2024.10.20 作业

1 教材 4-35 实现 4 位超前加法器

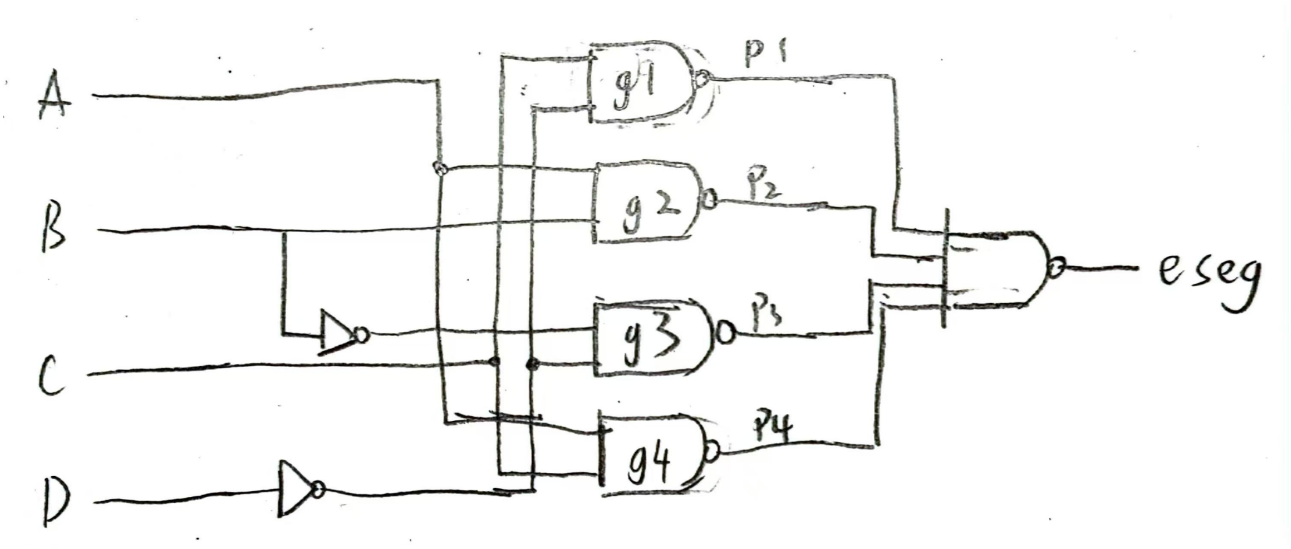
```
module HC283(
    input [3:0]A,
    input [3:0]B,
    input Cin,
    output [3:0]S,
    output Cout
    );
    wire [3:0]C;
    wire [3:0]P;
    wire [3:0]G;
    integer i;

    assign C[0] = Cin;

    assign P[0] = A[0] ^ B[0];
    assign P[1] = A[1] ^ B[1];
    assign P[2] = A[2] ^ B[2];
    assign P[3] = A[3] ^ B[3];

    assign G[0] = A[0] & B[0];
    assign G[1] = A[1] & B[1];
    assign G[2] = A[2] & B[2];
    assign G[3] = A[3] & B[3];

    assign S[0] = P[0] ^ C[0];
    assign S[1] = P[1] ^ C[1];
    assign S[2] = P[2] ^ C[2];
    assign S[3] = P[3] ^ C[3];

    assign C[0] = Cin;
    assign C[1] = G[0] || (P[0] & C[0]);
    assign C[2] = G[1] || (P[1] & (G[0] || (P[0] & C[0])));
```

```
    assign C[3] = G[2] ||
    (P[2] & (G[1] || (P[1] & (G[0] || (P[0] & C[0])))));
    assign Cout = G[3] || //装不下了，只能换行了
    (P[3] & (G[2] || (P[2] & (G[1] || (P[1] & (G[0] || (P[0] & C[0])))))));
endmodule
```

2 教材 4-36 根据下面的语言描述, 画出对应的逻辑电路图

```
module binaryToESeg;
    wire eSeg,p1,p2,p3,p4;
    reg A,B,C,D;
    nand g1(p1,C,~D),
         g2(p2,A,B),
         g3(p3,~B,~D),
         g4(p4,A,C),
         g5(eSeg,p1,p2,p3,p4);
endmodule
```

逻辑电路图如下



该电路的逻辑表达式是

$$Eseg = CD' + AB + B'D' + AC$$