 4) NOR gates using an active-high RESET. The RESET bit is fed as one of the inputs of all those NOR gates, and the other input (W+4) lines of NOR gates of NOR cell are fed with (W+4) bits of LUToutput in parallel. When word X = (0 0 0 0), the control circuit in Fig. (3), generates an active high RESET according to the logic expression: equation (6).

When RESET = 1, the outputs of all the NOR gates become 0, so that the barrel shifter is fed with (W+4) number of zeros. When RESET =0, the outputs of all the NOR gates become the complement of the LUT output-bits. Note that, keeping this in view, the product values are stored in the LUT in bit-inverted form.

4. Realization of digital FIR filter using proposed LUT based Multiplier

The Realization of digital FIR filter using proposed LUT based multiplier is done by using direct form realization structure of digital FIR filter. The equation, which defines the FIR filter with output sequence y[n] in terms of its input sequence x[n]: (4) Where x[n] is the input signal, y[n] is the output signal, h[k] is the coefficients of FIR filter frequency response, and N is the filter order. The fig. 3 shows the direct form realization of digital FIR filter. From this figure 3 the input X is delayed and given to multiplier each multiplier gives products corresponding to different filter coefficients and all these products are accumulated and give fir filter output. The proposed LUT multiplier is used in the above Fig. 2 in which each multiplier is having fixed filter coefficients ,the inputs are delayed and given to this LUT multiplier .A memory-unit of (2L/2) words of (W+L) bit width is used to store all the odd multiples of filter coefficient. The L-bit input word is mapped to (L-1) -bit LUTaddress by an encoder.The barrel-shifter is used to derive all the even multiples of filter coefficient. The required control-bits for the barrel shifter are derived by Control-circuit to perform the necessary shifts of the LUT output. RESET signal is generated by the same control circuit to reset the LUT output when X = 0.There by corresponding products which are stored in the LUT of particular input given to LUT based multiplier based circuit in Fig. 1 are obtained. These products are finally accumulated and give as FIR filter output based on number of taps for a given filter .The FIR filter realized using proposed LUT based multiplier is shown in fig (1).

5. EQUATIONS

equation(1)-> $d0=x2(x3+x1)+\overline{x3}(x0.x1+\overline{x0}.\overline{x1}.\overline{x2})$

equation(2)-> $d1=(\overline{x3}(x0.x2+\overline{x0}.\overline{x1}.\overline{x2}))+(x3.x1)$)

EQUATION(3)-> $s0 =x3(\overline{x2}+x1)+\overline{x0}(x1+\overline{x2})$

equation(4)-> $s1 =\overline{x1}(x3+\overline{x0})$ $f0 = (x3.x0)$

equation(5)

$$\mathrm{RESET} = \overline{(x_0 + x_1)} \cdot \overline{(x_2 + x_3)}$$

6 HELPFUL HINTS

6.1 Figures and Tables

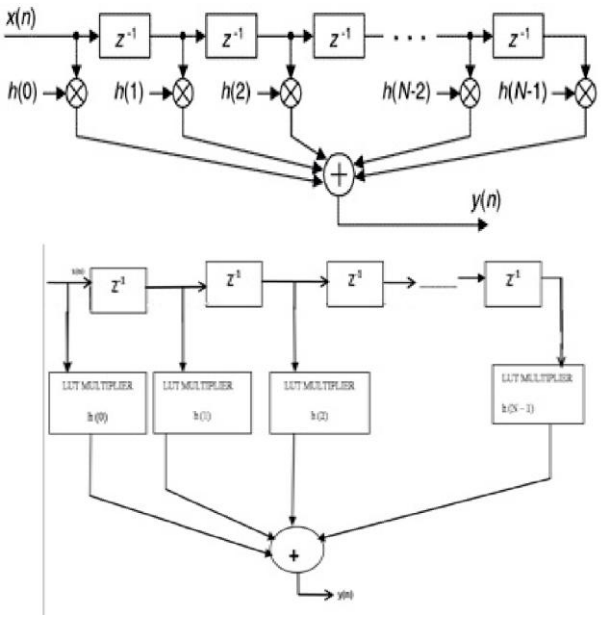

Fig. 1 Direct form realization of FIR
&
Realization of digital FIR filter using proposed LUT basedmultiplier

