module timer\_block

(

input wire clk

input wire n\_rst

input wire enable\_timer

output wire enable\_timer

output reg shift\_strobe

output reg packet\_done

);

reg [3:0] count\_out;

reg [3:0] count\_out\_2;

reg first\_rollover;

reg second\_rollover;

assign shift\_strobe = first\_rollover;

assign packet\_done = second\_rollover;

always @ (posedge clk, negedge n\_rst)

{

flex\_counter(.clk(clk), .n\_rst(rst), .clear(!enable\_timer), .count\_enable(enable\_timer), .rollover\_val(10), .count\_out(count\_out\_1), .rollover\_flag(first\_rollover))

flex\_counter(.clk(first\_rollover), .n\_rst(rst), .clear(!enable\_timer), .count\_enable(enable\_timer), .rollover\_val(9), .count\_out(count\_out\_1), .rollover\_flag(first\_rollover))

}