DESIGN SPEC DOCUMENT

ECE-593: Fundamentals of Pre-Silicon Validation

Maseeh College of Engineering and Computer Science

Winter, 2025

Project Name: Design, Implementation and Verification of Asynchronous FIFO

Members: Sai Charan Teja Balne

Akshay Kumar Medishetti

Steeva Murikipudi

Date:01/30/2026

Project Name	Asynchronous FIFO Design, Implementation, and Verification
Location	Portland, Oregon, USA
Start Date	January 2026
Estimated Finish Date	March 2026
Completed Date	

Prepared by: Team Number 8	
Prepared for: Prof. Venkatesh Patil	
Team Member Name	Email
Sai Charan Teja Balne	saibalne@pdx.edu
Akshay kumar Medishetti	amedishe@pdx.edu
Steeva Murikipudi	steeva@pdx.edu

Design Features:
Dual-Clock Domain Architecture: Enables safe data transfer between independent clock domains operating at different frequencies.
Gray Code Counters: Implements synchronous pointer counters to minimize metastability risks
Pointer Synchronization: Two-stage flip-flop synchronizers (ptr_sync) for reliable clock domain crossing
Status Flags: Full, Empty, and intermediate flag conditions (3/4 full, 1/4 empty) for flow control
Memory System: Dedicated FIFO memory with independent write and read address buses
Write/Read Control Logic: Separate controllers managing pointer updates and flag generation

Project Description:
An Asynchronous FIFO (First-In-First-Out) buffer is a critical digital design component that facilitates safe data transfer between two independent subsystems operating on different clock domains. This design ensures reliable communication by preventing data loss, duplication, and metastability issues when write and read clocks operate at different frequencies.

Depth Calculation
Given Specifications: 1. Sender writes data at 500 MHz with no idle cycles. 2. Maximum write burst: 450 items. 3. Receiver reads data at 500 MHz with 4 idle cycles between reads. 4. Each read operation takes 5 clock cycles (read + 4 idle). Calculation Steps: 5. Time to write one item: 1/500 MHz = 2.0 ns. 6. Total write time for 450 items: 450 × 2.0 ns = 900 ns. 7. Time to read one item: 5/500 MHz = 10 ns. 8. Items read during write burst: 900 ns / 10 ns = 90 items. 9. FIFO depth needed: 450 – 90 = 360 items. 10. Closest power of 2: 2^9 = 512 entries. 11. Address Width Required: 9 bits

Key Design Objectives:
12. Data Integrity: Ensure no data loss, corruption, or duplication across clock domains. 13. Metastability Mitigation: Implement proper synchronization mechanisms. 14. Flag Generation: Accurately detect full, empty, 3/4 full, and 1/4 empty conditions. 15. Scalability: Support parameterized data and address widths.

IMPORTANT SIGNALS/FLAGS

Write Domain Signals

Signal	Type	Width	Description
wclk	input	1	Write domain clock (500 MHz)
Wrst_n	Input	1	Write domain asynchronous reset (active low)
wen	Input	1	Write enable signal
Wdata[7:0]	Input	8	Data to be written into FIFO
wfull	output	1	FIFO full flag (cannot write more data)
waddr[8:0]	internal	9	Write address pointer (binary)
Wptr_gray[8:]	internal	9	Write pointer in Gray code format

Read Domain Signals

Signal	Type	Width	Description
rclk	Input	1	Write domain clock (500 MHz)
rrst_n	Input	1	Read domain asynchronous reset (active low)
ren	Input	1	Read enable signal
rdata	Output	8	Data read from FIFO
rempty	Output	1	FIFO empty flag (no data available)
raddr[8:0]	Output	9	Read address pointer (binary)
rptr_gray[8:0]	Output	9	Read pointer in Gray code format

Synchronized Pointer Signals

Signal	Domain	Direction	Description
rptr_gray_sync[8:0]	Write	Internal	Synchronized read pointer (in write clock domain)
wptr_gray_sync[8:0]	Read	Internal	Synchronized write pointer (in read clock domain)

