# IMPLEMENTATION OF HIGH SPEED AND LOW AREA RECONFIGURABLE FIR FILTER USING MULTIPLE CONSTSANT MULTIPLIER

# G.SOWMYA I.SUGITHA Dr. K. K. SENTHIL KUMAR

Student, ECE, Prince Shri Venkateshwara Padmavathy Engineering College, Tamilnadu, India Student, ECE, Prince Shri Venkateshwara Padmavathy Engineering College, Tamilnadu, India Associate professor, ECE, Prince Shri Venkateshwara Padmavathy Engineering College, Tamilnadu,

India

# ABSTRACT

The direct Form FIR filter is used for DSP applications where the filter order is fixed. Generally this filter consumes more area and power. To overcome this problem reconfigurable architecture of transpose form FIR filter which is similar to direct form FIR filter where only delay and multiplier unit switched positions is incorporated. The RFIR structure consist of Coefficient selection unit (CSU), register unit(RU),Inner product unit (IPU) and pipeline adder unit(PAU). The PAU is used to add output from the IPU unit .The CSU is used to store coefficient of all the filter to be used for the reconfigurable application. Here RFIR is designed using a IPU which consumes more area and delay. The modified RFIR is designed further to reduce the area and delay. The modified RFIR filter the IPU is replaced by the multiple constant multiplier (MCM). The MCM consumes less area, delay than compared methods.

**Keyword:** *Finite impulse response filter(FIR), Reconfigurable architecture, Multiple constant multiplier, etc.* 

# **1.INTRODUCTION**

Finite impulse response filter is the filter where the impulse response is of finite duration and it settles zero at finite time. This is in contrast to the infinitely impulse response which have feedback and response at infinite duration. For a FIR filter order N the output sequence is given as the sum of the recent inputs.

 $Y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] \cdots b_N x[N-1]$ 

 $= \sum_{i=0}^{N} x[N-i]$ 

1

- where x[n] is the input signal,
- y[n] is the output signal,
- N is the filter order; an Nth-order filter has (N+1)terms on the right-hand side
- $b_i$  is the value of the impulse response at the i'th instant for  $0 \le i \le N$  of an Nth-order FIR filter.

The output of the FIR filter is given in form Z transform of the FIR frequency response. The basic structure of the FIR filter of order N is given as follows



FIG 1 FIR Filter

Where  $z^{-1}$  is the delay element and x[n] is the input impulse and y[n] is the output of the FIR filter where the output is given in recurrence relation represented in the above equation. The above figure is given as the direct form FIR filter. The FIR is commonly used in the DSP application like equalization, software defined radio, and other digital communication applications.

#### **1.1.OBJECTIVE**

Generally the software defined radio require many large order filter, this large order filter occupies more area and delay is large. Here the design becomes complex to reduced this complexity the various techniques was introduced. The proposed structure of the FIR filter is designed as Reconfigurable FIR filter and the coefficients are fixed in order to reduce the multiplication complexity.

# **2.RECONFIGURABLE FIR FILTER**

Reconfigurable FIR filter is designed such that the it consist of a coefficient storage unit(csu), register unit (RU). Inner product unit(IPU) and a pipeline adder unit. The reconfigurable FIR filter is given as follows



Fig 2 Reconfigurable FIR filter architecture

3

4

The figure 2 shows the architecture of the RFIR which shows all the blocks. Here the CSU stores the coefficients of all the filter to be used. It is implemented using the LUT's, such that the filter coefficient is obtained in one clock cycle. The coefficient is given as  $c_m$  which is given by a equation

$$C_{m} = \{h(mL), h(mL+1), h(mL+2), \dots, h(mL+L-1)\}$$
 2

Where L is the block size.

