Design And Implementation of Parallel Processing Fir Filter Using Modified Booth Encoding

Dr. S. Selvakumar Raja[1], Mr. M. Mahipal[2]

Abstract: This project presents a Parallel Processing FIR filter or Full Parallel FIR Filter design implemented using the Modified Booth Encoder (MBE) to achieve high-speed performance. A novel hardware architecture employing fine-grained seamless pipelining is proposed, where pipeline registers are strategically placed not only between components but also across them. This ensures minimal gate delays and maximizes throughput. A precise critical path analysis at the gate level enables an optimal pipelining strategy tailored to throughput requirements. The results show that the fully-parallel FIR filter achieves very high throughput with a substantial reduction in area delay product (ADP) compared to existing systolic designs. The proposed design optimizes MBE encoding and parallel processing to achieve significant enhancements in speed, area, and power efficiency. FIR filters are crucial components in digital signal processing applications, but traditional designs often face limitations. The proposed design employs a pipelined architecture, leveraging MBE encoding to reduce computational complexity. This research investigates the potential of parallel processing and MBE encoding in FIR filter design analysing trade-offs between speed, area, and power consumption.

Keywords: consumption, proposed, complexity, requirements

1.Introduction:

The architecture implemented in this project is based on an 8-tap FIR filter with fixed coefficients. The structure is built using structural Verilog, ensuring modularity and clarity in design. Each input sample is passed through a series of delay registers to align with corresponding coefficients, and all tap operations are executed concurrently through dedicated MBE-based multiplier modules. This design is not only practical for FPGA and ASIC synthesis but also aligns well with current industry needs for high-speed DSP cores. Timing constraints are carefully managed through Static Timing Analysis (STA) using proper Synopsys Design Constraints (SDC), ensuring that the design operates reliably at a clock frequency of 100 MHz or higher. The final

Principal & Professor ECE Department[1], HOD, ECE Department[2]

Kakatiya Institute of technology and science for women, Manikbhandar, Nizamabad-503003, Telangana State, India.

design can serve as a core module in a variety of applications requiring fast convolution operations, such as digital receivers, real-time audio filters, or image enhancement pipelines. By combining the strengths of full-parallel FIR filtering and MBE-based multiplication, this project presents a robust solution to the growing demand for high-speed, low-latency digital filter implementations in modern electronics. In modern digital signal processing applications, Finite Impulse Response (FIR) filters are a foundational component due to their inherent advantages such as guaranteed stability and linear phase response. These characteristics make FIR filters indispensable in systems where signal fidelity, predictability, and design simplicity are crucial, such as wireless communication, image processing, biomedical signal analysis, and audio enhancement. However, as data rates and real- time processing requirements continue to increase in complexity and scale, traditional FIR filter architectures often struggle to meet high-speed performance demands. This project proposes a

high-speed, full-parallel FIR filter design that incorporates Modified Booth Encoding (MBE) to improve multiplication efficiency. Unlike serial or partially parallel implementations, the full-parallel architecture performs all coefficient multiplications simultaneously, significantly increasing the processing speed. The MBE technique further enhances performance by reducing the number of partial products generated during multiplication, thereby minimizing the overall logic depth and critical path delay. Modified Booth Encoding is especially effective when implemented in hardware because it compresses the partial products by encoding the multiplier bits into overlapping groups of three, which leads to faster and more area- efficient multiplication circuits. Integrating this technique into the FIR filter architecture enables the design to handle high-frequency inputs without compromising output accuracy or increasing the silicon footprint significantly.

2.Existing System

The existing FIR (Finite Impulse Response) filter architectures primarily rely on conventional multiplier-based designs, often implemented using straightforward parallel or serial computation techniques. These filters operate based on the convolution sum where the current output is derived from the weighted sum of current and past input samples. Typically, the multiplication operation is the most resource-intensive and time-consuming component of the FIR filter, especially when implemented on FPGA platforms where logic and delay resources are constrained. In traditional implementations, FIR filters are designed using multiply-accumulate (MAC) units that process input samples and filter coefficients sequentially or in a semi-parallel manner. While these structures are easy to implement and understand, they often suffer from limitations in terms of speed and hardware utilization. For instance, serial architectures result in longer processing delays due to the sequential

