Implementation of a 16-bit signed unsigned, radix-4, Booth multiplier with sleep mode

Debasmita Deoghuria
Department of Electronic System Engineering
Microelectronics and VLSI Design *Indian Institute of Science*
Bengaluru, India
ddebasmita@iisc.ac.in

Bibhore Goswami
Department of Electronic System Engineering
Electronic System Engineering *Indian Institute of Science*
Bengaluru, India
bibhoreg@iisc.ac.in

***Abstract*—This report presents a detailed algorithm for a Booth-encoded, two-stage pipelined, 16-bit sleep-enabled signed/unsigned multiplier. The design process, including both schematic and layout implementation, is thoroughly discussed and analyzed.**

***Index Terms*—radix 4, pipelined, wallace**

I. Introduction

A 16-bit multiplier takes two 16-bit inputs and produces a 32-bit output. The inputs and outputs can be interpreted as signed or unsigned numbers based on the control signal. Using Booth's algorithm, the number of partial products is reduced to 8 for signed and 9 for unsigned operations, enhancing efficiency. To accelerate addition, a Wallace tree architecture is employed [1].The final outputs of the Wallace tree are combined using a carry bypass adder, further enhancing the speed of the multiplier. Since subthreshold leakage is a significant concern in VLSI circuits, sleep-enabled PMOS header transistors are integrated into each sub-block to minimize power loss. Additionally, two D-flip-flops are incorporated to enable pipelined execution, improving overall performance.This design offers a faster and more energy-efficient multiplier compared to conventional multipliers, although it requires more area for implementation. The following sections provide a detailed explanation of each block, highlighting the design choices and their impact on performance and efficiency.

II. Partial Product Generator

A. Booth encoder

The radix-4 Booth algorithm is widely used to improve the performance of multiplier because it can reduce the number of partial products by half. This in turn increases the speed of the multiplier. Radix-2^r multipliers generate $\frac{N}{r}$ partial products, where each partial product depends on r bits of the multiplier. Reducing the number of partial products leads to a more compact and faster carry-save adder (CSA) array. For instance, a radix-4 multiplier produces $\frac{N}{2}$ partial products. Each partial product corresponds to 0, Y, $2Y$, or $3Y$, based on pairs of bits from the multiplier X. While $2Y$ can be computed through a simple left shift, calculating $3Y$ involves a more complex carry-propagate addition, requiring the sum of Y and $2Y$ before the generation of partial products begins. The multiplier is zero padded two times befor MSB and one time after LSB. The multplier is divided into groups of three as shown in 1.

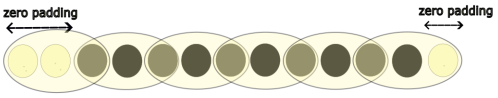

Fig. 1. Booth encoding algorithm

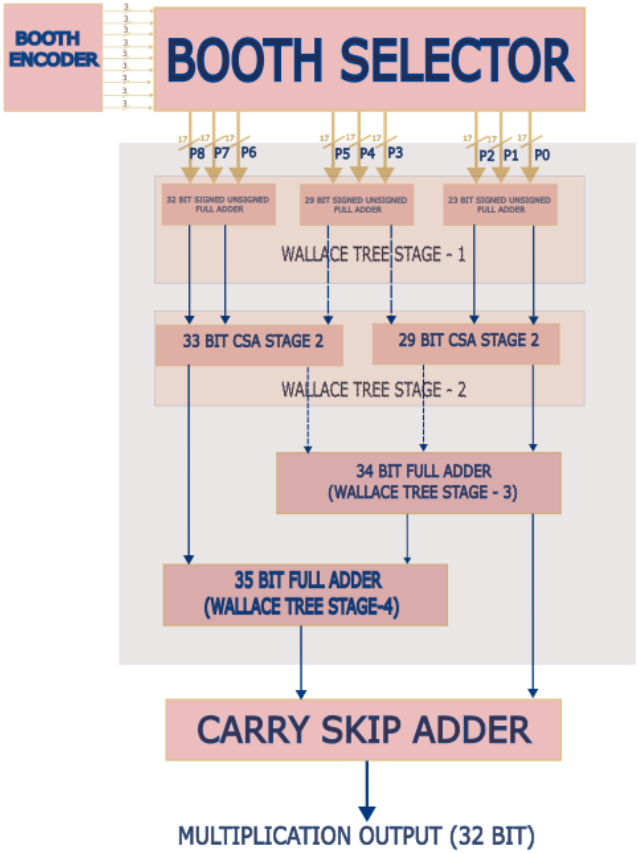

Fig. 2. 16-bit Multiplier signal flow

Identify applicable funding agency here. If none, delete this.