The register unit (RU) gets a input x at the  $k_{th}$  cycle and produces a output with respect to the block size L. The output of the register unit is given to the M IPU unit. Where M is given as N\L. The inner product unit gets input from the coefficient storage unit and the register unit. In IPU unit the matrix multiplication of the output from the CSU unit and RU is performed. Then the partial product produced at the inner product unit is given to the pipeline adder unit (PAU). All the IPU acts parallel and give the output to the PAU unit. The output of the PAU unit is given in recurrence relation.

$$Y_{k} = \sum_{m=0}^{M-1} r_{k}^{m}$$

 $r^{m}_{k}=S^{0}_{k-m}.c_{m}$ thus the Y<sub>k</sub> is given in the form as follows  $Y(z)=S^{0}(z)[z^{-1}((z^{-1}($ 

$$(z) = S^{0}(z)[z^{-1}((z^{-1}(z^{-1} c_{M-1} + c_{M-2}) + c_{M-3}) + \dots) + c_{1}) + c_{0}]$$

the equation 3 represents the output of the PAU unit. Where,

- $Y_k$  is the output of the signal
- S<sup>0</sup> is the signal from the register unit
- C<sub>m</sub> is the coefficient vector

The 4 represents the output signal of  $y_k$  given in the z-domain.

But in this implementation of the RFIR with inner product unit large area is required which is comparatively less compared to the N no of direct form FIR filter. Thus inorder to reduce the area and to increase the performance a method called multiple conatant multiplier is used.

### **3.MULTIPLE CONSTANT MULTIPLIER**

Multiple constant multiplication (MCM) is defined as a technique used to reduce the number of addition/subtraction operations required for the multiplication of multiple constants by an input variable . Multiplier Constant Multiplications is a way constant multiplications using a shift-adds architecture. It is first to define the constants under a particular number representation, and second, for the nonzero digits in the representation of the constant, is to shift the input variable according to the digit positions and add/subtract the shifted variable with respect to the digit value.



Fig 3 multiple constant multiplier

Fig 3 represents the basic MCM block where a input x is multiplied with a constant coefficient  $t_0, t_1, \ldots, t_n$  and the output is obtained parallelly .MCM uses shifter for multiplication in order to reduce the number of multipliers and adder used for calculating the partial products. This multiple constant multiplier can implemented only in the transpose form FIR filter. The transpose form FIR filter is similar to that of the direct form FIR filter only difference is that the delay and multiply unit is interchanged. So that the MCM can implemented easily on the FIR filter.

#### **3.1 RECONFIGURABLE FIR FILTER USING MCM**



The figure 4 represents the MCM based architecture of the reconfigurable FIR filter. The derivation of MCM units for transpose form block FIR filter is shown in the figure. For fixed-coefficient implementation, the CSU is no longer required, since the structure is to be tailored for only one given filter. Similarly, IPUs are not required. The multiplications are given to the MCM units for a low-complexity realization. The proposed method for MCM-based implementation of block FIR filter makes use a symmentry input matrix  $S_k^0$  to perform horizontal and vertical common subexpression elimination and to minimize the number of shift-add operations in the MCM blocks. MCM block produces the necessary product terms. The subexpression of the MCM blocks are shift adding the adder network to produce the inner-product values (rl,m), for  $0 \le l \le L - 1$  and  $0 \le m \le (N/L) - 1$  corresponding to the matrix product. The inner-product values are finally added in the PAU. In pipeline adder unit the

