1. 8-Priority encoder

설계 모듈의 구조를 설명하면 다음과 같다.

우선순위 인코더로, A7에 1이 들어오게 되면 A0-6에 뭐가오든 상관없이 값이 결정되고, A7에 값이 업고, A6에1이 들어오게 되면 A0-5에 뭐가 오든 상관없이 값이 결정되는 구조이다.

코드에 대해 설명하면, 다음과 같다.

module p\_encoder(A,Z,NR);

input [7:0] A;

output reg [2:0] Z;

output reg NR;

always @(A)

begin

if(A==8'b00000001) Z=3'b000; //조건에 맞추기 위해 if else구문을 사용했고, 그 순서가 중요하다

else if(A==8'b0000001x) Z=3'b001; //이는 A7=1이면, Z0=0,Z1=0,Z2=1을 의미한다

else if(A==8'b000001xx) Z=3'b010; //..이하 동일하다

else if(A==8'b00001xxx) Z=3'b011;

else if(A==8'b0001xxxx) Z=3'b100;

else if(A==8'b001xxxxx) Z=3'b101;

else if(A==8'b01xxxxxx) Z=3'b110;

else if(A==8'b1xxxxxxx) Z=3'b111;

else

begin

Z=3'bxxx; // Z0=x,Z1=x,Z2=x을 의미한다

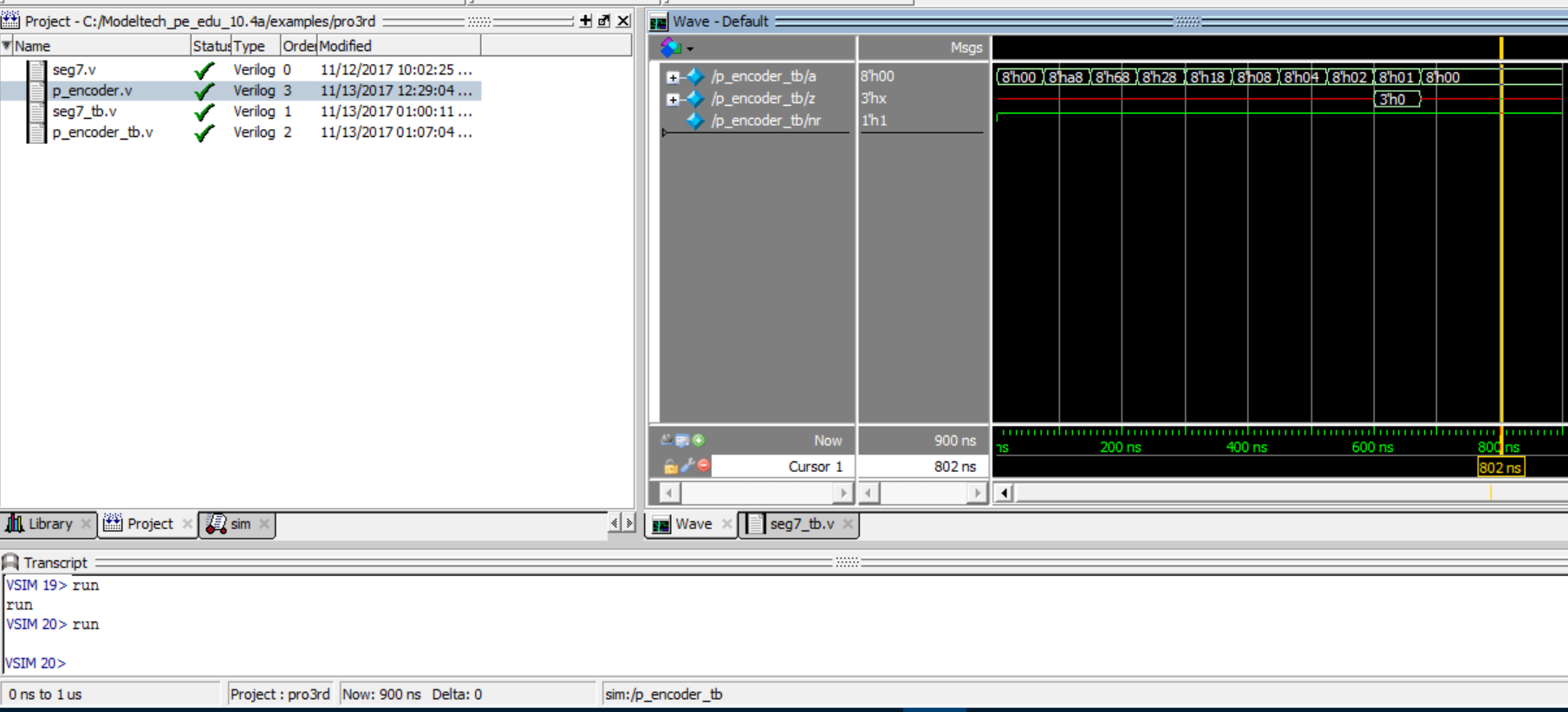
NR=1; //NR이 1일떄.

