FPGA-based implementation of floating point processing element for the design of efficient FIR filters

Tintu Mary John | Shanty Chacko

1School of Electrical Sciences, Electronics & Communication Engineering, Karunya Institute of Technology & Sciences, Coimbatore, India
2School of Electrical Sciences, Electrical & Electronics Engineering, Karunya Institute of Technology & Sciences, Coimbatore, India

Correspondence
Tintu Mary John, Electronics & Communication Engineering, School of Electrical Sciences, Karunya Institute of Technology & Sciences, Coimbatore, India.
Email: tintuajin@hotmail.com

Abstract
Numerous applications based on very large scale integration (VLSI) architecture suffer from large size components that lead to an error in the design of the filter during the stages of floating point arithmetic. Hence, it is necessary to change the architectural model that increases the design complexity and the time delay effect. The issue encountered in the VLSI architectures for finite impulse response (FIR) filter is the increased number of components, especially delay elements. For the VLSI architecture reconfigured with reduced register usage, this article provides the floating point processing element (FPPE) implementation with Cross-Folded Shifting. The proposed FIR filter system reduces the number of components in the circuit which increases the complexity and high delay rate in the logical operation. The system has a comparatively reduced delay rate and power consumption. Hence, an efficient fast architecture based on the FPPE method is developed in this paper.

1 | INTRODUCTION

Developments in electronic technology affecting the whole design structure have caused various difficulties to the digital systems. Hence, the design of architecture requires a clear idea about various aspects that lead to novel very large scale integration (VLSI) architecture [1]. The digital signal processing requires huge amount of data, continuous handling capacities, and broadened computational power. It prompts an expanded consideration towards the versatile structures with run-time reconfiguration capacities. Thus, various new applications and algorithms evolved to provide a feasible option for the architecture. The customary field-programmable gate array (FPGA)-based reconfigurable arrangements are no longer feasible, on account of their granularity and an enormous measure of steering overhead that bring about poor proficiency on silicon structure. At present, investigation has moved to designs of reconfigurable processing that offer command over a wide range of bits, one after another. The framework designs perform contingent upon the top-level array, interconnection scheme, and handling of the individual cell. On account of shared memory, the structures are basic to system throughput. Subsequently, the preparation of cells is the significant mainstays of the framework that are similarly essential to the total throughput. Hence, it is critical to advise extendable arithmetic processing units that permit integrated system structure so as to guarantee maximum throughput from a reconfigurable array-based architecture. The significant prerequisite of the media processing and the modern digital signal processing (DSP) application is the ability to handle floating point (FP). The reuse or extension of integer data path leads to a reduction in development time, execution of low-cost system, and feasible FPGA data paths.

The prevailing design methods in finite impulse response (FIR) structure mainly focus on the reduction of complexity in adder, whereas the complexity in a multiplier of the filter contributes a key part to the space and power. Most of the traditional methods do not provide a clear idea about the reduction in the memory footprint of the channel filter. But the block processing method enhances the memory footprint of channel filter where the output of the block is figured out in parallel.

The efficient reconfigurable architecture is used to derive an approach towards the major complexity in VLSI architecture such as, area, speed, and power. In the arithmetic calculation, most of the system struggles in the FP number estimation because of its exponential term.

The FP helps to predict the operation of both derivation and integration of that given signal.

The floating point processing element (FPPE) is an important unit for addition, subtraction, multiplication, and so forth of binary data in DSP applications. To design an FPPE
model, there are several techniques used for estimating the
Mantissa value and the exponential term, with minimum usage
of logical components. There are some limitations in those
techniques which lead to increase in memory usage and delay.

In DSP systems, issues may occur in calculating filter value
at various stages of FP. To overcome this drawback, some of
the system refers changing the architecture of that model. But
this will increase the design complexity and time delay effect.
The aim of reducing architecture is to gain reduction in power
loss and area. This work introduces the new concept of Cross-
Folded Shifting design for ALU unit. The look-up table (LUT)
device utilization summary is compared with previous works
based on the number of register and flip flops (FFs), power,
and delay in implementation.

The rest of this article is organized as follows: Section 2
explains the literature background of the FIR filter design.
The design and development of FPPE and cross-folded-
shifting-based architecture is presented in Section 3. The
hardware architecture of the proposed structure with pre-
vious methods is discussed in Section 4. Section 5 explains
the results, and Section 6 portrays the conclusion and the
future scope.