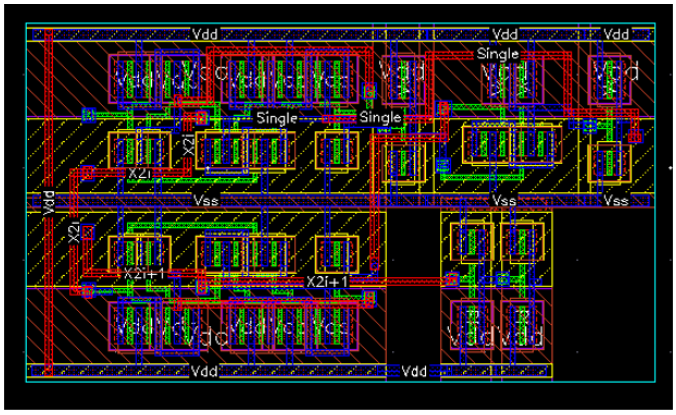

Fig. 3. Layout of Booth Encoder

B. Booth Selector

The Booth selector is a key component of the Radix-4 Booth encoder, which is commonly used in multipliers to reduce the number of partial products. It examines groups of three bits from the multiplier (including one overlap bit) to decide how to encode the multiplicand. In a Radix-4 Booth encoder, the multiplier bits are inputs that generate control signals called Single (S), Double (D), and Negative (N). These signals determine the partial products based on the encoding rules. This has been encoded in the table 1.

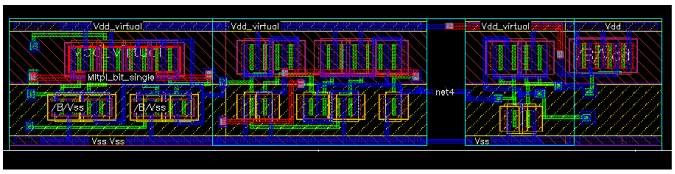

Fig. 4. Layout of Booth Selector

TABLE I
TRUTH TABLE FOR BOOTH ENCODER

X_{2i+1}	X_{2i}	X_{2i-1}	**Partial Product** (PP_i)	**S**	**D**	**N**
0	0	0	0	0	0	0
0	0	1	Y	1	0	0
0	1	0	Y	1	0	0
0	1	1	$2Y$	0	1	0
1	0	0	$-2Y$	0	1	1
1	0	1	$-Y$	1	0	1
1	1	0	$-Y$	1	0	1
1	1	1	0	0	0	1

III. WALLACE TREE ADDER

A. Mirror Full Adder With Sleep

Mirror full adders with **low-voltage threshold (LVT)** cells are incorporated into the design, enabling **power gating** to optimize performance. The use of LVT cells enhances the speed of the critical path in the system. Power gating is achieved using a **NOR gate** and **header PMOS high-voltage threshold (HVT)** transistors, effectively managing power consumption. At typical process-voltage-temperature (PVT) conditions, the design achieves a maximum input-to-carry delay of **53.32 ps** and an input-to-sum delay of **37.1 ps**. Multiple mirror adders operate in parallel to form a **Wallace Tree** structure, which efficiently reduces the partial products in multiplication operations. In the Wallace Tree, **parallel carry-skip adders**, constructed using mirror full adders, enable faster parallel computations.

B. Carry Save Adder

A Carry Save Adder (CSA) is a high-speed adder architecture designed to efficiently handle the addition of multiple binary numbers, commonly used in multiplication and other arithmetic operations. Unlike a ripple carry adder, where the carry must propagate through each bit position, the CSA reduces delay by generating partial sums and carries separately and deferring the final carry propagation to a later stage. In this structure of Wallace Tree we are using seven carry skip adders of different sizes each of which are generating the sum and carry outputs in parallel. The CSA is made of mirror full adder arrays which are implemented using the sleep signal. The area of the largest CSA adder (35-bits) is 3178 μm^2 .

