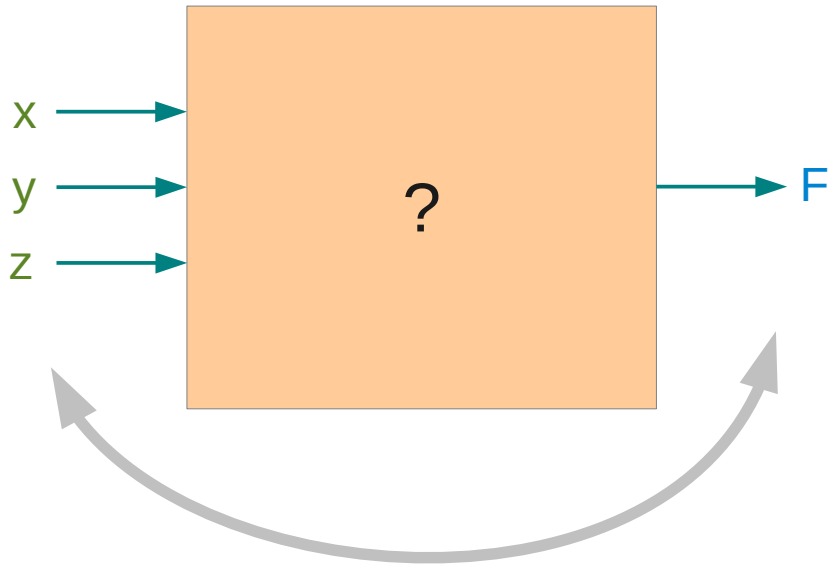
Encoders (A)

Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using OpenOffice and Octave.

