A Novel Low Power Multiply–Accumulate (MAC) Unit Design for Fixed Point Signed Numbers

N. Jithendra Babu and Rajkumar Sarma

Abstract In the emerging technologies the low power designs play a critical role of operations. Our proposed work is on the low power MAC unit that is used to find the fixed point signed numbers. The proposed design is to achieve high throughput and low power consumption. Our proposed work has various building blocks like firstly, Wallace tree multiplier since a multiplier is one of the key part for the processing of digital signal processing systems and secondly an accumulation block. Since the output from the multiplier and adder is to be efficient, we proposed a BCD block that is used to convert the output into BCD number. The overall MAC is performed in the cadence virtuoso 90 nm technology and performance analysis of each individual block is examined using the cadence virtuoso before designing the overall MAC unit. Power, delay and power-delay product are calculated using the Cadence Spectre tool.

Keywords BCD · Fixed point · Multiply–accumulate unit · Signed number

1 Introduction

MAC refers to the multiply–accumulate unit that is the basic block in many of the digital signal processing systems for basic arithmetic operations. For any real-time processing, to achieve the high throughput and speed, the MAC unit is the basic block. From the past decades, the concentration of designers are more focused on the speed but due to technological development, low power designs have been the major considerations in designing integrated circuits. The main investigation of our work is to design a low-power and high-speed multiplier–accumulator (MAC) unit. The MAC unit is the basic consideration in any of the digital filters. Due to this, MAC unit has now become the basic and essential building block for the processing

N. Jithendra Babu (✉) · R. Sarma
Lovely Professional University, Punjab, India
e-mail: jithendra.lpu@gmail.com

S.S. Dash et al. (eds.), *Artificial Intelligence and Evolutionary Computations in Engineering Systems*, Advances in Intelligent Systems and Computing 394,
DOI 10.1007/978-81-322-2656-7_62

systems. To design the low-power MAC unit we have considered the various architectures of the multipliers and adders. Since, the multiplier is a key element for the multiplications in the MAC unit and the accumulation unit is designed using the various considerations because the function of MAC unit is to multiply the *N*-bit input values from the memory location and accumulates the output of the multiplier with the previously accumulated value that is stored in the accumulator register.

Further this accumulation block is composed of various blocks as accumulation adder and accumulation register in which a register has been used for storing the data within the time synchronization, such that the register is designed using the low power hybrid flip-flop which reduces the overall power reduction of the system. This hybrid flip flop is designed using a novel ultra-low leakage CMOS circuit structure called sleepy stack. The function of mac unit is expressed as

$$Z = \sum_{i=0}^{n-1} a_i b_i$$

This paper is organized as follows. Section 2 describes the multiply–accumulate unit. Section 3 describes the proposed work of our design using the various techniques. Section 4 describes the proposed mac architecture for sign numbers. Section 5 describes the experimental analysis. Section 6 describes the power-delay analysis.

2 MAC Unit

The main concerns of the signal processing system are multiplications and additions, the multiplier and accumulator is the basic building block for any of the digital signal processing systems. For this, MAC architecture represents the various functions of the different blocks for the high speed throughout the system for an efficient processing of the signals. Basically, the MAC is a DSP processor which is used for high-speed signalling such that the different blocks of the MAC unit should also perform the high-speed operations. This can be achieved by considering the different parameters such as power, speed and area. The block diagram of the MAC architecture is represented in the Fig. 1 [1–7]. Here an *N*-input bits of two sets are fetched from memory and given as input to the multiplier block which performs the multiplication and produces the 2*N*-bit output as input to the resister block which stores the data and sends the data to next level as input to the to the adder. This adder performs the adding operation by adding the output from the register block with the previously accumulated value that is stored in the accumulator register. Thus, the output from the adder is given to the accumulator register and the overall output is taken from the output of the accumulator register which is stored in the feedback register for next step. In the architecture of the MAC unit the blocks is considered to be of a high speed and low power consumption. The multiplier and

adder circuit is designed by considering the various architectures and the final MAC unit is designed by using those individual blocks.

3 Basic Blocks in Our Design

3.1 Multiplier

In any of the signal processing systems the main concern depends on the multiplication of two numbers. So, multiplier plays a vital role in such type of signal processing systems.

While designing multiplier architecture we should first consider the speed (because for any digital system the speed is the main consideration) and the second important aspect is power. Due to this we have implemented the multiplier which is