end

end

endmodule

테스트 벤치를 통해 본 웨이브폼 결과는 다음과 같다.



테스트벤치 코드를 설명하면 다음과 같다.

module p\_encoder\_tb;

reg [7:0]a; //인풋은 reg

wire [2:0]z; //아웃풋은 wire

wire nr;

p\_encoder uut(a,z,nr);

initial begin

a[0]=0;a[1]=0;a[2]=0;a[3]=0;a[4]=0;a[5]=0;a[6]=0;a[7]=0; //0으로 입력값 초기화

#75 a[7]=1; a[5]=1; a[3]=1; //75ns - A7=1이면, 그 밖에 어떤 값이 와도 DC임을 확인

#75 a[7]=0; a[6]=1; //75ns – A7=0이고, A6=1이면, 그 밖에 어떤 값이 와도 DC임을 확인

#75 a[6]=0; a[5]=1; a[3]=1; //..이하 동일

#75 a[5]=0; a[4]=1;

#75 a[4]=0; a[3]=1;

#75 a[3]=0; a[2]=1;

#75 a[2]=0; a[1]=1;

#75 a[1]=0; a[0]=1;

#75 a[0]=0;

end

endmodule

1. 7-segment

설계모듈의 구조를 설명하면 다음과 같다.

0-9중 한 숫자가 들어오게 된다. 0000-0001-0010 (이런식으로, 4개의 입력을 통해) 이 숫자를LED로 나타낼 수 있도록 해주는 구조이다.

코드에 대해 설명하면, 다음과 같다.

module seg7(bcd,leds);

input[3:0] bcd;

output reg[6:0] leds;

reg a,b,c,d,e,f,g;

always @(bcd)

begin

leds={a,b,c,d,e,f,g}; //led와 reg{a,b,c,d,e,f,g}를 연결 시켜준다

case(bcd)

0:{a,b,c,d,e,f,g}<=7'b111\_1110; //bcd에 0이 들어오게 되면, a,b,c,d,e,f seg에 불빛을 키라는 의미

1:{a,b,c,d,e,f,g}<=7'b011\_0000; //bcd에 1이 들어오게 되면, b,c seg에 불빛을 키라는 의미. 나머지는 x

2:{a,b,c,d,e,f,g}<=7'b110\_1101; //..이하 동일

3:{a,b,c,d,e,f,g}<=7'b111\_1001;

4:{a,b,c,d,e,f,g}<=7'b011\_0011;

5:{a,b,c,d,e,f,g}<=7'b101\_1011;

6:{a,b,c,d,e,f,g}<=7'b101\_1111;

7:{a,b,c,d,e,f,g}<=7'b111\_0010;

8:{a,b,c,d,e,f,g}<=7'b111\_1111;

9:{a,b,c,d,e,f,g}<=7'b111\_1011;