2 | LITERATURE REVIEW

Digital FIR filters are basic components in many DSP
systems [1]. With the increasing developments in VLSI
technology, the real-time realization of FIR filter with less
hardware and less latency has become more and more
important. Since the complexity of implementation grows
with the length of filter, several algorithms have been
developed to realize an effective architecture for FIR filter
with reduced filter length using Specific Integrated Circuits
(ASIC) and FPGA platforms.

The architecture of two integer reconfigurable data paths is
proposed to achieve a distinct cycle of basic operations
(addition, subtraction, multiplication, and accumulation). More
difficult arithmetic and logical operations were performed to
work in multithreaded systems in this method. While performing
these ranges of operations, the data path has short and uniform
critical path. The data path is extendable, and they are
parameterized to support higher precision arithmetic as well as
software-assisted variable precision reconfigurable systems.
The procedure utilized static, domino, and information-driven
unique rationale (D3L) to execute 8-bit form of the number
information ways in IBM (90 nm) [2]. Here, the data paths
accomplish working frequencies in the scope of 1 GHz. A 24-
piece augmentation of the information in the gliding point
preparing component (FPPE) introduced another single pre-
cision. They are framed by the engineering and circuit exami-
nation on the whole number information ways. The average
force utilization of this FPPE is 6.5 mW, and its capacity is at a
recurrence of 1 GHz.

The attributes of the whole framework impact strongly on
the throughput, flexibility, and the expense bestowed by the
arithmetic processing components. On the off chance that the
plan calculation is capable, broadening its results and config-
uration reuse show an enormous decrease in the improvement
time. It upgrades the presentation of PE esteem. Xydis et al.
discussed about the effects of creating a productive program-
nable number juggling calculations to execute adaptable
structures [3]. This necessitates a steady interconnection plot
between numerous segments which is fundamentally engaged
to build up an inline adaptability that is permitted into the
engineering and accomplishes computational efficiency.
Despite the fact that this methodology indicated a huge trouble
in the interconnect arrangement is probably a bottle neck
performance on basis of power in large array systems.

Another methodology [4] incorporates adaptability into the
computational calculation that permits enormous cluster to
have the fundamental adaptability with an essential intercon-
nection scheme. The effect of advancement of digital arith-
metic on image processing frameworks is studied in this article.
The above-mentioned adaptability is accomplished by utilizing
a blend of calculation and circuit improvement that leads to
adaptability in chip design.

An innovative architecture for FP in multiple precision,
which is named as Multiply–Add Fused (MAF) unit design is
proposed in [5]. They accomplish FP of either single precision
double a time or double precision one time. When the module
is on the critical data path, the traditional double precision
MAF unit of each module is vectored, and it is shared between
multiple precision operations or by duplication of hardware
resources. It also extends to other FP operations such as,
multiple precision FP addition or multiplication.

The implementation of a new split-path of full adder
function that is considered as the strongest contender in
terms of performance, power efficiency as well as strong
drivability, is done in [6]. The functions of PROPAGATE
and GENERATE are used to realize full adders to obtain
an optimum solution from the performance and process
viability and also provide a reduction of capacitance at
critical nodes, high drivability, and robustness to process.
The adder used in this is far better than the conventional
dynamic domino adders in terms of reliability and robust-
ness to process variations.

A parameterized MAC unit used for DSP core of
embedded systems are described in the system described in [7].
The MAC unit offers a complete set of instructions for integer
and fractional data types. The placement parameters and their
architectural implementations are controlled in the current
generations. The organized physical assignment to this gen-
eration process ensures fast and predictable performance esti-
mation. They also provide good performance, predictable
quantitative analysis, and better optimizations than existing
methods for modern technologies. The above method is used
to govern an optimal DSP core architecture that allows a fast
and reliable estimation of the MAC unit to perform the
characteristics for various consequences.

The difficulty of a register in the direct-transpose forms of
structures of FIR filter is to discover the option of the register
reuses. The number of registers used in direct form is
reasonably lesser than the transpose form and it allows register
reuse in corresponding implementation. The architecture of Distributed Arithmetic (DA) for reconfigurable block-based FIR filter is capable of bigger block sizes and higher filter lengths. The system function equation is computed using direct form or transposes form structure. Both the forms need the same number of arithmetic components such as multipliers, adders etc. Therefore, the number of register bits for both forms are different. In this system, the upturn in the block size does not increase the number of registers. The structure of block size consumes less power than the existing structure for the same output rates.

