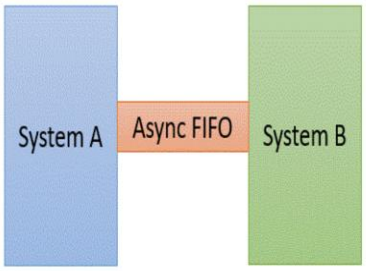
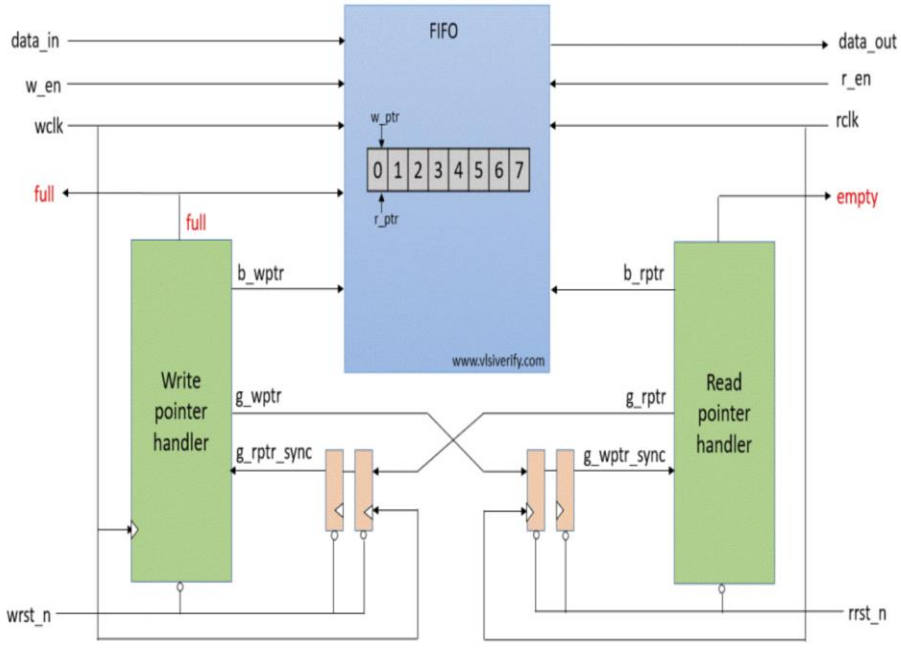
ASYNCHRONOUS FIFO

Asynchronous FIFO

In asynchronous FIFO, data read and write operations use different clock frequencies. Since write and read clocks are not synchronized, it is referred to as asynchronous FIFO. Usually, these are used in systems where data need to pass from one clock domain to another which is generally termed as 'clock domain crossing'. Thus, asynchronous FIFO helps to synchronize data flow between two systems working on different clocks.



Asynchronous FIFO Block Diagram



Asynchronous FIFO

Asynchronous FIFO Operation

In the case of synchronous FIFO, the write and read pointers are generated on the same clock. However, in the case of asynchronous FIFO write pointer is aligned to the write clock domain whereas the read pointer is aligned to the read clock domain. Hence, it requires domain crossing to calculate FIFO full and empty conditions. This causes metastability in the actual design. In order to resolve this metastability, 2 flip flops or 3 flip flops synchronizer can be used to pass write and read pointers. For explanation, we will go with 2 flip-flop synchronizers. Please note that a single "2 FF synchronizer" can resolve metastability for only one bit. Hence, depending on write and read pointers multiple 2FF synchronizers are required.

Design modules and TB :

TOP_module ;

FIFO_module ;

Read_pointer_handler ;

Write_pointer_handler ;

2FF_synchronizers_module ;

TOP_module_TB ;

TOOLS :

Simulation : Questasim . (using file.do to automation)

RTL_elaboration : Vivado .

Synthesis : Vivado .