Tutorial

4-input Shifter

In this section, you are going to design a 4-input Shifter. A 4-bit bus I[3:0] will be used for data inputs, and four other 1-bit inputs are used for the control signals F (fill), R (rotate/shift), D (direction), and En (enable signal). Bus Y[3:0] will show the output of the shifter.

Declare a Shifter

Create a Verilog module called shifter with inputs I, En, D, R, F and outputs Y as follows:

You will use if-else statement to implement the shifter.

```
module shifter (
    input [3:0] I,
    input D,
    input    R,
    input    F,
    input En,
    output reg [3:0] Y
);

always @ (I, En)
begin
    if (En == 0)
        Y <= I;
    else begin
        if (R == 0)
            Y <= (D == 0) ? {I[2:0], F} : {F, I[3:1]};
        else
            Y <= (D == 0) ? {I[2:0], I[3]} : {I[0], I[3:1]};
    end

    end
endmodule
```

Code explanation

The above Verilog code represents a 4-bit shifter module.

The module has the following inputs:

1. I (4-bit input): This is the input data to be shifted.
2. D (1-bit input): This input determines the direction of the shift. If D is 0, it represents a right shift, and if D is 1, it represents a left shift.

3. R (1-bit input): This input determines whether the shift is a logical shift or a rotate shift. If R is 0, it represents a logical shift, and if R is 1, it represents a rotate shift.
4. F (1-bit input): This input represents the fill bit to be inserted during the shift.
5. En (1-bit input): This input enables or disables the shift operation.

The module has one output:

1. Y (4-bit output): This is the output of the shift operation.

The **always** block is sensitive to changes in the **I** and **En** inputs.

1. If **En** is 0 (disabled), the output **Y** is assigned the value of input **I**. This means the shift operation is bypassed, and the original value is passed through.
2. If **En** is 1 (enabled), the shift operation takes place based on the values of **D**, **R**, and **F**.
    - If **R** is 0 (**logical** shift):
        - If **D** is 0 (**right** shift), the output **Y** is assigned the value **{I[2:0], F}**. This means the input data **I** is shifted right by **one** position, and the fill bit **F** is inserted at the most significant bit position.
        - If **D** is 1 (left shift), the output **Y** is assigned the value **{F, I[3:1]}**. This means the input data **I** is shifted left by one position, and the fill bit **F** is inserted at the least significant bit position.
    - If **R** is 1 (**rotate** shift):
        - If **D** is 0 (**right** shift), the output **Y** is assigned the value **{I[2:0], I[3]}**. This means the input data **I** is shifted right by one position, and the most significant bit is rotated to the least significant bit position.
        - If **D** is 1 (left shift), the output **Y** is assigned the value **{I[0], I[3:1]}**. This means the input data **I** is shifted left by one position, and the least significant bit is rotated to the most significant bit position.

Concatenation

In the shifter's behavioral code, {A,B} is used to concatenate two groups of signals into a bus. For example, `Y <= {I[2:0], F}` means `Y[3:1] <= I[2:0]` and `Y[0] <= F`.

The module performs a 4-bit shift operation based on the inputs **I**, **D**, **R**, **F**, and **En**. The output **Y** represents the shifted value according to the specified shift.