Truth Table

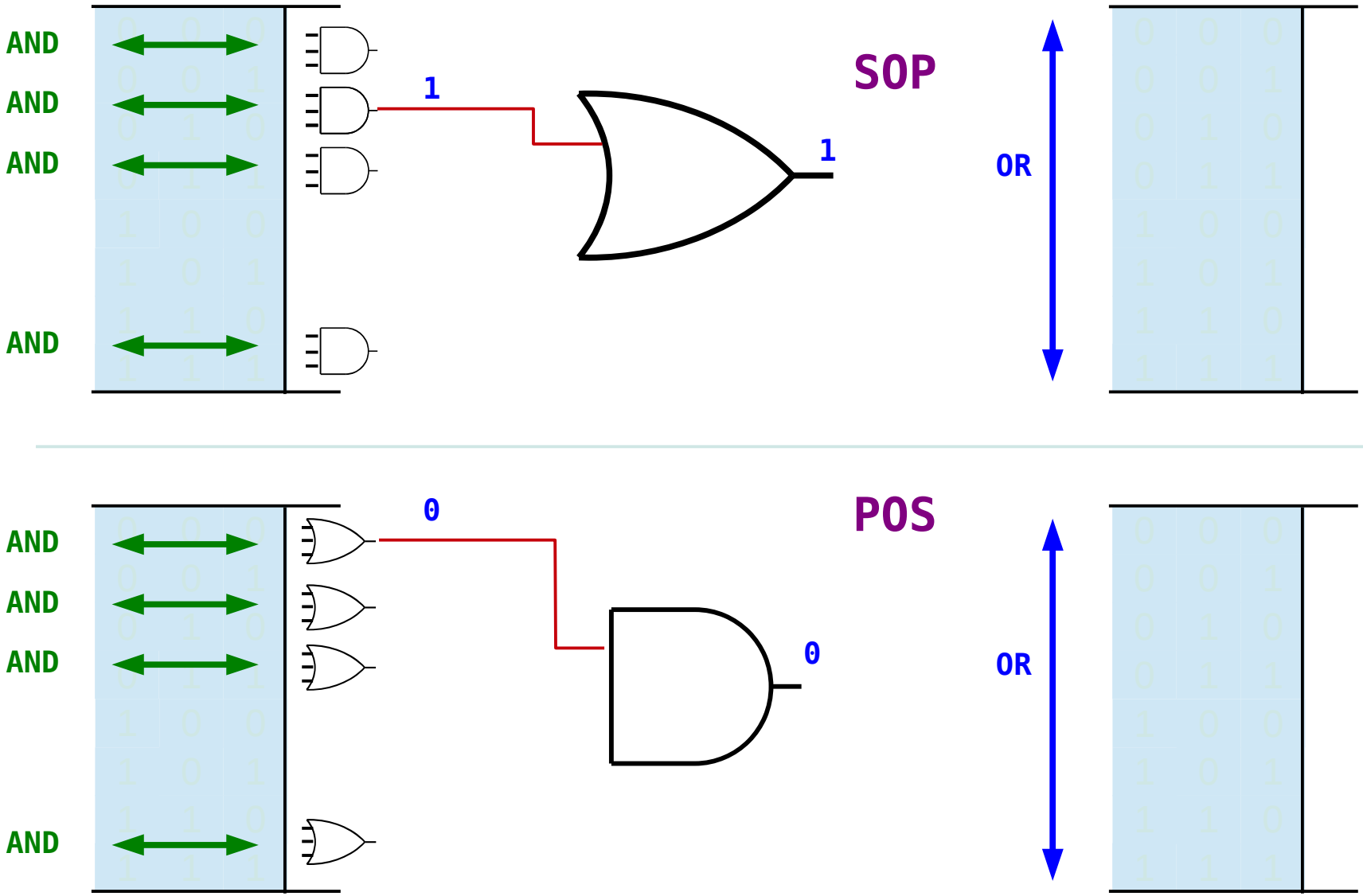


I/O relationship

x	y	z	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

inputs output

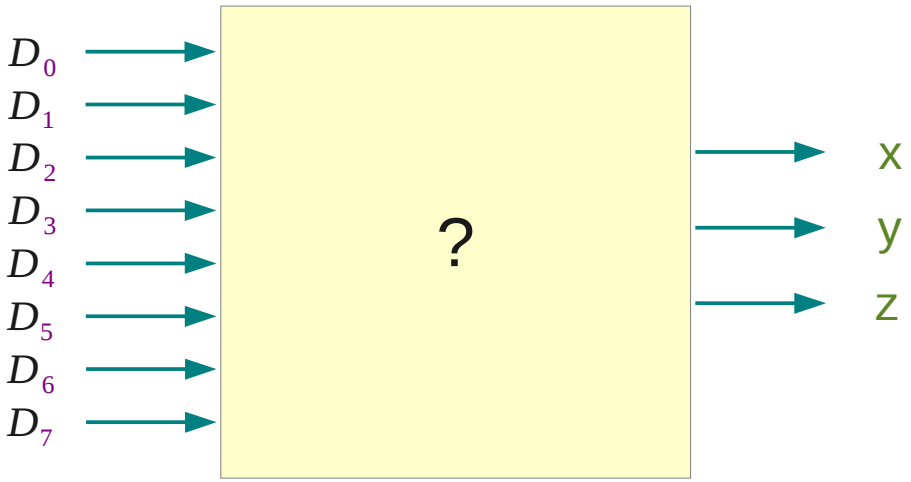
SOP and POS



Truth Table

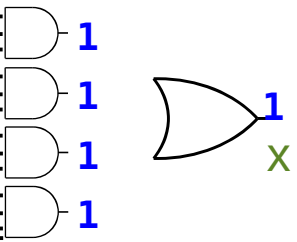
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

inputs output

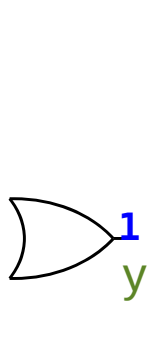


Truth Table and minterms

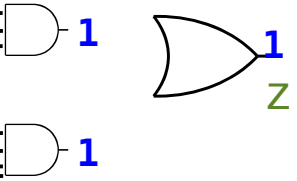
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x
1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	1	1



D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	y
1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	0	1	0	0	0	0	1
0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	1	1



D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	z
1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	1