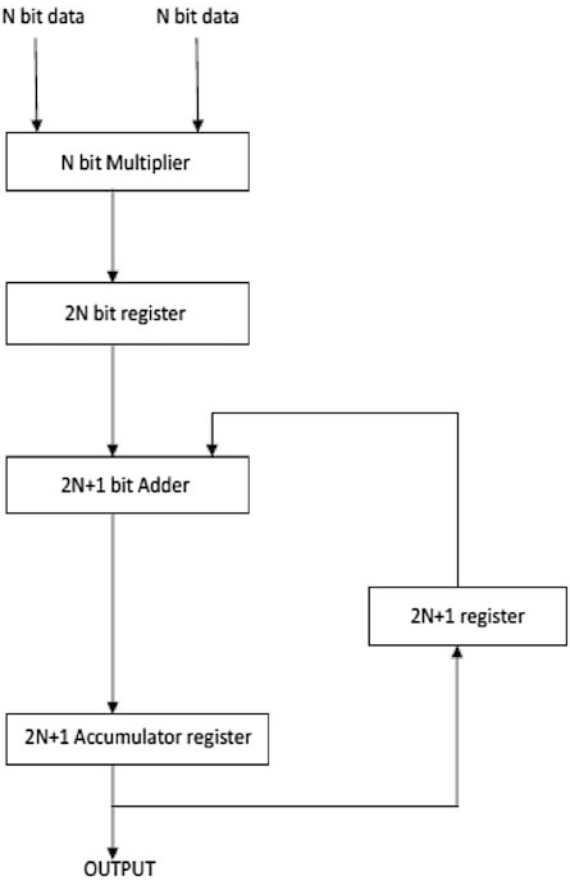

Fig. 1 Basic MAC unit [1]

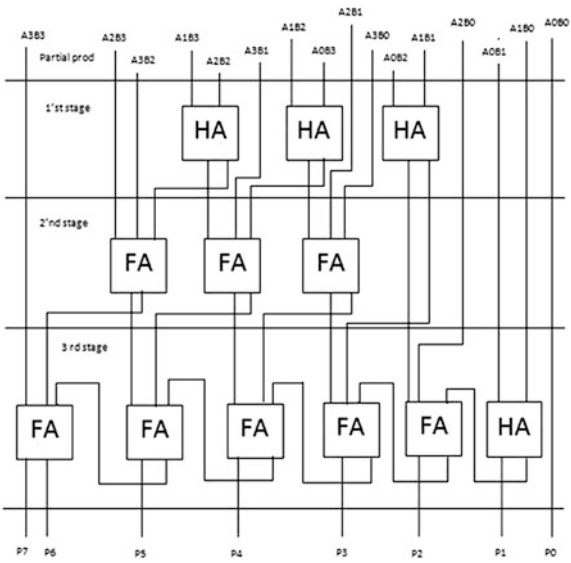

Fig. 2 Wallace tree multiplier [8]

based on Wallace tree algorithm known as Wallace tree multiplier [8]. The multiplier has efficient speed and less power dissipation when compared to other multiplier architectures. The Wallace tree multiplier architecture for a 4-bit is shown in Fig. 2 is:

1. At the very first step the partial products are multiplied by using the AND gates.
2. Half adder is used to add the two partial products.
3. At the last full adders are used which is used to produce the final result by using a simple Ripple carry architecture.

3.2 Binary to BCD

The proposed design is to convert the output of the multiplier block into the binary coded decimal format. Since, the multiplier multiplies two values that is fetched from memory and gives the output as binary format. So, there is a need for the conversion of binary values to BCD logic. The proposed designed is implemented using the basic logic gates as AND, XNOR and full adders. The output from the XNOR block passes through the AND gate whenever one of the input of XNOR gate is always high. Depending on the XNOR output the AND gate passes the BCD value as an output which are consecutively added by the full adders.

3.3 Adder

Adder is the most important part of the processing units such as arithmetic unit, signal processing unit, etc. In addition to these, the adders have important considerations in applications such as digital system architectures and processing units, etc. The basic arithmetic operations such as add, sub, multi and divide are implemented using adder as a basic building block [9].

Due to this, the design of a high-performance full adder with low power dissipation is necessary. The adder circuit is the basic block in our proposed work for adding the current output value from the multiplier block with the previously accumulated value that is stored in the accumulator register. Our adder is of 9 transistors which enhances high speed and low power dissipation throughout the circuit while compare to the other full adder architectures [9]. With the modification in W/L values of the transistors in the 9-transistor full adder circuit, provides better efficiency in the output. The modification of the W/L values in the adder circuit is done because the proposed architecture is scaled down to design in 0.9 m technology. The proposed adder is used to design our multiplier block and adder circuit that adds the output from the multiplier and previously accumulated value and is used in the design of our BCD block.

3.4 BCD Block

The proposed BCD block is to maintain the overall output of the MAC unit to be in the binary coded decimal format. The BCD block is designed using three 2:1 MUX, two 4:1 MUX, etc. The BCD block is used in two combinations in our proposed architecture as the first combination is used before our adder block which maintains the output of the first register block to be in BCD format and the second combination is used after the adder block which again helps in maintaining the overall output of the MAC unit to be in the binary coded decimal format. The proposed block diagram is shown in Fig. 3. The proposed design works according to the following combinations:

1. If the inputs i3 and i7 of BCD block are 0 s, then it is directly bypassed to the output of the BCD block.
2. If the i3 bit is 1 and i7 bit is 0 then input is converted by adding 0110 to the LSB at the first stage and OR with 1001 to the input LSB bit and considering it as a select line and adding 0110 to MSB bit at the second stage from the output of MUX block and passed to the output.
3. If the i3 bit is 0 and i7 bit is 1 then input is converted by adding 0110 to the MSB bit and OR with 1001 to the input MSB bit and using it as select line to MUX block and passing the output.

