

# DESIGN SPEC DOCUMENT

ECE-593: Fundamentals of Pre-Silicon Validation  
Maseeh College of Engineering and Computer  
Science  
Winter, 2025



## **Project Name:**

Design and Verification of an Asynchronous FIFO

## **Members:**

Pavan Gaddam, Gene Hu, Surya Teja Purma, Kai Roy

## **Date:**

4/20/2025

<b>Project Name</b>	Design and Verification of an Asynchronous FIFO
<b>Location</b>	Portland State University
<b>Start Date</b>	4/10/2025
<b>Estimated Finish Date</b>	5/25/2025
<b>Completed Date</b>	N/A

<b>Prepared by:</b> Team 5
<b>Prepared for:</b> Prof. Venkatesh Patil

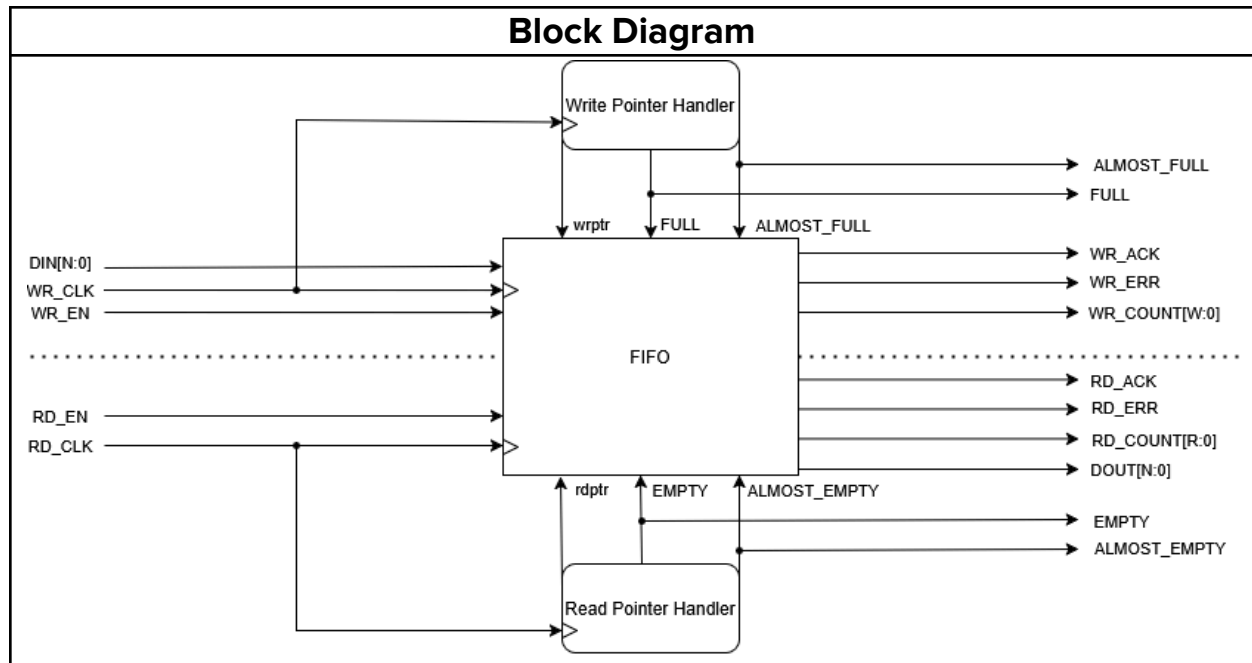
<b>Team Member Names</b>	<b>Emails</b>
Pavan Gaddam	pavang@pdx.edu
Gene Hu	ghu@pdx.edu
Surya Teja Purma	purma@pdx.edu
Kai Roy	roykai@pdx.edu

<b>Design Features</b>
Supports data widths up to 256 bits
Supports memory depths of up to 65,535 locations
Fully synchronous and independent clock domains for the read and write ports
Supports FULL and EMPTY status flags
Optional ALMOST_FULL and ALMOST_EMPTY status flags
Invalid read or write requests are rejected without affecting the FIFO state

Project Description
<p>The team will implement an asynchronous FIFO using SystemVerilog, and perform thorough testing to confirm correct functionality in accordance with design features.</p> <p>There will be a functional test plan drafted which will assign responsibilities to team members for their part in the verification process. These responsibilities include the DUT and its components, design documentation, testbench development, verification plan, writing tests, running simulations, generating reports, and debugging.</p>

Important Signals/Flags		
Signal Name	Signal Direction	Description
DIN[N:0]	input	Data input
WR_EN	input	Write enable (request)
WR_CLK	input	Clock for write domain operations (rising edge)
RD_EN	input	Read enable (request)
RD_CLK	input	Clock for read domain operations (rising edge)
AINIT	input	Asynchronous reset of all FIFO functions, flags, and pointers. The four FIFO flags will reset to active high, and deactivated on their next respective clock pulse
FULL	output	Full: no additional writes can be performed, synchronous to WR_CLK
ALMOST_FULL	output	Almost Full: only one additional write can be performed before FIFO is FULL, synchronous to WR_CLK
DOUT[N:0]	output	Data Output: synchronous to RD_CLK
EMPTY	output	Empty: no additional reads can be performed, synchronous to RD_CLK
ALMOST_EMPTY	output	Almost Empty: only one additional read can be performed before FIFO is EMPTY, synchronous to RD_CLK

Design Signals	
Signal Name	Description
wrptr	Pointer from the write pointer handler for keeping track of where in memory the FIFO should place its next write
rdptr	Pointer from the read pointer handler for keeping track of where in memory the FIFO should retrieve its next read



note: some signals in diagram were not implemented

References/Citations
[1] Xilinx Asynchronous FIFO V3.0 design specification