**FIFO Lab with Basys 3(Arty7)**

FIFO: first in first out, it is a register file with control circuitry.

A diagram of a computer

Description automatically generated

* Write point will write to FIFO first.
* Read pointer will read from FIFO.
* This will keep going until FIFO is empty again.
* Will not be able to write when FIFO is full.

UART: Universal asynchronous receiver/transmitter

* Defines a protocol or set of rules for exchanging serial data between two devices.
* Only uses 2 wires between TX and RX.

A diagram of a computer system

Description automatically generated

* FIFO block is after RX block.

A computer screen shot of a diagram

Description automatically generated

* Switches 2 1 0 for the 3 bits of data.
* LED0 will be for fifo empty
* LED15 will be for FIFO full
* LED7:5 will be for Data to read

Modules in this Labs:

1. Fifo\_test.v(top)
2. Debounce\_explicit.v
3. Fifo.v

Fifo\_test.v(TOP)

* The purpose of this module is to instantiate and wire the other two modules
* Debouce module will split into two instantiations.
  + One for read pointer
  + One for write pointer
* Fifo module will be instantiate as well with the following wires:
  + .clk(clk\_100MHz),
  + .reset(reset),
  + .write\_to\_fifo(write),
  + .read\_from\_fifo(read),
  + .write\_data\_in(sw),
  + .read\_data\_out(data\_out),
  + .full(full),
  + .empty(empty)

Debounce\_explicit.v

* Purpose:
  + This module is to set a counter for the button so that read and write does not switch multiple times because the signal bounces.
* By setting a counter period of ~40ms, the bounce is prevented.

Fifo.v

* Purpose:
  + This module is to declare register signals for read and write
  + Memory address array is also declared with 3 bits
* Operation:
  + Write enable operation with always block.
    - Read data only from the current read address.
    - Write enable only if ~fifo\_full
  + Next state always block.
    - Register and FIFO control logic
    - Write address = write buffer
    - Ready address = read buffer
  + Next state always block logic.
    - Write address pointer will increment.
    - Read address pointer will increment.
  + Case
    - 4 cases:
      * No button is pressed
      * Write button is pressed
      * Read button is pressed
      * Both buttons are pressed
  + Outputs
    - Assign full and empty signals

Constraints file:

* Purpose:
  + This file is to define the port requirements needed for this lab.
* Data in:
  + Switches will be used for size 3 bits data
  + 001: V17
  + 010: V16
  + 100: W16
* Data out:
  + 0 : U15 LED
  + 1: U14 LED
  + 2: V14 LED
* signal indicator:
  + Full: L1
  + Empty: U16 LED
* Button function:
  + Reset: U18
  + Write: T18
  + Read: U17

Observation:

* Set data into value 7, press write button, data will be written to FIFO address 0.
* Set data again to value 6, press write button, data will be written to next FIFO address 1.
* Press read button to read data value 7, data out will display value 6.
* Press read again to read data value 6, data out will display 7 and FIFO EMPTY signal.