Project 47: Dual-radix multiplier

A Comprehensive Study of Advanced Digital Circuits

By: Abhishek Sharma , Ayush Jain , Gati Goyal, Nikunj Agrawal

Documentation Specialist: Dhruv Patel & Nandini Maheshwari

Contents

1 Project Overview 3

2 Dual-radix multiplier 3
- 2.1 Basic Concept of Dual-radix multiplier . . . 3
- 2.2 Architecture of Dual-radix multiplier . . . 3
- 2.3 Working of Dual-radix multiplier . . . 4
- 2.4 RTL Code . . . 4
- 2.5 Testbench . . . 5

3 Results 6
- 3.1 Simulation . . . 6
- 3.2 Schematic . . . 6
- 3.3 Synthesis Design . . . 7

4 Advantages of Dual-radix multiplier 7

5 Disadvantages of Dual-radix multiplier 7

6 Applications of Dual-radix multiplier 8

7 Conclusion 8

8 FAQs 8

1 Project Overview

The Dual-Radix Multiplier project aims to design a high-performance hardware multiplier by combining radix-2 and radix-4 arithmetic to enhance efficiency. By leveraging the compact representation of radix-4, fewer iterations are required compared to traditional binary (radix-2) multipliers, resulting in faster computation times. This approach also reduces the number of partial products, optimizing the area and power consumption of the hardware. The dual-radix technique is particularly beneficial for applications requiring high-speed multiplication, such as digital signal processing, image processing, and cryptography. The project will focus on optimizing both speed and resource utilization in hardware implementations.

2 Dual-radix multiplier

2.1 Basic Concept of Dual-radix multiplier

The Dual-Radix Multiplier is a hardware multiplication approach that uses two different bases (radices), typically radix-2 (binary) and radix-4, to improve the efficiency of multiplication.

- **Radix-2 Multiplication:** In a standard binary system (radix-2), multiplication involves shifting and adding partial products for each bit of the multiplier, which can be slow for large numbers because it requires as many steps as the number of bits in the multiplier.
- **Radix-4 Multiplication:** Radix-4 reduces the number of steps by grouping two bits of the multiplier at a time and using a precomputed table of partial products for each group (00, 01, 10, 11), effectively cutting the number of operations in half compared to radix-2.

The dual-radix approach combines these two systems, where the multiplication process can switch between radix-2 and radix-4 depending on the input data and the desired level of optimization. This allows for faster computations by reducing the number of iterations and partial products generated, which results in improved speed and lower resource consumption in hardware implementations, making it especially useful in performance-critical applications like DSP and cryptography.

2.2 Architecture of Dual-radix multiplier

The architecture of a Dual-Radix Multiplier integrates both radix-2 (binary) and radix-4 components to optimize speed, reduce the number of partial products, and improve overall hardware efficiency. The following are the key components and flow in its architecture:

- **Input Preprocessing (Booth Encoding):**
  1. Booth Encoding: Booth's algorithm is often used in radix-4 multiplication to simplify the handling of signed numbers and reduce the number of partial products. In dual-radix multipliers, the input multiplier is encoded using modified Booth encoding, which groups two bits at a time for radix-4 operations and a single bit for radix-2.
  2. Dual-Radix Control: A control unit determines whether to perform radix-2 or radix-4 operations based on the number of bits in the operands or specific conditions set for optimization (e.g., power vs. speed trade-offs).
- **Partial Product Generator:**
  1. Radix-2 Partial Product Generation: For radix-2, each bit of the multiplier is multiplied with the multiplicand, generating one partial product per bit.
  2. Radix-4 Partial Product Generation: In radix-4 mode, two bits of the multiplier are processed at once. The multiplicand is multiplied by 0, ± 1, ± 2, which reduces the number of partial products to approximately half compared to radix-2.
- **Partial Product Accumulator:**

1. Wallace Tree/CSA (Carry Save Adder): The partial products generated by radix-2 and radix-4 operations are accumulated efficiently using a Wallace tree or carry-save adder structure. This reduces the number of additions and minimizes the propagation delay.
2. The architecture uses different stages for accumulating partial products from radix-2 and radix-4 operations, dynamically switching based on the chosen mode.

