Midterm Project

Building a 7-Segment Display Circuit

컴퓨터과학부

2017920064 강민철

1. 구현 목표

7 Segment Display 중 DP를 제외한 7개의 비트로 1~F까지 표현하기 위해

우선 어떤 Segment에서 불이 들어와야 할 지를 먼저 결정해야 할 것이다

이를 토대로 input 값이 각각 1~F일 때 어떤 Segment에서 불이 들어와야할지를 나타내는

Table을 작성해 보자면

in	out
0000000000000001	1111110
0000000000000010	0110000
0000000000000100	1101101
0000000000001000	1111001
0000000000010000	0110011
0000000000100000	1011011
0000000001000000	1011111
0000000010000000	1110000
0000000100000000	1111111
0000001000000000	1110011
0000010000000000	1110111
0000100000000000	0011111
0001000000000000	1001110
0010000000000000	0111101
0100000000000000	1001111
1000000000000000	1000111

이와 같이 작성될 것이다. 이 out을 Decoder7.cmp와 Seg7.cmp에 그대로 대입해준다

2. Encoder 구현 논리

우선 이 Encoder를 구현하는 방법을 크게 나누어 보자면

Not 16과 Not, 그리고 Or gate로 구현하는 방법,

그리고 4to2 encoder를 계층적으로 구성하는 방법

이렇게 두 가지 방법이 있다.

본 프로젝트에서는 전자의 방식을 차용해 Encoder를 만들었는데,

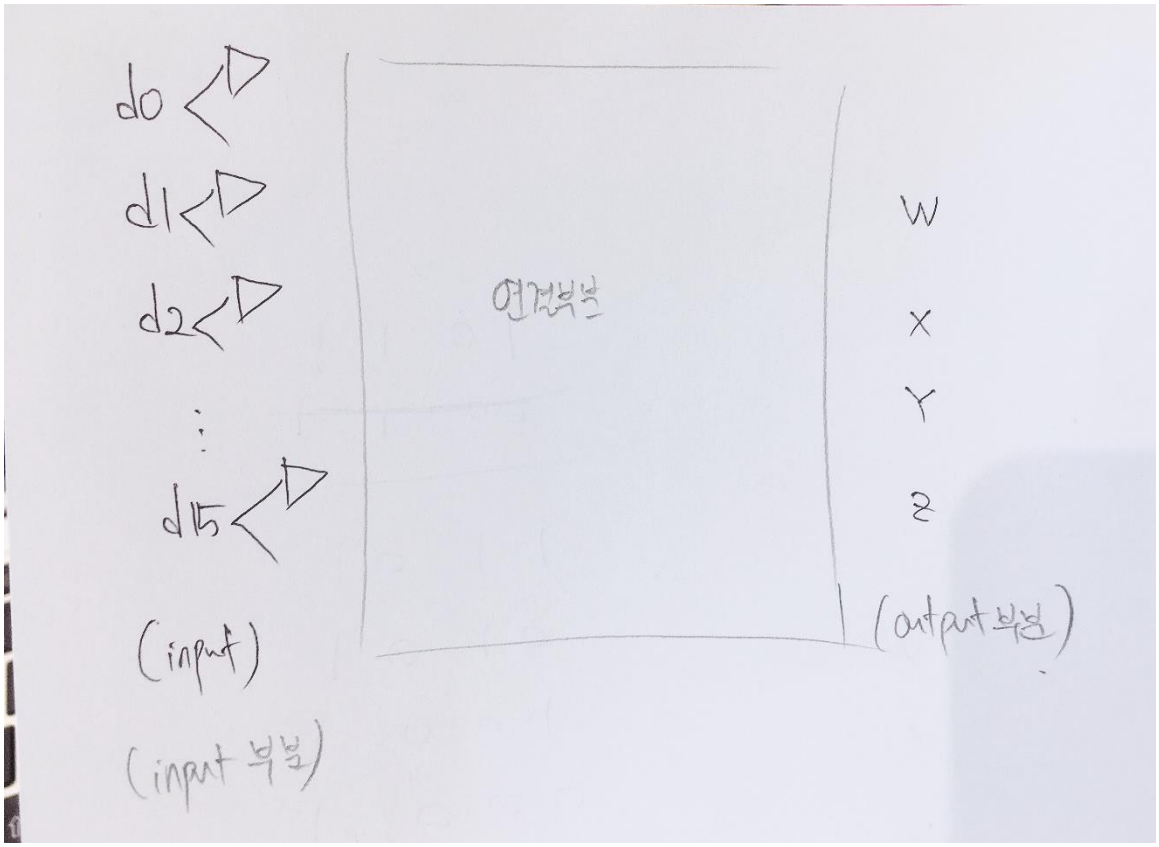
아래의 사진처럼 각 input에 대해 not과 buffer를 나누어 32개의 input을 만들고,

