ECE-593: Fundamentals of Pre-Silicon Validation

Maseeh College of Engineering and Computer Science

Winter, 2025

Project Name: Design and Verification of Asynchronous FIFO

Members: Vaishnavi Sharad Pandhare, Mrudula Chekuri, Venkat Sahith Reddy Cheedu, Rohith Sri Krishna Inti.

Date: 04/20/2025.

Project Name	Design and Verification Of Asynchronous FIFO.
Location	Portland.
Start Date	04/20/2025.
Estimated Finish Date	05/22/2025.
Completed Date	

Prepared by: Group-8	
Prepared for: Prof. Venkatesh Patil	
Team Member Name	Email
Vaishnavi Sharad Pandhare	pandhare@pdx.edu
Mrudula Chekuri	mrudulac@pdx.edu
Rohith Sri Krishna Inti	rohithin@pdx.edu
Venkat Sahith Reddy Cheedu	cheedu@pdx.edu

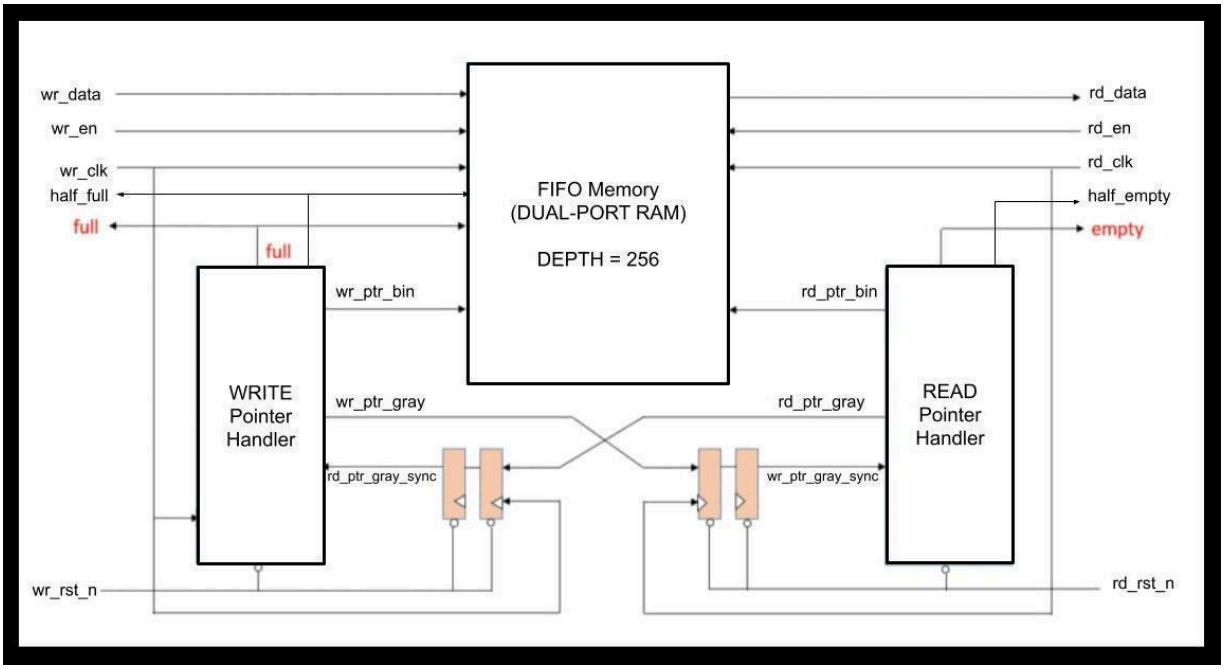
Design Features:
The read and write operations of FIFO are in different clock domains. The Depth and Width of the FIFO are parametrizable where depth indicates the no.of entries and address width is derived from depth. The dual-port memory uses register array to store data. Write logic and Read logic are used for writing and reading the data on the rising edge of wr_clk and rd_clk. Synchronizes read pointer to write domain and vice versa using 2 stage synchronizers. Full is asserted when the write pointer is one step ahead of the read pointer. Empty is asserted when the read pointer is one step ahead of the write pointer. Binary to Gray code conversion is used for pointer comparison across clock domains to prevent metastability.

Project Description:
This project focuses on the Design and Verification of an Asynchronous FIFO using SV and UVM for the design and testbench verification. It ensures safe clock domain crossing, reliable and robust data transfer to thoroughly verify its correctness, performance and corner cases using a scalable and reusable verification methodology. The FIFO is responsible for the data transfer between two clock domains operating asynchronously. The design module includes essential components such as Dual-clock operation for write and read sides, Pointer synchronization using Gray code, Status flag generation(full, empty, reset).

Important Signals/Flags
Control and Status flags: full – indicates when FIFO is full. empty – indicates when FIFO is empty. half_full – indicates when FIFO is 50% full. half_empty – indicates when FIFO is 50% empty. wr_en – write enable from write clock domain. rd_en – read enable from read clock domain . wr_ptr_bin, rd_ptr_bin – binary write and read pointers wr_ptr_gray, rd_ptr_gray – gray write and read pointers.

Design Signals
wr_clk – write clock domain which contros write side operations. wr_rst_n – write domain active low reset. wr_data – data to be written in FIFO. rd_clk – read clock domain which controls read side operations. rd_rst_n – read domain active low reset. rd_data – data to be read from FIFO.

Block Diagram



The above block diagram represents the architecture of the asynchronous FIFO which consists of a Dual-port RAM which has the depth of 256, a READ pointer handler and a WRITE pointer handler. wr_data, wr_en, wr_clk, rd_en, rd_clk are all the input signals to the FIFO memory and rd_data, half_full, full are output signals. Write pointer handler block manages write operations and generates write address pointers. Read pointer handler block manages read operations and generates read address pointers. Pointer synchronization blocks synchronize gray coded pointer across clock domains to prevent metastability.