A low power transistor level FIR filter implementation using CMOS 45 nm technology

M. Balaji, N. Padmaja

A low power transistor level FIR filter implementation using CMOS 45 nm technology

M. Balaji*

Research Scholar,
Department of ECE,
Jawaharlal Nehru Technological University Ananthapur,
Ananthapuramu, 515002, India
Email: balajichaitra3@gmail.com
*Corresponding author

N. Padmaja

Department of ECE,
Sree Vidyanikethan Engineering College,
Tirupati, 517101, Andhra Pradesh, India
Email: padmaja.n@vidyanikethan.edu

Abstract: Digital finite impulse response (FIR) filters are widely used in signal processing fields, due to their stability and linear-phase property. In this paper, the low area FIR filter is designed by proposing the optimal array multiplier (OAM) and optimal full adder (OFA) to minimise the resources. The number of adders used in the OAM is decreased by replacing all the half and full adders of conventional multiplier with the OFA. The buffer circuit is designed in the OFA for avoiding the noise, glitches and threshold issue. The proposed OAM-OFA-FIR architecture is designed using Cadence virtuoso software with 45 nm technology and it is analysed in terms of area, power and delay. The existing methods used to evaluate the OAM-OFA-FIR architecture are FIR filter design using Radix-2 algorithm and look-up-table carry select adder (LCSLA), and Vedic design (VD) and carry look-ahead adder (CLA). The area of the OAM-OFA-FIR architecture is 1755 um^2, which is less when compared to the existing methods.

Keywords: area; delay; FIR; finite impulse response filter; OAM; optimal array multiplier; OFA; optimal full adder; power.

Biographical notes: M. Balaji completed BTech in Electronics and Communication Engineering and MTech in Embedded Systems from JNT University, Hyderabad, India. He is currently pursuing his Doctor of Philosophy from JNT University Anantapur, with specialisation in VLSI Signal Processing. He has authored or coauthored 14 national and international research publications. His research interests include advance filter designs using novel VLSI design architectures.

N. Padmaja completed her BE from Mumbai University and MTech and PhD from S V University in the area of Atmospheric Radar Signal Processing. Presently, working as a Professor, Sree Vidyanikethan Engineering College (Autonomous), Tirupati. India. For her credit she is having 50 technical papers in various reputable international peer reviewed journals and conferences. Her areas of interests include VLSI signal processing, image processing and communication systems. She is a Life Member of IETE, IAENG, ISCA, ISTE and IACSIT.

1   Introduction

Digital filter is considered as one of the important fields in digital signal processing (DSP) where filters are used to remove the unwanted parts such as random noise or to obtain the essential information from the data i.e., information in the significant frequency range [1, 2]. In real time, DSP is mandatory in various applications such as military applications, speech signal processing, video surveillance, software defined radios, processing the satellite signals, medical image processing and so on [3]. These digital filters are further categorised into two categories such as infinite impulse response (IIR) and FIR filters [4]. FIR filter is always retained in the linear stage using the symmetric coefficient. Due to this feature, the FIR filter is considered the best choice for phase-sensitive applications which includes mastering, data communications, seismology and crossover filters [5, 6]. However, the IIR filters are highly sensitive to the coefficient quantisation that is based on the instability of the design. Meanwhile, these structures are difficult to design and they suffered from the limit cycles [7].

The multipliers are the basic components of the FIR filters, however, the filters with huge filter lengths required a huge amount of multipliers. This resulted in high power and area consumption as well as the complexity of the multiplier is higher than the complexity of adder [8, 9, 10]. Specifically, the multiplication is accomplished between one particular variable (i.e., input) and many constants (i.e., coefficients) which is referred to as Multiple Constant Multiplication (MCM). Subsequently, the MCM in the multiplier is used to reduce the number of adders, shifters and multipliers while designing the FIR filter [11, 12]. Since, the FIR filter is implemented by using the delay, adder and multiplier elements. In an FIR filter, the computations are minimised using fast Fourier transform algorithms, however, this is not possible in the IIR filter [13]. The design of low area FIR filter with an appropriate speed performance is essential because the increase in area results in higher hardware cost [14]. Generally, the applications of DSP are designed using the application specific integrated circuits (ASICs) or general-purpose DSP devices, because of their energy efficiency and higher processing speed [15]. Examples of designing technologies are complementary metal oxide semiconductor (CMOS) [16], organic field effect transistors [17], fin field-effect transistor (FinFET) [18, 19], etc. This research uses the CMOS technology to develop the filter design. Since, CMOS is extensively utilised for semiconductor devices, due to its simple operation, less fabrication stages, less area, less current and so on [16].

The important contributions of this research paper are given as follows:

- The combination of OAM and OFA is used to improve the performances of the FIR filter in terms of area, power and delay. The designed OAM uses only fewer amounts of full adders (i.e., three 4-bit OFA) for the addition of partial products. But, the conventional array multiplier requires 4 half adders and 4(4-2) full adders. Therefore, the OAM used in the FIR filter is used to minimise the area of the overall proposed architecture.

- Unlike conventional full adders, the OFA designed in the FIR filter has a buffer circuit to avoid the noises and glitches that occurred during the filtering process. This helps to obtain the signal output without any distortions.

- Further, the proposed architecture is designed and simulated using Cadence virtuoso software with 45 nm technology where the key parameters analysed in this research are area, power, and delay.

The overall organisation of the paper is given as follows: The existing works related to the FIR filters are described in Section 2. Section 3 provides the problem statement of this research. A detailed explanation of the proposed architecture is provided in Section 4. Section 5 provides the results and discussion of the proposed architecture. Finally, the conclusion is made in Section 6.

2   Related work

Sundar et al. [20] developed the enhanced DA architecture (EDA) based adaptive FIR filter. This EDA based FIR filter was designed using the compressor adder and multiplexers. Here, the 2 : 1 multiplexer and 4 : 2 compressors were used instead of ROM and accumulator respectively. The performance of the FIR filter was not varied, because of the truncation used in the EDA based filter. However, the EDA based FIR filter was required an extra multiplexer during the filtering process.

Jyothi et al. [21] presented the memory less distributed arithmetic (MLDA) for the design of FIR filter with residual number system. Since the residual number system was used to design the high speed DSP systems. The input and coefficients of MLDA were in the residual number form and the output generated by the MLDA was converted into binary value using the Chinese remainder theorem. Moreover, the area of the FIR filter was reduced using the compressor. However, this MLDA based FIR filter required more multiplexers at the filtering process.

Satish Reddy and Suresh [22] developed the reconfigurable FIR filter using the Radix-2 algorithm and look-up-table carry select adder (LCSLA). Here, the rewritable RAM-based LUT was utilised instead of the ROM-based LUT. The LCSLA was used to accomplish the addition operation and radix-2 algorithm was used to perform the multiplication in the FIR filter design. Moreover, the modified Radix 2 algorithm was used to minimise the partial products during multiplication. However, the area of the Radix 2-LCSLA-FIR filter was increased, because of the frequent usage of the Radix 2 during the filtering process. Additionally, this Radix 2 consumes high time during the filtering process.

Sumalatha et al. [23] presented the FIR filter based on the Vedic design (VD) and carry look-ahead adder (CLA). The multiplication operation was performed by using the

VD and the addition operation was done by using CLA. The developed FIR filter using VD and CLA was used for the noise removal over the ECG signal. Moreover, the combination of VD and CLA was used to enhance the speed of the filtering process. But, the vedic multiplier used in the multiplication was created a high latency in the FIR filter.