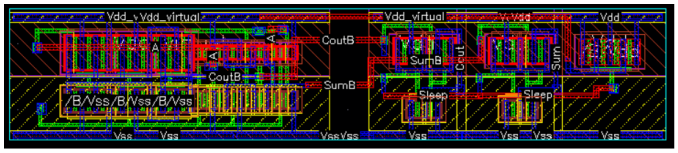

Fig. 5. Mirror adder with sleep layout

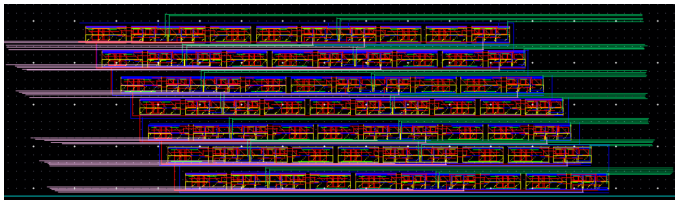

Fig. 6. Layout of a 32-bit Carry Save Adder

IV. CARRY SKIP ADDER

A Carry Skip Adder (CSA) is a specialized adder architecture designed to enhance the speed of binary addition by minimizing the delay caused by carry propagation. In conventional ripple carry adders, the carry bit must propagate through each bit position, resulting in significant delays for wide bit-width adders. The CSA addresses this by dividing the adder into smaller blocks and allowing carries to skip over these blocks, thereby reducing the overall delay.

We have used LVT cells in the critical path of the carry skip adder to increase the speed of the final addition. The remaining path is made of normal 1v transistors. The delay of the carry skip adder is 281ps under worst case PVT corner (SS, 0.9 V, 125 °C).

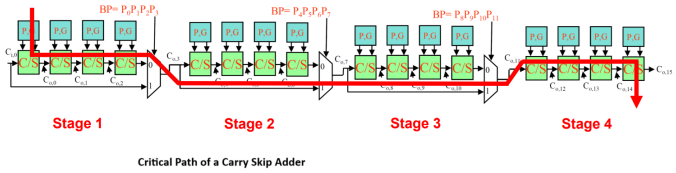

Fig. 7. Critical Path of a Carry Skip Adder

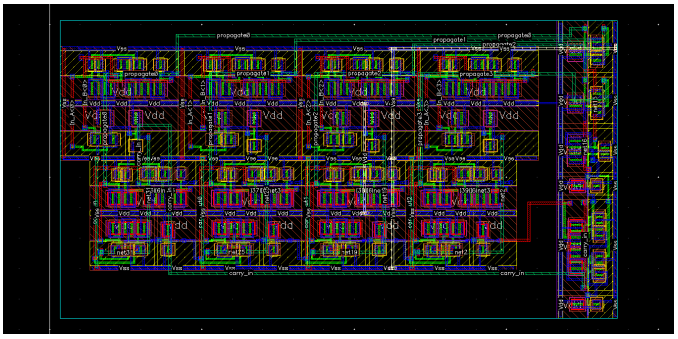

Fig. 8. Layout of 4-bit Carry Skip Adder

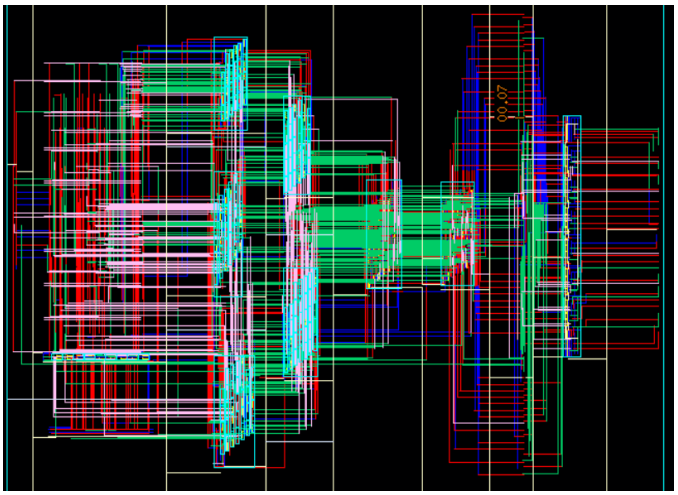

Fig. 9. Final Layout of the 16-bit Multiplier

The final layout of the multiplier has a maximum area of **0.117** mm^2 and uses metal layer upto Metal 5 (including power rails).

