IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023

DOI: 10.17148/IJARCCE.2023.12694

A REVIEW ON SYNCHRONOUS & ASYNCHRONOUS FIFO DESIGN

Shashank C Pai[1], Vishwitha A[2], Rakshath[3], Sathwik Bhat[4], Shreya[5]

Student, Electronics & Communication Engineering, MITE, Moodabidri, India[1]

Assistant Professor, Electronics & Communication Engineering, MITE, Moodabidri, India[2]

Student, Electronics & Communication Engineering, MITE, Moodabidri, India[3]

Student, Electronics & Communication Engineering, MITE, Moodabidri, India[4]

Student, Electronics & Communication Engineering, MITE, Moodabidri, India[5]

Abstract: Because of flexibility of application and highest performance, thrills, and middle end for an obtained extensive market. As a fundamental memory structure. The FIFO is widely used in FPGA based projects. But limited by the resources in chip and imperfections of development tools, the problem of insufficient memory while the overall capacity is often enough occurring in implementation of multi-channel FIFO. This project surveys various occasion applications of FIFO and puts forward the implementation of FIFO Memory Using Shift registers.

Keywords: FIFO, NoC, FPGA, Synchronous, Asynchronous,

I. INTRODUCTION

Data can be transferred through various communication channels, including copper wires, optical fibres, wireless channels, storage media, and computer buses. High-speed data transfer was achieved through parallel/serial transmission. Parallel transmission groups binary data and transmits them in frames/streaming using n wires. Serial transmission involves transmitting one bit at a time, requiring only one communication channel, which reduces transmission costs.

However, conversion devices are required at the interface between the sender and the line and between the line and the receiver. Serial transmission can be asynchronous or synchronous. FPGA tech is widely used due to its low cost, high performance, maturity, and short design cycle.

FIFO is a fundamental storage structure used as a data buffer for signal processing systems, communication bridges between networks of different data rates, and communication interfaces between modules in different clock domains.

Multi-channel FIFO combines FIFOs into four categories: SISO FIFO, SIPO FIFO, PISO FIFO, and PIPO FIFO. SISO FIFO channels are independent, while SIPO FIFO is written channel by channel but read out simultaneously. PISO FIFO is written in parallel but read channel by channel, and PIPO FIFO is written and read simultaneously in parallel.

II. LITERATURE REVIEW

Dr. Ashok Kumar K et.al.,[1] proposed a paper "Advanced FIFO Structure for Router in Bi-NoC". In this paper, the author proposes an advanced FIFO-based memory unit in NoC router to reduce the burden on the router and improve FIFO internal structure, thus improving the performance of Bi-NoC. The proposed work shows a 28% improvement in delay and 17% resource utilization compared to previous work. Power consumption increases through virtual channels at an advanced FIFO structure, so router components need improvement to enhance NoC performance.

Ameer M.S. qbdelhadi et.al.,[2] presented a paper on Synthesizable synchronization FIFOs gives a design of a low power consumption architecture for NoC, designers can reduce buffer power consumption. Existing techniques optimize virtual channel buffer power with buffer sharing, power gating, and Dynamic Voltage Frequency Scaling. This paper proposes two clock gating techniques, enabling clock gating at input channel BUFFERS, and applying clock gating on FPGA slices. The approach significantly improves FIFO buffer power in 2D NoC by 10.70% and reduces power consumption by approximately 39% when Xilinx intelligent clock gating is applied in the NoC.

DOI: 10.17148/IJARCCE.2023.12694

Ann Gordon-Ross et.al.,[3] proposed a paper titled "A One-Cycle FIFO Buffer for Memory Management Units in Manycore Systems". FIFO architectures require many registers, increasing power consumption and area overhead. The paper [3], proposed design leverages two-phase clocks, a two-ported SRAM memory 8T-Cell, and simplified flag circuitry to provide high-speed operation and large storage capacity. Future work will adapt the design for asynchronous behaviour.

