High-Speed Multiply–Accumulate Unit Based on Brent–Kung Adder with Pipelining

Ms. M. Pavitra[1], K. V. Sailaja[2], P. Keerthana[3], M. Harsha Vardhan [4], G. Sudheer Reddy[5], P. Vamsi[6]

[1]*Associate Professor, Dept. Of ECE, PBR VITS, Kavali, Nellore District, Andhra Pradesh, India*
[2-6]*UG Students, Dept. Of ECE, PBR VITS, Kavali, Nellore District, Andhra Pradesh, India*

***

Abstract -The Multiply-Accumulate (MAC) Unit is a crucial component in all DSP Applications, due to its ability to perform high speed arithmetic operations. This Project aims to design and implement an 8-bit MAC Unit capable of performing multiplication and accumulation operations. The MAC Unit employs the same multiplier unit implemented using different adder architectures while keeping the multiplier constant. Conventional MAC units use Ripple Carry Adders (RCA) and Carry Look-Ahead Adders (CLA), which suffer from high carry propagation delay and increased hardware complexity respectively. To overcome these limitations, parallel prefix adders are proposed, such as Kogge–Stone (KSA), Ladner–Fischer (LFA), Brent Kung Adder with Pipelining are integrated into the MAC unit. The structures were formed using Verilog Hardware Description Language (HDL) and implemented on Xilinx ISE, with simulation carried out using Model Sim.

Key Words: *MAC Unit, 8-bit, RCA, CLA, Kogge-Stone Adder (KSA), Ladner-Fischer Adder (LFA), Verilog HDL, FPGA, DSP, Parallel Prefix Adder, Vivado, Artix7*

1. INTRODUCTION

The Multiply-Accumulate (MAC) unit is one of the most critical components in digital signal processing (DSP) and modern computing architectures. It forms the computational backbone of numerous applications, ranging from digital filters and Fast Fourier Transforms (FFT) to image processing, wireless communication, and neural network accelerators. The MAC unit performs a fundamental arithmetic operation: it multiplies two input operands and adds the product to a previously stored sum. Mathematically, this operation is expressed as: Y=Y + (A x B)

where, A and *B* are input operands, and Y is the accumulated result. The efficiency of a MAC unit directly influences the throughput, speed, power consumption, and overall performance of the system in which it is deployed. In many high-performance DSP and embedded systems, MAC units account for a significant portion of the critical path delay and power consumption, making their design and optimization a primary research focus.

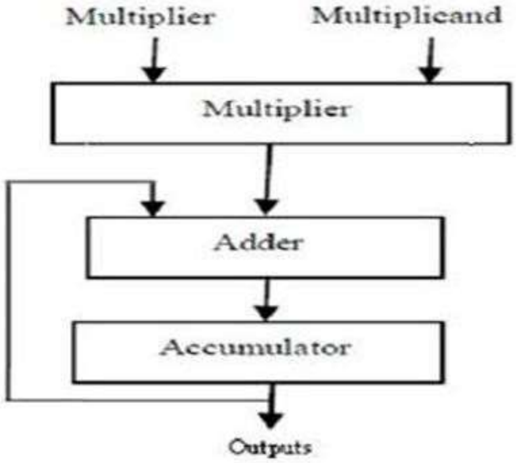


Fig.1.1: MAC Unit Layout

2. LITERATURE SURVEY

The design and optimization of Multiply-Accumulate (MAC) units and related arithmetic components play a crucial role in enhancing the performance of Digital Signal Processing (DSP) and VLSI systems. B. Hemalatha (2019) presented a MAC unit design using Verilog HDL, emphasizing efficient arithmetic operations to improve processing speed, hardware efficiency, and overall computational performance. Similarly, Sudhanya and Joy Vasantha Rani (2021) proposed hybrid iterative algorithms for FPGA placement optimization, focusing on minimizing wire length and reducing routing complexity, thereby improving resource utilization and execution efficiency.

Dimitrakopoulos et al. (2021) introduced Sum Propagate Adders aimed at optimizing carry propagation to reduce delay while maintaining manageable hardware complexity, significantly enhancing the speed of arithmetic circuits. Khare et al. (2016) developed a high-speed MAC unit using the Karatsuba multiplication technique, which reduces multiplication complexity by dividing large computations into smaller operations, resulting in improved computational speed. Shanthala et al. (2009) proposed a pipelined MAC architecture for VLSI implementation, where pipelining enables parallel execution of operations, increasing throughput and reducing delay. Furthermore, Harris (2003) presented a taxonomy of parallel prefix adders, including Kogge–Stone, Brent–Kung, and Ladner–Fischer architectures, analyzing them based on speed, area, and wiring complexity.

In addition, Poonguzhali et al. (2020) designed a Kogge–Stone adder using approximate compressors to reduce power consumption and hardware complexity while maintaining acceptable accuracy. Overall, these studies highlight that efficient design of MAC units, adders, and FPGA optimization techniques significantly improves performance, speed, and energy efficiency in modern digital and signal processing systems.