handling of inputs and coefficients, which is not ideal for high-speed or real-time applications. On the other hand, fully parallel architectures - though faster -incur higher hardware costs and complexity due to the duplication of multipliers and adders. Moreover, the multipliers used in conventional FIR filters typically rely on basic shiftand - add techniques or array-based multipliers, which are not optimized for speed or area. These approaches lead to increased critical path delays, directly affecting the overall throughput of the system. In high-order FIR filters with many taps, this becomes a significant bottleneck, making it difficult to meet stringent timing requirements in high-frequency applications. To improve performance, some existing systems have adopted pipelining and retiming techniques to shorten the critical path and enhance clock frequency. However, these methods often add latency and increase the control logic complexity, making them less suitable for latency- sensitive applications. Additionally, while pipelining helps to maintain high throughput, it does not address the core inefficiency of the multiplier unit. In terms of optimization techniques, the Modified Booth Encoding (MBE) algorithm has not been widely adopted in conventional FIR filter architectures despite its known advantages in reducing the number of partial products in multiplication. Most FIR filters still rely on traditional binary multiplication, which generates a large number of partial products, thereby increasing the power consumption, area, and delay. Overall, the existing FIR filter systems provide a foundation for signal processing but lack the advanced architectural and arithmetic optimizations needed for high-speed and efficient FPGA implementation. They fall short in exploiting modern parallel processing techniques and multiplier enhancements such as MBE, which can significantly reduce computational complexity and critical path delay. This limitation presents an opportunity to develop an improved FIR filter

3. Proposed System

Full Parallel FIR filter with Modified Booth Algorithm:

The proposed system introduces a Full Parallel FIR Filter architecture integrated with the Modified Booth Encoding (MBE) algorithm to significantly enhance the speed, efficiency, and performance of FIR filtering operations, particularly in FPGA and high-speed digital signal processing (DSP) applications. Traditional FIR filters are limited by the critical path delay introduced by sequential multiply-accumulate operations, especially when implemented with basic multiplier structures. To overcome this, the proposed methodology adopts a fully parallel structure, where each tap of the FIR filter performs its multiplication and addition operations independently and concurrently. This results in one output per clock cycle, greatly increasing throughput. The core innovation lies in the use of the Modified Booth Multiplier for each multiplication operation within the filter taps. The MBE technique reduces the number of partial products by encoding the multiplier operand, enabling the generation of fewer intermediate results. For example, compared to traditional binary multiplication, MBE reduces the partial products nearly by half, thus minimizing the delay and logic required in the partial product generation and addition stages. Each filter coefficient is multiplied with its corresponding delayed input sample using this optimized multiplication technique.

To implement this, the system uses a delay line structure of registers to store past input samples. These delayed samples are then fed into parallel multiplier blocks, each utilizing MBE logic to carry out fast multiplication with the pre-defined filter coefficients. The filter coefficients remain constant, allowing further optimization of the multiplier blocks, particularly in FPGA implementation where multipliers with constant operands can be hardwired efficiently.

Each MBE-based multiplier produces partial products that are structured using Partial Product Generators (PPG) and are passed to a Reduction Tree designed with full adders and half adders. This reduction stage uses techniques such as Wallace tree-like structures or Carry-Save Adders (CSA) to sum the partial products quickly, minimizing the propagation delay usually associated with long addition chains.
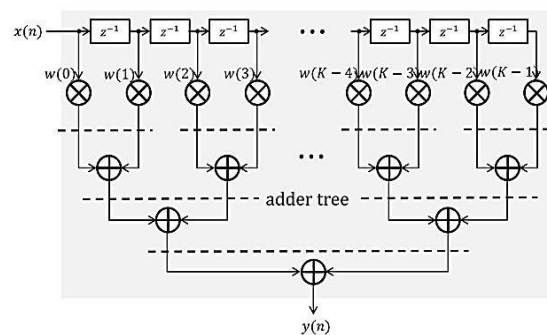

Fig:1 FIR Filter Design

The final outputs from all the multiplier units are then summed up using a high-speed adder tree to produce the final filtered output. The use of MBE not only accelerates multiplication but also significantly reduces the area and power consumption, as fewer logic elements are toggled during each operation.

In this architecture, no pipelining or retiming is used, ensuring minimal latency. This is particularly useful in applications where both high-speed computation and low-latency output are required. The architecture also simplifies timing analysis, as the fully parallel nature ensures uniform delays across data paths.

3.1 Modified Booth Encoding (MBE)

The Modified Booth Encoding (MBE)

algorithm is a well-established technique used to optimize the multiplication process in digital circuits, especially in high-speed and low-power applications like FIR filters. In conventional binary multiplication, each bit of the multiplier is used to generate a partial product, resulting in as many partial products as there are bits in the multiplier. This leads to increased computation time and power consumption, especially for wide- bit operations.