Functions x, y, z

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

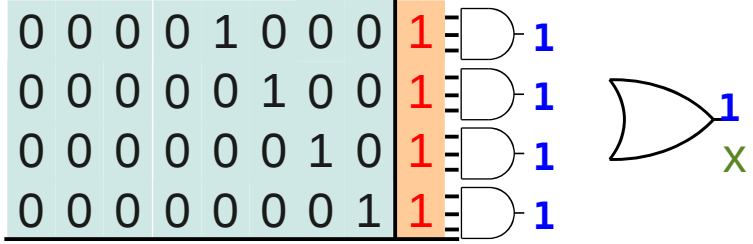
$$\bar{D}_0\bar{D}_1\bar{D}_2\bar{D}_3D_4\bar{D}_5\bar{D}_6\bar{D}_7 + \bar{D}_0\bar{D}_1\bar{D}_2\bar{D}_3\bar{D}_4D_5\bar{D}_6\bar{D}_7 + \bar{D}_0\bar{D}_1\bar{D}_2\bar{D}_3\bar{D}_4\bar{D}_5D_6\bar{D}_7 + \bar{D}_0\bar{D}_1\bar{D}_2\bar{D}_3\bar{D}_4\bar{D}_5\bar{D}_6D_7 = x$$

$$\bar{D}_0\bar{D}_1D_2\bar{D}_3\bar{D}_4\bar{D}_5\bar{D}_6\bar{D}_7 + \bar{D}_0\bar{D}_1\bar{D}_2D_3\bar{D}_4\bar{D}_5\bar{D}_6\bar{D}_7 + \bar{D}_0\bar{D}_1\bar{D}_2\bar{D}_3\bar{D}_4\bar{D}_5D_6\bar{D}_7 + \bar{D}_0\bar{D}_1\bar{D}_2\bar{D}_3\bar{D}_4\bar{D}_5\bar{D}_6D_7 = y$$

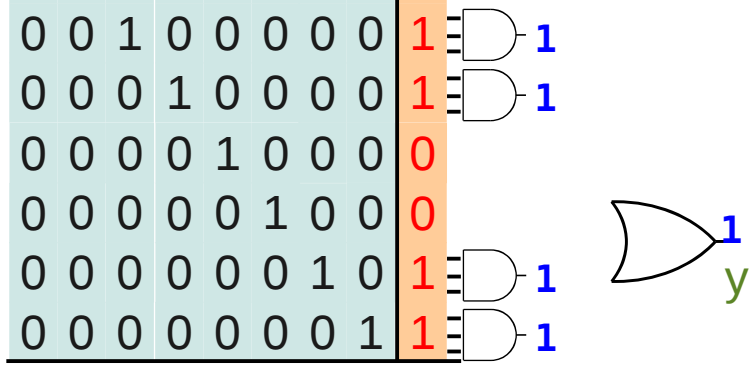
$$\bar{D}_0D_1\bar{D}_2\bar{D}_3\bar{D}_4\bar{D}_5\bar{D}_6\bar{D}_7 + \bar{D}_0\bar{D}_1\bar{D}_2D_3\bar{D}_4\bar{D}_5\bar{D}_6\bar{D}_7 + \bar{D}_0\bar{D}_1\bar{D}_2\bar{D}_3\bar{D}_4D_5\bar{D}_6\bar{D}_7 + \bar{D}_0\bar{D}_1\bar{D}_2\bar{D}_3\bar{D}_4\bar{D}_5\bar{D}_6D_7 = z$$

Another Functions x, y, z

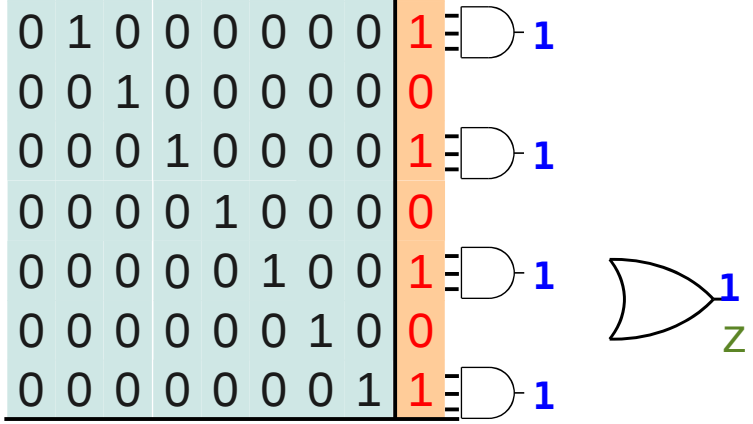
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x
1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	1	1