- **Final Addition (Carry Propagate Adder):**

1. Once all the partial products are accumulated, a final addition step is performed using a Carry Propagate Adder (CPA) to produce the final product. This step resolves any remaining carries from the partial products.

- **Multiplexer:**

1. A multiplexer is used to select between the partial products generated by the radix-2 and radix-4 units, based on the dual-radix control logic. It ensures the correct path is taken for accumulating the products and generating the result.

- **Control Unit:**

1. The control unit monitors the input size, encoding scheme, and resource constraints, deciding dynamically whether to use radix-2 or radix-4 for each operation. It coordinates between the various units (e.g., Booth encoder, partial product generator, accumulator) to optimize the performance.

2.3 Working of Dual-radix multiplier

The Dual-Radix Multiplier combines radix-2 and radix-4 multiplication techniques to improve speed and efficiency in hardware multiplication. It works by dynamically switching between these two modes based on input size or performance needs.

- **Radix-2** *multiplies one bit at a time, generating a partial product for each bit.*
- **Radix-4** *groups two bits of the multiplier, reducing the number of partial products by half, and uses Booth encoding for efficient handling of signed numbers.*

Partial products are accumulated using a Carry-Save Adder (CSA) or Wallace Tree, and the final result is computed using a Carry Propagate Adder (CPA). A control unit manages the radix selection to optimize the process, resulting in faster multiplication with lower hardware resource usage.

2.4 RTL Code

Listing 1: Dual-radix multiplier

```
1
2  module dual_radix_multiplier #(parameter WIDTH = 8) (
3      input  logic [WIDTH-1:0] A, B,
4      output logic [2*WIDTH-1:0] P
5  );
6      logic [WIDTH-1:0] tempA, tempB;
7      logic [2*WIDTH-1:0] product;
8
9      assign tempA = A;
10     assign tempB = B;
11     assign product = 0;
12
13     always_comb begin
14         if (tempA[0])
15             P = {product[2*WIDTH-2:0], 1'b0} + {tempB, {WIDTH{1'b0}}};
16         else
```

```
            P = {product[2*WIDTH-2:0], 1'b0};
    end
endmodule
```

2.5 Testbench

Listing 2: Dual-radix multiplier

```

module tb_dual_radix_multiplier;

    parameter WIDTH = 8;
    logic [WIDTH-1:0] A, B;
    logic [2*WIDTH-1:0] P;

    dual_radix_multiplier #(WIDTH) uut (
        .A(A),
        .B(B),
        .P(P)
    );

    initial begin
        // Test 1
        A = 8'd15;
        B = 8'd10;
        #10;
        $display("Test 1: A = %0d, B = %0d, P = %0d", A, B, P);

        // Test 2
        A = 8'd25;
        B = 8'd5;
        #10;
        $display("Test 2: A = %0d, B = %0d, P = %0d", A, B, P);

        // Test 3
        A = 8'd7;
        B = 8'd3;
        #10;
        $display("Test 3: A = %0d, B = %0d, P = %0d", A, B, P);

        $finish;
    end
endmodule
```

