module shiftreg(input wire clk, load, cin, input wire[7:0] load\_sig, output wire cout, output wire[7:0] out\_sig); wire[7: 0] tmp; wire[7: 0] regw;

assign out\_sig = regw; assign cout = regw[0];

mux2 mx2i0 (regw[1], load\_sig[0], load, tmp[0]); mux2 mx2i1 (regw[2], load\_sig[1], load, tmp[1]); mux2 mx2i2 (regw[3], load\_sig[2], load, tmp[2]); mux2 mx2i3 (regw[4], load\_sig[3], load, tmp[3]); mux2 mx2i4 (regw[5], load\_sig[4], load, tmp[4]); mux2 mx2i5 (regw[6], load\_sig[5], load, tmp[5]); mux2 mx2i6 (regw[7], load\_sig[6], load, tmp[6]); mux2 mx2i7 (cin, load\_sig[7], load, tmp[7]);

df dfi0 (clk, tmp[0], regw[0]); df dfi1 (clk, tmp[1], regw[1]); df dfi2 (clk, tmp[2], regw[2]); df dfi3 (clk, tmp[3], regw[3]); df dfi4 (clk, tmp[4], regw[4]); df dfi5 (clk, tmp[5], regw[5]); df dfi6 (clk, tmp[6], regw[6]); df dfi7 (clk, tmp[7], regw[7]); endmodule

module circuit(input wire clk, in, reset, output wire xor\_out); wire dff\_out,or\_out; dfr f2(clk,reset,or\_out,dff\_out); or2 f1(in,dff\_out,or\_out); xor2 f3(in,dff\_out,xor\_out); endmodule

module serialcompliment(input wire clk, reset, input wire[7:0] load\_sig, outpu t wire[7:0] out\_sig); wire t1, t2;

shiftreg sri0 (clk, reset, t1, load\_sig, t2, out\_sig); circuit ci0 (clk, t2, reset, t1); endmodule