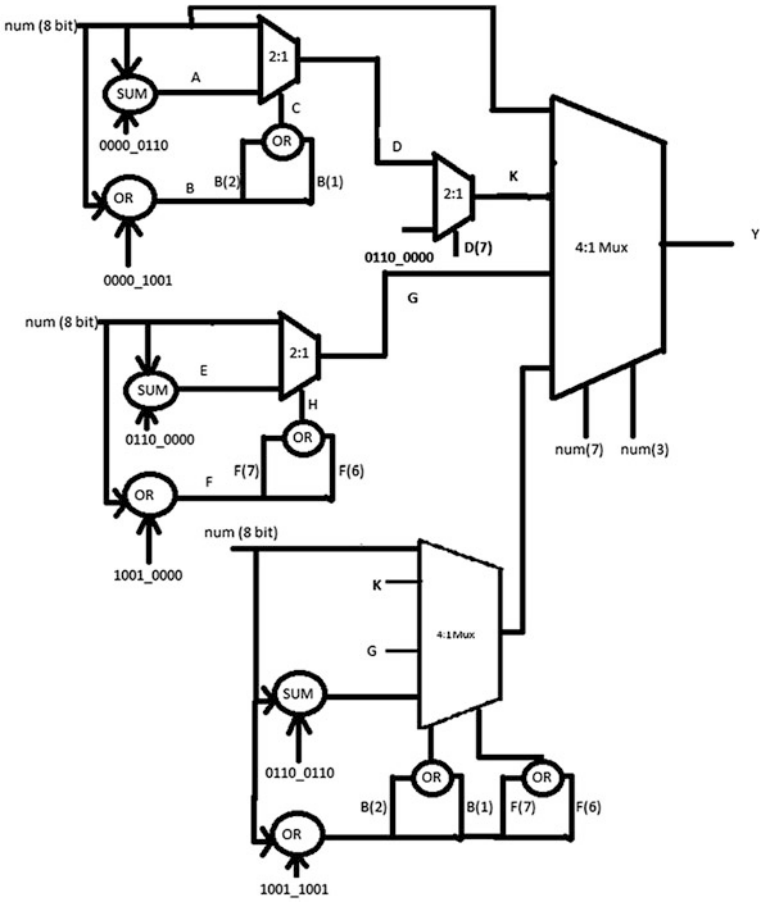

Fig. 3 BCD block

4. If the i3 and i7 bit is 1 then we are checking for all the combinations that are explained in the above steps by using a 4:1 MUX as first three inputs and last input will be by adding 01100110 to the input at the first stage and OR with 10011001 to the input and using it as a select line to pass the output.

3.5 Sleepy Stack Inverter

The sleepy stack inverter is a combination of two techniques one is forced stack and the other is sleep transistor technique [10]. As we know that for any low power design the main consideration of the work is to reduce the power dissipation and to eliminate leakage currents that occur while the transistor is in off mode.

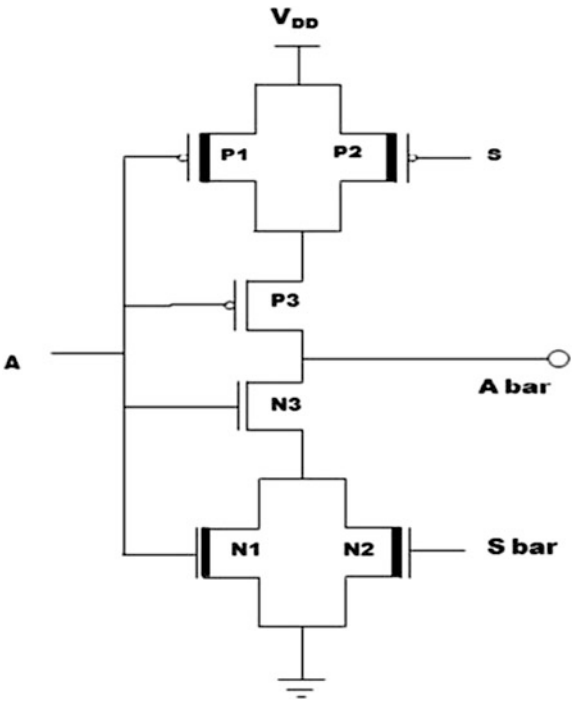

Fig. 4 Sleepy stack inverter [11]

To eliminate this effect in our proposed design we are using the sleepy stack technique [11]. The block diagram is shown in Fig. 4.

3.6 D Flip-Flop

The D flip-flop here we are using is designed using the sleepy stack technique which enhances high switching speed and less power dissipation and low leakage currents through the register unit [11]. In the digital circuits the storing of data is done with the flip flops or latches. So, the design of D flip-flop has to be maintained at low power and high switching speed. The flip-flop design used in our architecture has two phases of operation [11]. The design is based on a conventional dual dynamic flip-flop which enhances the high switching speed throughout the circuit [12]. The function of d flip-flop is that it maintains the output which follows the input whenever our clock pulse is high. So the flip-flop designed here is a positive edge triggered hybrid node flip-flop [11]. The block diagram is shown in Fig. 5.

