2025
Basic RTL Practice
Verilog-HDL

Deadline: 2025/08/04 23:59

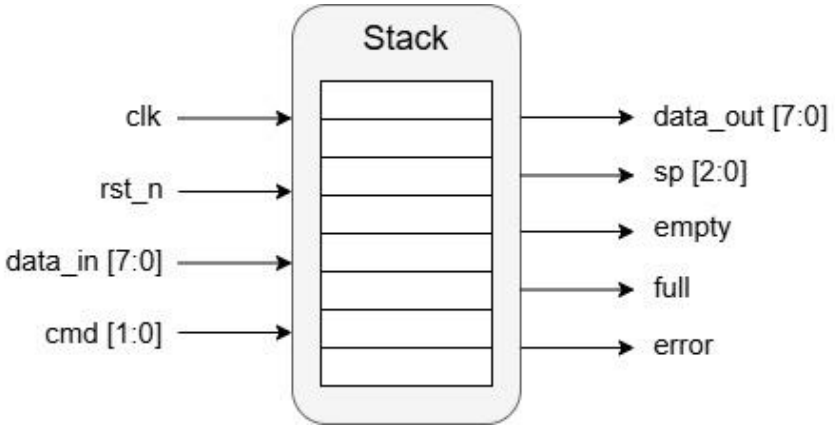
Data preparation

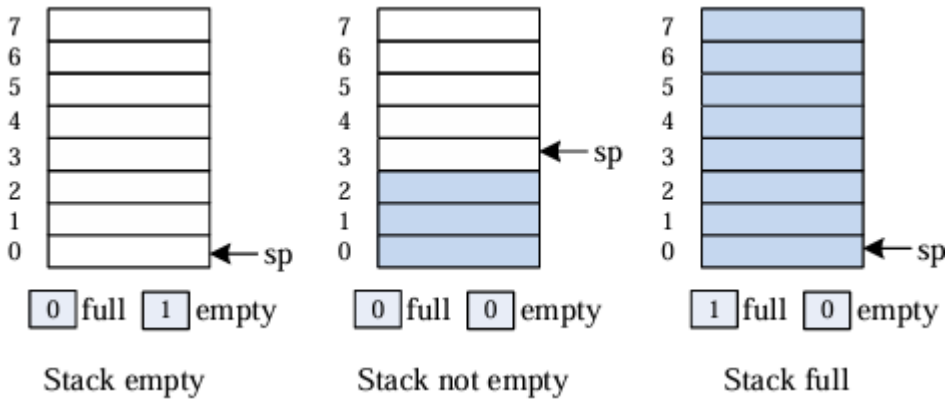
Unpack basic_practice.tar with the following command

```
tar    -xvf    2025_practice.tar
```

Topic: Stack Module Design

Use Verilog-HDL to design the ***RTL-Level*** stack module as below. The stack consists of a memory module of 8X8, a 3-bit stack pointer “sp”, and three stack flags, “full”, “empty” and “error”. “sp” always points to the next available entry for push operations. The command will be given by a 2-bit “cmd”. For push operation, your stack module will store 8-bit input data “data_in” into your memory module. For pop operation, your stack module will output 8-bit “data_out” to the testbed. Note that the output flags “full” is set to high when your stack is full, “empty” is set to high when your stack is empty, and “error” is set to high after receiving invalid push or pop operations.





Input “cmd” and “data_in” will be given by our testbed at the negative edges of the clock. The function of encoded “cmd” is as follow:

cmd	00	01	10	11
operation	No operation	clear	push	pop

Note that the input data will be set to 0 except the push operation.

Please update your outputs (“sp”, “data_out”, “empty”, “full”, “error”) at the next positive edge of the clock after receiving “data_in” and “cmd” (i.e. half cycle for you to update your internal stack status and outputs). Testbed will sample and compare your outputs with the golden data at the next negative edge.

Please set your output data to 0 except the pop operation.

Please use the positive triggered D-Flip Flops with asynchronous negative reset to design your sequential circuits.

Please read the README file in the folder p2 to get more information.

Note: Run the following command to check your design with no latch

First run the command dc_shell to get into design_compiler

```
$ dc_shell
```

Then run the command read_verilog to check your design

```
dc_shell>  read_verilog  stack.v
```

Please make sure that your file is read successfully and there is no any latch in your design

Use command exit to quit the dc_shell

```
dc_shell>  exit
```