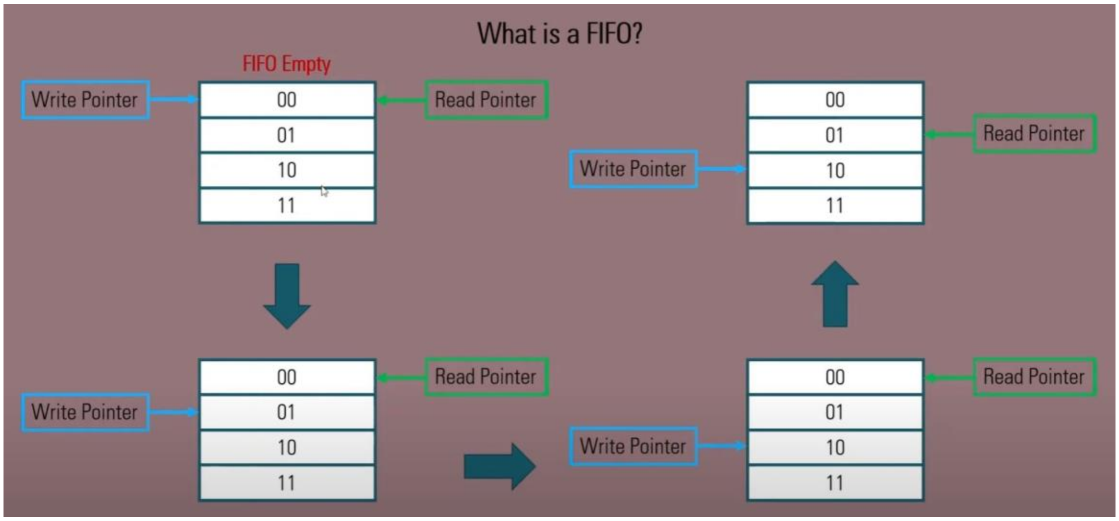
FIFO Lab with Basys 3(Arty7)

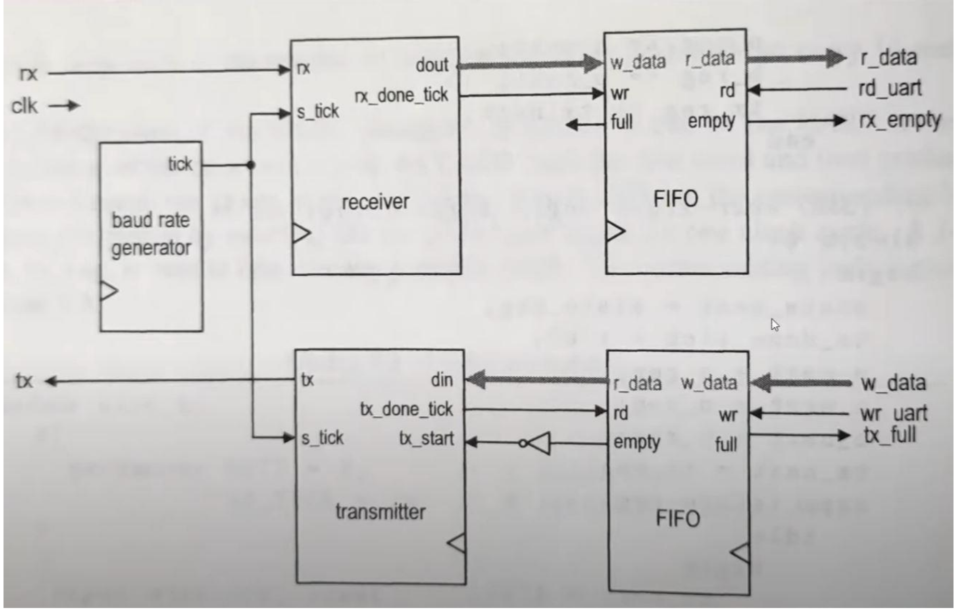
FIFO: first in first out, it is a register file with control circuitry.



- 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.

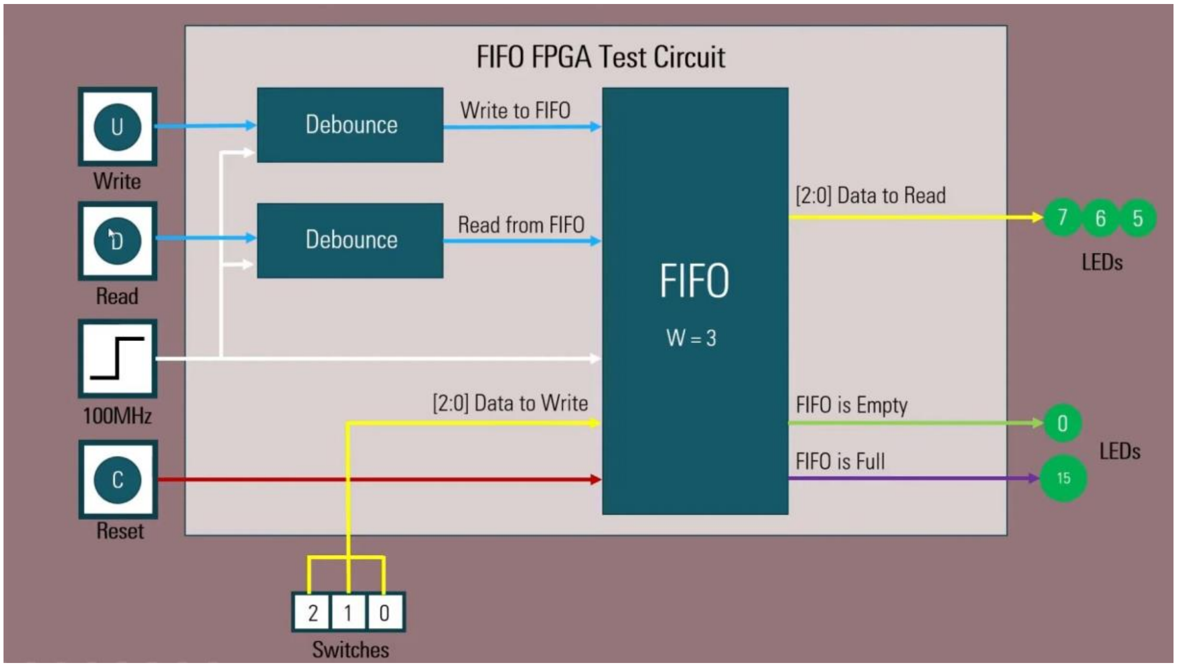
UART: Universal asynchronous receiver/transmitter

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



- FIFO block is after RX block.

-



- 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