Odugu [24] adopted the Circular Symmetry for FIR filter (CS-FIR) and it was designed by using the approximate compressors and multipliers. Here, an exact $4:2$ compressor was altered as $4:2$ compressor by avoiding few inputs. Therefore, the amount of transistors in the CS-FIR was minimised based on the CMOS mixed logic gates, dual value logic and transmission gate logic. However, a huge amount of computations was created high delay during the filtering process.

Vijetha and Naik [25] developed the block FIR filters using distributed arithmetic (DA) structure for the decision feedback equaliser (DFE). The designed DA-based FIR filter was inbuilt in the feedforward of DFE and feedback filter of concurrent DFE. Accordingly, the feed-forward filters were used to remove the intersymbol interference errors and feedback filters were used to remove the remaining noises. Hence, the throughput rate was increased using this block FIR filter. However, the DA-based FIR filter was utilised a high amount of hardware to acquire high throughput.

NagaJyothi and Sridevi [26] presented the reconfigurable offset-binary code (OBC) DA-based FIR filter with a shared LUT updating approach at decimation filter. The developed DA using the shared LUT approach was used to decrease the huge memory requirement of higher order filters. The splitting of LUT was done by the shared LUT approach where the coefficients were separated into small length vectors that decreased the LUT size. However, the decimation filter was used three different filters such as corrector filter, cascaded integrated comb (CIC) filter and half-band filter for converting the high-frequency signal into a low-frequency signal which leads to an increase in the overall area.

Padmavathy et al. [27] developed the fast FIR filter to eliminate the noise in the Electro Cardiogram (ECG). The 8-bit multiplier was designed based on Vedic Mathematics with the combination of UrdhvaTiryagbhyam sutra. The designed 8-bit multiplier was used to generate the partial products. Subsequently, the ripple carry adder (RCA) was used in the multiplier to accomplish the addition over the partial products. This fast FIR filter was used to reduce the delay and area during the implementation. However, the developed fast FIR filter was failed to retrieve the data precisely, when the input was affected by the noise.

Rammohan et al. [28] presented the low complexity architecture of the FIR filter for digital hearing aid applications. Moreover, approximately $4:2$ compressor adders were developed for the memoryless DA-based FIR filter structure. Since, the compressors were used to minimise the number of additional operations in the desired circuit. The designing of memoryless DA based on the compressor adders was used to minimise the power consumption for the applications of hearing aid. This filter design also required three more filters such as CIC, half band and corrector to achieve the appropriate signal.

3   Problem statement

The problems found from the existing researches along with the solutions provided by the proposed architecture are stated in this section.

The DA based FIR filter [25] consumes a high amount of hardware resources for enhancing the data rate. For an effective FIR filter, the hardware resources should be less, because a high amount of hardware resources causes higher delay and power consumption. The data acquired from the fast FIR filter [27] is degraded when the given input is affected by the noise. Additionally, the frequent usage of Radix 2 leads to an increase in the overall area of the Radix 2-LCSLA-FIR filter. The developed Radix 2 [22] and vedic multiplier [23] creates a delay during the computation process.

Solution:

The area of the proposed architecture is minimised by using the OFA in the addition process and OAM in the multiplication process. The area of the FIR filter is minimised because the OAM uses fewer amounts adders compared to the conventional array multiplier. According to that, the delay and power of the proposed architecture are decreased by minimising the area. On the other hand, the buffer circuit used in the OFA helps to avoid noises during the filtering process.

4   Proposed architecture

In the proposed architecture, the combination of OAM and OFA is used for enhancing the 4-tap FIR filter operation. In that, the 4×4 OAM is designed with fewer adders for accomplishing the multiplication operation between the input and coefficients. Further, the 4-bit OFA is used to add the partial products inside the OAM and the 8-bit OFA is used to add the multiplied value acquired from the different taps. The buffer circuit used in the OFA helps to avoid the threshold problem, noise and glitches that occurred during the filtering process. The schematic representation of the proposed architecture is given in Figure 1.

Figure 1   Schematic representation of the proposed architecture (see online version for colours)
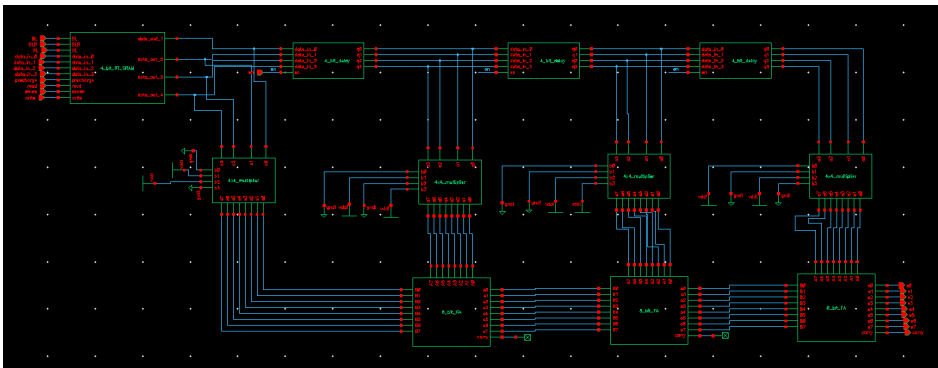
The steps processed in the proposed architecture are given as follows:

Step 1: The supply voltage of overall OAM-OFA-FIR architecture is $0.9\,V$ and the coefficients given to the filter design are $\{6,5,3,10\}$. The 4-bit input data such as $data\_in\_0, data\_in\_1, data\_in\_2, data\_in\_3$ are given to the 4-bit SRAM cell along with the Bit Line (BL), Bit Line Bar (BLB), Word Line (WL), read, sense and

write signals. This 4-bit SRAM cell delivers the given input (*data _ in*) to the 4-bit delay element and 4×4 OAM circuit.

Step 2: 4-bit delay element is used to store the *data _ in* and it is used for the next tap. The stored *data _ in* is given as input to the next tap 4-bit delay element and 4×4 OAM circuit.

Step 3: The input from the 4-bit SRAM cell are $a0 = q0$, $a1 = q1, a2 = q2$ and $a3 = q3$. The 4×4 OAM also receives the 4-bit coefficients as one more input which is read from the register. The coefficients are $b0 = gnd1, b1 = vdd1, b2 = vdd1$ and $b3 = gnd1$, therefore the given 4-bit coefficients are $\{0,1,1,0\}$.

Step 4: Further, the multiplied values from the first tap and second tap are added by using the 8-bit OFA. Similarly, the operation is carried out for all taps and it provides the 8-bit output $s0 - s7$ along with the carry output.

The explanation about the individual modules used in the proposed architecture along with its schematic architecture is provided in the following sections.

4.1   4-bit SRAM cell

In this proposed architecture, different inputs are given to the SRAM such as BL, BLB, WL, input data $(data \_ in \_ 0, data \_ in \_ 1, data \_ in \_ 2, data \_ in \_ 3)$, read, sense and write. Since the input data is stored in the SRAM and it is further processed in the filtering operation. The SRAM cells are developed to execute the read/ write operation or to hold one value. Each function enables the different parts of the SRAM architecture. The 4-bit SRAM cell used in the proposed architecture is the combination four 1-bit SRAM cells and 1-bit SRAM cell is designed using the 6T SRAM. The architecture of the 1-bit SRAM and 6T SRAM cells are shown in Figures 2 and 3 respectively.