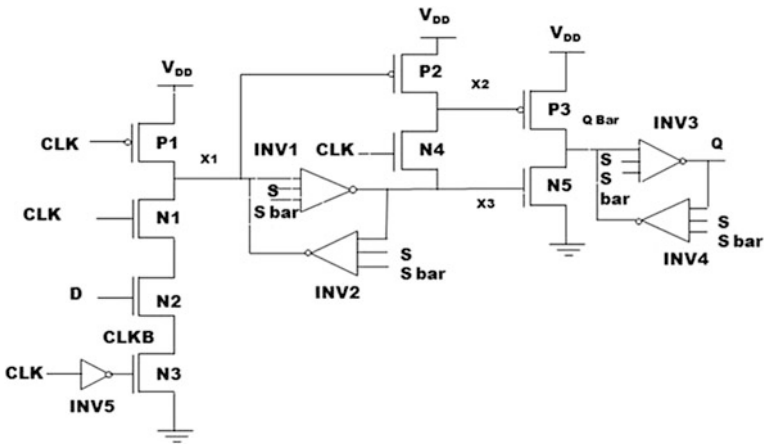

Fig. 5 Hybrid flip-flop [11]

3.7 Accumulator Register

The term accumulator or register refers to a flip-flop or latches which is used to store the one bit data. In our proposed work the accumulator register is used to store the outputs from the multiplier output which is an 8-bit and the overall MAC output which is an 12-bit used for the successive addition of the previous consecutive values with the adder circuit. The accumulator register consists of D flip-flop, basic gates that are operated by clock synchronization. The register cell has 3 inputs and 1 output. The inputs are D, Wsel and Rsel and Q will be the output. The flip-flop will store the input value when Wsel is 1, if Rsel signal is 1, then the flip-flop will pass the value that is stored using a tristate buffer to the output. In our proposed work the D flip-flop is implemented using a technique called DDFF, which is a hybrid flip-flop that uses sleepy stack technique for less power dissipation and low leakage currents which occurs during the storing of data in the register block [1]. The block diagram is shown in Fig. 6.

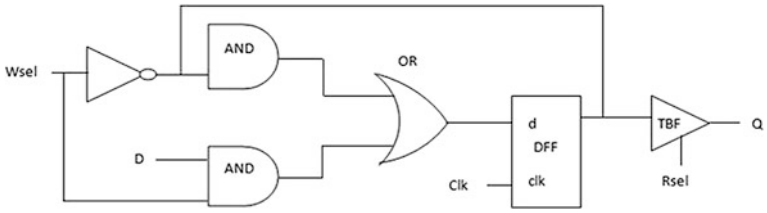

Fig. 6 1-bit register

4 Modified Architecture for Fixed Point Unsigned and Signed Numbers

4.1 Unsigned Number Architecture

Our proposed MAC design function is that it multiplies the two values and then adding with the previously accumulated values that is stored in the accumulator register. Here we have proposed two architectures, one is for unsigned number and the other is for signed number. The architecture for unsigned number will be as follows. The inputs of two sets of 4-bit BCD are fetched from the memory and are given to the multiplier and produce an output of 8-bit. The output from the multiplier is passed through the 8-bit register which stores the data for a certain time and reads the data to the next level during the next transition time. The output from the register block is passed to the BCD block which helps in retaining the inputs to be in BCD format and passed to the adder circuit. The adder circuit performs the addition operation with the previously accumulated values that are stored in the register and sends the data again to the second BCD block which helps in maintaining the overall output of the MAC to be in the BCD format through the output of the accumulator register. Here the BCD block, adder and the second register combination is of 12-bit. The proposed block diagram is shown in Fig. 7.

4.2 Signed Number Architecture

Here in the proposed design the multiplier multiplies the two sets of BCD value and passes the value to the adder at which the input for the adder circuit will be positive number or negative number. Depending on the sign bits that are taken at input sets will be ex-or with each other and defines that the product can be directly bypassed or complemented. The term 0 in the MSB indicates the positive number and the term 1 in the MSB bit indicates the negative number. So there is a need for eliminating the negative numbers and to make it as a positive number.

The modified architecture of MAC deals with the fixed point sign numbers. Our proposed modified architecture has the same architecture as if used in the fixed point representation. There is a small variation that the proposed architecture deals with the sign numbers so necessary blocks are synchronized into the fixed point architecture. The proposed architecture additionally uses a binary to BCD converter block and three multiplexer blocks which deal with the sign numbers. The function is that the product coming from the BCD block will be 10 s complemented and discharged through the MUX block by using sign bit as a select line and the previous accumulated value that is stored in the register to be 10 s complemented