3. EXISTING SYSTEM

The Multiply-Accumulate (MAC) unit is a crucial component in digital signal processing (DSP) systems, performing two essential arithmetic operations: multiplication and accumulation. MAC units are widely used in applications such as digital filters, convolution operations, Fast Fourier Transforms (FFT), digital communication systems, and neural network accelerators. The core operation of a MAC unit involves multiplying two operands and adding the product to a previously stored sum, a function that directly affects the overall speed, power, and efficiency of the system. Consequently, the choice of adder architecture plays a critical role in MAC unit performance.

In traditional MAC designs, the multiplier is often combined with simple and widely used adder architectures such as the Ripple Carry Adder (RCA) and the Carry Look-Ahead Adder (CLA). These adder designs represent a fundamental trade-off between simplicity, speed, and hardware complexity. The RCA, in particular, is popular due to its minimal circuit complexity and low area overhead. It operates by computing the sum of each bit sequentially, propagating the carry from the least significant bit (LSB) to the most significant bit (MSB). This sequential carry propagation results in a linear delay that increases with the bit-width of the operands.

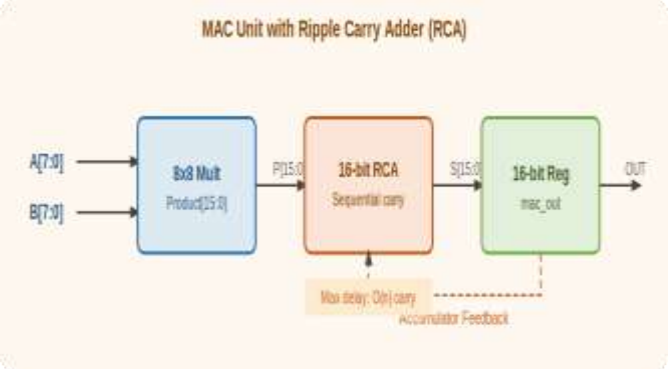


Fig.3.1 : Structure of 8-bit MAC with RCA

The Carry Look-Ahead Adder (CLA) addresses the primary limitation of the RCA by reducing carry propagation delay. Instead of waiting for the carry to propagate sequentially, the CLA uses generate and propagate logic to predict carry signals in parallel. For each bit position, a generate signal indicates whether that bit will generate a carry regardless of the input carry, while a propagate signal indicates whether a carry will pass through the bit. By combining these signals across all bit positions, the CLA can compute carry outputs for multiple bits simultaneously, significantly reducing the critical path delay.

This allows the MAC unit to operate at a higher clock frequency compared to an RCA-based design. However, this improvement comes at the cost of increased hardware complexity, larger silicon area, and higher dynamic power consumption. Additionally, for wider bit-widths, the logic required for carry computation grows rapidly, making the design more challenging to implement efficiently.

Fig.3.2: Structure of 8-bit MAC with CLA

Simulation and synthesis of RCA and CLA-based MAC units are commonly performed using Verilog Hardware Description Language (HDL) on FPGA platforms such as Xilinx ISE or Vivado. Timing analysis from these tools highlights the critical path delay imposed primarily by the adder, while power analysis provides estimates of both static and dynamic power consumption. For RCA-based MAC units, the delay dominates due to sequential carry propagation, whereas CLA-based MAC units achieve reduced delay at the expense of increased power consumption due to additional logic gates and switching activity. The results of such simulations provide insights into the trade-offs between speed, power, and area for different adder configurations.

Another important aspect of existing MAC designs is the impact of operand bit-width on performance. As DSP applications evolve, higher bit-width operations (such as 16-bit, 32-bit, or 64-bit) are increasingly required for greater precision and accuracy. In RCA-based MAC units, the linear growth of delay with bit-width becomes a significant bottleneck, while in CLA-based units, the carry logic complexity increases substantially. These factors highlight the limitations of traditional adder designs when scaling MAC units to higher precision or wider data paths, a critical consideration for modern high-performance DSP systems.

Finally, although RCA and CLA-based MAC units provide an acceptable trade-off between simplicity, speed, and power, they do not fully address the demands of high-speed, low-power DSP systems in emerging applications such as machine learning accelerators, real-time image processing, and communication systems.

Their limitations in scalability, throughput, and energy efficiency motivate research into alternative adder architectures and MAC designs that can achieve faster operation without significantly increasing area or power consumption.

In summary, the existing method of MAC units using RCA and CLA adders is widely adopted in DSP systems due to its simplicity and proven functionality. RCA-based MAC units are advantageous for low-power, low-speed applications, while CLA-based MAC units provide improved speed at the cost of higher complexity and power.