D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	y
1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	0	1	0	0	0	0	1
0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	1	1



D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	z
1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1
0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	1



$$D_4 + D_5 + D_6 + D_7 = x$$

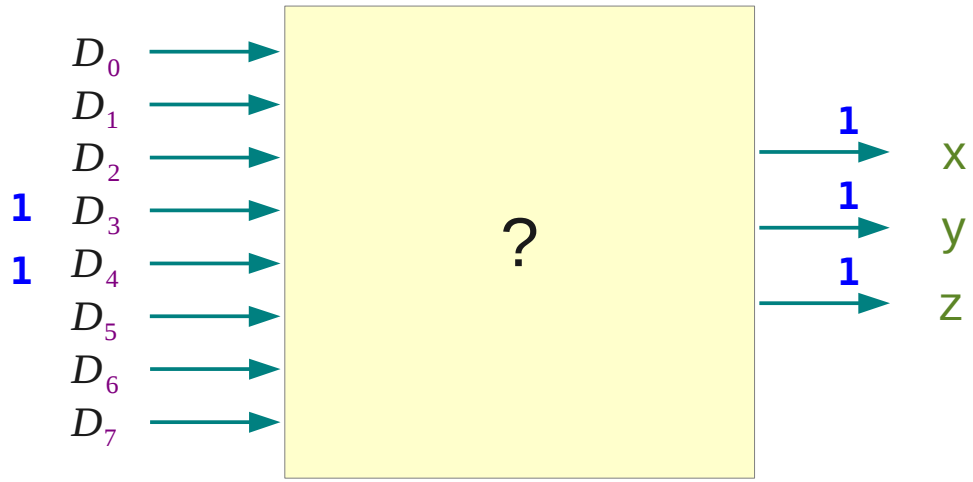
$$D_2 + D_3 + D_6 + D_7 = y$$

$$D_1 + D_3 + D_5 + D_7 = z$$

Problems

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

inputs output



Priority Encoder: Truth Table

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z	v
1	0	0	0	0	0	0	0	0	0	0	1
X	1	0	0	0	0	0	0	0	0	1	1
X	X	1	0	0	0	0	0	0	1	0	1
X	X	X	1	0	0	0	0	0	1	1	1
X	X	X	X	1	0	0	0	1	0	0	1
X	X	X	X	X	1	0	0	1	0	1	1
X	X	X	X	X	X	1	0	1	1	0	1
X	X	X	X	X	X	X	1	1	1	1	1
0	0	0	0	0	0	0	0	X	X	X	0

inputs

output

Priority Encoder: Truth Table

D_0	D_1	D_2	D_3	x	y	v
1	0	0	0	0	0	1
X	1	0	0	0	1	1
X	X	1	0	1	0	1
X	X	X	1	1	1	1
0	0	0	0	X	X	0

```
if (d3) xy=11
else if (d2) xy = 10;
else if (d1) xy = 01;
else if (d0) xy = 00;
```

D_0	D_1	D_2	D_3	x	y	v
1	0	0	0	0	0	1
0	1	0	0	0	1	1
1	1	0	0	0	1	1
0	0	1	0	1	0	1
0	1	1	0	1	0	1
1	0	1	0	1	0	1
1	1	1	0	1	0	1
0	0	0	1	1	1	1
0	0	1	1	1	1	1
0	1	0	1	1	1	1
0	1	1	1	1	1	1
1	0	0	1	1	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1
0	0	0	0	X	X	0

D_0	D_1	D_2	D_3	x	y	v
0	0	0	0	X	X	0
0	0	0	1	1	1	1
0	0	1	0	1	0	1
0	0	1	1	1	1	1
0	1	0	0	0	1	1
0	1	0	1	1	1	1
0	1	1	0	1	0	1
0	1	1	1	1	1	1
1	0	0	0	0	0	1
1	0	0	1	1	1	1
1	0	1	0	1	0	1
1	0	1	1	1	1	1
1	1	0	0	0	1	1
1	1	0	1	1	1	1
1	1	1	0	1	0	1
1	1	1	1	1	1	1

