# Lab Exercise 3: Register file with 2 read and 1 write ports

#

A register file is an array of registers or a small memory. Data is read from or written into a register file through read and write ports, respectively. The number of ports determines how many read/write operations can take place concurrently. The most common form of register file used in processors is one with 2 read ports and 1 write port. Such a register file allows two operands for an operation to be read and one result to be written concurrently.

This exercise aims at designing and implementing a small register file of size 8x4, with 2 read and 1 write ports. Here 8 is the number of registers and 4 is the size of each register (number of bits). The figure on right shows inputs and outputs of such a register file.
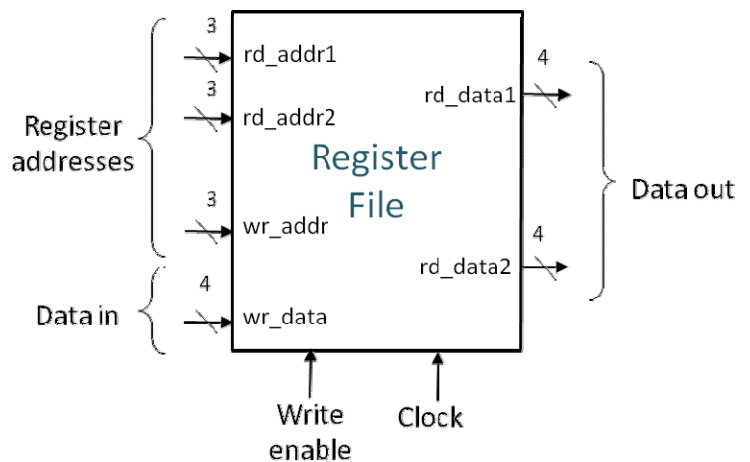

All inputs (except clock) could be connected to slide switches and outputs could be connected to LED displays. Clock input could be connected to a push button ~~through debouncing logic~~. This arrangement could be used for entering data into individual registers and displaying their contents.

To demonstrate concurrent access to all the ports, connect the register file to a 4-bit XOR circuit and a multiplexer as shown on right.

The multiplexer selects whether the data to be written into the register file comes from the switches or from XOR output, as specified by the Mode input. Connect Mode to a slide switch. Address inputs are not shown in this figure for clarity.
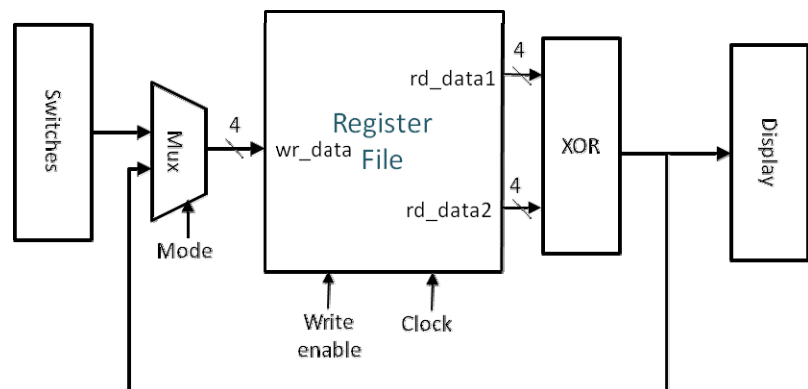
For demonstration, enter some data into the registers through switches. Then perform the following sequence of XOR operations and check that the resulting value of D is same as A.

C <= A XOR B
D <= C XOR B

Also, make note of the following:
1. Reading operation will be purely combinational.
2. Reading will happen continuously (there is no read enable signal).
3. Writing will be a single clock operation (no latches in address path).

4. If write address is same as one of the read addresses, the new data written will appear in the output immediately.
5. If mode is '0' then data is taken from switches, else taken from XORed data (of read output ports).

Pin Name	Direction	Purpose
clock	Input (from push button)	Clock for writes
write_enable	Input	Write enable
mode	Input	Decides write data source
wr_addr(2:0)	Input	Write Address
rd_addr1(2:0)	Input	Read Address (Port 1)
rd_addr2(2:0)	Input	Read Address (Port 2)
wr_data(3:0)	Input	Write Data
rd_data1(3:0)	Output	Read Data (Port 1)
rd_data2(3:0)	Output	Read Data (Port 2)
rd_data1_xor_data2(3:0)	Output	XOR of Read Data
Top Level Module Name	lab3_reg_file	
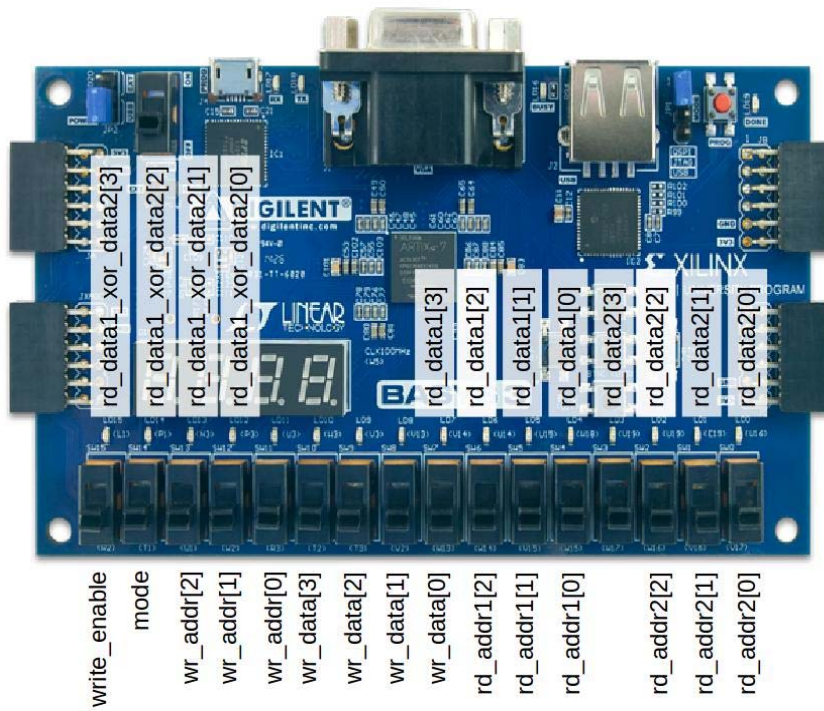
The clock signal is mapped to push button - BTNC (centre button).

D - Flip Flop (FF):
Please use the following, D-FF from the library.