Xu, Jinfu, et.al., [4] published a paper titled "An unified online fault-tolerant mechanism for FIFO in network-on-chip router." As multi-core network-on-chip (NoC) designs become more complex, router failures caused by production defects, process deviation, material radiation, and aging are becoming more serious. online fault-tolerant mechanism for FIFO in NoC routers, which includes a fault detection algorithm and reliable circuit architecture. The proposed mechanism improves throughput and latency with injected faults, and the area overhead is increased by 12%.

Ghoshal, Bibhas, et.al., [5] published paper titled "In-field test for permanent faults in FIFO buffers of NoC routers." An on-line transparent test technique for detecting latent hard faults in NoC router is first input first output buffers is presented in [5], The proposed technique involves repeating tests periodically to prevent the accumulation of faults. A prototype implementation of the proposed test algorithm has been integrated into the router channel interface and has been tested with synthetic self-similar data traffic. An additional on-line test technique for the routing logic has also been proposed, which considers utilizing the header flits of data traffic movement in transporting the test pattern.

MAshour, Haytham, et.al., [6] published paper titled "Design, simulation and realization of a parameterizable, configurable and modular asynchronous FIFO." presents the design and simulation of an asynchronous FIFO that is parameterizable in data interface width and memory depth. The FIFO flag thresholds are re-configurable at run time and are using a modular design approach in its implementation and in interfacing with other system components. The design approach followed is modular in its architecture which means each module in the architecture is implementing an isolated and self-contained set of functions that can be reused in any other system. This approach resulted in the asynchronous FIFO itself becoming a modular design offering a modular interface to the other system level components. The reconfigurability of the FIFO at run time finds its applications in applications like rate matching, re-configurable hardware, and data streaming applications.

Saleh Abdel-Hafeez et.al.,[7] proposed a paper titled "A One Cycle Asynchronous FIFO Queue Buffer Circuit". An energy-efficient asynchronous (clockless) FIFO memory design that operates on the handshake signalling that is Request and Acknowledge signals is presented in. The design generates the Acknowledge signal based on the received Request signal using a new asynchronous circuit that controls the operation of the FIFO. Thus, the design can be considered as a potential technology of choice for low-power applications such as IoT communication solutions. The design can operate 5X faster at the same supply voltage compared to prior work. The design's total power consumption is 2 mW with a total transistor count of 34,470 at 65 nm and 1 V power supply.

Sharmaa, Shruti et.al.,[8] Published a paper gives asynchronous and synchronous design topologies of FIFO being constructed based on a scheme where data movement is avoided in the FIFO. Pipelining is a practical technique for high-performance digital system design that is used to maintain parallelism and increase system throughput. The principles of synchronous and asynchronous logics are based on strict protocols, which are timed-driven and demand-driven. To cope with variations in process, voltage, and temperature, most works use a handshaking process that specifies a token of request or acknowledgement. Transparent latches in the pipeline allow the next data item to enter, and asynchronous design is integrated to avoid critical delays, clock skew, and power consumption control.

The four-phase handshake protocol is used to achieve an asynchronous FIFO environment, and the early acknowledgement protocol is an improvement above the normal four-phase protocol. Asynchronous communication protocols for control and synchronization in integrated circuits are a clear requisite for the semiconductor industry, and the design approach based on globally asynchronous locally synchronous (GALS) in integrated circuits is being emphasized. The presented method for designing asynchronous pipelines avoids predictable syntax directed translation tactics, and a data-driven design style is used to synthesize an asynchronous control system that can easily integrate into conventional synchronous design styles.

Gengting Liu et.al., [9] presented a study about the usage of packet switched NoC. The use of packet switched NoC architectures and GALS networks to integrate design blocks in different clock domains is discussed. It proposes a new asynchronous interface FIFO design optimized for FPGA implementation, which utilizes D-type flip flops for fast NRZ synchronization and Johnson-encoded asynchronous pointers to improve performance. The design is immune to layout delays and skew and is designed as macros for easy implementation and improved portability. The proposed asynchronous FIFO can be applied to other m-of-n protocols and an extended ASIC version can be developed for ASIC GALS systems.