3 | DESIGN

The main objective of this work is the IC chip design of the FIR filter using Xilinx software tool for filtering any signal. By default, any data signal has noises, which need to be removed using some kind of filter. A new compact integrated chip using FPPE as an important unit for addition, subtraction, and multiplication in ALU as well as the binary data usage in DSP applications, is designed.

In a DSP processing system, there may occur some error in calculating filter value, in the FP stage. To overcome this drawback, in some of the systems, the architecture of that model is changed. But this increases the degree of complexity and time delay. The objective of reducing architecture is to achieve reduction in power loss and also in area. Here, we introduce the new concept of Cross Folded Shifting design for ALU unit. The LUT device utilization summary is compared here based on previous works, with number of registers, number of FFs, speed, power, and delay in the implementation.

There are many methods and approaches to design an FIR filter such as Distributed Arithmetic (DA) [8], Canonic Sign Digit (CSD) [9], and Modified Processing Element [10]. The new compact integrated chip designed using FPPE is an important unit for addition, subtraction, multiplication; for ALU; and for the binary data which are used in the computational processes. The architecture uses the new concept of Cross Folded Shifting, for the ALU unit.

A DSP system always produces an output $y(n)$ at every time instants. The transformed output may have the effect of noise and can be degraded. The resources used in the transformed system are adder, register, and multiplexers in which the functional block adder makes severe possibilities of the degradation. In this approach, the adder block is therefore reduced.

3.1 | FPPE multiplier

A single-precision floating-point number inhabits 32 bits; so, there is a concern about the size of the mantissa and the size of the exponent. These chosen sizes provide a range of roughly $\pm 10^{-38}$ to $10^{38}$.

Figure 1 shows the steps to multiply two floating point numbers [13].

3.2 | Folding

Folding is a transformation technique used in DSP architecture implementation for minimizing the number of functional blocks. Here, the new concept of Cross Folded Shifting design is applied for implementing the processing unit [14].

Figures 2 and 3 explain the process of folding and folding with a factor of 2, respectively. A system produces $y(n)$ at each unit time. The transformed system yields $y(n)$ in each $2^l$ where each $2^l$ increase $1n$, index of $y$. The resources used in Figure 2
are adders, and it is transformed in Figure 3 with one adder, one register, and three multiplexers. The functional block, adder, is therefore reduced. Normally, ALUs can shift the operand by 1-bit position, whereas the more complex ALUs that use barrel shifters, allow the operand to be shifted by an arbitrary number of bits. For every single shift operation, the bit shifted out of the operand appears on to be carried out and the value of the bit shifted into the operand depends on the type of the shift.

In any digital signal processing system, there may occur error in calculating the filter value while handling the FP arithmetic. To overcome this, the system requires changing of its architecture. But it upturns the design complexity and the time delay effect. The design of architecture must consider vast reduction in power loss and in area.

Here, the work is benefited by the new concept of Cross-Folded Shifting design for the ALU unit. The usage of FFs and LUTs in the proposed architecture got reduced. Thus the power consumption is reduced due to the reduction in the number of the components. This design is synthesised in VERILOG language and implemented in the FPGA.

4 | ARCHITECTURE

The FPPE present in the architecture fasten up the process which yield the desired outcomes and the process can be implemented as in the flowchart given in Figure 4.

The register which yields the logical output holds the combination of blocks at each stage. The overpass of bit sequence from input to output will optimise the delay block due to this link formation in register.

The architecture uses minimum number of logical components and registers. The shifter and adder logical...
architecture structure also considerably reduce the accumulator size, the FFs and the Look up tables. The article presents a multiplier free FIR filter which is aimed by the adder and shifter architecture followed with the FP enabled accumulator.

The FPPE in the block diagram which reduces the number of multiplication and accumulation process which in turn reduces the logic blocks.

Figure 4 explains the working of the FPPE block. The logic blocks consist of register blocks for data storage, and other arithmetic and logic operations. Figure 5 shows the MATLAB implementation model of the proposed approach.