TABLE II
AREA AND DELAY CONSUMPTION COMPARISON

Layout	Booth Encoder	Booth Selector	Carry-Save	Carry-Skip
Delay	20 ps	17 ps	53 ps	614ps
Area	291 μm^2	437 μm^2	1735.8 μm^2	1541 μm^2

V. PIPELINING

Pipelining is a technique used in digital circuits and computer architecture to improve the throughput of a system by overlapping the execution of multiple operations. It divides a process into smaller, independent stages, each of which performs part of the overall task. Data flows through these stages in sequence, similar to an assembly line, allowing multiple tasks to be in progress at the same time. In this architecture we have done two stage pipelining using the D Flip Flop provided to us. The t_{pdf} of the two intermediate combinational blocks between the pipelined stages are **873ps** and **281ps**.The t_{cq} delay and t_{setup} of the flip flop is **190ps** and **41ps** respectively. After capturing the inputs, we get the multiplication after two clock cycles but decreases the throughput allowing multiple processes being executed at a particular time. The delay also reduces. A basic block diagram of our pipeline implementation is shown in fig10

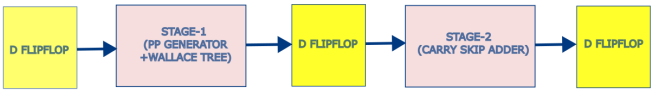

Fig. 10. Pipelining architecture of 16 bit multiplier

VI. POWER GRIDING

To provide supply voltage (Vdd) and common ground (Vss) to each and every subblock properly and to reduce IR drop across the wires, power grid is utilised. Metal 5 and Metal 6 are used for power griding technique. Alternative rows(Metal 5) and columns(Metal 6) of Vdd and Vss rails are implemented. Individuals= blocks of each stage has been placed at the unit cells of the power grid. This has been illustrated in the diagram shown in fig 11.

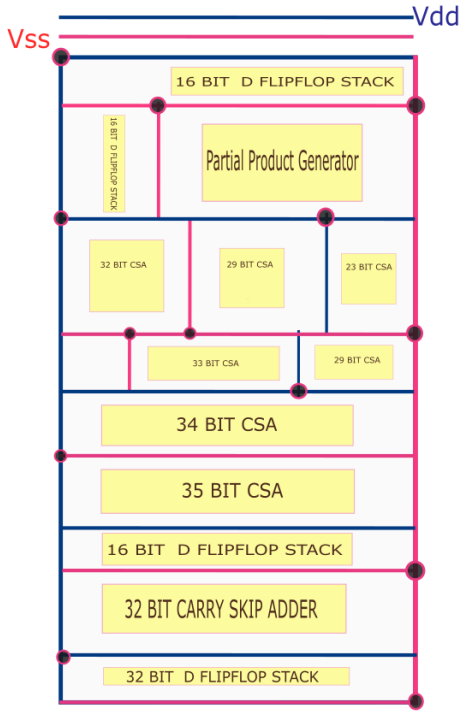

Fig. 11. Power Grid in Multiplier Cell

VII. TESTING

FO4 reference is used to characterise the delay of the multiplier block. Input shaping, load and load-on-load circuitry are emulated. Three major process corners (SS, FF, TT) are used to get the range of operating parameters.The electrical and physical conditions for the three process corners are mentioned below: A test bench for the delay characterization

TABLE III
DEFINITION OF THREE MAJOR PVT CORNERS.

	Process	Voltage	Temperature
Best	FF	1.1 V	-40 °C
Typical	TT	1.0 V	27 °C
Worst	SS	0.9 V	125 °C

is shown in the figure 12.