4. PROPOSED SYSTEM

To overcome the limitations of conventional MAC units using Ripple Carry Adder (RCA) and Carry Look-Ahead Adder (CLA), the proposed method integrates advanced parallel prefix adders, specifically Kogge-Stone, Ladner-Fischer and Brent Kung adder with pipelining as the accumulation component within the MAC architecture. The main objective of this approach is to achieve high-speed arithmetic computation while maintaining reasonable power consumption and area overhead, making the design suitable for high-performance DSP applications such as real-time signal processing, image filtering, and neural network accelerators.

The Kogge-Stone Adder is a parallel prefix adder well-known for its minimal logic depth and balanced tree structure. Unlike RCA and CLA, which compute carry sequentially or with block-level propagation, Kogge-Stone calculates carry signals in a logarithmic number of stages relative to the bit-width. This structure allows simultaneous computation of carry for all bit positions, dramatically reducing the critical path delay. When integrated into an 8-bit MAC unit, the Kogge-Stone adder significantly increases the maximum operating frequency, enhancing the throughput of DSP operations. However, the trade-off is higher hardware complexity and a larger number of logic gates, which increases routing requirements and area on the silicon.

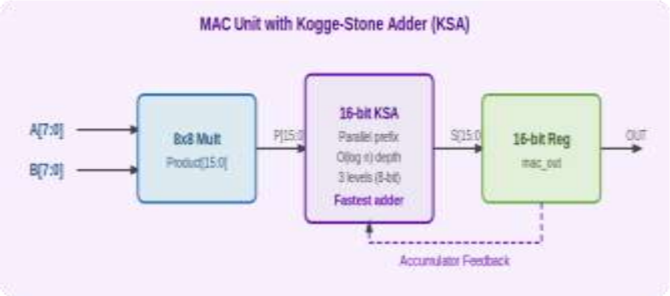


Fig.4.1: Structure of 8-Bit MAC with Kogge-Stone Adder

The Ladner-Fischer Adder is another parallel prefix adder designed to balance speed and hardware complexity. It reduces the number of prefix nodes and interconnections compared to the Kogge-Stone Adder while maintaining a logarithmic delay for carry computation. This results in slightly slower performance than Kogge-Stone but with reduced area and wiring congestion, making it an attractive choice for systems that require high speed without excessive resource utilization.

In the proposed MAC architecture, the Ladner-Fischer adder provides a middle ground, offering improved performance over CLA while limiting hardware overhead compared to Kogge-Stone.

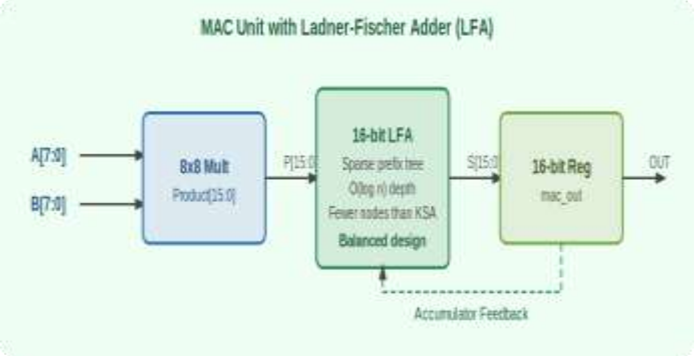


Fig.4.2: Structure of 8-bit MAC with Ladner-Fischer adder

The Brent–Kung adder is a type of parallel prefix adder designed to reduce hardware complexity and wiring requirements while maintaining efficient carry computation. Compared to other prefix adders such as the Kogge–Stone and Ladner–Fischer adders, the Brent–Kung adder uses a more balanced tree structure with fewer prefix nodes and interconnections.

In the proposed method, the MAC unit consists of a common 8-bit multiplier paired with either a Kogge-Stone or Ladner-Fischer adder or Brent-Kung adder with pipelining in a modular architecture. This modularity allows easy substitution of the adder component, enabling designers to select the optimal configuration based on specific application requirements. The multiplier is implemented using an array-based approach, generating partial products through AND gates and reducing them with carry-save adders.

While the parallel prefix adder-based MAC units in the existing proposed method achieve improved critical path delay over RCA and CLA, they are limited by the throughput of a single-cycle combinational data path. Each MAC operation must complete the full multiply-accumulate chain within one clock cycle, bounding the maximum frequency to the critical path through the multiplier, adder, and accumulator in series.

This proposed extension introduces a 5-stage synchronous pipeline to the Brent-Kung (BK) MAC unit. By inserting registers between computational stages, the critical path is reduced to the longest single-stage path, enabling a higher operating frequency and improved throughput for streaming DSP applications such as FIR filters, convolution, and neural network inference.