# 4 RESULTS AND COMPARISON

4.1 Results of IPU based FIR filter

4.1.1 Simulation output

se wave - default		
Messages		
	01 00000)00000	2 200000
abcde/h7 000000	01 00000001 00000010 000	300001 X00000010 X0000001 X000000 X0 X0000001 X000000010 X00000001 X00000010 X00000001 X000000010 X
🗖 🔷 /abcde/h8 000000	11 0000 000000100 000000	011 X0000100 1000000011 X00000000000000
🖸 🐣 /abcde/h9 000000	01 ) (00000001 ( )0000	KONT X X00000001 X X00000001 X X00000001 X X00000001 X X00000001 X     X00000001 X     X000000001 X     X000000001 X     X000000001 X     X000000001 X     X000000001 X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X000000001     X     X00000000     X     X00000000
/abcde/h10 000000		
/abcde/h11 000001		
Abcde/h12 000000	11 )00 (0 (0 (00	1 Yao Y. YaoYoYooYoYooYoYoYoYoYoYoYoYoYoYoYo
/abcde/h13 000000	01 000 job0 joc00 jocq	
/abcde/h14 000000	10 0000 (000000 (000000)	😰 X20000 X200000.10 👔 20000 X200000.10 X20000 X200000.10 🕱 20000 X200000.00 X20000
Abcde/h15 000001	00 (00000101 (00000100	00000101 00000100 00000101 00000100 00000101 00000100 00000101 00000100 0
/sbcde/y0 000000	0000100011 (())(())(())(()()()()()()()()()()()()	
Abcde/y1 000000		
/abcde/y2 000000		X X00XX X00IX X I X X I X X X X X X X X X X X X
(a) (a) (a) (b) (b) (c) (c) (c) (c) (c) (c) (c) (c) (c) (c		<u></u>
/abcde/uut/dk St1		
/abcde/uut/rst St0		
Abcde/uut/x0 000000	91. ( <b>6</b> )0777) <b>(6</b> )0777( <b>6</b> )	⊈Xo‡Xo‡Xo‡Xo1X ‡Xo1 10‡X 1Xo‡101Xo‡Xo1 10‡Xo1 Xo1
Abcde/aut/x1 000001	00 (00(000)00	🟚 (oso
/abcdie/uut/x2 000001	01 (00000 (0000 (000	še, jacad jacedo jaced jaced jaced jacedo jacedo jacedo jacedo jacedo jacedo j
//abcde/ucit/x3 000000	11 20000 [00000010 [000	30011 X00000010 X00000011 X00000010 X00000010 X00000010 X00000011 X00000010 X00000011 X0
Abcde/uut/h0 000001	00 (00000	§a., joooq., joocos., jooco., joocoq., joocoq., jooco., jooco., jooco., jooco., jooco., jooco., jooco., j
/sbcde/uut/h1 000000	10 20000 (0000000 1 (000	10010 X0000001 X0000010 X0000001 X00000010 X00000001 X00000001 X00000001 X00000010 X00000010
/abcde/uut/h2 000000	11 00)0)00(0(00	1 Yoo. Y Yoo. Yo Yoo. Yo Yoo. Yo Yoo. Yo. Y
Abcde/uut/h3 000001	01 00000000	. Xo. Xod. Io. Ioo. Io. Ioo. Io. Ioo. Xo. Xo. Xod. Ioo. Xo. Xod. Xo. Xod. Xo. Xod. Xo. Xod. Xod
- Ebols Aut Ed 000000	te teo teo teo teo	<mark>d Yon t</mark> ana Yan Nan Nan Yan Tan Yang Nan Yang Yang Yang Yang Yang Yan t
Now	187790 ps 187000 p	187200 ps 187400 pa 187600 ps 187800 ps
Cursor 1	187030 ps 1870	30 ps

Fig 5 Modelsim output for reconfigurable FIR filter

By following the procedure to run the Modelsim the simulation results are obtained. Figure 5 represents simulation results of reconfigurable FIR filter

#### 4.1.2 Area

Start +□ & ×	🙀 🖹 Design Overview 🖉	Design Goal:	Balanced	2	• Routing Results:	Al Sonais
Design 🗗 🛃 🕹	C C Servery	Design Strategy:	Ninx Default Suniode	edi	•Timing Constraints	e Al Carut
👕 Vex: 🖲 🎇 Indementation 🔿 🞆 Simulation	Module Level Utilization	Environment:	System Settings		+Final Timing Score	d (Timing
Hearthy	Timing Constraints     Pinout Report     Gock Report			Device Utilization S	oninary	
B B W type	Static Timing	Logic Utilization		Used	Available	Utilization
8	Parser Mestages	Number of Sice Flip Flops	6	110	4,895	2%
	A Synthesis Messages	Number of 4 input LUTs		354	4,896	3%
2	Map Messages	Number of occupied Sices		301	2,448	4%
<u>*</u>	Place and Route Messages	Number of Sices contain	ring only related logic	101	101	100%
	Timing Messages	Number of Sices contain	ning unvelated logic	0	101	0%
	All implementation Messages	Total Number of 4 input (1)	Ts .	336	4,896	3%
No Processes Running	Detailed Reports	Number used as logic		:254		
Processes: yyy	a) Synthesis Keport	Number used as a route	-thu			
10 View RTL Schematic	Design Properties	Number of bonded 108s		98	172	96%
Check Suntax	Ontonal Design Summary Contents	Number of BUFGMUNs			24	4%
Generate Post-Synthesis	9 Show Clock Report	Number of MULT 181 18530	8	11	12	91%
B (20) Implement Design	9 Show Failing Constraints	Average Fanout of Non-O	ock Nets	1.77	1	
E Map	Show Warrings					
Original Place & Route     Original Place & Route     Original Place & Route				Performance Sur	imary	

