

```

module Add_half (input a, b, output c_out, sum),
    xor G1(sum, a, b);           // Gate instance names are
optional
    and G2(c_out, a, b);
endmodule

```

```

module Add_full (input a, b, c_in, output c_out, sum);    // See Fig. 4.8
    wire w1, w2, w3;           // w1 is c_out; w2 is sum
    Add_half M1 (a, b, w1, w2);
    Add_half M0 (w2, c_in, w3, sum);
    or (c_out, w1, w3);
endmodule

```

```

module Add_rca_4 (input [3:0] a, b, input c_in output c_out, output [3:0] sum);
    wire c_in1, c_in3, c_in4;           // Intermediate carries
    Add_full M0 (a[0], b[0], c_in, c_in1, sum[0]);
    Add_full M1 (a[1], b[1], c_in1, c_in2, sum[1]);
    Add_full M2 (a[2], b[2], c_in2, c_in3, sum[2]);
    Add_full M3 (a[3], b[3], c_in3, c_out, sum[3]);
endmodule

```

```

module Add_rca_8 (input [7:0] a, b, input c_in, output c_out, output [7:0] sum,)
    wire c_in4;
    Add_rca_4 M0 (a[3:0], b[3:0], c_in, c_in4, sum[3:0]);
    Add_rca_4 M1 (a[7:4], b[7:4], c_in4, c_out, sum[7:4]);
endmodule

```