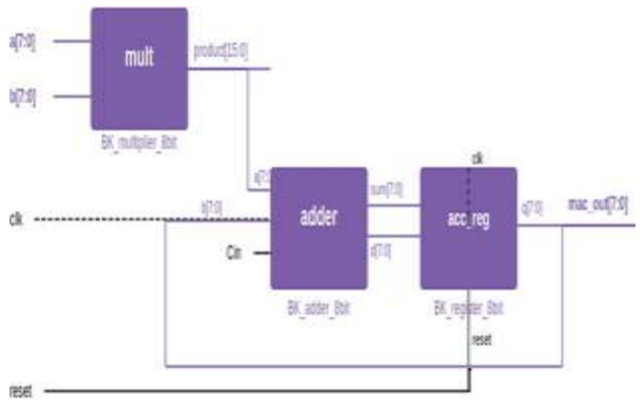


Fig.3.3: Structure of 8Bit MAC Unit with Brent Kung Adder (pipelined)

The Brent–Kung with pipelining MAC unit also demonstrates efficient power utilization due to its reduced number of prefix nodes and simplified interconnection structure. Compared to Kogge–Stone adders, the Brent–Kung adder requires fewer logic resources, which helps in limiting switching activity and overall power consumption. Although the power consumption is slightly higher than RCA because of the parallel prefix operations, the Brent–Kung architecture still provides an effective balance between power efficiency and computational speed.

Another advantage of the proposed method is scalability. The parallel prefix structure of both Kogge-Stone and Ladner-Fischer adders allows the MAC unit to be extended to higher bit-widths, such as 16-bit or 32-bit, without linear increases in carry propagation delay. This makes the proposed design adaptable for high-precision DSP applications and emerging workloads that require larger operands, such as high-resolution signal processing or deep learning accelerators.

In summary, the proposed MAC unit using Kogge-Stone and Ladner-Fischer adders offers a significant improvement over conventional RCA and CLA-based designs. By leveraging parallel prefix adders, the MAC unit achieves reduced carry propagation delay, higher throughput, and improved scalability. The Kogge-Stone adder maximizes speed at the expense of area, while the Ladner-Fischer adder provides a balanced solution with moderate area and power consumption. Overall, the proposed method provides a high-performance and flexible MAC unit suitable for modern DSP applications requiring both speed and energy efficiency.

Compared to the Kogge–Stone, Ladner–Fischer, and conventional adders, the Brent–Kung architecture requires fewer prefix nodes and reduced wiring complexity, resulting in more efficient hardware utilization. The pipelined Brent–Kung MAC unit therefore achieves faster operation while maintaining moderate area and power consumption.

Hence, the Brent–Kung adder provides an effective balance between speed, hardware efficiency, and scalability, making it a suitable choice for high-performance DSP and FPGA-based applications.

ADVANTAGES

- Minimum Critical Path Delay (Pipelined BK — Best Overall)
- Balanced Speed-Area Trade-off (LFA)
- Modular and Parameterizable Architecture
- Improved Power-Delay Product (PDP)
- Direct Performance Comparison Platform

APPLICATIONS

- Digital Signal Processing (DSP)
- Image and Video Processing
- Neural Network Inference Accelerate
- Wireless Communication
- Embedded Systems and IOT

5. RESULTS AND DISCUSSIONS

It presents the Vivado synthesis results for all five 8-bit MAC unit implementations: Ripple Carry Adder (RCA), Carry Look-Ahead Adder (CLA), Brent-Kung Adder (BK), Kogge-Stone Adder (KSA), and Ladner-Fischer Adder (LFA). For each variant, the Vivado-generated schematic, resource utilization (Slice LUTs), timing analysis (critical path delay), and power analysis are presented. All designs target the Artix-7 xc7a100t-1csg324 FPGA.

Fig.1: Simulation Results of 8-bit MAC with Ripple Carry Adder

Total on-chip power for the RCA MAC is 19.107 W: dynamic power 18.853 W (99%) and device static 0.254 W (1%). Within dynamic power, I/O contributes 14.505 W (77%), signals 2.334 W (12%), and logic 2.014 W (11%). Junction temperature is 60.7°C with a thermal margin of 24.3°C.


Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power:	**17.65 W**
Design Power Budget:	**Not Specified**
Power Budget Margin:	**N/A**
Junction Temperature:	**58.0°C**
Thermal Margin:	27.0°C (14.2 W)
Effective θJA:	1.9°C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Low

Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power

Dynamic:	17.414 W	(99%)
Signals:	1.690 W	(10%)
Logic:	1.294 W	(7%)
I/O:	14.431 W	(83%)
Device Static:	0.236 W	(1%)