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 6, June 2023
DOI: 10.17148/IJARCCE.2023.12694

Ameer M.S. qbdelhadi et.al.,[10] presented a paper on Synthesizable synchronization FIFOs. The issue related to modern chip design due to clock distribution challenges is discussed Globally asynchronous, locally synchronous (GALS) design provides timing independence, simplifying closure, supporting design reuse, and enabling energy efficiency. For efficient, high-performance systems, optimizing the asynchronous interfaces between timing domains is critical. The current trend is to partition systems-on-chip designs into multiple clock domains, leading to large numbers of clock-domain crossings. GALS design enables asynchronous network-on-chip (ANoC) and synchronizing FIFOs to interface between domains, improving performance.

III. SUMMARY

The papers reviewed above discuss a variety of methods for designing and implementing FIFO buffers. FIFO buffers are used in a wide variety of applications, including network-on-chip routers, memory management units, and asynchronous communication systems. The review also discusses about several different design challenges, such as power consumption, fault tolerance, and testability. They also present several different solutions to these challenges. One of the key challenges in designing FIFO buffers is power consumption. The papers discuss several different techniques for reducing power consumption, such as using architectural alternatives, reducing the number of clock cycles per operation, and using low-power components. Another key challenge in designing FIFO buffers is fault tolerance. An insight is provided on several different techniques for improving fault tolerance, such as using redundant components, error detection and correction, and online fault-tolerance mechanisms. The review depicts the importance of testability in FIFO buffer design and several different techniques for testing FIFO buffers, such as using boundary-scan testing, functional testing, and structural testing.

IV. CONCLUSION

Overall, the review provides a comprehensive overview of the design and implementation of FIFO buffers. They discuss several different challenges and solutions, and they provide valuable insights for engineers who are designing or implementing FIFO buffers.

REFERENCES

[1] Dr Ashok Kumar and V. Karunakar Reddy. "*Advanced FIFO Structure for Router in Bi-NoC*." 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS). IEEE, 2021.
[2] Sharma, Ashish, et al. "*Reducing fifo buffer power using architectural alternatives at rtl*." 2016 20th International Symposium on VLSI Design and Test (VDAT). IEEE, 2016
[3] Gordon-Ross, Ann, Saleh Abdel-Hafeez, and Mohammed Hammam Alsafrjalni. *"A One-Cycle FIFO Buffer for Memory Management Units in Manycore Systems."* 2019 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). IEEE,2019
[4] Xu, Jinfu, et al. *"An unified online fault-tolerant mechanism for FIFO in network-on-chip router."* 2016 13th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT). IEEE, 2016
[5] Ghoshal, Bibhas, et al. *"In-field test for permanent faults in FIFO buffers of NoC routers."* IEEE Transactions on VLSI Systems 24.1 (2015): 393-397
[6] MAshour, Haytham. *"Design, simulation and realization of a parameterizable, configurable and modular asynchronous FIFO."* 2015 Science and Information Conference (SAI) IEEE, 2015.
[7] Abdel-Hafeez, Saleh, and Muhannad Quwaider. "*A one-cycle asynchronous FIFO queue buffer circuit*." 2020 11thInternational Conference on Information and Communication Systems (ICICS). IEEE, 2020
[8] Sharmaa, Shruti. "*Implementation of an RTL synthesizable asynchronous FIFO for conveying data by avoiding actual data movement via FIFO*." 2015 6th International Conference on Computing, Communication and Networking Technologies (ICCCNT). IEEE, 2015
[9] Liu, Gengting, et al. *"Asynchronous interface FIFO design on FPGA for high-throughput NRZ synchronisation."* 201727th International Conference on Field Programmable Logic and Applications (FPL), IEEE, 2017.
[10] Abdelhadi, Ameer MS. "*Synthesizable synchronization FIFOs utilizing the asynchronous pulse-based handshake protocol*." 2020 IEEE Nordic Circuits and Systems Conference (NorCAS). IEEE, 2020.