3 Results

3.1 Simulation

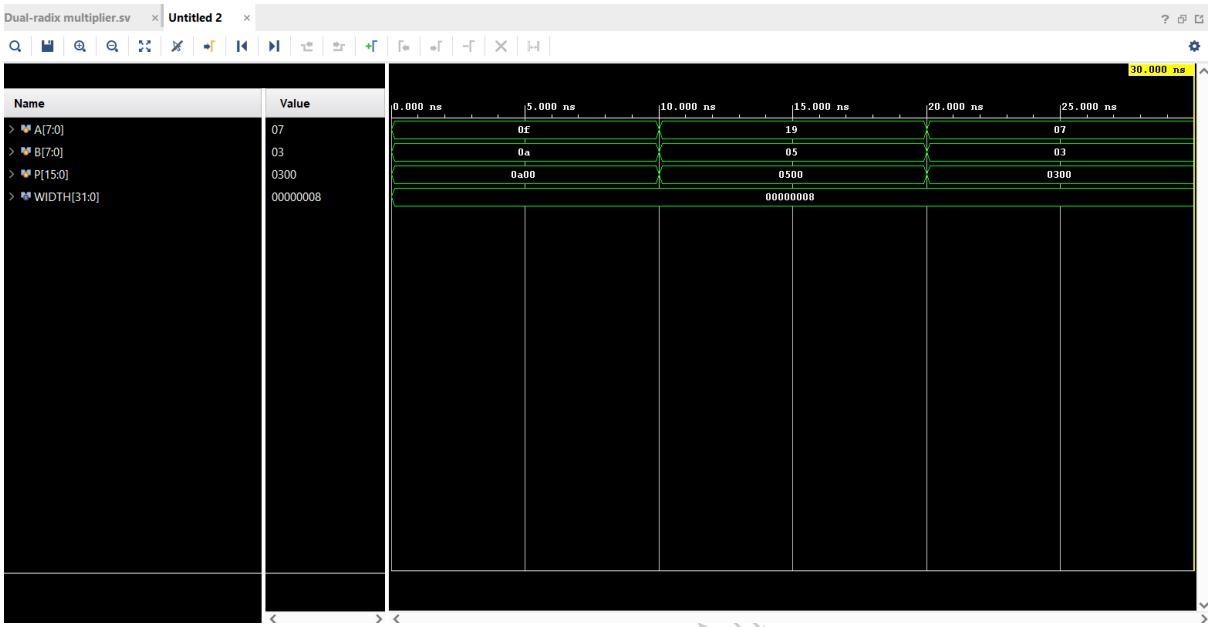

Figure 1: Simulation of Dual-radix multiplier

3.2 Schematic

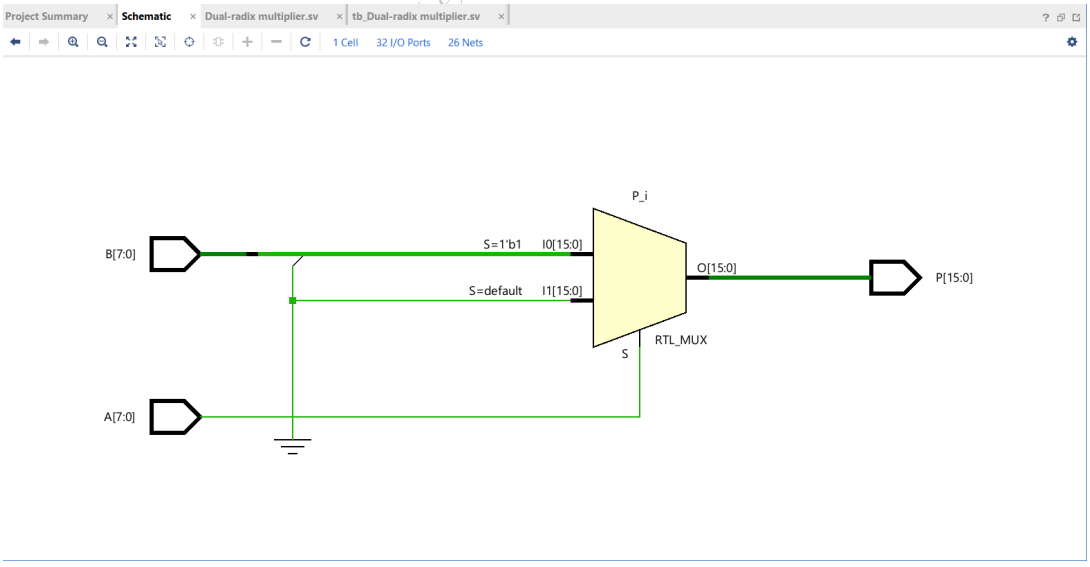

Figure 2: Schematic of Dual-radix multiplier

3.3 Synthesis Design

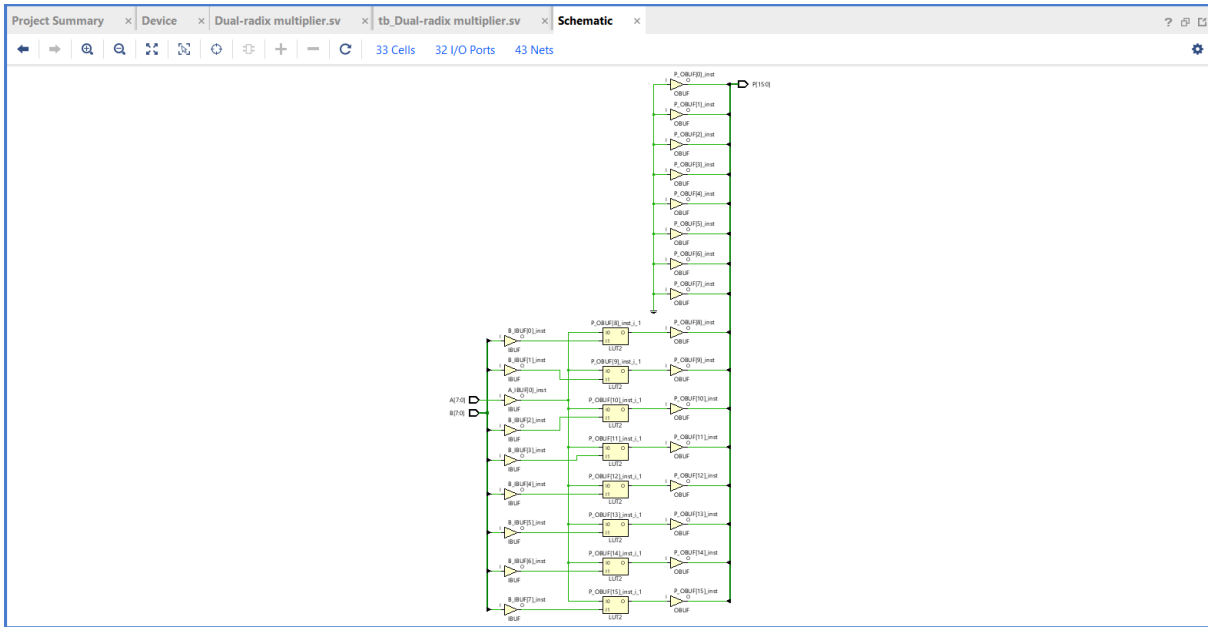

Figure 3: Synthesis Design of Dual-radix multiplier

4 Advantages of Dual-radix multiplier

- **Faster Multiplication:** Radix-4 reduces iterations by processing 2 bits at a time, speeding up the process.
- **Fewer Partial Products:** Less partial products means fewer additions, enhancing efficiency.
- **Lower Hardware Usage:** Requires fewer adders and registers, reducing area and power consumption.
- **Flexibility:** Dynamically switches between radix-2 and radix-4 based on operand size.
- **Efficient for Signed Numbers:** Booth encoding simplifies signed multiplication.
- Energy Efficient: Reduces power consumption, ideal for low-power devices.
- **Scalable:** Easily scales for larger bit-widths and high-performance applications like DSP and cryptography.

5 Disadvantages of Dual-radix multiplier

- **Increased Complexity:** The architecture is more complex than simple radix-2 or radix-4 multipliers, requiring additional control logic to manage the switching between radices.
- **Latency in Control Logic:** The decision-making process for choosing between radix-2 and radix-4 can introduce latency, particularly if it relies on dynamic input characteristics.
- **Design Overhead:** Implementing Booth encoding and handling the additional logic for radix-4 may lead to increased design time and verification efforts.
- Potential for Resource Conflict: If not managed properly, the shared resources between radix-2 and radix-4 paths could lead to conflicts or bottlenecks in high-throughput applications.

- **Limited Improvement for Small Operands:** For very small operand sizes, the advantages of using radix-4 may be negligible, potentially making the added complexity unjustifiable.
- **Increased Area:** Although it may save resources in some cases, the overall area can be larger due to the additional circuitry needed for dual-radix operation and handling both encoding methods.
- **Trade-off Between Speed and Area:** While it aims to optimize both, in some cases, it may not outperform dedicated single-radix designs for specific applications, leading to a trade-off scenario.