Fig.2: Power Report of 8-bit MAC with Ripple Carry Adder

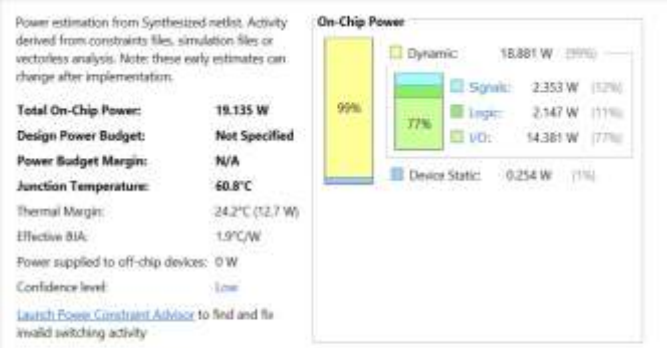
The critical path for the RCA MAC runs through 17 logic levels with a worst-case total delay of 9.781 ns (Path 1: from a[1] to acc_reg/q_reg[15]/D). This is the highest delay among all variants due to the 16-stage sequential carry chain in the 16-bit RCA accumulator. The high fanout of 15 and 17-18 routing levels confirm the long critical path.

Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement
Path 1	∞	17	18	15	a[1]	acc_reg/q_reg[15]/D	9.781	2.562	7.219	∞
Path 2	∞	16	17	15	a[1]	acc_reg/q_reg[14]/D	9.329	2.457	6.872	∞
Path 3	∞	15	16	15	a[1]	acc_reg/q_reg[13]/D	8.869	2.355	6.514	∞
Path 4	∞	14	15	15	a[1]	acc_reg/q_reg[12]/D	8.403	2.247	6.156	∞
Path 5	∞	13	14	15	a[1]	acc_reg/q_reg[11]/D	7.940	2.142	5.798	∞
Path 6	∞	12	13	15	a[1]	acc_reg/q_reg[10]/D	7.477	2.037	5.440	∞
Path 7	∞	11	12	15	a[1]	acc_reg/q_reg[9]/D	6.994	1.932	5.062	∞
Path 8	∞	10	11	15	a[1]	acc_reg/q_reg[8]/D	6.331	1.827	4.504	∞
Path 9	∞	10	11	15	a[1]	acc_reg/q_reg[7]/D	5.851	1.830	4.021	∞
Path 10	∞	8	9	15	a[1]	acc_reg/q_reg[6]/D	4.898	1.638	3.260	∞

Fig.3 : Delay Report of 8-bit MAC with Ripple carry Adder

Fig.4: Simulation Results of 8-bit MAC with Carry Look Ahead Adder

Total on-chip power for the CLA MAC is 19.136 W: dynamic 18.882 W (99%) and static 0.254 W (1%). Breakdown: I/O 14.381 W (77%), signals 2.353 W (12%), logic 2.147 W (11%). Junction temperature 60.8°C, thermal margin 24.2°C.


Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power:	**19.135 W**
Design Power Budget:	**Not Specified**
Power Budget Margin:	**N/A**
Junction Temperature:	**60.8°C**
Thermal Margin:	24.2°C (12.7 W)
Effective θJA:	1.9°C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Low

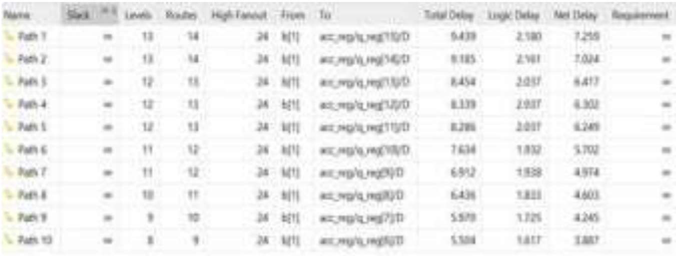
Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power

Dynamic:	18.881 W	(99%)
Signals:	2.353 W	(12%)
Logic:	2.147 W	(11%)
I/O:	14.381 W	(77%)
Device Static:	0.254 W	(1%)

Fig.5: Power Report of 8-bit MAC with Carry Look Ahead Adder

The CLA MAC critical path spans 13 logic levels with a worst-case delay of 9.422 ns (Path 1: from a[2] to acc_reg/q_reg[15]/D). This represents a 3.7% improvement over RCA (9.781 ns). The high fanout of 24 reflects the parallel carry logic. The reduction in logic levels from 17 (RCA) to 13 (CLA) confirms the effectiveness of parallel carry generation.



Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement
Path 1	∞	13	14	24	b[1]	acc_reg/q_reg[15]/D	9.439	2.180	7.259	∞
Path 2	∞	13	14	24	b[1]	acc_reg/q_reg[14]/D	9.185	2.161	7.024	∞
Path 3	∞	12	13	24	b[1]	acc_reg/q_reg[13]/D	8.454	2.037	6.417	∞
Path 4	∞	12	13	24	b[1]	acc_reg/q_reg[12]/D	8.339	2.037	6.302	∞
Path 5	∞	12	13	24	b[1]	acc_reg/q_reg[11]/D	8.286	2.037	6.249	∞
Path 6	∞	11	12	24	b[1]	acc_reg/q_reg[10]/D	7.634	1.932	5.702	∞
Path 7	∞	11	12	24	b[1]	acc_reg/q_reg[9]/D	6.912	1.938	4.974	∞
Path 8	∞	10	11	24	b[1]	acc_reg/q_reg[8]/D	6.436	1.833	4.603	∞
Path 9	∞	9	10	24	b[1]	acc_reg/q_reg[7]/D	5.970	1.725	4.245	∞
Path 10	∞	8	9	24	b[1]	acc_reg/q_reg[6]/D	5.504	1.617	3.887	∞

Fig.6: Delay Report of 8-bit MAC with Carry Look Ahead Adder

Fig.7: Simulation Results of 8-bit MAC unit with Kogge Stone Adder

Total on-chip power for the KSA MAC is 18.912 W: dynamic 18.661 W (99%), static 0.251 W (1%). Breakdown: I/O 14.488 W (78%), signals 2.265 W (12%), logic 1.908 W (10%). Junction temperature 60.4°C, thermal margin 24.6°C — the second-best thermal performance.

Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power: **19.077 W**
Design Power Budget: **Not Specified**
Power Budget Margin: **N/A**
Junction Temperature: **60.7°C**
Thermal Margin: 24.3°C (12.8 W)
Effective θJA: 1.9°C/W
Power supplied to off-chip devices: 0 W
Confidence level: Low
Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power
Dynamic: 18.824 W (99%)
Signals: 2.321 W (12%)
Logic: 1.957 W (11%)
I/O: 14.505 W (77%)
Device Static: 0.253 W (1%)

Fig.8: Power Report of 8-bit MAC with Kogge Stone Adder

The KSA MAC achieves the shortest critical path among all five variants: 12 logic levels and 8.943 ns maximum delay (from b[5] to acc_reg/q_reg[15]/D). This is an 8.6% improvement over RCA and 5.1% improvement over CLA. The reduced fanout (28) compared to BK (29) confirms the KSA's more balanced signal distribution.

Name	Slack ^1	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement
Path 1	∞	12	13	26	[illegible]	acc_reg/q_reg[15]/D	8.922	2.037	6.885	∞
Path 2	∞	12	13	26	b[6]	acc_reg/q_reg[14]/D	8.846	2.056	6.790	∞
Path 3	∞	12	13	26	b[6]	acc_reg/q_reg[13]/D	8.782	2.037	6.745	∞
Path 4	∞	12	13	26	b[6]	acc_reg/q_reg[12]/D	8.585	2.058	6.527	∞
Path 5	∞	11	12	26	b[6]	acc_reg/q_reg[11]/D	8.131	1.932	6.199	∞
Path 6	∞	11	12	26	b[6]	acc_reg/q_reg[10]/D	7.878	1.932	5.946	∞
Path 7	∞	10	11	26	b[6]	acc_reg/q_reg[9]/D	7.428	1.848	5.580	∞
Path 8	∞	9	10	26	b[6]	acc_reg/q_reg[8]/D	6.714	1.722	4.992	∞
Path 9	∞	8	9	26	b[6]	acc_reg/q_reg[7]/D	6.023	1.617	4.406	∞
Path 10	∞	7	8	25	b[1]	acc_reg/q_reg[6]/D	4.969	1.533	3.436	∞

Fig.9: Delay Report of 8-bit MAC with Kogge Stone Adder

Fig.10: Simulation results of 8-bit MAC unit with Ladner Fischer

Adder

The LFA MAC achieves the lowest total on-chip power: 17.651 W — a 9.1% reduction compared to the highest-power BK variant (19.406 W). Dynamic power is 17.415 W (99%) and static 0.236 W (1%). Breakdown: I/O 14.431 W (83%), signals 1.690 W (10%), logic 1.294 W (7%). Junction temperature is the lowest at 58.0°C with the best thermal margin of 27.0°C.

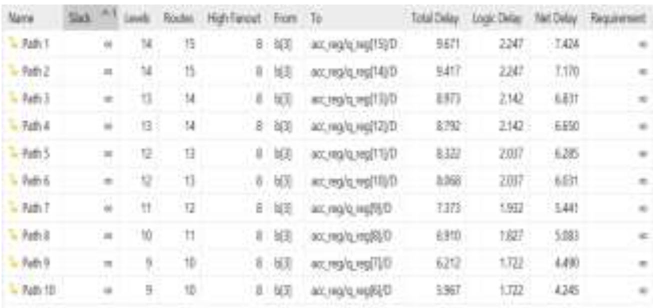
Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power: **18.999 W**
Design Power Budget: **Not Specified**
Power Budget Margin: **N/A**
Junction Temperature: **60.5°C**
Thermal Margin: 24.5°C (12.9 W)
Effective θJA: 1.9°C/W
Power supplied to off-chip devices: 0 W
Confidence level: Low
Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power
Dynamic: 18.746 W (99%)
Signals: 2.309 W (12%)
Logic: 1.950 W (10%)
I/O: 14.487 W (78%)
Device Static: 0.252 W (1%)

Fig.11: Power Report of 8-bit MAC with Ladner Fischer Adder

The LFA MAC critical path spans 14 logic levels with a worst-case delay of 9.671 ns (from b[3] to acc_reg/q_reg[15]/D). Despite the sparse prefix tree, the delay is slightly higher than KSA and BK due to the Gray Cell fanout in the final carry extraction stage. The low high-fanout value of 8 (compared to 29 for BK) confirms the reduced wiring complexity.