5 | RESULTS AND DISCUSSIONS

An optimal FIR filter architecture using Xilinx Verilog is designed and synthesised. The programming is done in Verilog for a Spartan 6-100 T FPGA as target device. In this design, the number of logic gates, registers and counters have been designed in such a way that the architecture must be with lowest number of gates. This causes significant reduction in area and number of FFs.

In the LUT device utilization summary, it has low level of LUT and FFs, which is taken as the amount of the logic blocks in terms of area. From the LUT device utilization summary, it is noted that the FPPE method has considerably less power of simulation when compared to other methods. Here the Look up Tables, FFs and IOB’s has less power and hence the total power is reduced. From the Maximum time display in the synthesis report, it can be said that the speed of the process is increased, and the total time executed will be reduced.

The performance analysis and comparison of the design with previous architectures are shown in Table 1.

| Architecture                          | Delay (ns) | Area (logic blocks) | Power (mW) |
|---------------------------------------|------------|---------------------|------------|
| Basic                                 | 12.697     | 204                 | 8.14       |
| Sequential distributed arithmetic [20]| 6.418      | 123                 | 8.71       |
| Parallel distributed arithmetic       | 3.686      | 245                 | 8.14       |
| Canonic sign digit [21]               | 6.312      | 123                 | 15.2       |
| Modified PE [23]                      | 4.232      | 112                 | 5.07       |
| Proposed (FPPE)                       | 3.487      | 82                  | 6.526      |

Abbreviation: FPPE, floating point processing element.

Table 1

5 | RESULTS AND DISCUSSIONS

An optimal FIR filter architecture using Xilinx Verilog is designed and synthesised. The programming is done in Verilog for a Spartan 6-100 T FPGA as target device. In this design, the number of logic gates, registers and counters have been designed in such a way that the architecture must be with lowest number of gates. This causes significant reduction in area and number of FFs.

In the LUT device utilization summary, it has low level of LUT and FFs, which is taken as the amount of the logic blocks in terms of area. From the LUT device utilization summary, it is noted that the FPPE method has considerably less power of simulation when compared to other methods. Here the Look up Tables, FFs and IOB’s has less power and hence the total power is reduced. From the Maximum time display in the synthesis report, it can be said that the speed of the process is increased, and the total time executed will be reduced.

The performance analysis and comparison of the design with previous architectures are shown in Table 1.

The various design approaches for the design of FIR filters are compared here. The plots are shown below portray the comparison study on these filter structures with the proposed design. In Figure 6a, the area has been reduced considerably in the proposed FPPE design. CSD and MPE have similar reduction in area, but they are inferior while considering the power and delay comparisons. Thus, for FPPE architecture the delay is considerably reduced and can be used for high-speed applications.

Figure 6b portrays that the power consumption is reduced, but the MPE (modified processing element) method is better in this case. The other metrics such as area and delay are better for the proposed method compared to other methods.
Minimum usage of logical components reduces the amount of register usage. The change in the structure of shifter and adder logical architecture will reduce the accumulator size and the FFs and LUTs. This will improve the speed of the process and thus reduces the delay as in Figure 6c.

In the proposed FPPE cross folding architecture, while comparing the performance matrices, it can be noted that the area, power, and delay factors are optimized. Hence, a multiplier free FIR filter is presented here which is done by using the adder and shifter followed with an accumulator to establish the FP Processing.

6 | CONCLUSION

The difficulties in VLSI architectures such as delay and power consumption limit the selection of application areas. The issue encountered in the VLSI architecture of filter design is the increased number of components. For the VLSI architecture reconfigured with reduced register usage, this article provided the FPPE implementation. However, the FIR filter system has a large amount of delay components in the circuit, which increased the complexity and high delay rate in the logical operation. This article proposed an FPPE architecture based upon cross folded shifting help to get the efficient filter structure in accordance with speed, power and area. Future work will include extensive analysis of the FP units to identify more design trade-offs.