Priority Encoder: Truth Table

X	1	1	1
0	1	1	1
0	1	1	1
0	1	1	1

$$x = D_2 + D_3$$

X	1	1	0
1	1	1	0
1	1	1	0
0	1	1	0

$$y = D_3 + D_1 \bar{D}_2$$

0	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

$$v = D_0 + D_1 + D_2 + D_3$$

D_0	D_1	D_2	D_3	x	y	v
0	0	0	0	X	X	0
0	0	0	1	1	1	1
0	0	1	0	1	0	1
0	0	1	1	1	1	1
0	1	0	0	0	1	1
0	1	0	1	1	1	1
0	1	1	0	1	0	1
0	1	1	1	1	1	1
1	0	0	0	0	0	1
1	0	0	1	1	1	1
1	0	1	0	1	0	1
1	0	1	1	1	1	1
1	1	0	0	0	1	1
1	1	0	1	1	1	1
1	1	1	0	1	0	1
1	1	1	1	1	1	1

pencoder.v

```
`timescale 1ns/100ps


module pencoder();
  reg  D0, D1, D2, D3;
  wire P;

  //Invert the sel signals
  not (D2b, D2);

  // 3-input AND gate

  and (P, D1, D2b);

  or (x, D2, D3);
  or (y, D3, P);
  or (v, D0, D1, D2, D3);
```

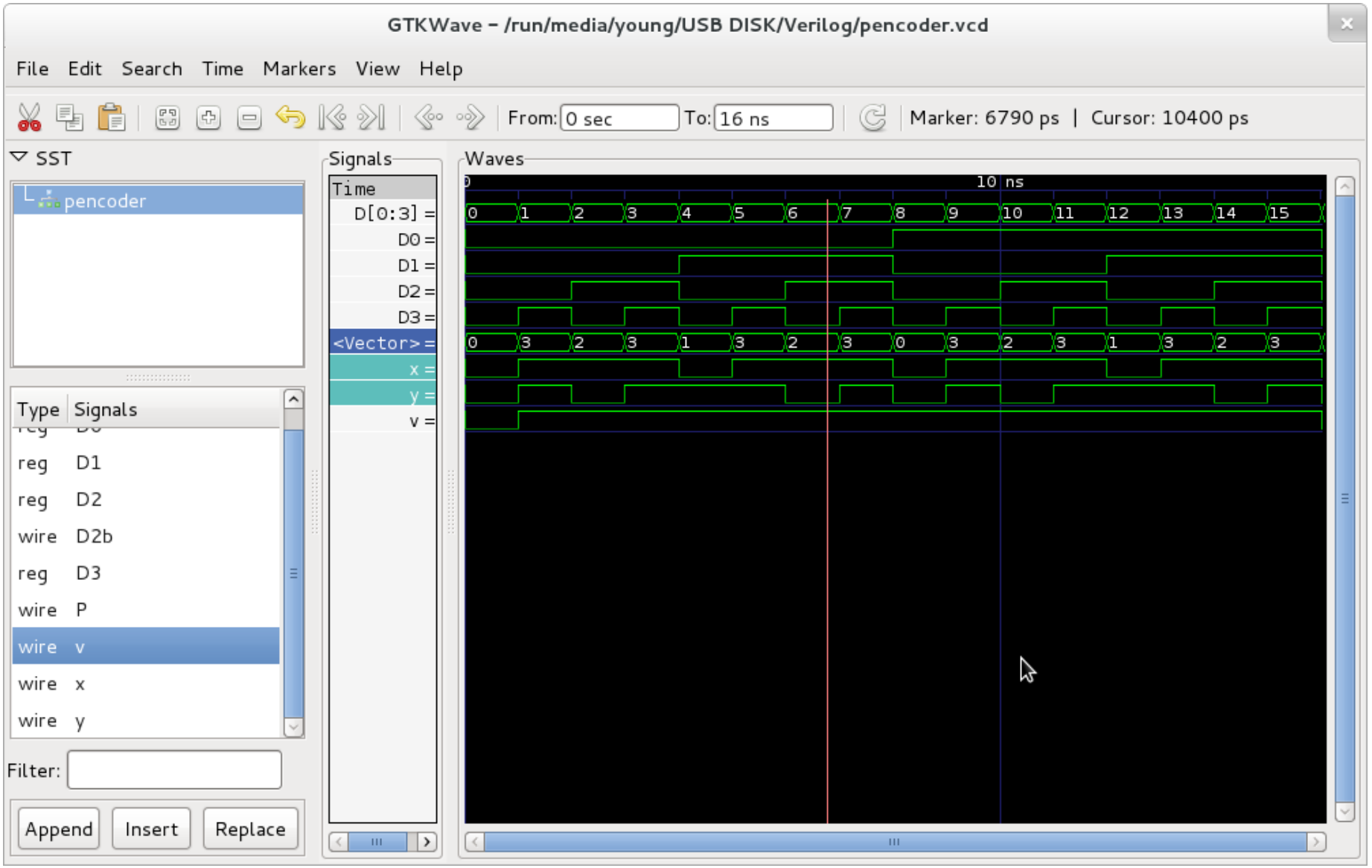
```
// Testbench Code goes here
initial begin
  $dumpfile("pencoder.vcd");
  $dumpvars(0, pencoder);

  $monitor ("[D0, D1, D2, D3] = %b %b %b %b [x y] = %b %b v = %b",
   D0, D1, D2, D3, x, y, v);

  D0 = 0;
  D1 = 0;
  D2 = 0;
  D3 = 0;



  #16 $finish;
end


always #8 D0 = ~D0;
always #4 D1 = ~D1;
always #2 D2 = ~D2;
always #1 D3 = ~D3;


endmodule
```

