A Resource-Efficient 1024-Point Pipeliend MSC FFT Using Time-Multiplexed Constant Multiplication

First A. Author, *Member, IEEE,* Second B. Author, *Fellow, IEEE,* and Third C. Author, *Student Member, IEEE*

***Abstract*—This paper**

***Index Terms*—FFT, constant multiplication, time-multiplexing, FPGA, ASIC, low-complexity, signal processing.**

I. Introduction

II. Time-Multiplexed Constant Multiplication

III. Proposed MSC Architecture

An resource-efficient 4-parallel 1024-point pipeliend MSC FFT architecture is proposed according to the method desribed in Section II. Fig. 1 shows the overall architecture of the proposed design, which mainly consists of memory-based permutation module and two identical radix-2^5 modules.

A. Data Management

B. Radix-2^5 Module

C. optimization of Rotators

D. Memory-based Permutation Circuit

IV. Experimental Results and Comparison

To evaluate the efficacy of the proposed 4-parallel 1024-point MSC FFT architecture, we implemented it on a Xilinx Virtex 7 XC7VX330TFFG1157 FPGA platform and compared it against five state-of-the-art designs [1]–[5].

We employ the same FFT size, word length(WL) and degree of parallelism for a fair comparison, i.e., N = 1024, WL = 16bit and P = 4. Table I presents a detailed comparison of hardware resource utilization and performance characteristics between the proposed 4-parallel 1024-point MSC FFT architecture and contemporary designs. The key metrics include the number of slices, LUTs, flip-flops (FFs), DSPs, Block-RAMs, maximum clock frequency (f_{CLK}), throughput(Th.), latency (in cycles and microseconds), signal-to-quantization noise ratio (SQNR), power consumption (P), and normalized power (NP). All architectures listed in the table are implemented on either Virtex-6 (V6) or Virtex-7 (V7) FPGA platforms.

In terms of hardware resources utilization, the proposed MSC architecture achieves competitive area efficiency. It utilizes only 12 DSPs and 4 BRAMs, significantly fewer than those required by previous MDC-based architectures. [1]–[3]. Among recent high-radix FFT designs, the proposed MSC architecture achieves the lowest Slices count, reducing area by 8.5% compared to the prior MSC implementation [5] and by 44% relative to the CM-based design [4].Although the LUTs count of proposed architecture is comparable to [5] , the proposed design reduces flip-flops usage by 17.3%, indicating that the use of time-multiplexed constant multiplication contributes to a more resource-efficient implementation.

Regarding operating frequency and power efficiency, the proposed implementation operates at 493 MHz, outperforming all MDC-based counterparts [1]–[3] and recent MSC implementation [5]. This enables a throughput of 1820 MS/s, surpassing [5] by 8.3%. The latency is measured at 307 clock cycles (0.62 μs), offering a balanced trade-off between pipeline depth and processing speed. In terms of power, the design consumes 1.16 W, slightly higher than [5] (0.98 W) but 31% lower than [4] (1.68 W). Critically, the normalized power metric (NP = power per MHz) of proposed design is 2.35 mW/MHz, nearly identical to [5] (2.33 mW/MHz), confirming that the optimization using time-multiplexed constant multiplication contributes to low-power operation across frequency scaling.

To further quantify the trade-offs between hardware cost, speed, and energy efficiency, we introduce two composite metrics: the resource-delay product (RDP = Slices $\cdot$ T_{CLK}) and the power-delay product (PDP = Power $\cdot$ T_{CLK}), where $T_{\text{CLK}} = 1/f_{\text{CLK}}$. These metrics jointly capture area–time and power–time efficiencies, respectively, providing a more holistic view of design quality beyond isolated figures. The proposed MSC architecture achieves the lowest RDP among all compared designs, with a value of 2995.94, representing a 22.1% reduction over the previous best MSC implementation [5] and a 29.2% improvement over the CM-based approach [4]. As for PDP, the proposed design reports 2352.94 mW·ns, which is comparable to [5] and 5% lower than [4] —note that lower PDP is better. The near-parity in PDP between our work and [5], combined with significantly better RDP, demonstrates that the time-multiplexed constant multiplication offers a more balanced optimization across resources, speed, and power dimensions.

With respect to numerical accuracy, the SQNR of the proposed MSC FFT (49.46 dB) is comparable to that of [5] (50.16 dB), differing by less than 1 dB, yet considerably higher than the 40.30 dB reported in [1]. This demonstrates that the time-multiplexed constant multiplication strategy employed in the MSC architecture effectively preserve signal fidelity despite resource reduction.

V. Conclusion

This brief presents a 4-parallel 1024-point MSC FFT architecture optimized by time-multiplexed constant multiplication. Experimental results on a Xilinx Virtex-7 FPGA show that the proposed architecture outperforms several state-of-the-art FFT implementations in terms of resources efficiency, operating frequency, power consumption, and accuracy. The use of time-multiplexed constant multiplication proves to be an effective strategy for resource-efficient FFT implementations.