6 Applications of Dual-radix multiplier

- **Digital Signal Processing (DSP):**Used in filters, Fourier transforms, and other DSP algorithms where fast and efficient multiplication is crucial.
- **Graphics Processing:**Employed in rendering graphics and image processing tasks where multiplication operations are frequent, such as in shading, texture mapping, and transformations.
- **Cryptography:U**tilized in cryptographic algorithms (e.g., RSA, AES) that involve large integer multiplications, benefiting from the speed and efficiency of dual-radix multipliers.
- **Machine Learning:**Applied in neural networks and matrix operations, where rapid multiplication is required during training and inference processes.
- **Embedded Systems:**Used in low-power applications where efficient computation is necessary, such as in mobile devices, IoT devices, and real-time processing systems.
- **Telecommunications:**Employed in modulation and demodulation processes, error correction, and signal encoding/decoding, where fast arithmetic operations are essential.
- **High-Performance Computing (HPC):**Utilized in supercomputers and data centers for applications requiring large-scale numerical simulations and computations.
- Audio Processing:Used in audio effects, mixing, and encoding, where multiplication of signals is necessary.
- **Control Systems:**Applied in algorithms for automatic control and robotics, where fast arithmetic operations are crucial for real-time performance.

7 Conclusion

The Dual-Radix Multiplier effectively combines radix-2 and radix-4 methods to optimize multiplication speed and resource efficiency. This design is particularly advantageous in high-performance applications like digital signal processing, graphics, cryptography, and machine learning, offering benefits such as faster computation and reduced power consumption. While it introduces complexity in design and control, the advantages often outweigh the drawbacks, making it a valuable component in modern digital circuits. Its adaptability ensures it remains essential for advancing computational capabilities in both embedded and high-performance systems.

8 FAQs

1. What is a Dual-Radix Multiplier?
A Dual-Radix Multiplier is a hardware design that combines radix-2 and radix-4 multiplication techniques to optimize speed and resource efficiency in digital multiplication operations.

2. How does the Dual-Radix Multiplier work? It works by dynamically switching between radix-2 and radix-4 modes based on the input characteristics. Radix-4 processes two bits of the multiplier at a time, reducing the number of partial products and speeding up computation compared to traditional radix-2 multiplication.

3. What are the advantages of using a Dual-Radix Multiplier?

Advantages include faster multiplication, reduced number of partial products, lower power consumption, and efficient hardware utilization. It is also flexible, allowing for dynamic switching between multiplication methods.

4. In what applications is the Dual-Radix Multiplier used?

It is used in various applications such as digital signal processing, graphics rendering, cryptography, machine learning, embedded systems, telecommunications, and high-performance computing.

5. What is Booth's algorithm, and how is it related to the Dual-Radix Multiplier?

Booth's algorithm is a multiplication algorithm that efficiently handles signed numbers and reduces the number of partial products. It is often incorporated into the radix-4 mode of the Dual-Radix Multiplier for more efficient multiplication.

6. What are the disadvantages of the Dual-Radix Multiplier?

Disadvantages include increased design complexity, potential latency in control logic, and higher area requirements due to additional circuitry. It may also provide limited benefits for very small operand sizes.

7. How does the Dual-Radix Multiplier compare to traditional multipliers?

Compared to traditional radix-2 or radix-4 multipliers, the Dual-Radix Multiplier offers improved speed and efficiency, especially for larger operands, while also introducing some complexity in design.

8. Can the Dual-Radix Multiplier be implemented in hardware?

Yes, it can be implemented in hardware using digital circuit design techniques, often in Field-Programmable Gate Arrays (FPGAs) or Application-Specific Integrated Circuits (ASICs) for high-performance applications.

9. What are the performance considerations for implementing a Dual-Radix Multiplier?

Performance considerations include the trade-off between speed and area, the choice of control logic, the handling of carry propagation, and the overall complexity of the design.

10.Is the Dual-Radix Multiplier suitable for low-power applications?

Yes, the Dual-Radix Multiplier can be suitable for low-power applications, as it reduces the number of operations and partial products, leading to lower power consumption compared to simpler multipliers.