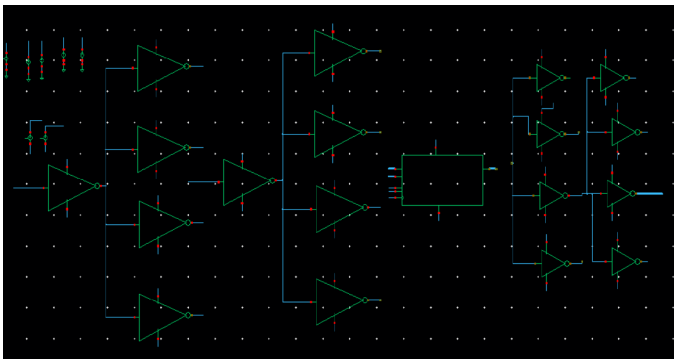

Fig. 12. Test bench with FO4 reference inverter setup

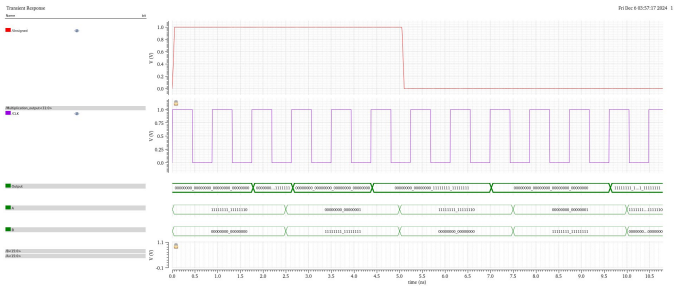

Fig. 13. Test bench output waveform

VIII. PERFORMANCE

A. Delay:

To characterise the maximum propagation delay the inputs are given as X=1111111111111111 and Y=0000000000000001. The output is -1 i.e. 11111111111111111111111111111111. This transition needs maximum no of changes in the output bits. Pre and post layout delays are as following:

Therefore, the multiplier imparts a maximum delay of **6.519 ns** at the worst PVT corner (SS, 0.9 V, 125 °C).

TABLE IV
MAXIMUM FO4 DELAY AT DIFFERENT PVT CORNERS.

	FF	TT	SS
Pre-layout	495.9 ps	871 ps	1.4902 ns
Post-layout	1.96 ns	2.68 ns	4.01 ns

B. Clock Speed:

The maximum clock frequency f_{clk} that can be applied to a sequential circuit containing a combinational logic block is defined as:

$$f_{clk} = \frac{1}{t_{pd}}$$

The maximum clock frequency at different PVT corners are:

TABLE V
MAXIMUM CLOCK SPEED AT DIFFERENT PVT CORNERS.

	FF	TT	SS
Pre-layout	2 GHz	1.15 GHz	700 MHz
Post-layout	510 MHz	373 MHz	250 MHz

The post-layout multiplier can support a maximum clock speed of 510 MHz, but at the best PVT corner. But it is safe to use it below **250 MHz**, measured the worst PVT corner.

C. Power calculation:

- Without sleep: The major portion of power consumption is dynamic power consumption. So characterisation of dynamic power is very much important. The following table summaries the dynamic power consumption($\mu watt$) of the circuit"

TABLE VI
POWER CONSUMPTION WITH SLEEP SIGNAL DISABLED

	FF	TT	SS
Pre-layout	600.3 μW	473.4 μW	567.7 μW
Post-layout	987.25 μW	822 μW	657.7 μW

- With sleep: To save power consumption, sleep PMOS transistors are implemented in each block of booth selector and mirror adder where we have used LVT transistors. Enabling sleep signal reduces the power consumption which can be verified from the following table:

TABLE VII
POWER CONSUMPTION WITH SLEEP SIGNAL ENABLED

	FF	TT	SS
Pre-layout	350.72 μW	228.81 μW	294.4 μW
Post-layout	489.25 μW	457.3 μW	394.62 μW

Thus the multiplier dissipates a power of **987.25** μW at best PVT corner (FF, 1.1 V, -40 °C). With power gating we can achieve a power gating of maximum of 51.8 percentage at the best PVT corner. The leakage power of the multiplier at best PVT (FF, 1.1 V, -40 °C) is **657 nW** in sleep disabled mode and **100 nW** in sleep enabled mode giving us a **84.7** percentage power gating in leakage power.

IX. SUMMARY

The 16-bit radix-4 Booth multiplier has been designed by implementation of Wallace tree algorithm for high-speed operation. The schematic consists of four major blocks – Booth selectors, FA arrays, carry skip adders and D-Flip Flop. The layout of the multiplier has 5 metal layers and total area of 0.117 mm2. The multiplier works perfectly in temperature range of -40 °C to 125 °C and can tolerate voltage fluctuations of 10gives maximum FO4 delay 4.01 ns, clock speed of 250 MHz at the worst PVT corner with two stage pipelining and consumes maximum power of 987.25 μW at the best PVT corner. This architecture works for both signed and unsigned multiplications and also has sleep enabled mode. ***All stages and top level are DRC and LVS clear***.

X. CONCLUSION

The process of schematic and layout design offered valuable exposure to industry-level practices for digital IC design and optimization. During layout creation and RC extraction, challenges related to DRC (Design Rule Check) and LVS (Layout Versus Schematic) drove the exploration of more efficient placement and routing strategies, ultimately enhancing the circuit's overall performance. Incorporating high-speed logic families and optimised auto-routing at the top level, however, could further boost performance and reduce the required area.

REFERENCES

[1] N. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*. USA: Addison-Wesley Publishing Company, 4th ed., 2010.