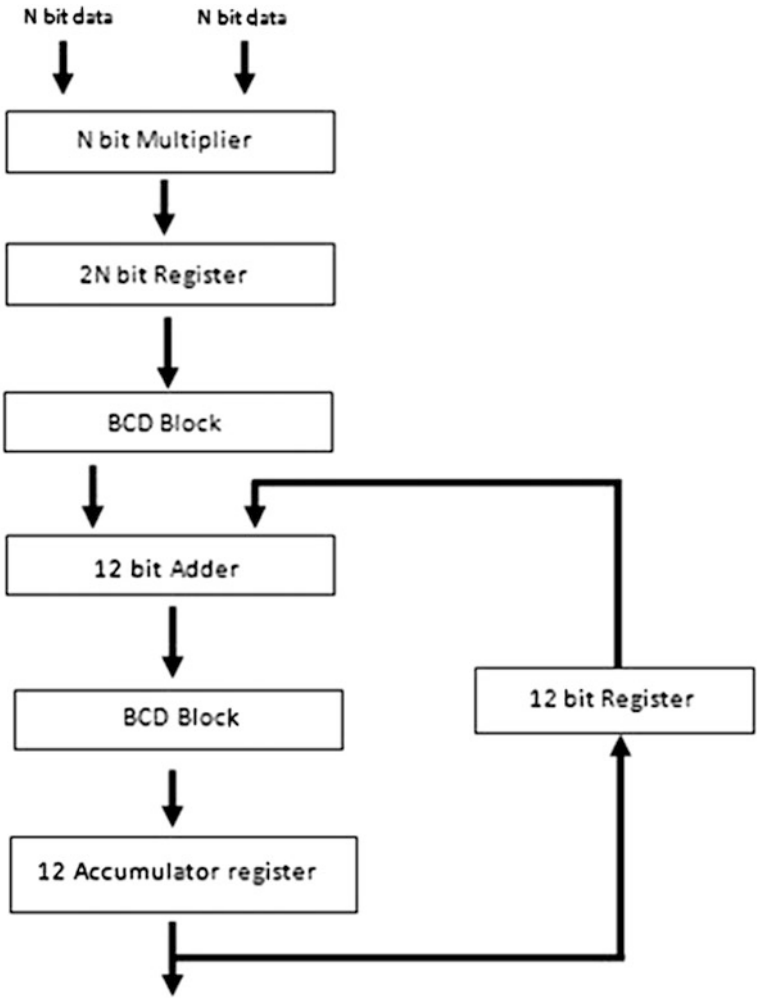

Fig. 7 Proposed MAC for unsigned numbers

and passed through another MUX by using the higher most MSB as a sign bit. The outputs from the MUX blocks are added using an adder circuit and it is passed through the MUX by taking the other input as complement of adder output by using 10 s complement and sign bit from the higher most MSB bit of the adder output and the MUX output is passed to the BCD block as done in the fixed point architecture. The MAC architecture for signed numbers is shown in Fig. 8.

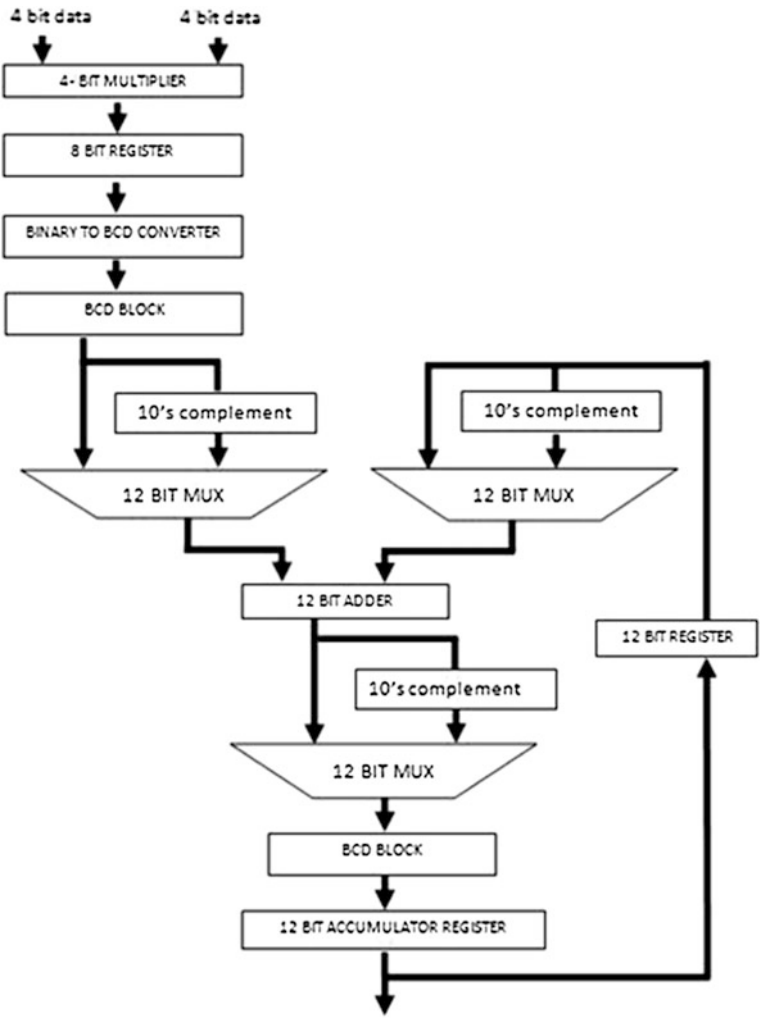

Fig. 8 Proposed MAC for signed numbers

5 Experimental Work

5.1 Unsigned Architecture

The entire MAC unit is implemented in cadence virtuoso at 90 nm technology. First the individual blocks that are the key parts of the mac unit have been designed at the virtuoso and the performance has been verified by considering the parameters power and delay. At last the individual blocks are connected according to the architecture which produces the overall architecture of our fixed point

