

```
////////////////////////////////////  
////////////////////////////////////  
//  
// Company:  
// Engineer:      Ralnikov Vadim Dmitryevich  
//  
// Create Date:   15:00 23/02/2019  
// Design Name:  
// Module Name:   test_bench  
// Project Name:  
// Target Devices:  
// Tool versions:  
// Description:    test_bench for shiftreg8_1  
//  
////////////////////////////////////  
////////////////////////////////////
```

```
`timescale 1ns / 100ps  
`default_nettype none
```

```
module test_bench;
```

```
parameter PERIOD = 10;
```

```
wire          CLK;  
// in, u[ 1], Clock freq 100MHz  
wire          SLOAD;  
// in, u[ 1], register sync loading  
wire          EN;  
// in, u[ 1], enable work 1 -enable  
wire          DIR;  
// in, u[ 1], direction setter, 1 - left, 0 - right  
wire [ 7 : 0 ] DATA;  
// in, u[ 8], input data, parallel  
wire          RST;  
// in, u[ 1], reset input, 0 - reset async  
  
wire          Q;  
// out, u[ 1], normal polarity
```

```
reg                r_clk    = 1'b0;
reg    [ 7 : 0 ] r_data    = 8'b0;
reg                r_en     = 1'b0;
reg                r_dir    = 1'b0;
reg                r_sload  = 1'b0;
reg                r_rst    = 1'b1;

assign CLK        = r_clk;
assign DATA      = r_data;
assign EN         = r_en;
assign DIR        = r_dir;
assign SLOAD      = r_sload;
assign RST        = r_rst;

initial begin
    forever
        #(PERIOD/2) r_clk = ~r_clk;
end

initial begin

    #100;

    @(posedge CLK);

    #2;

    r_data = 8'b11110101;
    r_en   = 1;
    r_sload = 1;
    r_rst  = 1;

    #100;

    r_rst = 0;
```

```
#30;

@(posedge CLK);
r_rst = 1;
r_en = 1;
r_sload = 0;
r_dir =1;

#100;

@(posedge CLK);

#2;

r_data = 8'b11110101;
r_en = 1;
r_sload = 1;
r_rst = 1;

#30;

@(posedge CLK);
r_rst = 1;
r_en = 1;
r_sload = 0;
r_dir =1;

#200;
```

end

```
shiftreg8_1
  shiftreg8_1_inst
  (
    .CLK                                (CLK) ,
```

```
        // in, u[ 1], Clock freq 100MHz
        .SLOAD                                (SLOAD) ,
        // in, u[ 1], register sync loading
        .EN                                    (EN) ,
        // in, u[ 1], enable work 1 -enable
        .DIR                                    (DIR) ,
        // in, u[ 1], direction setter, 1 - left, 0
        - right
        .DATA                                    (DATA) ,
        // in, u[ 8], input data, parallel
        .RST                                    (RST) ,
        // in, u[ 8], reset input, 0 - reset async

        .Q                                    (Q)
        // out, u[ 1], normal polarity
    );
endmodule
```