# Combinational Logic Design Process

*Truth Table to Gates*

- Design a ROUNDER circuit with a 3-bit input A (represented as three inputs A2, A1, A0) and a 3-bit output R (represented as three outputs R2, R1, R0) where the output of the circuit is the input value A rounded up to the nearest multiple of 2. For example, if the input A is 5 (101), the output R will be 6 (110). If the input A is equal to 7 (111), the output should be 0 (000). Provide the Boolean equations for each of the three outputs.

# Verilog - Combinational Logic

*Behavioral Descriptions of Combination Logic*

- Create a Verilog description for the ROUNDER circuit.

```
module ROUNDER(A2, A1, A0, R2, R1, R0);
   input A2, A1, A0;
   output reg R2, R1, R0;

   always @(A2, A1, A0) begin
      R2 <= ~A2 &  A1 & A0 |  A2 & ~A1 & ~A0 |
             A2 & ~A1 & A0 |  A2 &  A1 & ~A0;
      R1 <= ~A2 & ~A1 & A0 | ~A2 &  A1 & ~A0 |
             A2 & ~A1 & A0 |  A2 &  A1 & ~A0;
      R0 <= 0;
   end
endmodule
```

- Are there any problems with the Verilog code?

```
module WhatIsThis(I1, I0, En, D3, D2, D1, D0);
   input I1, I0, En;
   output reg D3, D2, D1, D0;

   always @(I1, I0, En)
   begin
      D3 <= 0; D2 <= 0; D1 <= 0; D0 <= 0;

      if (EN) begin
         if (I1==0 && I0==0) begin
            D0 <= 1;
         end
         else if (EN && I1==0 && I0==1) begin
            D1 <= 1;
         end
         else if (I1==1 && I0==0) begin
            D2 <= 1;
         end
         else begin
            D3 <= 1;
         end
   end
endmodule
```

- What circuit does the following Verilog describe?

```
module WhatIsThis(I1, I0, En, D3, D2, D1, D0);
   input I1, I0, En;
   output reg D3, D2, D1, D0;

   always @(I1, I0, En)
   begin
      D3 <= 0; D2 <= 0; D1 <= 0; D0 <= 0;

      if (EN) begin
         if (I1==0 && I0==0) begin
            D0 <= 1;
         end
         else if (EN && I1==0 && I0==1) begin
            D1 <= 1;
         end
         else if (I1==1 && I0==0) begin
            D2 <= 1;
         end
         else begin
            D3 <= 1;
         end
   end
endmodule
```