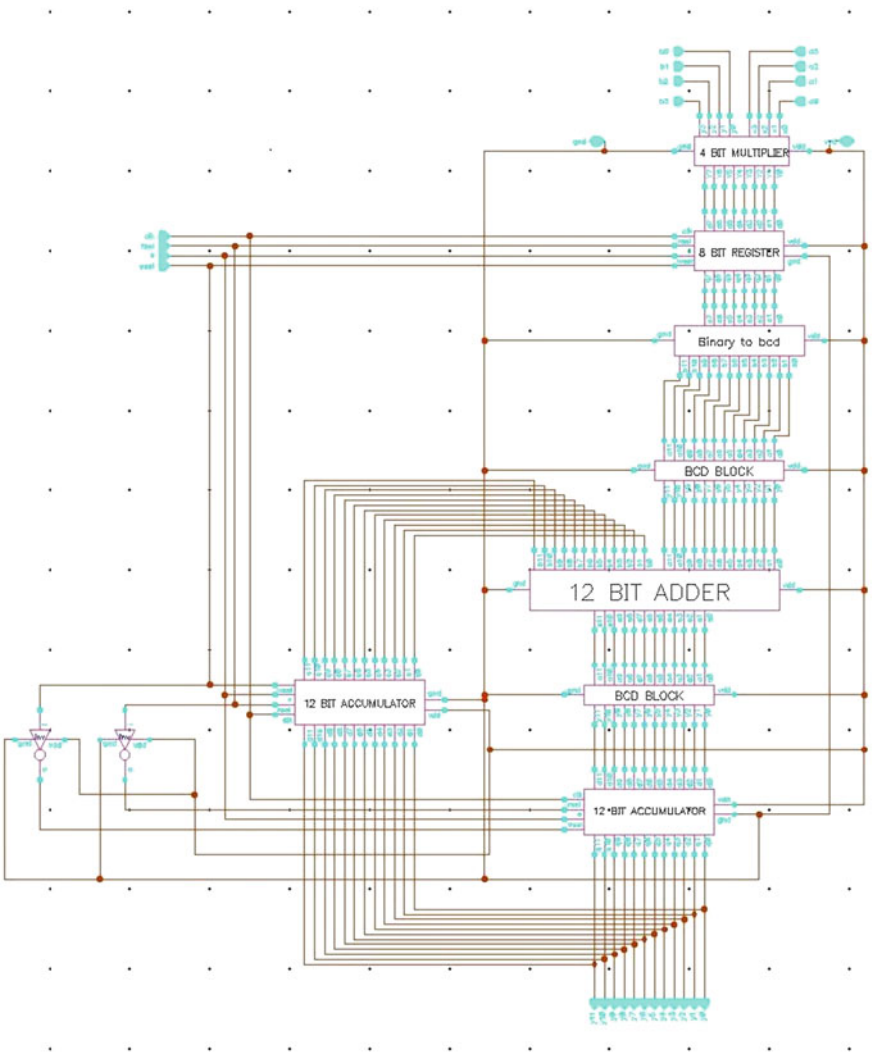

Fig. 9 MAC schematic for fixed point unsigned number

representation. The MAC unit is designed at the cadence virtuoso 90 nm technology and the following considerations are done. The overall time period of our proposed design is to be considered at 100 ns and our input bits are considered to be varied from the 40 to 20 ns and the clock gating for the register unit is taken as a positive edge clock. The Wsel and Rsel of the register unit are inverting and varying with the time response of the input bits. The working of MAC unit is as follows the multiplier multiplies the two sets of 4-bit values and produces the output as an input to the register block during the time response of 20 ns. In the register unit during the

positive edge clock Wsel is to taken as 1 which writes the input data to my register block at 0–20 ns after the completion of writing the data into the register block Wsel is disabled and Rsel is enabled which reads the data at 21–40 ns and passes the output to the binary to BCD block. The binary to BCD block converts the output of the multiplier to the BCD format where it is in binary format. The BCD block maintains the exact BCD logic and passes the value to the adder. The adder circuit adds the current input with the previously accumulated value that is stored in the register and sends to the BCD block. Here again the BCD block maintains the input to be in BCD value in the same time response and passes to the 12-bit accumulator register which reads the data from 41 to 60 ns which is the overall output of our fixed point Mac unit shown in Fig. 9.

5.2 Signed Architecture

The entire proposed MAC unit for the signed numbers is implemented in cadence virtuoso at 90 nm technology shown in Fig. 10. First the individual blocks that are the key parts of the mac unit has been designed at the virtuoso and the performance has been verified by considering the parameters power and delay. At last the individual blocks are connected according to the architecture which produces the overall architecture of our fixed point signed number representation.

