# Lab 3. Simple DMA Design

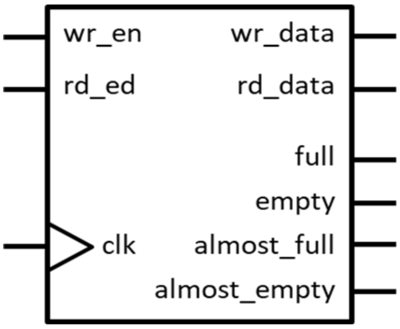
## I. Purpose

The purpose of this experiment is to learn about the role of FIFO and DMA, and to implement a simple DMA and verify its functionality with an FPGA.

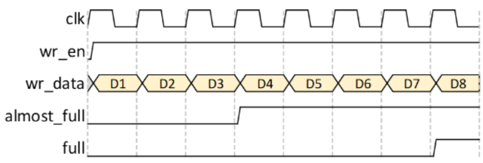
## II. Backgrounds

### What is FIFO?

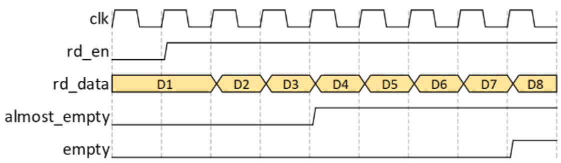
**First In, First Out** (FIFO) is a method for organizing the manipulation of a data structure (data buffer) where the oldest entry, or "head" of the queue, is processed first. FIFO consists of a set of read/write pointers, storage and control logic. Storage may come in different types of memory (e.g., SRAM, flip-flops, and latches). For typical FIFO design, a dual-port SRAM is used, where one port is dedicated to writing and the other to reading. Note that, when read/write enable signal is asserted, data are presented at the same clock cycle.



**FIFO**



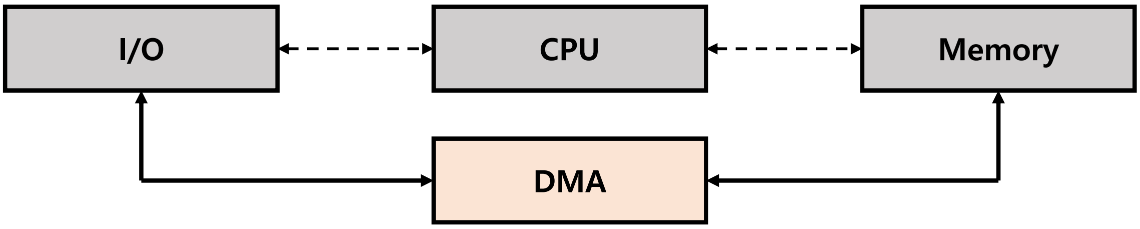
*Example of FIFO's Data Write Operation*



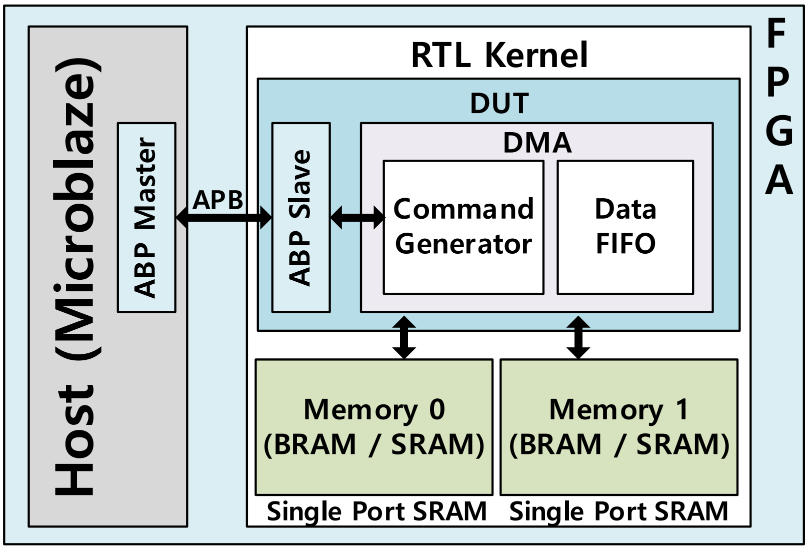
*Example of FIFO's Data Read Operation*

### What is DMA?

DMA is a separate engine to offload data movement between memory from CPU. DMA operates through the following process: First, CPU describes the transfer information to the DMA. This information includes source address of memory, destination address of memory, transfer size of data to move, etc. After that, when the CPU sends a start signal, DMA performs memory read / writes without help of CPU. After DMA finish its work, DMA completes transfer by sending interrupt to CPU. During DMA work. CPU can do other job during the DMA operation, and DMA can communicate with memory faster than CPU can do.



## III. Lab Procedure



### Problem 3A. Design FIFO

1) Design your own FIFO module with System Verilog
   - A. **Note that we would verify your design for the transfer size 1~32.**
2) Verify your module through simulation
3) Understand the FIFO design with your module

### Problem 3B. Design simple DMA

1) Design your own DMA that includes Problem 3A's FIFO
2) Verify operation of simple DMA through simulation
3) Compile, synthesize, implement your DMA on FPGA
4) Please follow the lab guide file for more detailed information

### Problem 3C. Verify of DMA on FPGA

1) Verify/demo your system on FPGA

**For design details, please refer to the provided supplemental material**

## IV. Final Report

Followings **should** be included in the report

1. Screen capture of simulation result of DMA (FIFO operation, data verification scenario)
2. Screen capture of synthesis & implementation result
3. Screen capture of register setting (host – RTL kernel communication process)
4. Picture of LED result should be included in the final report
5. Discussion for the following questions
    1) Describe each module of your own RTL code.
    2) The FIFO used in this project has only one clock input, which is synchronous clock. If you want to design asynchronous FIFO that has different clock signals for data read and data write, what needs to be modified?
       Draw block diagram and explain your design. You can use FSM for explaining your design. (you don't have to write codes)
    3) In simple DMA project, we did not consider the granularity of data in memory for simplicity.
       Explain what are the considerations and how to solve if we want DMA to operate properly even for the non-aligned data.
       (For example, in case of Source Address = 0x0001_0003,
       Destination Address = 0x0002_0014, and size = 8)

## V. References