MBE addresses this inefficiency by grouping bits of the multiplier in overlapping sets of three, allowing the algorithm to analyse more information at once and reduce the number of partial products by nearly half. The encoding rule is based on radix-4 Booth recoding, which interprets the multiplier bits in groups and maps them to values ranging from -2 to +2. This drastically reduces the number of additions or subtractions needed during multiplication.

The table below shows the Modified Booth encoding logic based on a 3-bit window:

Multiplier Bits (y_{i+1}, y_i, y_{i-1})	Operation
000	0
001	+1 × multiplicand
010	+1 × multiplicand
011	+2 × multiplicand
	-2 × multiplicand
	-1 × multiplicand
	-1 × multiplicand
	0

Encoding to obtain m=2 partial products using m-bit MBE

THE KEY ADVANTAGES OF USING MBE IN FIR FILTER DESIGN ARE:

Reduction in Partial Products: With fewer partial products, the circuit requires fewer additions, leading to faster computation and lower switching activity.

Speed Enhancement: Since fewer operations are performed in parallel, the propagation delay through the multiplier is reduced.

Area Efficiency: The logic needed to handle fewer partial products is compact, saving valuable area in FPGA implementations.

Power Efficiency: Reduced switching and shorter critical paths lower dynamic power consumption.

In the context of the FIR filter design, each tap's multiplication is implemented using the MBE technique. The MBE module takes the delayed input sample and a fixed coefficient (tap weight) and produces a set of partial products efficiently. These are then fed into an adder tree (using full adders and half adders) for final summation. Moreover, since the filter coefficients are constant and known beforehand, further optimizations can be applied to the MBE logic - like pre-computing some partial results or simplifying hardware for specific coefficient values - resulting in even greater efficiency in a Full Parallel FIR architecture. Thus, the use of MBE in FIR filtering is crucial for achieving the design goal: a high-speed, low-area, and power-efficient filter architecture suitable for real-time digital signal processing on FPGA platform
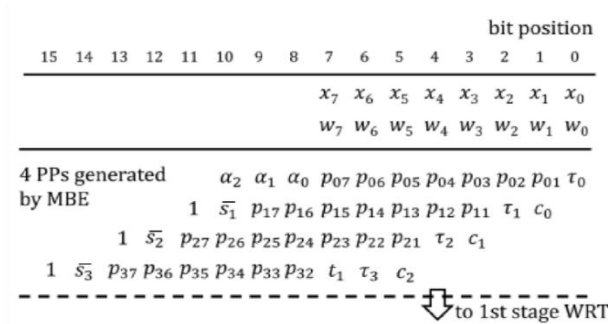
Fig: 2 Four partial products generated by an 8-bit MBE

4.2 Full Parallel Fir Filter:

The proposed architecture of the Full Parallel FIR Filter leverages the principles of parallelism and Modified Booth Encoding (MBE) to significantly enhance speed and throughput. This section explains the working of the proposed FIR filter by referencing Figures 2, 4, and 5 of the original paper, which collectively illustrate the dataflow, multiplication mechanism, and optimized partial product accumulation strategy.

This figure represents the top-level design of the full parallel FIR filter. Unlike traditional or pipelined FIR filters that process one tap per clock cycle or use time-sharing logic, this architecture performs all tap multiplications simultaneously in parallel. Each input sample is passed through a series of delay registers (flip-flops) to generate a sequence of delayed input values corresponding to the number of filter taps. In the case of an 8-tap FIR filter, 7 delay elements are used to hold previous inputs. Each of these delayed values is then multiplied concurrently with a fixed filter coefficient using dedicated multiplier blocks.

To boost performance, the multipliers in this architecture are designed using the Modified Booth Encoder, reducing the number of partial products and thus the critical path delay. After multiplication, all the products are fed into an adder tree structure for accumulation. This highly parallel dataflow ensures minimal latency and supports high-speed filtering required for real-time signal processing
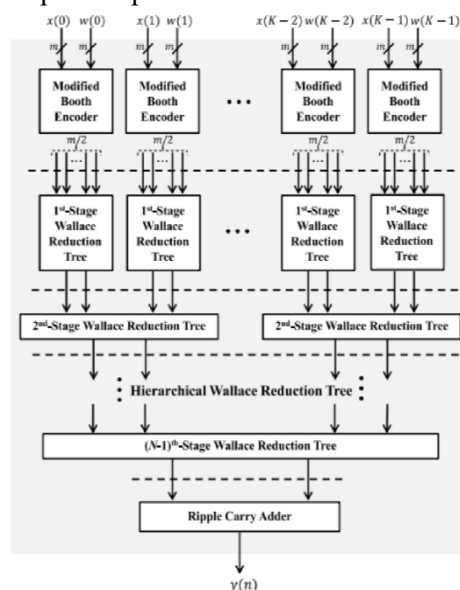