pencoder.v

```
`timescale 1ns/100ps


module pencoder();
  reg  D0, D1, D2, D3;
  wire P;

  //Invert the sel signals
  not (D2b, D2);

  // 3-input AND gate

  and (P, D1, D2b);

  or (x, D2, D3);
  or (y, D3, P);
  or (v, D0, D1, D2, D3);
```

```
/*
  #1 {D0,D1,D2,D3} = 4'b0000; // 0
  #1 {D0,D1,D2,D3} = 4'b0001; // 1
  #1 {D0,D1,D2,D3} = 4'b0010; // 2
  #1 {D0,D1,D2,D3} = 4'b0011; // 3
  #1 {D0,D1,D2,D3} = 4'b0100; // 4
  #1 {D0,D1,D2,D3} = 4'b0101; // 5
  #1 {D0,D1,D2,D3} = 4'b0110; // 6
  #1 {D0,D1,D2,D3} = 4'b0111; // 7
  #1 {D0,D1,D2,D3} = 4'b1000; // 8
  #1 {D0,D1,D2,D3} = 4'b1001; // 9
  #1 {D0,D1,D2,D3} = 4'b1010; // 10
  #1 {D0,D1,D2,D3} = 4'b1011; // 11
  #1 {D0,D1,D2,D3} = 4'b1100; // 12
  #1 {D0,D1,D2,D3} = 4'b1101; // 13
  #1 {D0,D1,D2,D3} = 4'b1110; // 14
  #1 {D0,D1,D2,D3} = 4'b1111; // 15
*/
```

Priority Encoder: Truth Table



binary encoder

```
module encoder(D, E);

input [7:0] D;
output [2:0] E;
reg [2:0] E;

  always @(D)
    begin
      if (D == 8'b00000001) E = 0;
      else if (D == 8'b00000010) E = 1;
      else if (D == 8'b00000100) E = 2;
      else if (D == 8'b00001000) E = 3;
      else if (D == 8'b00010000) E = 4;
      else if (D == 8'b00100000) E = 5;
      else if (D == 8'b01000000) E = 6;
      else if (D == 8'b10000000) E = 7;
      else E = 3'bX;
    end

endmodule
```

priority encoder

```
module pencoder(D, E);

input [7:0] D;
output [2:0] E;
reg [2:0] E;

   always @(D)
      begin
         if (D[0]) E = 0;
         else if (D[1]) E = 1;
         else if (D[2]) E = 2;
         else if (D[3]) E = 3;
         else if (D[4]) E = 4;
         else if (D[5]) E = 5;
         else if (D[6]) E = 6;
         else if (D[7]) E = 7;
         else E = 3'bX;
     end

endmodule
```

References

[1] http://en.wikipedia.org/