

D:/ModelsimProjects/cntr.v	
Ln#	
1	module cntr#(
2	parameter w=8,
3	parameter iv = {w{1'b0}}
4)(
5	input clk,
6	input rst_b,
7	input c_up,
8	input clr,
9	output [w-1:0] q;
10	wire [w-1:0]d;
11	assign d = q+{(w-1){1'b0}},1'b1};
12	rgst #(
13	.w(w),
14	.iv(iv))u_rgst(
15	.clk(clk),
16	.rst_b(rst_b),
17	.d(d),
18	.ld(c_up),
19	.clr(clr),
20	.q(q)
21);
22	endmodule
23	
1	module decodificator #(
2	parameter w = 2
3)(
4	input [w-1:0] s,
5	input e,
6	output reg [(1<<w)-1:0] y
7);
8	
9	always @(*) begin
10	y = 'b0;
11	y[s] = e;
12	end
13	endmodule
14	
15	
Ln#	
1	module pkt_mux(
2	input [63:0] msg_len,
3	input [31:0]pkt,
4	input pad_pkt,
5	input zero_pkt,
6	input hi_mgln,
7	input lo_mgln,
8	output [31:0]o;
9	
10	assign o = pad_pkt ? {1'b1,31'd0} : (zero_pkt ? 32'd0 : (hi_mgln ? msg_len[63:32] : (lo_mgln ? msg_len[31:0] : pkt)));
11	
12	endmodule
13	

Ln#	
1	module reg_fl(
2	input [3:0]s,
3	input [31:0]d,
4	input en,
5	input clk,rst_b,
6	output [511:0]q);
7	
8	wire [15:0] o;
9	
10	decodificator#
11	(.w(4))
12	decl
13	(.s(s),
14	.e(en),
15	.y(o));
16	generate
17	genvar i;
18	for(i=0;i<16;i=i+1)
19	begin :vect
20	rgst#
21	(.w(32),
22	.iv(32'd0))
23	regis
24	(.d(d),
25	.clk(clk),
26	.rst_b(rst_b),
27	.ld(o[i]),
28	.clr(1'b0),
29	.q(q[32*i+31:32*i]));
30	end
31	endgenerate
32	endmodule
33	

Ln#	
1	module rgst #(
2	parameter w = 8,
3	parameter iv = {w{1'b0}})
4	(
5	input clk,
6	input rst_b,
7	input [w-1:0] d,
8	input ld,
9	input clr,
10	output reg [w-1:0] q
11);
12	
13	always @ (posedge clk, negedge rst_b)
14	if (!rst_b)
15	q <= iv;
16	else if (clr)
17	q <= iv;
18	else if (ld)
19	q <= d;
20	endmodule
21	

D:/ModelsimProjects/sum.v	
Ln#	
1	module sum#(
2	parameter w=8
3	(
4	input [w-1:0] a,
5	input [w-1:0] b,
6	output [w-1:0] o
7);
8	
9	assign o = a+b;
10	endmodule
11	
12	

D:/ModelsimProjects/unitate.v	
Ln#	
1	module unitate(
2	input st_pkt,
3	input lo_mgln,
4	input hi_mgln,
5	input zero_pkt,
6	input pad_pkt,
7	input inc_mgln,
8	input clr,
9	input c_up,
10	input clk,
11	input rst_b,
12	input [31:0] pkt,
13	output [3:0] ldx,
14	output [511:0] blk);
15	
16	wire [63:0] q;
17	wire [63:0] d;
18	wire [31:0] d2;
19	
20	sum#(.w(64))
21	sum1(.a(64'd32),
22	.b(q),
23	.o(d));
24	
25	cntr#(.w(4))
26	cntrl(
27	.clk(clk),
28	.rst_b(rst_b),
29	.c_up(c_up),
30	.clr(clr),
31	.q(ldx));
32	
33	
34	rgst#(.w(64))
35	rgst1(.d(d),
36	.clk(clk),
37	.rst_b(rst_b),
38	.ld(inc_mgln),
39	.clr(clr),
40	.q(q));

```

rgst#(.w(64))
rgst1(.d(d),
.clk(clk),
.rst_b(rst_b),
.ld(inc_mgln),
.clr(clr),
.q(q));

pkt_mux pkt_mux1(
.msg_len(q),
.pkt(pkt),
.pad_pkt(pad_pkt),
.zero_pkt(zero_pkt),
.hi_mgln(hi_mgln),
.lo_mgln(lo_mgln),
.o(d2));
reg_fl reg_fl1(
.s(ldx),
.d(d2),
.en(st_pkt),
.clk(clk),
.rst_b(rst_b),
.q(blk));
endmodule

```

Testbench:

D:/ModelsimProjects/unitate_tb.v

Ln#	
1	module unitate_tb(
2	output reg st_pkt,
3	output reg lo_mgln,
4	output reg hi_mgln,
5	output reg zero_pkt,
6	output reg pad_pkt,
7	output reg inc_mgln,
8	output reg clr,
9	output reg c_up,
10	output reg clk,
11	output reg rst_b,
12	output reg [31:0] pkt,
13	output [3:0] ldx,
14	output [511:0] blk);
15	
16	unitate DUT(
17	.st_pkt(st_pkt),
18	.lo_mgln(lo_mgln),
19	.hi_mgln(hi_mgln),
20	.zero_pkt(zero_pkt),
21	.pad_pkt(pad_pkt),
22	.inc_mgln(inc_mgln),
23	.clr(clr),
24	.c_up(c_up),
25	.rst_b(rst_b),
26	.clk(clk),
27	.pkt(pkt),
28	.ldx(ldx),
29	.blk(blk));
30	
31	initial begin
32	rst_b=1'b0;
33	clr=1'b0;
34	st_pkt=1'b1;
35	c_up=1'b1;
36	clk=1'b0;
37	inc_mgln=1'b1;
38	pad_pkt=1'b0;
39	zero_pkt=1'b0;
40	hi_mgln=1'b0;
41	lo_mgln=1'b0;
42	pkt=32'h01234567;
43	#1

```
42     pkt=32'h01234567;
43     #1
44     rst_b=1'b1;
45     #10 clk=~clk;
46     #10 clk=~clk;
47     pkt=32'haabbccdd;
48     #10 clk=~clk;
49     #10 clk=~clk;
50     inc_mgln=1'b0;
51     pkt=32'h0123fedc;
52     #10 clk=~clk;
53     #10 clk=~clk;
54     #10 clk=~clk;
55     pad_pkt=1'b1;
56     pkt=32'h56788765;
57     #10 clk=~clk;
58     #10 clk=~clk;
59     #10 clk=~clk;
60     zero_pkt=1'b1;|
61     pad_pkt=1'b0;
62     pkt=32'h00110022;
63     #10 clk=~clk;
64     #10 clk=~clk;
65     hi_mgln=1'b1;
66     zero_pkt=1'b0;
67     pkt=32'hffff0000;
68     #10 clk=~clk;
69     #10 clk=~clk;
70     #10 clk=~clk;
71     lo_mgln=1'b1;
72     hi_mgln=1'b0;
73     pkt=32'ha0a0a0a0;
74     #10 clk=~clk;
75     #10 clk=~clk;
76     end
77     endmodule
78
```