The MAC unit is designed at the cadence virtuoso 90 nm technology and the following considerations are done. The overall time period of our proposed design is to be considered at 100 ns and our input bits are considered to be varied from 40 to 20 ns and the clock gating for the register unit is taken as a positive edge clock. The Wsel and Rsel of the register unit are inverting and varying with the time response of the input bits. The working of MAC unit is as follows the multiplier multiplies the two sets of 4-bit values and produces the output as an input to the register block during the time response of 20 ns.

In the register unit during the positive edge clock Wsel is to taken as 1 which writes the input data to my register block at 0–20 ns. After the completion of writing the data into the register block Wsel is disabled and Rsel is enabled which reads the data at 21–40 ns and passes the output to the binary to BCD block. After the conversion of binary input to the BCD output the values are sent to the BCD block. The BCD block maintains the exact BCD logic and passes the value to the multiplexer where our original product will sent or the compliment of the product will be passed through the MUX where the select line will be the AND of B0 and B3 bits of the XOR output of signed bits. At this time there will be parallel execution of the previously accumulated value whether to send the accumulated value or the compliment of it through the MUX depending on the B3 and B0 of the higher MSB of the accumulated value. The outputs from the MUX are passed through the adder circuit. The adder circuit adds the outputs of the two MUX and

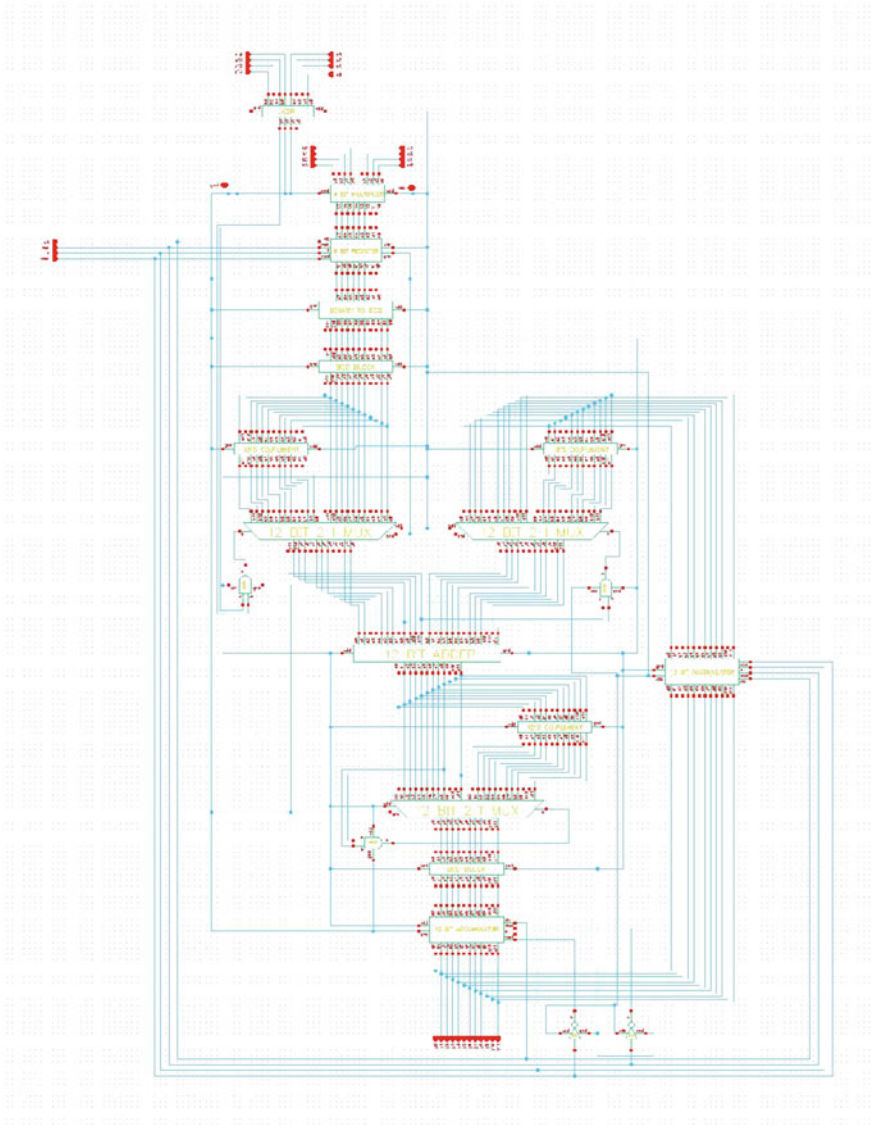

Fig. 10 MAC schematic for fixed point signed number

sends to the BCD block depending on the higher MSB bits through the MUX. Here again the BCD block maintains the input to be in BCD value in the same time response and passes to the 12-bit accumulator register which reads the data from 41 to 60 ns which is the overall output of our fixed point Mac unit.