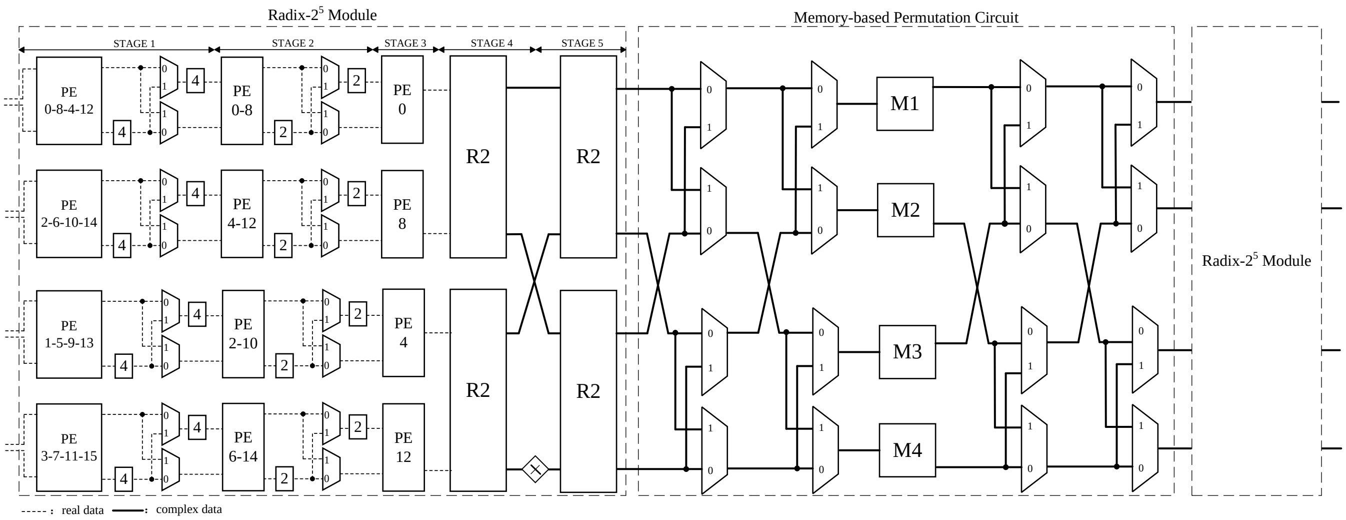

Fig. 1. abot

TABLE I
Comparison of Hardware Resources and Performance for 4-Parallel 1024-Point Pipeliend FFT Implemented on FPGA

	[1]	[2]	[3]	[4]	[5]	Proposed
Architecture	MDC	MDC	MDC	CM	MSC	MSC
Radix	2^5	2^2	2	2^5	2^5	2^5
WL	16	16	16	16	16	16
Slices	1420	1351	-	2631	1615	1477
LUTs	-	-	4116	-	4682	4629
FFs	-	-	1920	-	5910	4887
DSPs	16	48	72	12	12	12
Block-RAMs	12	12	0	0	4	4
f_{CLK} (MHz)	253	227	380	680	420	493
Th. (MS/s)	1012	910	1520	2720	1680	1820
Latency (cyc.)	265	285	767	394	300	307
Latency (µs)	1.04	1.25	2.02	0.58	0.71	0.62
SQNR (dB)	40.30	-	-	-	50.16	49.46
P (W)	-	-	-	1.68	0.98	1.16
NP ($\frac{\text{mW}}{\text{MHz}}$)	-	-	-	2.47	2.33	2.35

-:Not provided

TABLE II
Comparison of Resource-Delay Product (RDP) and Power-Delay Product (PDP). PDP = Power · T_{CLK} (mW·ns) , RDP = Slices · T_{CLK} (slices · ns)

	[1]	[2]	[3]	[4]	[5]	Proposed
Architecture	MDC	MDC	MDC	CM	MSC	MSC
Radix	2^5	2^2	2	2^5	2^5	2^5
PDP(mW·ns)	-	-	-	2470.61	2333.38	2352.94
RDP	5612.69	5951.56	-	3869.15	3845.32	**2995.94**

-: Not provided

References

[1] M. Garrido, S.-J. Huang, and S.-G. Chen, "Feedforward fft hardware architectures based on rotator allocation," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 2, pp. 581–592, 2018.

[2] M. Garrido, M. Acevedo, A. Ehliar, and O. Gustafsson, "Challenging the limits of fft performance on fpgas (invited paper)," in *2014 International Symposium on Integrated Circuits (ISIC)*, 2014, pp. 172–175.

[3] A. X. Glittas, M. Sellathurai, and G. Lakshminarayanan, "A normal i/o order radix-2 fft architecture to process twin data streams for mimo," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 6, pp. 2402–2406, 2016.

[4] M. Garrido and P. Malagón, "The constant multiplier fft," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 1, pp. 322–335, 2021.

[5] Z. Kaya and M. Garrido, "Optimized 4-parallel 1024-point msc fft," *IEEE Access*, vol. 12, pp. 84 110–84 121, 2024.