Fig 3 Structure of *K*-tap reference full-parallel FIR filter
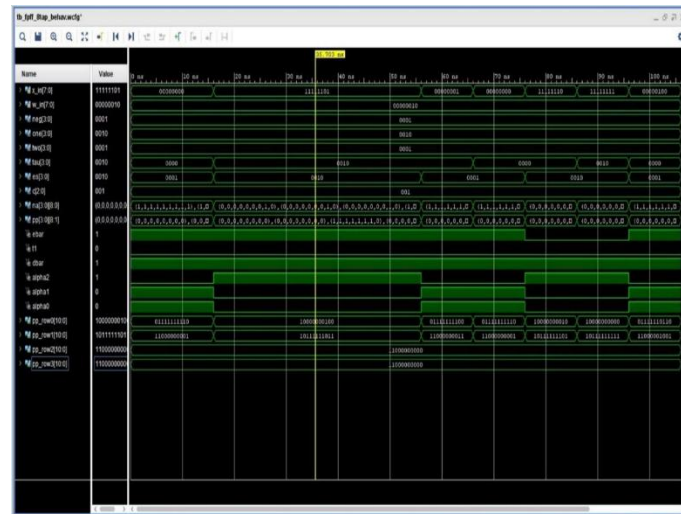
5.Simulation Results
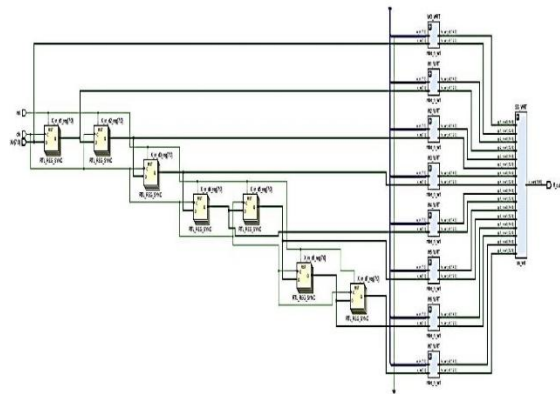

Fig 4 Simulation Results Modified Booth Encoder

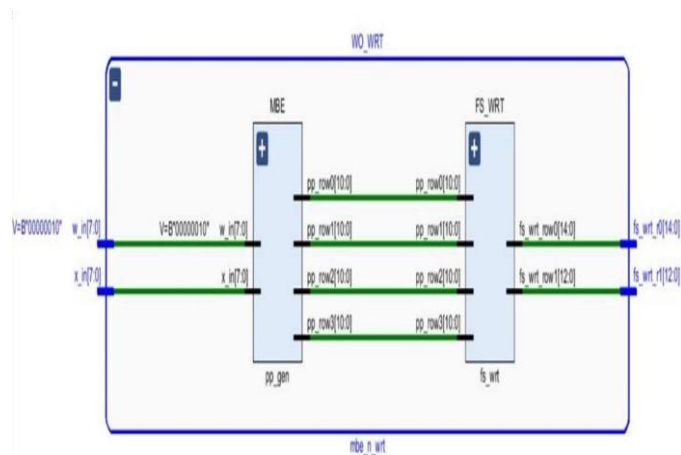Fig 5 Schematic Full Parallel FIR Filter

Fig 6 Schematic MBE and WRT

6.Applications and Advantages

Advantages

The proposed FIR filter design, built on a fully parallel processing structure and enhanced by Modified Booth Encoding (MBE), offers significant improvements in speed, area, and performance efficiency over conventional FIR filter architectures. By integrating MBE for multiplication and optimizing data flow through parallelism, the design delivers high throughput suitable for real-time digital signal processing on FPGA platforms.

High-Speed Computation

The Modified Booth Encoding (MBE) technique reduces the number of partial products in multiplication, thereby accelerating the overall computation time. Full parallelism enables simultaneous execution of all multiplication operations across multiple taps of the FIR filter, reducing latency and improving response time. The use of high-speed partial product reduction techniques, such as Wallace tree and ripple-carry adders, minimizes critical path delays'-based implementation allows dedicated hardware resources for each tap, facilitating real- time processing.

Reduced Power Consumption

The reduced partial product count from MBE leads to lower switching activity and hence minimizes dynamic power consumption. Full parallelism removes the need for clocked pipelining stages, which can contribute to clock tree power, further improving power efficiency, Optimized logic design for multiplication and addition avoids unnecessary logic transitions, reducing overall power usage.