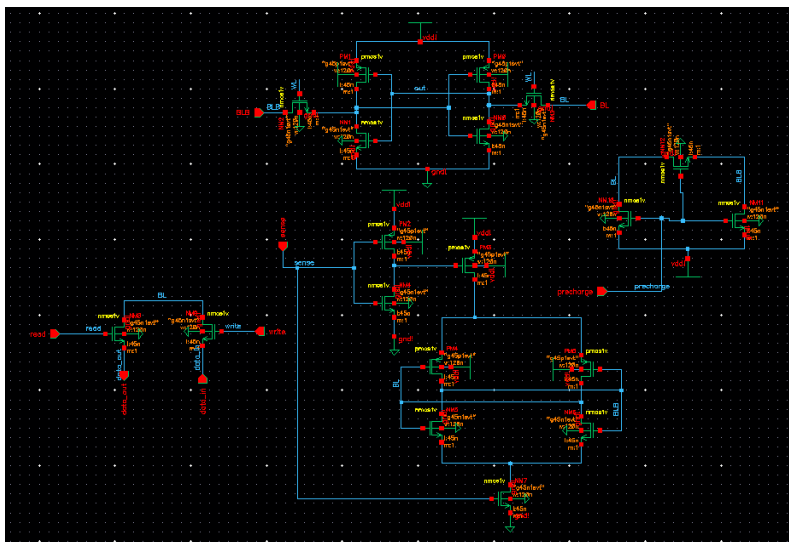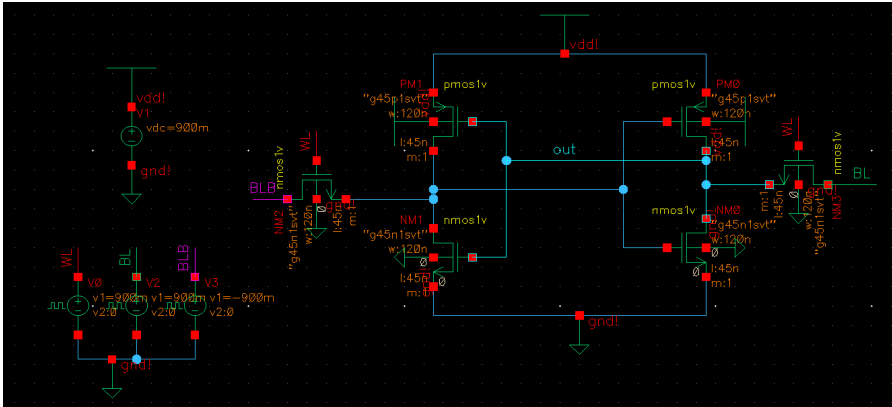**Figure 2**   Architecture of 1-bit SRAM cell (see online version for colours)

Figure 3   The architecture of 6T SRAM circuit (see online version for colours)

Generally, the SRAM cells have two nodes such as Q for storing a bit and complementary Qb node. Additionally, the 1-bit SRAM cell is used, where a n-bit memory requires an extra circuit for the precise operation of read/ write operation. Additionally, the 1-bit SRAM uses the row decoder, sense amplifier and the write circuit. The nodes Q and Qb of 6T cells are used only when the write and read operation. The design of 1-bit SRAM has required totally 19 transistors and four 1-bit SRAM is used to create the 4-bit SRAM. The pass transistors are controlled by enabling the Word Line (WL) using the row decoder. Next, the pass transistors are used to allow access to the internal loop inverters for read/ write data. The write circuit charges one BL and discharges the other for setting the required values in the BL and complementary Bit Line B (BLB) to save a bit value. Next, the WL is activated by the row decoder and the required value is written in the nodes Q and Qb. The bitlines BL and BLB are required to be pre-charged for reading a bit. If the WL is enabled, the node which contains the '0' is discharging the connected BL and creates the voltage variation between the BLs. Accordingly, the voltage variation between the BLs is sensed using the sense amplifier that is used to complete the read operation as well as the designed sense amplifier delivers the desired cell value as output. This 4-bit SRAM cell delivers 4-bit output such as $data\_out\_1, data\_out\_2, data\_out\_3$ and $data\_out\_4$ and these 4-bit values are delivered to the 4-bit delay element and 4×4 OAM.

4.2   4 bit delay

The 4-bit delay element is used to store the inputs acquired from the 4-bit SRAM cell. Since the 1-bit delay element is initially designed followed by four 1-bit delay element are combined using cell view for creating the 4-bit delay element. The design of 1-bit delay element comprises five NAND gates as shown in Figure 4. Figure 5 shows the NAND gate used in the delay element.

Figure 4 Design of delay element (see online version for colours)
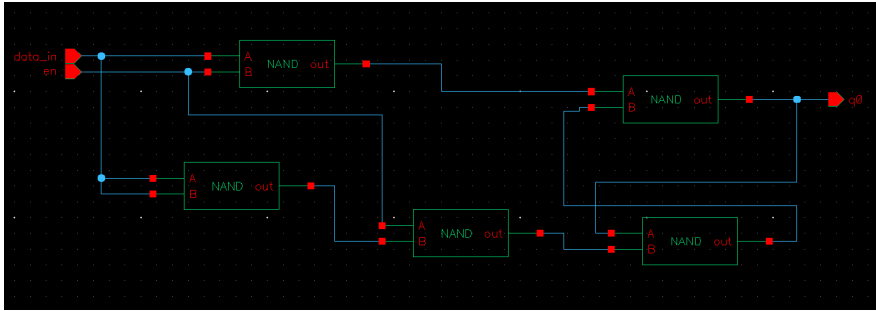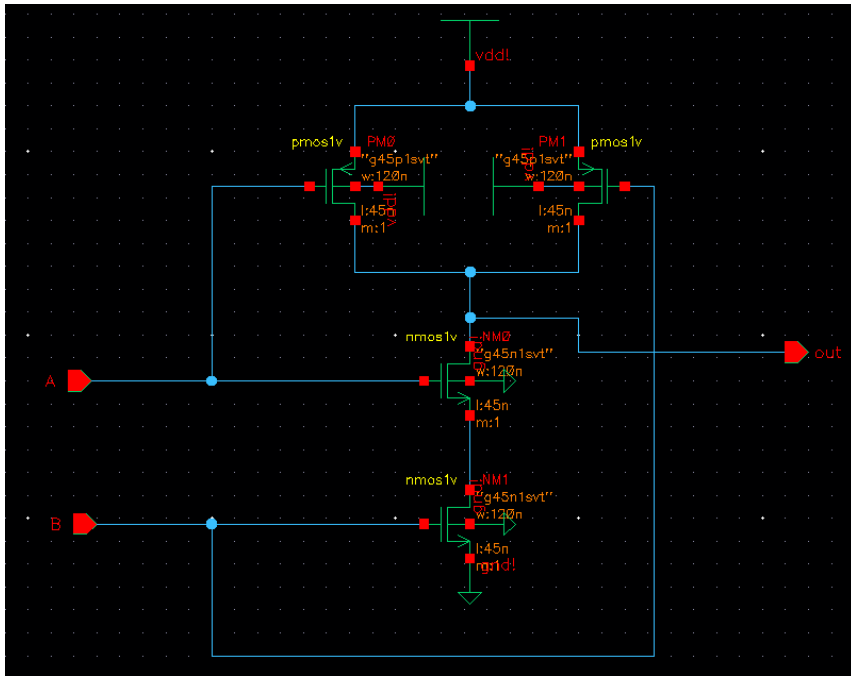