As shown in the figure 6 the number of occupied in the reconfigurable IPU based FIR filter is 110 out of 2489. The number of LUT's is 164

#### 4.1.3 Delay

Sert + □ € × Pengn E ×	000	Static Tening     Erron and Warrings     Parter Messages     Synthesis Messages	•	Clock Information: Clock Signal   Clock Buffer(FF name)		Load	7
All         Hierarchy           Image: State stat	の目標	A Translation Messages     A Map Messages     Prace and Route Messages     Timing Messaget     Timing Messaget     A Implementation Messages     December Lenote		cik i BUFGP Asynchronous Control Signals Information:	t	117	-
Cl 22	100	Synthesis Report     Transtation Report     Map Report     Map Report     Pour-PAR Static Timing Report     Pour-PAR Static Timing Report     Pour-Paport     State Report		No asynchronous control signals found in this design Timing Summary, Speed Grade: -4 Minimum period: 8.161ns (Maximum Frequency: 122.534001)			
C Processe yyy Vew RTL Schematic View Technology Schem. C Check Syntax C Generate Post-Synthesis C C Internet Design C Transite C Transite C C C C C C C C C C C C C C C C C C C		Secondary Reports Advanced HDs, Synthesis Report Low Level Synthesis Partition Report Devoet Utilization Summary Partition Resource Summary Cost Minumeton Annuthronous Control Siznata.	3 6	Minimum imput arrival time before clock: 3.796ns Maximum output required time after clock: 11.296ns Maximum combinational path delay: 13.332hs Timing Detail: All values displayed in nanoseconds (ns)			

As shown in the figure 7 the delay produced by using innerproduct unit is 21.07ns.

#### 4.2 Results of MCM based FIR filter 4.2 1 Simulation output

7.4.1	Simulation	Julpul	
se wave	- default		
	Messages		
e-4	/abcde/h6	00000101	росса, урасса, тосаан, тоссан, уроссан, урасан, тоссан, тоссан, тоссан, урассан, тоссан, тоссан, тоссан, уроссан, уроссан, уроссан, уроссан, уроссан, уроссан, ур
B-4	/abcde/h7	00000001	(00000001) (00000010)(000001) (00000010 (00000000) (000000001) (000000010 (00000001) (000000110 (00000011) (00000010)
<b>D-</b>	/abcde/h8	00000011	0000 000000100 00000011 00000100 00000011 000000
•	/abcde/h9	00000001	X X00000001 X
D-4	/abode/h10	00000010	
64	/abcde/h11	00000101	
6-4	/abcde/h12	00000011	3001p3pa1pa1pa1p1p1p
B-4	/abcde/h13	00000001	200 japo j
<b>B-</b>	/abcde/h14	00000010	2000 (20000 (200000 (200000 (200000.10 ) (20000 (20000 (20000 (20000 (200000 (20000
6-4	/abcde/h15	00000100	300000 101 300000 100 100000 101 300000 100 30000 101 300000 101 300000 101 300000 101 300000 101 300000 100 X
<b>B-</b>	/abcde/y0	0000000000100011	
<b>B-</b>	/abcde/y1	000000000000000101	
<b>B-</b>	/abcde/y2	00000000000101010	<u>X X XX X X X X X X X00XX X00XX X X X </u>
	/abcde/y3	0000000000100110	
	/abcde/uut/dk	541	
	/abcde/uut/rst	StD	
•	/abcde/uut/x0	00000001	1 Ya1 1 Yaa1 Yaa1 Ya1 Ya1 Ya1 X 1 Ya1 X Ya1 Ya1 Ya1 Ya1 Ya1 Ya1 Ya1 Ya
<b>D-</b>	/abcde/uut/x1	00000100	po(acb)(aco)(ac
B-4	/abcde/uut/x2	00000101	Хреофа Хрова Хрова Хрова Хрова Хрова Хрова Храва Храва Хрова
<b>B-</b>	/abcde/uut/v3	00000011	20000
<b>B-</b>	/abcde/uut/h0	00000100	Трасфа Хасааа Хасафа Хасааф Хасааа Хасаааф Хасааа
<b>B-</b>	/abcde/uut/h1	00000010	00000 (0000000 1 (000) 00 / 000000 1 / 000000 10 / 000000 10 / 0000000 1 / 0000000 10 / 0000000 1 / 000000 10 / 0000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 0000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 000000 10 / 0000000 10 / 000000 10 / 0000000 10 / 0000000 10 / 0000000 10 / 0000000 10 / 0000000 10 / 0000000 10 / 0000000 10 / 00000000
<b>B-</b>	/abcde/uut/h2	00000011	<u>xao</u>
<b>B-</b>	/abcde/uut/h3	00000101	5001010010110110110110110010101
	Febrate Aret But	020000.10	Van fana Van Van Jana Van Van Van Van Van Van Van Van Van
1000	Novi	187790 ps	187000 ps 187200 ps 187400 pa 187600 ps 187800 ps
	Cursor 1	187030 ps	197030 ps