Name	Slack ^1	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement
Path 1	∞	14	15	8	b[3]	acc_reg/q_reg[15]/D	9.671	2.247	7.424	∞
Path 2	∞	14	15	8	b[3]	acc_reg/q_reg[14]/D	9.417	2.247	7.170	∞
Path 3	∞	13	14	8	b[3]	acc_reg/q_reg[13]/D	8.973	2.142	6.831	∞
Path 4	∞	13	14	8	b[3]	acc_reg/q_reg[12]/D	8.792	2.142	6.650	∞
Path 5	∞	12	13	8	b[3]	acc_reg/q_reg[11]/D	8.322	2.037	6.285	∞
Path 6	∞	12	13	8	b[3]	acc_reg/q_reg[10]/D	8.068	2.037	6.031	∞
Path 7	∞	11	12	8	b[3]	acc_reg/q_reg[9]/D	7.373	1.932	5.441	∞
Path 8	∞	10	11	8	b[3]	acc_reg/q_reg[8]/D	6.910	1.827	5.083	∞
Path 9	∞	9	10	8	b[3]	acc_reg/q_reg[7]/D	6.212	1.722	4.490	∞
Path 10	∞	9	10	8	b[3]	acc_reg/q_reg[6]/D	5.967	1.722	4.245	∞

Fig.12: Delay Report of 8-bit MAC with Ladner Fischer Adder

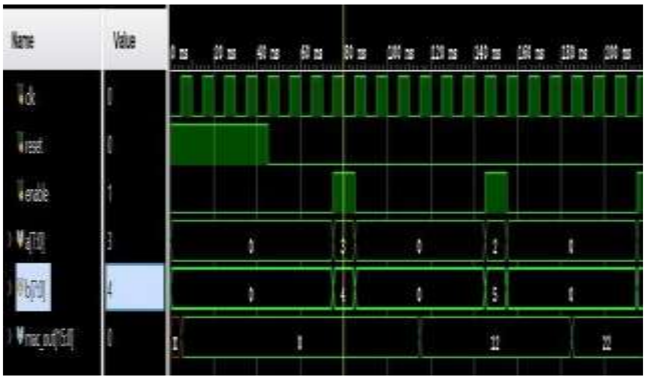


Fig.13 : Simulation results of 8-bit MAC with Brent Kung Adder

Total on-chip power for the BK MAC is 19.406 W — the highest among all five variants, due to higher switching activity in the BK carry tree. Dynamic power: 19.149 W (99%), static: 0.257 W (1%). Breakdown: I/O 14.410 W (75%), signals 2.501 W (13%), logic 2.238 W (12%). Junction temperature 61.3°C, thermal margin 23.7°C.

Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power: **19.22 W**
Design Power Budget: **Not Specified**
Power Budget Margin: **N/A**
Junction Temperature: **60.9°C**
Thermal Margin: 24.1°C (12.7 W)
Effective θJA: 1.9°C/W
Power supplied to off-chip devices: 0 W
Confidence level: Low
Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power
Dynamic: 18.965 W (99%)
Signals: 2.478 W (13%)
Logic: 1.937 W (10%)
I/O: 14.550 W (77%)
Device Static: 0.255 W (1%)

Fig.14: Power Report of 8-bit MAC with Brent Kung Adder

The BK MAC critical path spans 13 logic levels with a worst-case delay of 9.275 ns (from a[2] to acc_reg/q_reg[14]/D). This is a 5.2% improvement over RCA and 1.6% improvement over CLA. The high fanout of 29 is the highest among all variants, reflecting the BK tree's wider broadcast structure.

