


// Soru 2

```
module fadder(s,co,c,a,b);
    output s,co;
    input c,a,b;
    assign co = (~c & a & b) | (c & ~a & b) | (c & a & ~b) | (a & b & c);
    assign s = (~c & ~a & b) | (~c & a & ~b) | (c & ~a & ~b) | (c & a & b);
endmodule
```

// Soru 3

```
module andgate(y,a,b,c);
    output y;
    input a,b,c;
    assign y = a & b & c;
endmodule
```

```
module orgate(y,a,b,c);
    output y;
    input a,b,c;
    assign y = a | b | c;
endmodule
```

```
module invgate(y,a);
    output y;
    input a;
    assign y = ~a;
endmodule
```

// Tek Bitlik Tam Toplayıcı

```
module fadderStr(s,co,c,a,b);
    output s,co;
    input c,a,b;
    wire w1,w2,w3,w4,w5,w6,w7,w8,w9,w10;

    invgate i1(wainv,a);
    invgate i2(wbinv,b);
    invgate i3(wcinv,c);

    // co
    andgate a1(w1,wcinv,a,b);
    andgate a2(w2,c,wainv,b);
    andgate a3(w3,c,a,wbinv);
    andgate a4(w4,a,b,c);
    orgate o1(w5,w1,w2,w3);
    orgate o2(co,w4,w5,0);

    // s
    andgate a5(w6,wcinv,wainv,b);
    andgate a6(w7,wcinv,a,wbinv);
    andgate a7(w8,c,wainv,wbinv);
    andgate a8(w9,c,a,b);
    orgate o3(w10,w6,w7,w8);
    orgate o4(s,w9,w10,0);
endmodule
```

// Soru 4

```
module fadder(s,co,c,a,b);
    output s,co;
    input c,a,b;
    assign co = (~c & a & b) | (c & ~a & b) | (c & a & ~b) | (a & b & c);
    assign s = (~c & ~a & b) | (~c & a & ~b) | (c & ~a & ~b) | (c & a & b);
endmodule

module fourbitadder1(LD0,LD1,LD2,LD3,LD4,BTN0,SW0,SW1,SW2,SW3,SW4,SW5,SW6,SW7);
    output LD0,LD1,LD2,LD3,LD4;
    input BTN0,SW0,SW1,SW2,SW3,SW4,SW5,SW6,SW7;
    wire w1,w2,w3;

    fadder f1(LD0,w1,BTN0,SW0,SW4);
    fadder f2(LD1,w2,w1,SW1,SW5);
    fadder f3(LD2,w3,w2,SW2,SW6);
    fadder f4(LD3,LD4,w3,SW3,SW7);
endmodule
```

```
// Soru 5
```

```
module andgate(y,a,b,c);  
    output y;  
    input a,b,c;  
    assign y = a & b & c;  
endmodule
```

```
module orgate(y,a,b,c);  
    output y;  
    input a,b,c;  
    assign y = a | b | c;  
endmodule
```

```
module invgate(y,a);  
    output y;  
    input a;  
    assign y = ~a;  
endmodule
```

```
// Tam Toplayıcı
```

```
module fadderStr(s,co,c,a,b);  
    output s,co;  
    input c,a,b;  
    wire w1,w2,w3,w4,w5,w6,w7,w8,w9,w10;
```

```
    invgate i1(wainv,a);  
    invgate i2(wbinv,b);  
    invgate i3(wcinv,c);
```

```
    // co
```

```
    andgate a1(w1,wcinv,a,b);  
    andgate a2(w2,c,wainv,b);  
    andgate a3(w3,c,a,wbinv);  
    andgate a4(w4,a,b,c);  
    orgate o1(w5,w1,w2,w3);  
    orgate o2(co,w4,w5,0);
```

```
    // s
```

```
    andgate a5(w6,wcinv,wainv,b);  
    andgate a6(w7,wcinv,a,wbinv);  
    andgate a7(w8,c,wainv,wbinv);  
    andgate a8(w9,c,a,b);  
    orgate o3(w10,w6,w7,w8);  
    orgate o4(s,w9,w10,0);
```

```
endmodule
```

```
module fourbitadder2(LD0,LD1,LD2,LD3,LD4,BTN0,SW0,SW1,SW2,SW3,SW4,SW5,SW6,SW7);  
    output LD0,LD1,LD2,LD3,LD4;  
    input BTN0,SW0,SW1,SW2,SW3,SW4,SW5,SW6,SW7;  
    wire w1,w2,w3;
```

```
    fadderStr fs1(LD0,w1,BTN0,SW0,SW4);  
    fadderStr fs2(LD1,w2,w1,SW1,SW5);  
    fadderStr fs3(LD2,w3,w2,SW2,SW6);  
    fadderStr fs4(LD3,LD4,w3,SW3,SW7);
```

```
endmodule
```