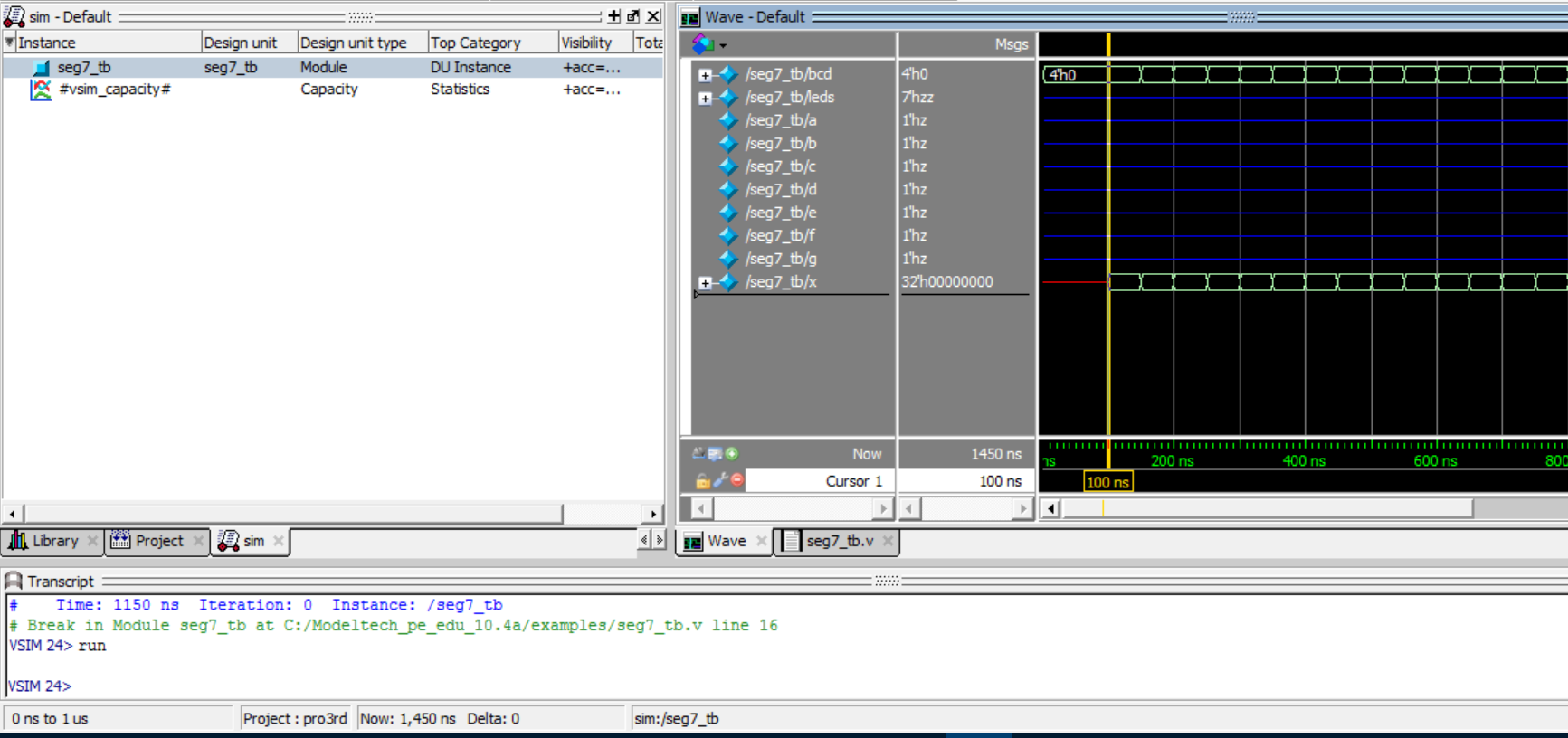
default:{a,b,c,d,e,f,g}<=7'b000\_0000; //디폴트값: 아무곳도 빛이 안들어와 있음

endcase

end

endmodule

테스트 벤치를 통해 본 웨이브폼 결과는 다음과 같다.



테스트벤치 코드를 설명하면 다음과 같다.

module seg7\_tb;

reg [3:0]bcd;

wire [6:0]leds;

wire a,b,c,d,e,f,g;

integer x; //반복해서 확인하려는 생각에 변수 설정

initial begin

bcd=0; //입력값 초기화

#100; //for global reset to finish – 사실 여기선 딱히 필요없는 듯 하다. 그래서 100ns빨간

for(x=0;x<=20;x=x+1) //반복문 시작 (20번)

#50 bcd=bcd+1; //50초마다 bcd는+1됨

$stop; //반복문 끝

end

endmodule