w, x, y, z 에 해당하는 output에 맞게

연결부분에서 이어주기만 하면 된다

w,x,y,z 를 구현하는 과정을 제외한 don't care condition은

문제 조건에서 무시해도 무관하다고 하였으므로 고려하지 않았다



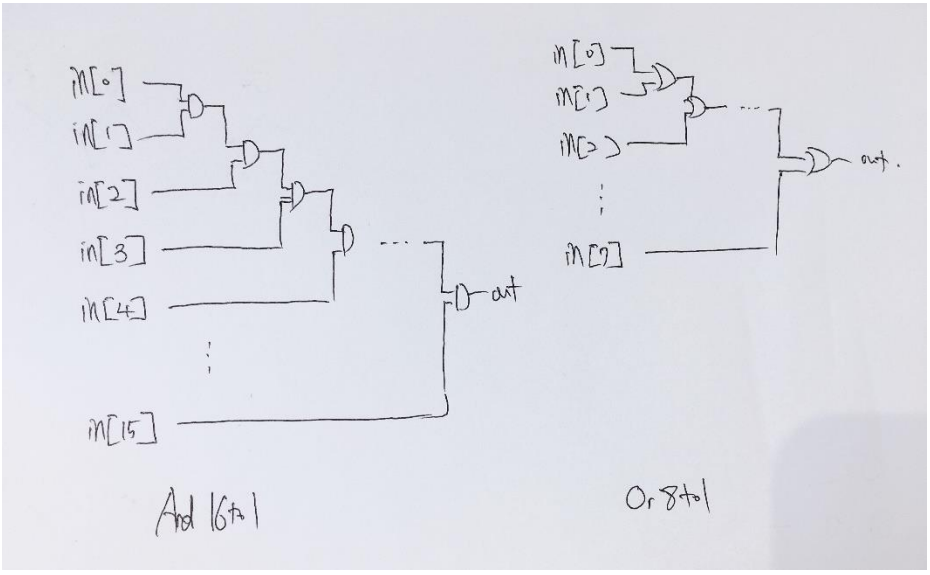
32개의 input 부분을 만드는 코드

```
Not(in=in[0], out=nd0);
    Not(in=in[1], out=nd1);
    Not(in=in[2], out=nd2);
    Not(in=in[3], out=nd3);
    //중략

    Not(in=in[12], out=nd12);

    Not(in=in[13], out=nd13);

    Not(in=in[14], out=nd14);
    Not(in=in[15], out=nd15);



    Not(in=nd0, out=d0);
    Not(in=nd1, out=d1);
    Not(in=nd2, out=d2);
    //중략
    Not(in=nd14, out=d14);
    Not(in=nd15, out=d15);
```