ORCID
Tintu Mary John ✉ https://orcid.org/0000-0003-2191-2597

REFERENCES
1. Antonion, A.: Digital Filters: Analysis, Design and Applications. McGraw-Hill, New York (1993)
2. Rafati, R., Fakhraie, S.M., Smith, K.C.: A 16-bit barrel-shifter implemented in data-driven dynamic logic. IEEE Trans. Circuits Syst. I Reg. Papers. 53, 2194–2202 (2006)
3. Xydis, S., Economakos, G., Pekmestzi, K.: Designing coarse-grain reconfigurable architectures by inlining flexibility into custom arithmetic data-paths. Integr. VLSI J. 42, 486–503 (2009)
4. Mohammad, K., Agaian, S., Hudson, F.: Implementation of digital electronic arithmetic and its application in image processing. Comput. Electr. Eng. 36, 424–434 (2010)
5. Huang, L., et al.: A new architecture for multiple-precision floating-point multiply-accumulate unit design. In: 18th IEEE Symposium on Computer Arithmetic, pp. 69–76 (2007)
6. Purohit, S., Margala, M.: Investigating the impact of logic and circuit implementation on full adder performance. IEEE Trans. Very Large Scale Integr. Syst. 20, 1327–1331 (2012)
7. Gierenz, V., Panis, C., Nurmi, J.: Parameterized MAC unit generation for a scalable embedded DSP core. Microprocess. Microsyst. 34, 138–150 (2010)
8. Mohanty, B.K., et al.: A high performance VLSI architecture for reconfigurable FIR filters. Integr. VLSI J. 34, 37–46 (2006)
9. Rajolia, A., Kaur, M.: Finite impulse response(FIR) filter design using canonical signed digit (CSD) Indian J. Sci. Res. 2(7) (2013)
10. Reddy, K.S., et al.: A modified approach for reconfigurable FIR filter architecture. In: IEEE Region 10 Conference TENCON 2014, Bangkok, Thailand, pp. 1–5 (2014)
11. Nagalexmi, K., Vasu Naik, B.: Design and implementation floating point multiplier design using combined booth and dadda algorithms. Int. J. VLSI Syst. Commun. Syst. 03(04) (2015)
12. Sudhakar, V., Murthy, N.S., Anjaneyulu, L.: Area Efficient Pipelined Architecture for Realization of FIR filter using Distributed Arithmetic. vol. 31, IPCSIT Press, Singapore
13. Sangwan, D., Yadav, M.K.: Design and implementation of adder/subtractor and multiplication units for floating point arithmetic. Int. J. Electron. Eng. 197–203 (2010)
14. Parhi, Keshab K., Wang, C.Y, Brown, A.P.: Synthesis of control circuits in folded pipelined DSP architectures. IEEE Journal of Solid-State Circuits. 27(1), 29–43 (1992)
15. Mohanty, B.K., Kumar Meher, P.: A high-performance FIR filter architecture for fixed and reconfigurable applications. IEEE Trans. Very Large Scale Integr. Syst. 24(2) (2016)
16. Sharma, A., Kumar, S.: VLSI implementation of pipelined FIR filter. Int. J. Innov. Res. Electric. Electron. Instr. Control Eng. 1(3) (2013)
17. Thamizharasan, V., Parthipan, V.: An efficient VLSI architecture for FIR filter using computation sharing multiplier. Int. J. Comput. Appl. 54(14) (2012)
18. Yagain, D., Krishna, A.V.: FIR filter design based on retiming & automation using VLSI design metrics. International Conference on Technology, Informatics, Management Engineering & Environment (2013)
19. Cheng, K.H., Sahni, S.: VLSI architectures for the finite impulse response filter. IEEE J. Sel. Area. Commun. 20(5) (2016)
20. Nair, P.P., Mary John, T.: Optimized FIR filter using distributed parallel architectures for audio application. Int. J. Comput. Technol. Appl. 8(3) (2017)
21. Yamada, M., Nishihara, A.: Design of FIR digital filters with CSD coefficients having power-of-two DC gain and their FPGA implementation for minimum critical path. IEICE Trans. Fundam. Electron. Commun. Comput. Comput. 84(8), 1997–2003 (2001)
22. Gupta, S., Kurmi, U.: A review – design of area and power efficient digital FIR filter based on faithfully rounded truncated 12-bit constant. Int. J. Comput. Appl. 149(6), (2016)
23. Agarwal, D., Reddy, K.S., Sahoo, S.K.: FIR filter design approach for reduced hardware with order optimization and coefficient quantization. In: International Conference Intelligent Systems and Signal Processing, (ISSP), Gujarat, India, pp. 293–296 (2013)

How to cite this article: John TM, Chacko S. FPGA-based implementation of floating point processing element for the design of efficient FIR filters. IET Comput. Digit. Tech. 2021;15:296–301. https://doi.org/10.1049/cdt.2.12010