Figure 5 NAND gate (see online version for colours)

The NAND gate (i.e., NOT-AND) is generally a logic gate that generates the false output, only when all the inputs are true. Since the output of the NAND is inverse of the AND gate. More specifically, the low output (0) is obtained, when all the gate inputs are high (1); If any of the gate inputs is low (0), the NAND gate resulted in high (1) output. The NAND gate is expressed in equation (1) and its trust table is shown in Table 1.

$$Out = \overline{A.B} \tag{1}$$

The implementation of NAND gate requires four transistors such as 2 PMOS and 2 NMOS transistors. The overall power supply of $0.9V$ is used to provide the *Vdd* supply of all sub modules of the OAM-OFA-FIR architecture.

Table 1   Truth table of NAND gate

A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0

In LP mode, the substrate of the PMO & PM1 is connected to the source terminal and the body of the NM0 & NM1 is connected to the ground as shown in Figure 1(a). The *Vdd* given to the PMOS transistors are considered as a source. The PMOS provides the output as *Vdd* , when the gate terminal receives the supply voltage of zero. Consider, the input *A* is zero, the *Vdd* is conducted to the drain using the body. Specifically, the PM0 and NM0 transistors are turned ON and OFF, when the input *A* is zero. For example, zero is given as input to both the *A* and *B* inputs to the LP NAND gate. Accordingly, the PM0 & PM1 are turned ON and NM0 & NM1 are turned OFF. Then, the outputs of the PM0 and PM1 closed and the $Vdd = 0.9$ V is conducted to the NM0 as well as there is no conductivity in NM0 & NM1, because the input value is zero. Further, the outputs are combined and it delivers the high output i.e., 1. Similarly, this LP NAND gate provides the same output for the remaining three input pairs mentioned in the trust table. Therefore, the delay element provides the four different outputs such as $q0, q1, q2$ and $q3$ for the input values $data\_in\_0, data\_in\_1, data\_in\_2$ and $data\_in\_3$. The outputs $q0, q1, q2$ and $q3$ are given as input to the successive 4-bit delay element (for next tap) and also to the 4×4 OAM.

4.3   Multiplication using OAM

The optimal array multiplier (OAM) used in the proposed architecture is used to multiply the output from the 4-bit SRAM $(q0, q1, q2, q3)$ and coefficients read from the register $(b0, b1, b2, b3)$. Figure 6 shows the schematic representation of the OAM. The main benefit of the array multiplier is easy to design the architecture as 1-bit adders are connected in the array. In this 4×4 OAM, the AND gate calculates the partial products of a_i and b_i . Subsequently, the output from the AND gate are added using the 4-bit OFA. However, the conventional array multiplier uses the 4 half adders and 4(4-2) full adders for accomplishing the addition operation, thus results in higher area consumption. The designed OAM is required only three 4-bit OFA to perform the addition operation which further decreases the area of the proposed architectures. Therefore the major components of the OAM are AND gates and 4-bit OFA are explained in the following section.

Figure 6 The architecture of 4×4 optimal array multiplier (see online version for colours)
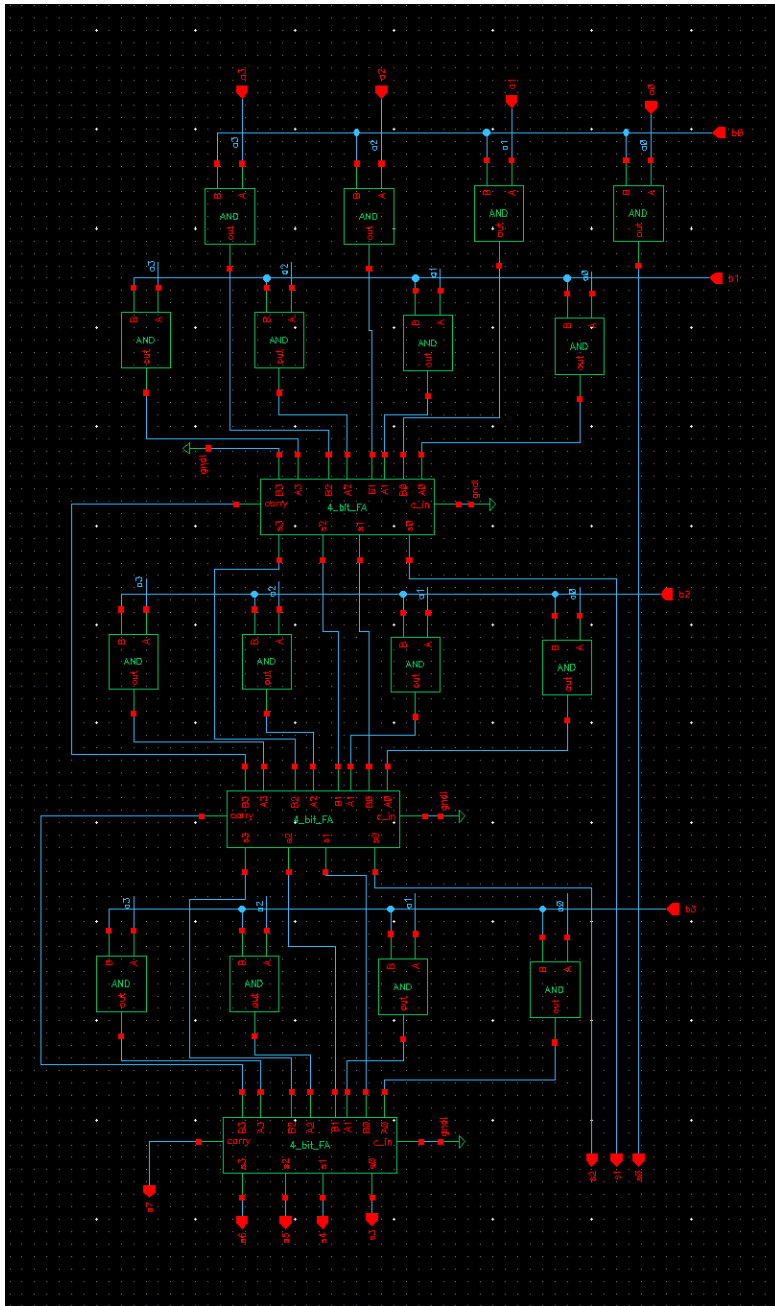
4.3.1 AND gate

The AND gate of OAM is used to compute partial products between the given inputs. AND gate is generally one of the basic logic gate which implements the logical conjunction. This AND gate provides the output high when all the inputs are high.

