```
// the circuit can be used as a delay block
//
// in this set up it decreases the frequency of input signal change to output change by a factor of 3


////////////
module HW1Q7( input logic async_sig,
        input logic outclk,
        output logic out_sync_sig,
        input logic VCC,
        input logic GND);

wire QA,QB,QC;
wire Q_1;


FDC A(.D(VCC),.Q(QA),.C(async_sig),.clr(Q_1));
FDC B(.D(QA),.Q(QB),.C(outclk),.clr(GND));
FDC C(.D(QB),.Q(out_sync_sig),.C(outclk),.clr(GND));
FDC FDC_1(.D(out_sync_sig),.Q(Q_1),.C(outclk),.clr(GND));

endmodule


//////////////////////////////////////////////////



module FDC( D, Q, C, clr);

input logic C, D, clr;  // D is in // C is clock // clr is reset
output logic Q; // Q is out


always_ff@( posedge C, posedge clr) begin

if (clr) // if reset
Q<=0;  // out is zero

else

Q<=D; // else out is in

end


endmodule

////////////////////////////////
module HW1Q7_TB();


reg async_sig_tb;
reg outclk_tb;
wire out_sync_sig_tb;
reg VCC_TB, GND_TB;


HW1Q7 HW1Q7TB( .async_sig(async_sig_tb), .outclk(outclk_tb),
        .out_sync_sig(out_sync_sig_tb),
        .VCC(VCC_TB), .GND(GND_TB));


initial forever begin
outclk_tb = 0;
#1;
outclk_tb = 1;
#1;
```

```
69
70    end
71
72    initial forever begin
73
74    async_sig_tb = 0;
75    #2;
76    async_sig_tb = 1;
77    #2;
78
79    end
80
81    initial begin
82
83    VCC_TB = 1;
84    GND_TB = 0;
85
86    end
87
88    endmodule
```