### Fig 8 Modelsim output of MCM based FIR filter

By following the procedure to run the Modelsim the simulation results are obtained. Figure 8 represents simulation results of reconfigurable FIR filter

		108 Properties	Internet and a second	N-3428E-111420		. THERE	INCOM					
📗 Vesc 🖲 🎇 Inglementation 🔿 🚟 Se	nuletion	Module Level Utilization	Product Version:	ISE 12.1		•Warnings:	233 Wat					
[a] Herarchy		Timing Constraints	Design Goak	Balanced		+Routing Results:	Al Sona					
22 - E 1000	C	Clock Report	Design Strategy:	Silox Default Juriocke	a	Timing Constraints	Al Const					
	-	1 Static Timing	Environment:	System Settings		•Final Timing Score	0 (Dimit)					
al	E M	Parser Messages	-	Device Utilization Summary								
		Map Messages	Logic Utilization		Used	Available	Utilization					
		Place and Route Messages	Number of Sice Fip Flops		40	4,896	1%					
		Timing Messages	Number of 4 input LUTs		101	4,895	25					
		All Implementation Message	Number of occupied Stores		71	2,441	25					
Ro Phoceases Running		Detailed Reports	Number of Sices contain	ning only related logic	21	71	300%					
C Processes yyy	~	Synthesis Report	+ Number of Sices cantan	ring unrelated logic	9	71	175					
C View Technology Scher	5. C	Design Properties	Total Number of 4 input ().	JTB.	105	4,896	25					
Check Syntax		C Enable Message Filtering	Number used as logic		101	C						
- B COO Implement Design		Show Clock Report	flumber used as a route	e-thru	4							
		Show Failing Constraints	Fiumber of bonded (CBs		58	172	56%					
# 2303 Place & Route		Show Warnings	108 Flip Flogat		1							
Generate Programming File		and the state of the state	Yiumber of BUPGMUKs		1	24	4%					
# W Configure Target Device			Average Pagest of Net-C	ind liets.	2.70							

Fig 9 Area of MCM based FIR filter

As shown in the figure 9 the number of occupied slices in MCM based FIR filter is 71 out of 2488. The number of LUT's is 101 out of 4896