6 Power-Delay Analysis

The MAC unit is stimulated in cadence spectre tool and analysed the power, delay and power-delay product using the spectre stimulator of each individual block and the overall MAC unit. The MAC unit is stimulated at a supply voltage of 0.9, 1.2, 1.8 V and power, delay are calculated at the three different supply voltages are shown in Tables 1, 2 and 3.

Table 1 At 0.9 V

Block	Power	Delay	PDP
4-bit multiplier	28.58E-6	20.09E-9	574.17 fs
8-bit register	7.583E-6	20.05E-9	152.03 fs
Binary to BCD	1.76E-3	5.888E-9	10.362 fs
BCD	476.5E-6	1.37E-9	652.80 fs
Adder	21.31E-6	106.3E-12	2265.25 as
12-bit register	7.11E-6	20.06E-9	142.62 fs
Unsigned MAC	1.033E-3	40.2E-9	41.52 ps
Signed MAC	5.507E-3	207.1E-12	1140.4 fs

Table 2 At 1.2 V

Block	Power	Delay	PDP
4-bit multiplier	95.1E-6	20.12E-9	1913 fs
8-bit register	16.13E-6	20.05E-9	323.4 fs
Binary to BCD	4.694E-3	1.308E-9	6.13452 ps
BCD	1.206E-3	869.8E-12	1048.9 fs
Adder	53.03E-6	75.73E-12	4015.9as
12-bit register	17.13E-6	20.05E-9	343.4 fs
Unsigned MAC	2.974E-3	80.13E-9	237.9 ps
Signed MAC	14.24E-3	40.16E-9	571.87 ps

Table 3 At 1.8 V

Block	Power	Delay	PDP
4-bit multiplier	410.6E-6	20.13E-9	8265.3 fs
8-bit register	181.1E-6	20.05E-9	3631 fs
Binary to BCD	17.17E-3	2.671E-9	45.86 ps
BCD	4.658E-3	657.9E-12	3064.4 fs
Adder	183.2E-6	58.79E-12	10770.3 as
12-bit register	143.6E-6	20.05E-9	2879.18 fs
Unsigned MAC	13.29E-3	80.1E-9	1064.5 ps
Signed MAC	51.52E-3	40.11E-9	2066.46 ps

7 Conclusion

The proposed MAC architectures for the fixed point signed and unsigned numbers are implemented at the cadence 90 nm technology. As the technology is scaled down there is a need to individually analyse the blocks of the MAC unit. The different blocks of MAC were designed and analysed to determine the power and delay. Since MAC is a digital signal processing element the design of the MAC in the cadence virtuoso environment is done successfully by varying the W/L ratios of the transistors wherever needed by doing the parametric analysis. The power, delay and power-delay products are calculated using the cadence spectre for unsigned number and signed number architecture by transient analysis. In our proposed design the inputs and output waveforms are huge and is difficult to put in the paper and the power, delay and power-delay product analyses are done based on the above waveforms. The proposed work of our paper will be on the floating point MAC architecture and the clock synchronization for the overall MAC.

References

1. Shanthala S, Kulkarni SY. VLSI design and implementation of low power MAC unit with block enabling technique. Eur J Sci Res. ISSN 1450-216X.
2. Kharate AB, Grumble PR. VLSI design and implementation of low power mac for digital FIR filter. Int J Electr Commun Comput Eng. 4(2) REACT-2013, ISSN 2249071X.
3. Sen A, Mitra P, Data D. Low power mac unit for DSP processor. Int J Recent Technol Eng (IJRTE). 1(6);ISSN: 2277–3878, January 2013.
4. Abdelgawad A. Low power multiply accumulate unit (MAC) for future wireless sensor networks. 978-1-4673-4637-5/13/ 2013 IEEE.
5. Deepak S, Kailath BJ. Optimized MAC unit design. 978-1-4673-5696-1/12/ 2012 IEEE.
6. Jagadesh P. Design of high performance 64 bit MAC unit. In: 2013 International conference on circuits, power and computing technologies [ICCPCT-2013], 978-1-4673-4922-2/13/ 20 13 IEEE.
7. Francis T, Joseph T, Antony JK. Modified MAC unit for low power high speed DSP application using multiplier with bypassing technique and optimized adders. IEEE 31661s.
8. Shanthala S, Prasanna Raj C, Kulkarni SY. Design and VLSI implementation of pipelined multiply accumulate unit. In: IEEE International conference on emerging trends in engineering and technology, ICETET-09.
9. Mishra S et al. On the design of high-performance CMOS 1-bit full adder circuits. In: International conference on VLSI, communication and instrumentation (ICVCI). 2011.
10. Park JC, Mooney III VJ. Sleepy stack leakage reduction. IEEE Trans Very Large Scale Integ (VLSI) Syst. 2006;14(11).
11. Evangelene H. A novel low power hybrid flip op using sleepy stack inverter pair. In: Science and information conference, London, UK. August 27–29, 2014.
12. Abdelgawad A, Bayoumi M. High speed and area efficient MAC unit for digital signal processing applications. 1-4244-0921-7/07 @ 2007 IEEE.