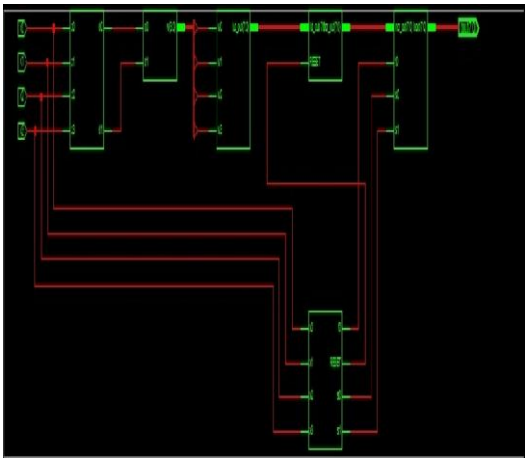

Fig. 2 Proposed LUT multiplier

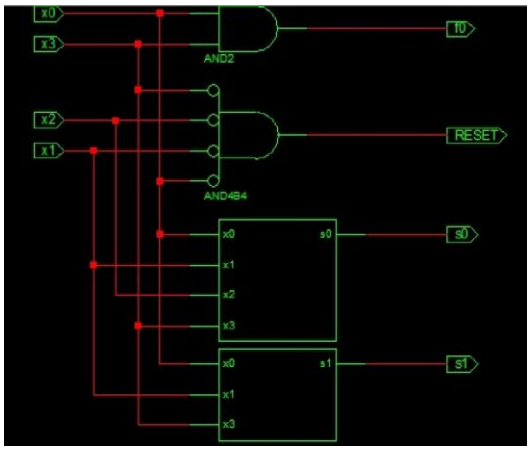

Fig. 3. **control circuit**

Address (d1d0)	Word symbol	Stored value	Input (x3x2x1x0)	Product Value	No of shifts	Control (s1) (s0)	(f0)
00	P0	A	0001	A	0	00	0
			0010	2^1XA	1	01	0
			0100	2^2X A	2	10	0
			1000	2^3X A	3	11	0
			1001	2^3X A	3	11	1
01	P1	3A	0011	3A	0	00	0
			0110	2^1X3A	1	01	0
			1100	2^2X3A	2	10	0
			1101	2^2X3A	2	10	1
10	P2	5A	0101	5A	0	00	0
			1010	2^1X5A	1	01	0
			1011	2^1X5A	1	01	1
11	P3	7A	0111	7A	0	00	0
			1110	2^1X7A	1	01	0
			1111	2^1X7A	1	01	1