Name	Slack ^1	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requ
Path 1	∞	12	12	23	b_reg_reg[1]/C	mult_reg_reg[14]/D	8.146	1.661	6.485	
Path 2	∞	12	12	23	b_reg_reg[1]/C	mult_reg_reg[13]/D	8.127	1.661	6.466	
Path 3	∞	12	12	23	b_reg_reg[1]/C	mult_reg_reg[15]/D	8.127	1.661	6.466	
Path 4	∞	11	11	23	b_reg_reg[1]/C	mult_reg_reg[12]/D	7.735	1.556	6.179	
Path 5	∞	11	11	23	b_reg_reg[1]/C	mult_reg_reg[11]/D	7.441	1.556	5.885	
Path 6	∞	10	10	23	b_reg_reg[1]/C	mult_reg_reg[10]/D	6.771	1.451	5.320	
Path 7	∞	9	9	23	b_reg_reg[1]/C	mult_reg_reg[9]/D	6.339	1.365	4.974	
Path 8	∞	9	9	23	b_reg_reg[1]/C	mult_reg_reg[8]/D	5.998	1.349	4.649	
Path 9	∞	8	8	23	b_reg_reg[1]/C	mult_reg_reg[7]/D	5.355	1.241	4.114	
Path 10	∞	7	7	23	b_reg_reg[1]/C	mult_reg_reg[6]/D	4.708	1.136	3.572	

Fig.15: Delay Report of 8-bit MAC with Brent kung Adder

6. CONCLUSION AND FUTURESCOPE

8-bit MAC unit was designed and analysed using various adders in Verilog HDL and simulated using Model sim. The analyzed results were summarized in the following Table 5.1

Table .1: Comparison Table of all Five Adders

SYSTEM	AREA (LUTs)	POWER (W)	DELAY (ns)
MAC with RCA	100	17.65	9.781
MAC with CLA	151	19.13	9.439
MAC with Kogge Stone	150	19.07	8.922
MAC with Ladner Fischer	196	18.99	9.671
MAC with Brent Kung	144	19.22	8.143

Thus, 8 - bit MAC unit was designed and analysed with various adders. It is concluded that the performance of the MAC unit varies based on area, power and delay parameters. The Ripple carry Adder, Carry Look Ahead Adder, Kogge Stone Adder, Ladner Fischer Adder, Brent Kung with Pipelining comparatively. Brent Kung with Pipelining high speed operation.

Overall, RCA is best for low area and power, while Brent–Kung and Kogge–Stone adders are preferred for high-speed MAC operations. Therefore, the Brent-Kung Adder with pipelining is the most suitable for high speed operations.

FUTURE SCOPE

The project can be extended for higher speeds by replacing the unsigned multipliers with highspeed multipliers like Wallace tree, Booth multiplication, Dadda tree etc. to reduce the multiplier critical path delay independently of the accumulator adder. Further the MAC architecture can be extended to 16-bit, 32-bit, and floating-point operand widths using parameterized Verilog generate blocks. Clock gating and operand isolation techniques can be incorporated to reduce dynamic power consumption during idle MAC cycles, applying the concepts from BIST power optimization to the MAC data path.

REFERENCES

1. D. H. S. S. K. B. Hemalatha, "Design of MAC Unit for DSP Applications using Verilog HDL," 2019.

2. Sudhanya, P., and SP Joy Vasantha Rani. "Wire-length and run-time optimization in FPGA placement using hybrid iterative algorithms." Journal of Circuits, Systems and Computers 30, no. 05 (2021): 2150081.

3. Sudhanya, P., Joy Vasantha Rani, S.P. (2020). "Adaptive Particle Swarm Optimization Based Wire-length Minimization for Placement in FPGA", New Trends in Computational Vision and Bio-inspired Computing. Springer, Cham.

4. R. Chikkani, B. M. and Y. J. M Shirur, "VLSI Implementation of Multiply and Accumulate Unit using Distributed Arithmetic," Bioscience Biotechnology Research Communication, vol. 13, pp. 212-217, 2020.

5. G. Dimitrakopoulos, K. Papachatzopoulo and V. Paliouras, "Sum Propagate Adders," IEEE Transactions on Emerging Topics on Computing, 2021.
6. M. Sultana, B. Arunalatha and P. Ramyaraju, "MAC Unit Design using Multiplier & Ripple Carry Adder," International Journal of VLSI System Design and Communication Systems (IJVDCS), vol. 03, no. 10, pp. 1538 1540, 2015.

7. N. Khare, D. Rao, and R. Mohan, "VLSI Implementation of High-Speed MAC Unit Using Karatsuba Multiplication Technique," Journal of Network Communications and Emerging Technologies (JNCET), vol. 6, no. 1, 2016.

8. Farooqui and V. Oklobdzija, "General data-path organization of a MAC unit for VLSI implementation of DSP processors," in IEEE, Monterey, CA, USA, 2002.

9. K. B. R. Dinesh, R. Vinoth and M. Kasyap, "Design and Implementation of High Speed 32-bit MAC Unit," in Journal of Physics: Conference Series, 2023.

10. M. S. Shanthala, C. Raj and S. Kulkarni, "Design and VLSI Implementation of Pipelined Multiply Accumulate Unit," in International Conference on Emerging Trends in Engineering and Technology, ICETET, 2009.

11. Laxman, N. S. S. Reddy and B. R. Naik, "Design and implementation of hybrid logic based MAC unit using 45 nm technology," ELSEVIER, vol. 6, 2023.