```
module register8bit(
  /* Inputs */
  input clk,
  input reset,
  input carry_in,
  input [7:0] data_in,

  /* Outputs */
  output reg [7:0] data_out,
  output reg carry_out,
  output reg over_flow
);

  /* Initialization */
  initial begin
      #10 data_out = 0;
      #10 carry_out = 0;
  end

  /* Activate on rising clock edge or reset */
  always @ (posedge clk | reset) begin
    if (reset) begin // Reset outputs
      data_out = 8'b00000000;
      carry_out = 0;
    end
    else begin // Set outputs
      data_out = (data_in * 2) + carry_in;
      carry_out = ~& data_in[7|6];
    end
  end
  /* Handle Overflow */
  assign over_flow = data_out[7];

endmodule
```