	0 0 X 0 6 X 3 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1		A A A A A A A A A A A A A A A A A A A		P ≥ 2 ≤ 2 ≤ 2					
四 四 湯 前 三	n	0000	Clock Report Static Timing Errors and Warnings Parser Messages Synthesis Messages Translation Messages Map Messages	0	OTE: THESE TINING HUMBERS FOS ACCURATE TIMING GENERATED AFTER FLACT (lock Information:	ARE ONLY INFORMATI E-and-ROU	A SYNTHESIS ESTIMATE. ON PLEASE REFER TO THE T TE.	-+	CE REP	08T
in.		12	<ul> <li>Place and Route Messages</li> <li>Timing Messages</li> </ul>	c	ločk Bignal		Clock buffer(FF name)	-	Load	4
ø		64	Bitgen Messages	e	1k		1 BURGP	1	44	1
			Detailed Reports     Synthesis Report     Translation Report     Map Report	A	aynchronoua Control Signa	1s Inform	ationi			
	(2) No Processes Running		<ul> <li>Place and Route Report</li> <li>Post-PAR Static Timing Report</li> </ul>	11	o salucuronons control at	duers ton	nd in chis besign			
ic,	Processes: yyy		Prover Report	¥1 T	iming Summary:					
	Besign Ukilities Uker Constraints Synthesize - XST Wew RTL Schematic Wew Technology Sche Check Syntas Generate Post-Synthesi		Advanced HDI, Synthesis Report Low Level Synthesis Partition Report E-Fixed Report Device Dilutation Summary Partition Resource Summary Clock Information		<ul> <li>Speed Grade: -4</li> <li>Minimum period: 4.198ms (Maximum Frequency: 235.194001s) Minimum input arrival time before clock: 5.078ms Maximum cutput required time after clock: 11.064ms Maximum combinational path delay: 8.409ms</li> </ul>					

As shown in the figure 10 the delay produced by using MCM unit is 16.14ns.

# 4.3 Comparison of results

4.3.1Using table

The result of various multiplier used are described in the following table

Reconfigurable FIR filter	Are	_	
	LUT's	Slices	Delay
IPU based			
	164	110	21.07 ns
MCM based			
	101	43	16.14 ns

TABLE 4.1 Results of various reconfigurable implementation

From the above table it is clear that the Area ,Delay and power are reduced respectively for each multiplier . Reconfigurable FIR filter with IPU consumed more Area and Delay . The multiple constant multiplier have reduced area and power.

### 4.3.2 Using graph

- X-Axis denotes various multipliers.
- Y-Axis denotes the values.



Fig 11 Area comparison

Figure 11 represents the area of various implementation of FIR using a inner product unit and MCM. The number of slices used and LUT's in IPU based FIR filter used is 164 and 110 respectively. The number of slices used and LUT's in MCM based FIR filter used is 101 and 43 respectively.



Fig 14 comparison graph on Delay

Figure 12 represents the delay of various implementation of FIR using a inner product unit and MCM. The delay IPU based and MCM based FIR filter is 21.07ns and 16.14ns respectively.

#### **5 CONCLUSION**

An area efficient and high speed multiple constant multiplier is proposed. Multiple constant multiplier consumes slices ,of delay when compared to the inner product unit. This multiple constant multiplier is incorporated in the RFIR. This proposed reconfigurable FIR filter with multiple constant multiplier is compared with the reconfigurable FIR filter with inner product unit. The results show that the proposed reconfigurable FIR filter with multiple constant multiplier is filter with multiple constant multiplier is compared with the reconfigurable FIR filter with inner product unit. The results show that the proposed reconfigurable FIR filter with multiple constant multiplier offers better performance than the conventional reconfigurable FIR filter with a inner product unit. Thus this proposed FIR filter con be used in the digital communication applications like the software defined radio.

# **6 REFERENCE**

- 1. Daniel Llamocca, Marios Pattichis and G. Alonzovera (2010) 'Partial reconfigurable FIR filtering system using Distributed arthematic'International Journal of Reconfigurable computing volume 2010, Article ID357978,14 pages.
- 2. J.Park, W.Jeong, H.Mahmoodi-Merimand, Y.Wang, H.Choo and K.Roy(2004) 'Computation Sharing programmable FIR filter for Low –Power and High Performance applications'IEEE j.Solid State Circuits ,Volume 39, no.2, pp.348-357,
- 3. Kuan-hung Chen and Tzi-Dar Chiueh (2006) 'A Low power digital based reconfigurable FIR filter'IEEE Transaction circuit system II,Exp.Breifs,vol.53,no.8,pp.617-621.
- 4. Mahesh.R. and Vinod.A.P. (February 2010) 'New reconfigurable architectures for implementing FIR filter with Low complexity'IEEE Transaction circuit system II,Breifs,vol 29,no.2,pp.275-288.
- Pramod Kumar Meher and Yu Pan (2014) 'Bit-Level Optimization of Adder Trees For Multiple Constant Multiplier for Efficient FIR filter Implementation'IEEE transaction on circuit system I,vol.61,no.2,pp.455-462.