Otherwise, the AND gate returns the output as low. Equation (7) and Table 2 show the expression and truth table for the AND gate respectively.
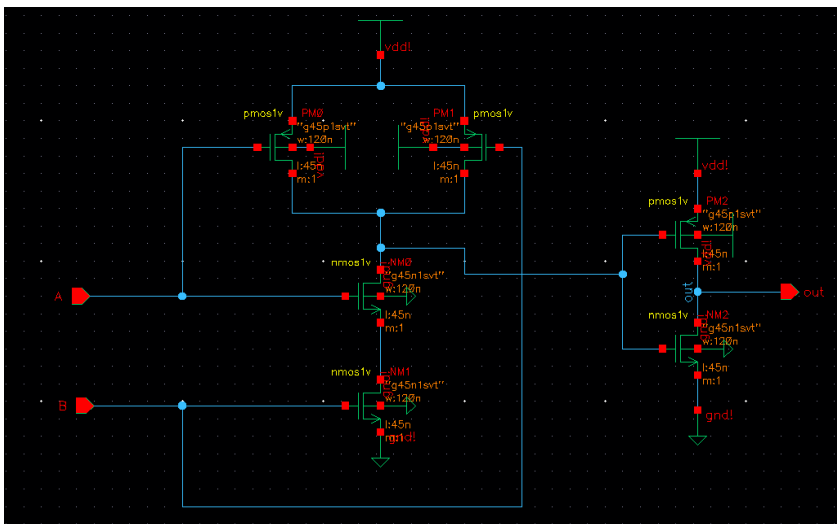
$$Out = A.B \tag{7}$$

The design of AND gate using CMOS requires three PMOS transistors and three NMOS transistors. Figure 7 shows the design of AND gate for LP mode.

Table 2   Truth table of AND gate

A	B	Out
0	0	0
0	1	0
1	0	0
1	1	1

Figure 7   Architecture of AND gate (see online version for colours)

The design of AND gate is generally the NAND gate followed by the inverter. Consider the input given to the terminal A and B are 0. Similar to the NAND gate, the first part of the circuit provides high output. Then the output high is given to the transistors PM2 (PMOS) and NM2 (NMOS). Accordingly, the transistor PM2 and NM2 are turned OFF and ON respectively. So there is no conductivity between the transistors, which provides the output low as like the truth table.

4.3.2   *Optimal full adder*

A 4-bit OFA is used in the OAM circuit for summing the partial products acquired from the AND gates. Initially, the 1-bit 10T full adder circuit is developed followed by 4-bit OFA is designed using the cell view by using the four 1-bit full adders. Figure 8 shows the architecture of a 1-bit full adder where it comprises of 9 PMOS transistors and 9 NMOS transistors. The truth table of 1-bit OFA is shown in Table 3. Unlike the

conventional 1-bit full adder, the 1-bit OFA has an additional circuit namely a buffer circuit at the end of the adder. This buffer circuit is designed using the 4 PMOS and 4 NMOS transistors which are used to avoid the threshold problem, unwanted noises and glitches occurred during the filtering process.

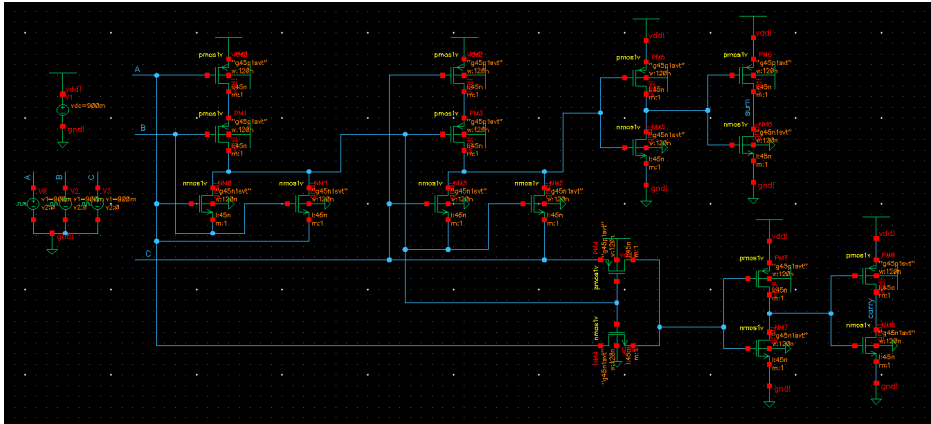**Figure 8** Architecture of 1-bit optimal full adder (see online version for colours)

Table 3 The truth table of 1-bit OFA

A	B	C	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1

4.4 Addition using OFA

The 8-bit OFA is used to add the multiplied values from first tap $B0 - B7$ and multiplied values from second tap $A0 - A7$, where the 8-bit OFA is generated by combining eight 1-bit full adders using cell view. The designed 8-bit OFA generates the 8-bit output value i.e., $s0 - s7$ and it is processed in the next tap. Likewise, all the 4-taps of the proposed architecture are done and it provides the final output.

5 Results and discussion

This section provides the results and discussion of the proposed architecture along with its waveform analysis. The proposed architecture has been designed using the Cadence virtuoso software with 45 nm technology and the system is operated with 4 GB RAM and a 500 GB hard disk. For a 4-tap filter, the 4-bit of input and 4-bit filter coefficient are

given to the 8-bit value at the FIR filter output. On the other hand, the analogue design environmental (ADE) window is used to analyse the power and delay of the proposed architecture.

5.1   Performance analysis

Area, power and delay are considered as the key parameters used for analysing the proposed architecture. In this section, the performances are analysed for 6T SRAM cells, 1-bit OFA and the overall proposed FIR filter. The process of computing the area, power and delay is defined as follows:

a   *Area calculation*

Generally, the area is considered as one of the essential parameters during the implementation. Because, recent researches always pursue an effective circuit with less area consumption. The area of the circuit mainly depends on the transistor area where the area calculation for the single transistor is mentioned in equation (3).

$$Single\,transistor\,Area\left(nm^2\right) = Width\left(nm\right) \times Length\left(nm\right) \tag{4}$$

Where, the length and width of the single transistor are 45 nm and 120 nm respectively. Therefore, the area of the single transistor used in the proposed architecture is 5400 nm i.e., 5.4 um^2.

b   *Power calculation and delay calculation*

The power of the designed architectures is analysed using the ADE window present in the Cadence software. The power waveform of the circuit is obtained based on the following instructions i.e., $ADE\,window \rightarrow Tools \rightarrow results\,browser \rightarrow power$. Meanwhile, the delay is calculated based on the rising edge difference between the input and output.

5.1.1   *Analysis of area, power and delay for 6T SRAM cell*

The input conditions for read operation of the 6T SRAM cell design are $WL = 0.9$ V, $BL = 0.9$ V, $BLB = -0.9$ V and $out = 0.9$ V. The design of 6T SRAM cell design requires 2 PMOS and 4 NMOS transistors, hence totally it requires 6 transistors as shown in Figure 3. Consequently, the area of the 6T SRAM cell is 6×120 nm $\times 45$ nm $= 5400$ nm $= 32.4$ um^2.

Figures 9–12 show the read waveform, read delay, read power and average read power for the 6T SRAM cell respectively. The 6T SRAM provides the output (*out*) similar to the BL, when the WL is enabled during the read operation. For example, the WL is enabled between the 6–12 ns, therefore the output of 6T SRAM is (0,1) as similar to the BL. The output acquired from the 6T SRAM cell is used to calculate the delay and average power as shown in Figures 10 and 11. Therefore, the delay and average power of the 6T SRAM cell are 0.06 ns and 199 uW respectively.