Fig 4: LUT Words and Products Values for Input Word Length L = 4

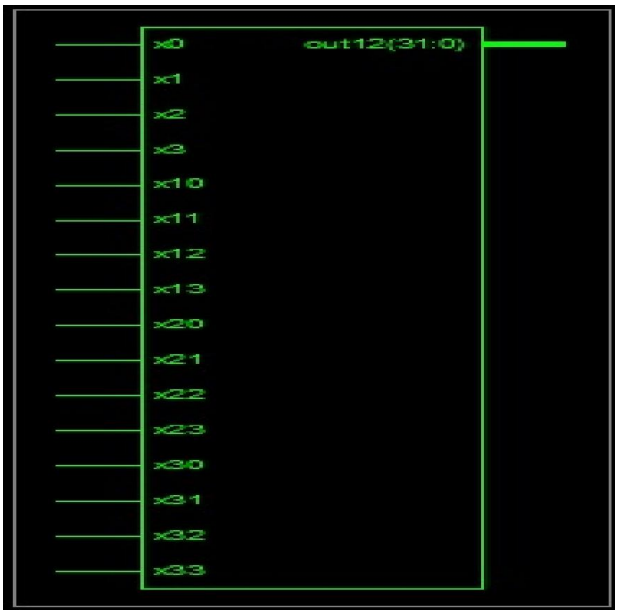

Fig. 5 Top Module

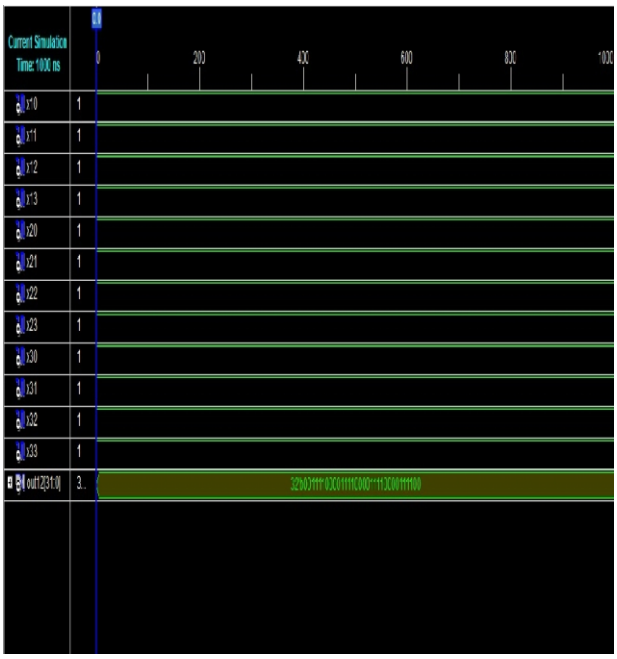

Fig. 7. Simulation Results of Top Module

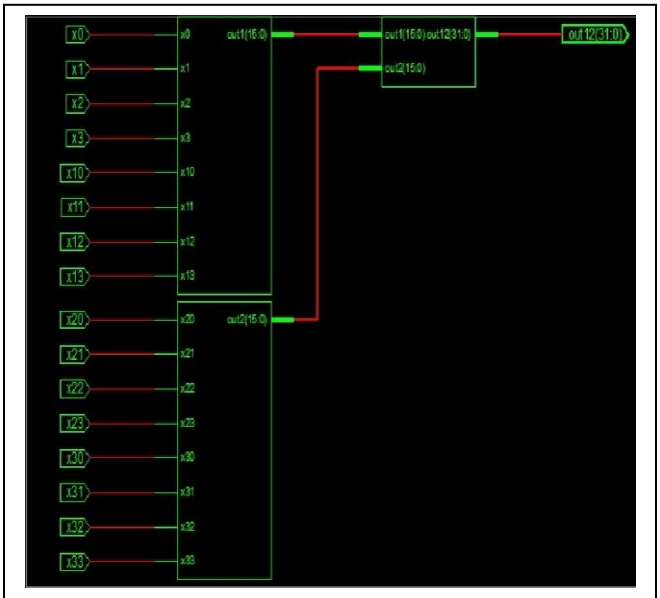

Fig. 6.

6.2 Snthesis Report

```
=========================================
================================
*                Final Report                *
=========================================
  ================================

  Final Results
  RTL Top Level Output File Name      : main01.ngr
  Top Level Output File Name      : main01
  Output Format      : NGC
  Optimization Goal            : Speed
  Keep Hierarchy      : YES
  Target Technology            : Automotive 9500XL
  Macro Preserve      : YES
  XOR Preserve      : YES
  Clock Enable            : YES
  wysiwyg                : NO

  Design Statistics
  # IOs                  : 48

  Cell Usage :
  # BELS                 : 508
  #  AND2                : 172
  #  AND3                : 4
  #  AND4                : 8
  #  GND                 : 4
  #  INV                 : 200
  #  OR2                 : 112
```

```
#  OR3                    : 4
#  VCC                    : 4
# Tri-States              : 32
#  BUFE             : 32
# IO Buffers              : 48
#  IBUF                   : 16
#  OBUF                   : 32

=========================================
==============================

CPU : 5.44 / 5.59 s | Elapsed : 6.00 / 6.00 s
--
>
```

Total memory usage is 180536 kilobytes

7.CONCLUSION

Advanced approaches to LUT–based–multiplication are suggested to reduce the LUT-based–multiplication are suggested to reduce the LUT-size over that of conventional design. By odd-multiple storage scheme and successive addition ,for address –length 4, the LUT size is reduced to half by using a two-stage logarithmic barrel shifter and adder .Therefore LUT multipliers could be used highspeed hardware implementation of digital filters and also for memory-based implementation of cyclic and linear convolutions, sinusoidal transforms ,and inner- product computation.The performance of memory based structures,with different adder and memory implementations could be studied in future for different DSP applications.

8.REFERENCES

[1] G.Mirchandani, R.L.Zinser Jr., J.B.Evans, "a new adaptive noise cancellation scheme in the presence of crosstalk [speech signals],"IEEE Trans Circuits Syst. II Analog Digital Signals Process vol .39,no. 10, pp.681-694, Oct 1995.

[2] H-R.Lee ,C-W Jen C-M Liu,"On the design automation of memory-based VLSI architectures for FIR filters"IEEE Trans.Consum. Electron. vol.39, no. 3,pp.619-629,Aug 1993.

[3] S.A.White, "Applications of the distributed arithmetic to digital signal processing :A total review," IEEE ASSP.

[4]P.K.Meher,"New approach to LUT implementation and accumulation for memory based multiplication," in proc. 2009 IEEE Int.Symp.Circuits Syst., ISCAS'09, May 2009, pp. 453-456.