Applications

The proposed Parallel Processing FIR Filter Design with MBE (Modified Booth Encoder) Implementation is well-suited for a wide range of real-time digital signal processing applications. Its high-speed, power-efficient, and resource-optimized design makes it ideal for FPGA-based implementations in critical domains such as communications, biomedical engineering, artificial intelligence, and embedded systems.

Wireless Communication Systems

Used in digital baseband processing for filtering signals in LTE, 5G, and Wi-Fi communication systems.

Enhances signal clarity by removing noise and unwanted frequency components in radio receivers and transmitters. Facilitates real-time modulation and demodulation processes in software-defined radios (SDR).

Biomedical Signal Processing

Applied in electrocardiogram (ECG) and electroencephalogram (EEG) processing to filter out high-frequency noise and artifacts. Improves medical imaging techniques, such as MRI and ultrasound, by enhancing signal-to-noise ratio (SNR).

Conclusion

In this project, a high-speed, full parallel FIR filter architecture utilizing the Modified Booth Encoder (MBE) algorithm was designed and implemented to meet the increasing demands for efficient digital signal processing in real-time applications. Traditional FIR filter implementations often suffer from performance bottlenecks due to the sequential nature of multiplication and accumulation operations. By incorporating the Modified Booth Encoding technique, the proposed design significantly reduces the number of partial products and optimizes the multiplication process, thereby enhancing overall speed and computational efficiency. The proposed architecture is based on a fully parallel structure, eliminating the need for pipelining or retiming techniques and instead relying on careful critical path optimization. This results in improved throughput without compromising the accuracy or stability of the filtering process. Furthermore, the design leverages modular

components such as full adders, half adders, and ripple-carry adders integrated into the partial product summation logic, ensuring systematic and scalable hardware implementation

Compared to conventional multiplier-based FIR filters, the proposed design offers substantial advantages in terms of computation speed, reduced latency, and better resource utilization on FPGA platforms. The Modified Booth Encoder plays a central role in accelerating the multiplication operations, making the FIR filter highly suitable for applications in wireless communication, audio processing, biomedical instrumentation, and AI-enabled embedded systems. Through the integration of parallel processing and MBE techniques, the design not only achieves high performance but also supports scalability for higher-order filters. Simulation and synthesis results demonstrate that the proposed FIR filter achieves its design goals of low delay, high throughput, and efficient resource usage, thereby proving its applicability in modern, high- performance digital systems.

This work lays a strong foundation for future enhancements, including low-power design optimization, reconfigurable filter architectures, and integration with adaptive filtering techniques. Overall, the project successfully delivers a robust and efficient FIR filter architecture tailored for high-speed digital signal processing using modern FPGA technology.

8.References

[1] S. Majhi, A. Dandapat, and R. Mahapatra, "Design of Very High-Speed Pipeline FIR Filter Through Precise Critical Path Analysis," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 2, pp. 346-349, Feb. 2014.

[2] M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*, 2nd ed., Prentice Hall, 2003.

[3] D. A. Patterson and J. L. Hennessy, *Computer Organization and Design: The Hardware/Software Interface*, 5th ed., Morgan Kaufmann, 2014.

[4] H. T. Bui, Y. Wang, and Y. Jiang, "Design and analysis of low-power 10-transistor full adders using novel XOR–XNOR gates," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 49, no. 1, pp. 25–30, Jan. 2002.

[5] K. K. Parhi, *VLSI Digital Signal Processing Systems: Design and Implementation*, John Wiley & Sons, 1999.

[6] S. Lin and D. Costello, *Error Control Coding*, 2nd ed., Pearson Prentice Hall, 2004.

[7] B. W. Parkinson, "A Review of Booth's Algorithm and Its Application in Computer Architecture," *Computer Design and Applications Journal*, vol. 5, pp. 20–27, 2011.

[8] C. H. Roth and L. L. Kinney, *Fundamentals of Logic Design*, 7th ed., Cengage Learning, 2013.

[9] X. Liang, R. F. Demara, "High-throughput and Low-power Multipliers Using Dynamic Operand Gating," *IEEE Transactions on VLSI Systems*, vol. 16, no. 3, pp. 312–321, March 2008.

[10] Z. Huang and M. Ercegovac, "High-Performance Low-Power Left-to-Right Array Multiplier Design," *IEEE Transactions on Computers*, vol. 54, no. 3, pp. 272–283, March 2005.