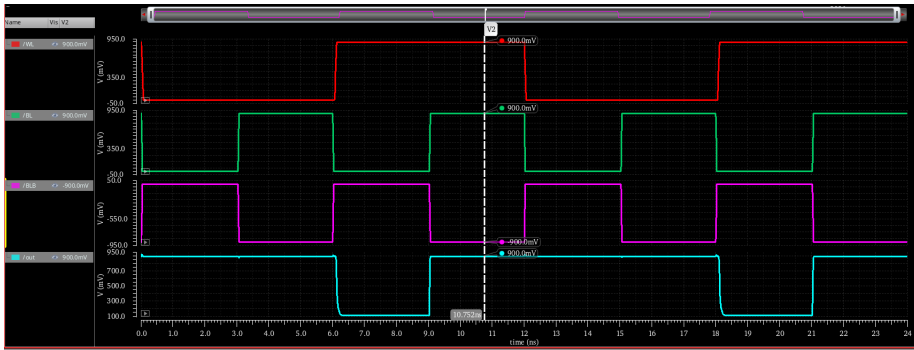**Figure 9** Read waveform of 6T SRAM cell (see online version for colours)

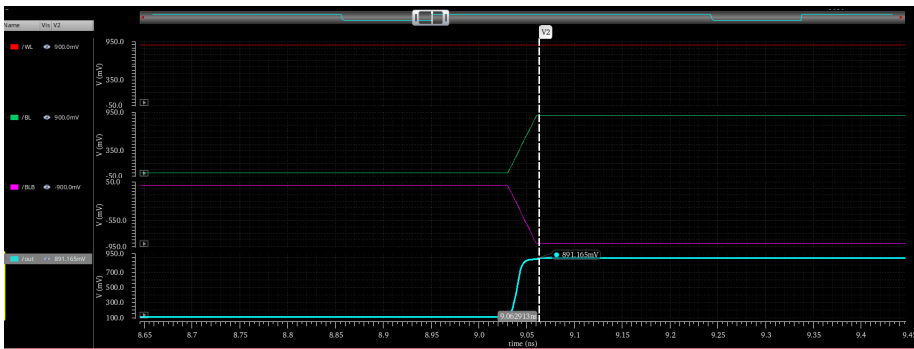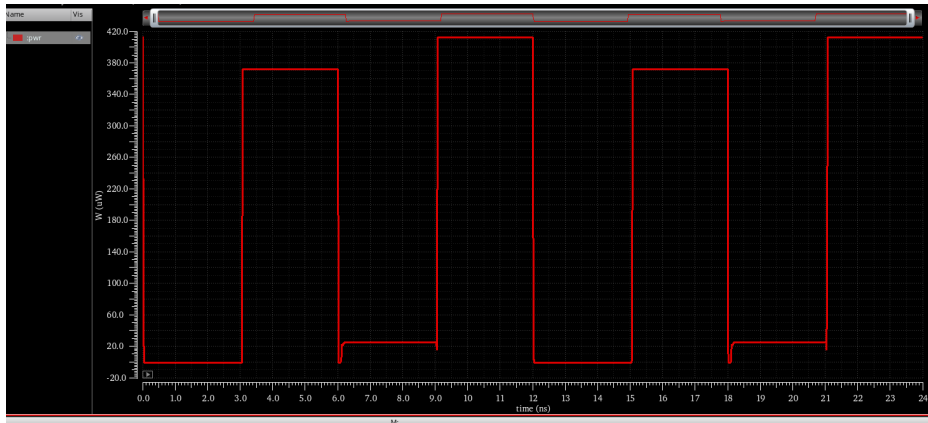**Figure 10** Read delay of 6T SRAM cell (see online version for colours)

Figure 11 Read the power of 6T SRAM cell (see online version for colours)

The input conditions for write operation of the 6T SRAM cell design are $WL = 0.9V, BL = -0.9V$ and $BLB = 0$. This leads to flip the value which is read by the 6T SRAM cell. The write waveform, write delay and average write power for 6T SRAM cell are shown in Figures 13–15 respectively. Similar to the read operation, the write operation also provides the output similar to the BL, when the WL is enabling in the

circuit. For example, the output of write operation is $(0,1)$, when the WL is enabled between the 6–12 ns. Additionally, the write delay and average power of the 6T SRAM cell are 0.058 ns and 204.2 uW respectively.

Figure 12 Average read power of 6T SRAM cell (see online version for colours)
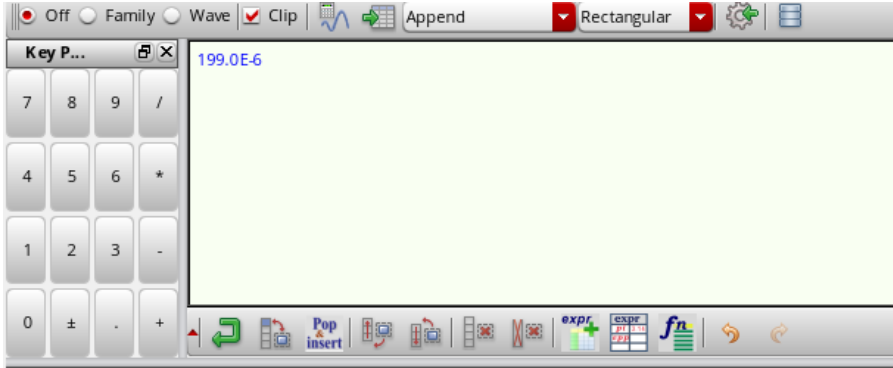

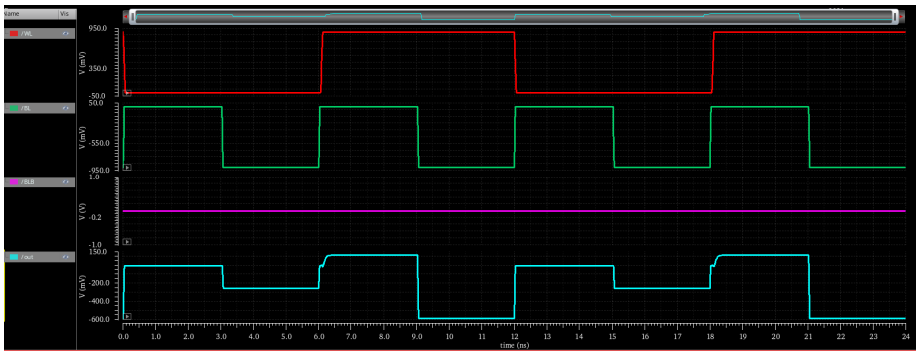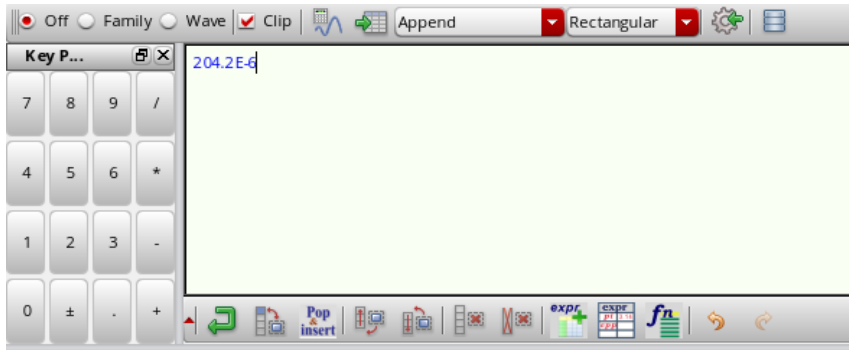**Figure 13** Write waveform of 6T SRAM cell (see online version for colours)

Figure 14 Write delay of 6T SRAM cell (see online version for colours)

Figure 15 Average write power of 6T SRAM cell (see online version for colours)

5.1.2 Analysis of area, power and delay for optimal full adder

The OAF used in the proposed architecture processes three inputs such as $A, B \& C$ and it provides two outputs such as $Sum \& Carry$. The output waveform of the OAF is shown in Figure 16 for verifying it with the given truth table (i.e., Table 3).