Flag Definitions

1. Full Flag (wfull):

Asserted when write pointer catches up to the synchronized read pointer

Prevents data overwriting by blocking write operations

Condition: wptr_gray == {~rptr_gray_sync[8:8], rptr_gray_sync[7:0]}

2. Empty Flag (rempty):

Asserted when read pointer matches synchronized write pointer

Prevents reading from empty FIFO

Condition: rptr_gray == wptr_gray_sync

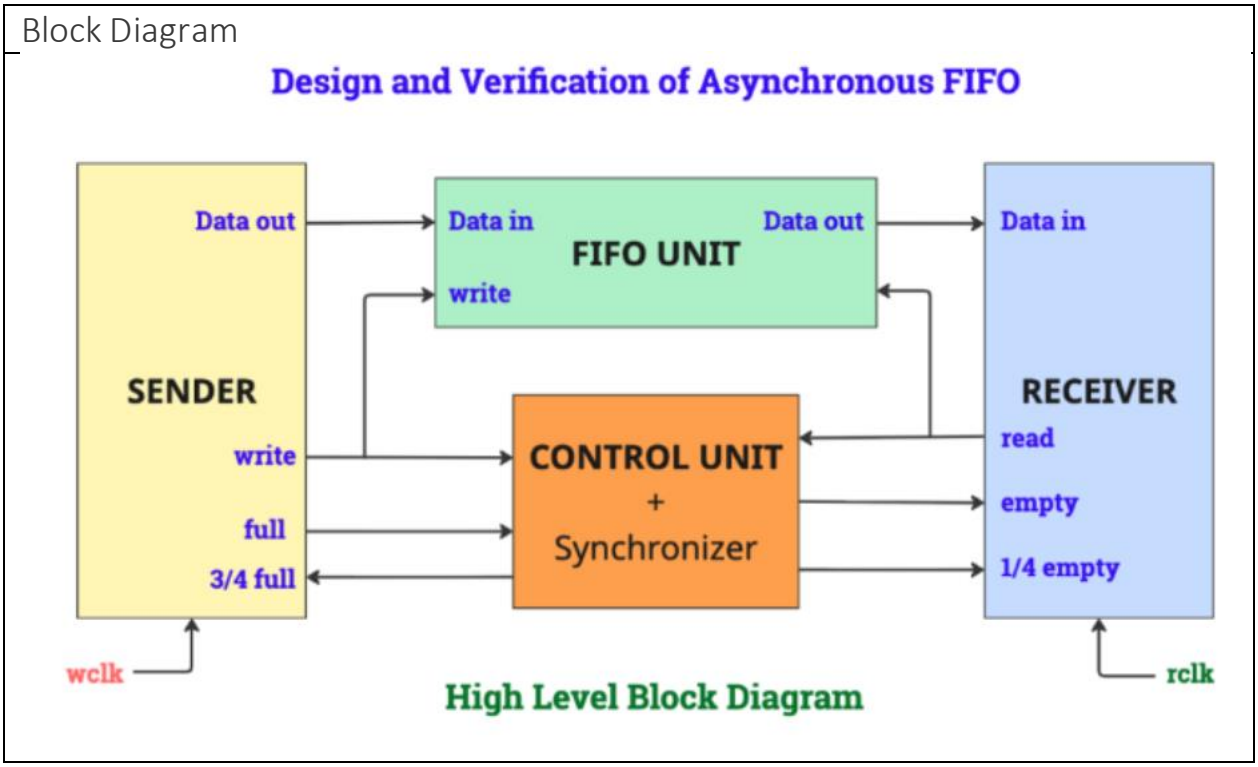
3. 3/4 Full Condition:

Triggered when FIFO occupancy exceeds 75% (>384 of 512 entries)

4. 1/4 Empty Condition:

Triggered when FIFO occupancy drops below 25% (<128 of 512 entries)

Block Diagram



References/Citations

1. Cliff Cummings. "Asynchronous & Synchronous Reset Design Techniques - Part Deux." SNUG San Jose, 2003.
2. Cliff Cummings. "Simulation and Synthesis Techniques for Asynchronous FIFO Design." SNUG San Jose, 2002.
3. Sunburst Design. "Asynchronous FIFO Design." Technical Documentation.
4. .Accellera SystemVerilog Language Reference Manual (IEEE 1800-2017).
5. Asynchronous FIFO Design – rfwireless-world
6. FIFO Architecture, Functions, and Applications – Texas Instruments
7. Asynchronous FIFO – VLSI Verify