**Figure 16** Output waveform for OFA (see online version for colours)
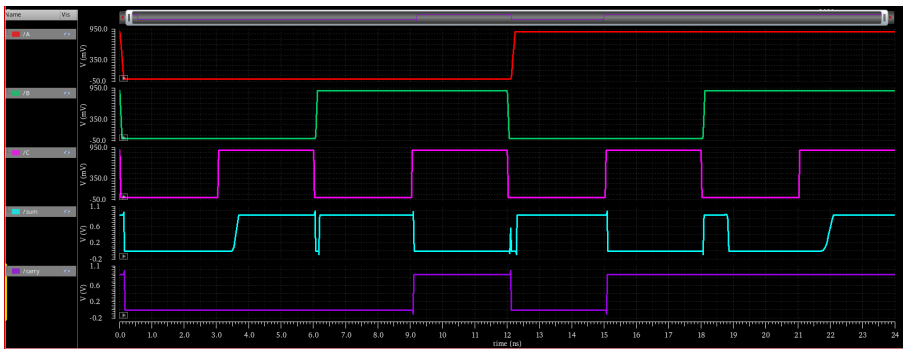
Consider, the inputs $A, B \& C$ are given in the time of 3 ns. From Figure 16, it is concluded that the OFA works as per the truth table given in Table 3. For example, the input given at the 6-9 ns is $(0,1,0)$ and the output from the OFA is $(1,0)$. Hence, the proposed OFA satisfies the operation of a full adder with less resources.

5.1.3 Analysis of area, power and delay for overall FIR filter

The FIR filter output waveform shown in Figure 17 is used for verification purpose, where the 8-bit input is provided to take 8-bit output. The 8-bit input i.e., $\{1,0,0,1,0,0,0,0\}$ and the carry input 0 are given to the proposed architecture. Subsequently, the proposed architecture provides the 8-bit output of $\{1,1,0,0,1,1,0,1\}$ with the carry output of 0. The FIR filter provides all 8-bit output, only when the circuit is designed without any error. However, the proposed architecture provides all the 8-bit values at the output, hence it is proven that the FIR filter is designed without any error.

Figure 17 FIR filter output waveform (see online version for colours)
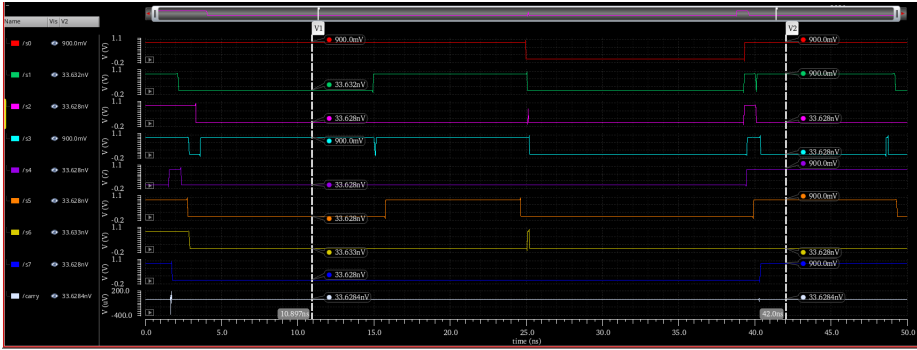
The performance analysis of the proposed architecture is shown in Table 4. Here, the performance of the area, power and delay are analysed with the run time of 24 ns. The performance of the proposed architecture is extensively varied from the 6T SRAM and OFA, because the 6T SRAM and OFA are the individual modules used to design the proposed architecture. The developed OAM-OFA-FIR can be used in the medical signal processing (e.g., ECG signal filtering) to diagnosis the health status of patients. In that, the ECG signal will be taken as input to the Cadence software for filtering the noise from the signals.

Table 4   Performance analysis of proposed architecture

	Design	Performances			
		Transistor count	*Area* (um^2)	*Power* (uW)	*delay* (ns)
6T SRAM	Read operation	6	32.4	199	0.06
	Write operation			204.2	0.058
	OFA	18	97.2	321	0.108
	Proposed FIR	325	1755	453.7	2.104

5.2   *Comparative analysis*

This section shows the comparative analysis of the proposed architecture. There are three existing methods such as Radix 2-LCSLA [22], VD-CLA [23] and CS-FIR [24] are used to evaluate the proposed architecture. This Radix 2-LCSLA [22], VD-CLA [23] and CS-FIR [24] are taken for comparison, because these three methods are implemented using the Cadence virtuoso software with 45 nm technology. The comparative analysis of the proposed architecture is shown in the following Table 5.

From Table 5, it is concluded that the proposed architecture achieves better performance than the Radix 2-LCSLA [22], VD-CLA [23] and CS-FIR [24]. For example, the area of the proposed architecture is 1755 um^2, which is less when compared to the Radix 2-LCSLA [22], VD-CLA [23] and CS-FIR [24]. The area of the proposed architecture is minimised by decreasing the number of adders used in the multiplication process. Because the conventional multiplier requires a high amount of adders for performing the multiplication. The smaller area of the OAM-OFA-FIR results in the less

power consumption and less delay than the Radix 2-LCSLA [22], VD-CLA [23] and CS-FIR [24]. Moreover, the buffer circuit used in the OFA is used to avoid noises, glitches and threshold issues. Hence, the proposed FIR filter with the combination of OAM and OFA provides better performance.

Table 5 Comparative analysis of the proposed architecture

Methods	Area (um^2)	Power (uW)	delay (ns)
Radix 2-LCSLA [22]	9426	85.15222	0.169
VD-CLA [23]	2302	74.143	0.6024
CS-FIR [24]	27,599	1211.98	10.348
OAM-OFA-FIR	1755	53.7	2.104

6 Conclusion

In this paper, the combination of OAM and OFA is used for developing an effective FIR filter architecture. Here, the adders in the OAM are decreased by replacing the half and full adders of the conventional multipliers with the designed OFA. Meanwhile, the buffer circuit used in the OFA is used to avoid noises, glitches and threshold issues. The reduction in hardware resources helps to minimise the area, power and delay of the overall FIR filter architecture. Therefore, the designed OAM-OFA-FIR architecture is used to minimise the area while avoiding the noises that occurred during the filtering process. From the performance analysis, it is concluded that the OAM-OFA-FIR architecture outperforms the Radix 2-LCSLA, VD-CLA and CS-FIR. The area of the OAM-OFA-FIR architecture is 1755 um^2, it is less when compared to the Radix 2-LCSLA, VD-CLA and CS-FIR. In future, the low power techniques such as clock gating or power gating can be used for minimising the overall power consumption of the FIR architecture.

References

1   Zahoor, S. and Naseem, S. (2017) 'Design and implementation of an efficient FIR digital filter', *Cogent Eng.*, Vol. 4, No. 1, p.1323373.

2   Kolawole, E.S., Ali, W.H., Cofie, P., Fuller, J., Tolliver, C. and Obiomon, P. (2015) 'Design and implementation of low-pass, high-pass and band-pass finite impulse response (FIR) filters using FPGA', *Circuits and Systems*, Vol. 6, No. 02, p.30.

3   Vinitha, C.S. and Sharma, R.K. (2019) 'New approach to low-area, low-latency memory-based systolic architecture for FIR filters', *J. Information Optim. Sci.*, Vol. 40, No. 2, pp.247–262.

4   Kumar, P., Shrivastava, P.C., Tiwari, M. and Dhawan, A. (2018) 'ASIC implementation of area-efficient, high-throughput 2-D IIR filter using distributed arithmetic', *Circuits Syst. Signal Process*, Vol. 37, No. 7, pp.2934–2957.

5   Pandey, B., Pandey, N., Kaur, A., Hussain, D.A., Das, B. and Tomar, G.S. (2019) 'Scaling of output load in energy efficient FIR filter for green communication on ultra-scale FPGA', *Wirel. Pers. Commun.*, Vol. 106, No. 4, pp.1813–1826.

6   Kumar, P., Shrivastava, P.C., Tiwari, M. and Mishra, G.R. (2019) 'High-throughput, area-efficient architecture of 2-D. Block FIR filter using distributed arithmetic algorithm', *Circuits Syst. Signal Process*, Vol. 38, No. 3, pp.1099–1113.

7   Vandenbussche, J.J., Lee, P. and Peuteman, J. (2015) 'Multiplicative finite impulse response filters: implementations and applications using field programmable gate arrays', *IET Signal Proc.*, Vol. 9, No. 5, pp.449–456.

8   Chowdari, C.P. and Seventline, J.B. (2020) 'Systolic architecture for adaptive block FIR filter for throughput using distributed arithmetic', *Int. J. Speech Technol.*, Vol. 23, No. 3, pp.549–557.

9   Mohanty, B.K., Meher, P.K., Singhal, S.K. and Swamy, M.N.S. (2016) 'A high-performance VLSI architecture for reconfigurable FIR using distributed arithmetic', *Integration*, Vol. 54, pp.37–46.

10 Kalaiyarasi, D. and Reddy, T.K. (2019) 'Design and implementation of least mean square adaptive FIR filter using offset binary coding based distributed arithmetic', *Microprocess. Microsyst.*, Vol. 71, p.102884.

11 Hatai, I., Chakrabarti, I. and Banerjee, S. (2015) 'An efficient constant multiplier architecture based on vertical-horizontal binary common sub-expression elimination algorithm for reconfigurable FIR filter synthesis', *IEEE Trans. Circuits Syst. I Regul. Pap.*, Vol. 62, No. 4, pp.1071–1080.

12 Arumugam, N. and Paramasivan, B. (2020) 'A novel microprogrammed reconfigurable parallel VHBCSE based FIR filter for wireless sensor nodes', *Wirel. Pers. Commun.*, Vol. 115, No. 3, pp.2197–2210.

13 Mahalakshmi, R. and Sasilatha, T. (2018) 'An improved digital FIR filter design using fast FIR algorithm and modified carry save addition', *Natl. Acad. Sci. Lett.*, Vol. 41, No. 3, pp.147–150.

14 Sakthimohan, M. and Deny, J. (2020) 'An optimistic design of 16-tap FIR filter with radix-4 booth multiplier using improved booth recoding algorithm', *Microprocess. Microsyst.*, p.103453.

15 Aranda, L.A., Reviriego, P. and Maestro, J.A. (2017) 'A comparison of dual modular redundancy and concurrent error detection in finite impulse response filters implemented in SRAM-based FPGAs through fault injection', *IEEE Trans. Circuits Syst. II Express Briefs* , Vol. 65, No. 3, pp.376–380.

16 Mukherjee, D. and Reddy, B.R. (2019) 'Algorithm design, software simulation and mathematical modelling of subthreshold leakage current in CMOS circuits', *Int. J. Comput. Complex. Intell. Algorithms*, Vol. 1, No. 2, pp.129–144.

17 Parameshwara, S. and Renukappa, N.M. (2017) 'Influence of interface structure on performance of organic field effect transistors', *Int. J. Nanotechnol.*, Vol. 4, Nos. 9–11, pp.775–792.

18 Li, Y., Huang, W.T., Chen, C.Y. and Chen, Y.Y. (2015) 'Upper/lower–side random dopant fluctuation on 16–nm–gate HKMG bulk FinFET', *Int. J. Nanotechnol.*, Vol. 12, Nos. 1–2, pp.126–138.

19 Zhang, K., Liu, Y., Zhu, H., Zhao, C., Ye, T. and Yin, H. (2015) 'Doping profile optimisation in bulk FinFET channel and source/drain extension regions for low off–state leakage', *Int. J. Nanotechnol.*, Vol. 12, Nos. 1–2, pp.111–125.

20 Sundar, P.P., Ranjith, D., Karthikeyan, T., Kumar, V.V. and Jeyakumar, B. (2020) 'Low power area efficient adaptive FIR filter for hearing aids using distributed arithmetic architecture', *Int. J. Speech Technol.*, Vol. 23, No. 2, pp.287–296.

21 Jyothi, G.N., Sanapala, K. and Vijayalakshmi, A. (2020) 'ASIC implementation of distributed arithmetic based FIR filter using RNS for high speed DSP systems', *Int. J. Speech Technol.*, Vol. 23, pp.259–264.

22 Satish Reddy, K. and Suresh, H.N. (2020) 'A low-power VLSI implementation of RFIR filter design using radix-2 algorithm with LCSLA', *IETE J. Res.*, Vol. 66, No. 6, pp.741–750.

23   Sumalatha, M., Naganjaneyulu, P.V. and Prasad, K.S. (2019) 'Low power and low area VLSI implementation of Vedic design FIR filter for ECG signal de-noising', *Microprocess. Microsyst.*, Vol. 71, p.102883.

24   Odugu, V.K. (2021) 'An efficient VLSI architecture of 2-D finite impulse response filter using enhanced approximate compressor circuits', *Int. J. Circuit Theory Appl.*, Vol. 49, No. 11, pp.3653–3668.

25   Vijetha, K. and Naik, B.R. (2020) 'High performance area efficient DA based FIR filter for concurrent decision feedback equalizer', *Int. J. Speech Technol.*, Vol. 23, No. 2, pp.297–303.

26   NagaJyothi, G. and Sridevi, S. (2020) 'High speed low area OBC DA based decimation filter for hearing aids application', *Int. J. Speech Technol.*, Vol. 23, No. 1, pp.111–121.

27   Padmavathy, T.V., Saravanan, S. and Vimalkumar, M.N. (2020) 'Partial product addition in vedic design-ripple carry adder design fir filter architecture for electro cardiogram (ECG) signal de-noising application', *Microprocess. Microsyst.*, Vol. 76, p.103113.

28   Rammohan, S.R., Jayashri, N., Bivi, M.A., Nayak, C.K. and Niveditha, V.R. (2020) 'High performance hardware design of compressor adder in DA based FIR filters for hearing aids', *Int. J. Speech Technol.*, Vol. 23, No. 4, pp.807–814.