그리고 연결부분에서 묶어주는 과정에서

Helper pin인 And16to1과 Or8to1을 추가적으로 구현하여 활용하였다

And16to1과 Or8to1은 각각 아래 사진과 같은 게이트를 구현한 것이다



이제, W, x, y, z (output들)에 input pin을 연결하는 과정은 그냥 1을 And16to1로 묶어 Or8to1로 내보내는 과정에 불과해 진다.

```
// w 값 도출
    And16to1(in[0]=nd0 , in[1]=nd1 , in[2]=nd2 , in[3]=nd3 , in[4]=nd4 , in[5]=nd5 , in[6]=nd6 , in[7]=nd7,
in[8]=d8 , in[9]= nd9, in[10]=nd10 , in[11]=nd11 , in[12]=nd12 , in[13]= nd13, in[14]= nd14, in[15]=nd15, out=w0);
    And16to1(in[0]=nd0 , in[1]=nd1 , in[2]=nd2 , in[3]=nd3 , in[4]=nd4 , in[5]=nd5 , in[6]=nd6 , in[7]=nd7,
in[8]=nd8 , in[9]=d9, in[10]=nd10 , in[11]=nd11 , in[12]=nd12 , in[13]= nd13, in[14]= nd14, in[15]=nd15, out=w1);
    And16to1(in[0]=nd0 , in[1]=nd1 , in[2]=nd2 , in[3]=nd3 , in[4]=nd4 , in[5]=nd5 , in[6]=nd6 , in[7]=nd7,
in[8]=nd8 , in[9]= nd9, in[10]=d10, in[11]=nd11 , in[12]=nd12 , in[13]= nd13, in[14]= nd14, in[15]=nd15, out=w2);
    And16to1(in[0]=nd0 , in[1]=nd1 , in[2]=nd2 , in[3]=nd3 , in[4]=nd4 , in[5]=nd5 , in[6]=nd6 , in[7]=nd7,
in[8]=nd8 , in[9]= nd9, in[10]=nd10 , in[11]=d11, in[12]=nd12 , in[13]= nd13, in[14]= nd14, in[15]=nd15, out=w3);
    And16to1(in[0]=nd0 , in[1]=nd1 , in[2]=nd2 , in[3]=nd3 , in[4]=nd4 , in[5]=nd5 , in[6]=nd6 , in[7]=nd7,
in[8]=nd8 , in[9]= nd9, in[10]=nd10 , in[11]=nd11 , in[12]=d12, in[13]= nd13, in[14]= nd14, in[15]=nd15, out=w4);
    And16to1(in[0]=nd0 , in[1]=nd1 , in[2]=nd2 , in[3]=nd3 , in[4]=nd4 , in[5]=nd5 , in[6]=nd6 , in[7]=nd7,
in[8]=nd8 , in[9]= nd9, in[10]=nd10 , in[11]=nd11 , in[12]=nd12 , in[13]=d13, in[14]= nd14, in[15]=nd15, out=w5);
    And16to1(in[0]=nd0 , in[1]=nd1 , in[2]=nd2 , in[3]=nd3 , in[4]=nd4 , in[5]=nd5 , in[6]=nd6 , in[7]=nd7,
in[8]=nd8 , in[9]= nd9, in[10]=nd10 , in[11]=nd11 , in[12]=nd12 , in[13]= nd13, in[14]=d14, in[15]=nd15, out=w6);
    And16to1(in[0]=nd0 , in[1]=nd1 , in[2]=nd2 , in[3]=nd3 , in[4]=nd4 , in[5]=nd5 , in[6]=nd6 , in[7]=nd7,
in[8]=nd8 , in[9]= nd9, in[10]=nd10 , in[11]=nd11 , in[12]=nd12 , in[13]= nd13, in[14]=nd14, in[15]=d15, out=w7);


    Or8to1(in[0]=w0, in[1]=w1, in[2]=w2, in[3]=w3, in[4]=w4, in[5]=w5, in[6]=w6, in[7]=w7, out=out[3] );
```

x, y, z 또한 위와 같은 방식으로 구현하면 된다

3. Decoder 구현 논리

각각의 minterm들을 해당 output에 맞게 연결하여 내보내기만 하면 된다

```
    // Use the following pins (w, notw, x, notx, y, noty, z, notz) for easier
implementation
    Not(in=in[3],out=notw); // NOT(w)
    Not(in=notw,out=w);     // w (==out[3])
    Not(in=in[2],out=notx); // NOT(x)
    Not(in=notx,out=x);     // x (==out[2])
    Not(in=in[1],out=noty); // NOT(y)
    Not(in=noty,out=y);     // y (==out[1])
    Not(in=in[0],out=notz); // NOT(z)
    Not(in=notz,out=z);     // z (==out[0])
```

문제에서 제시된 위 input pin들을 활용하여 연결하는 과정은 encoder를 구현하는 과정과 대동소이하다

4. 최종 결과

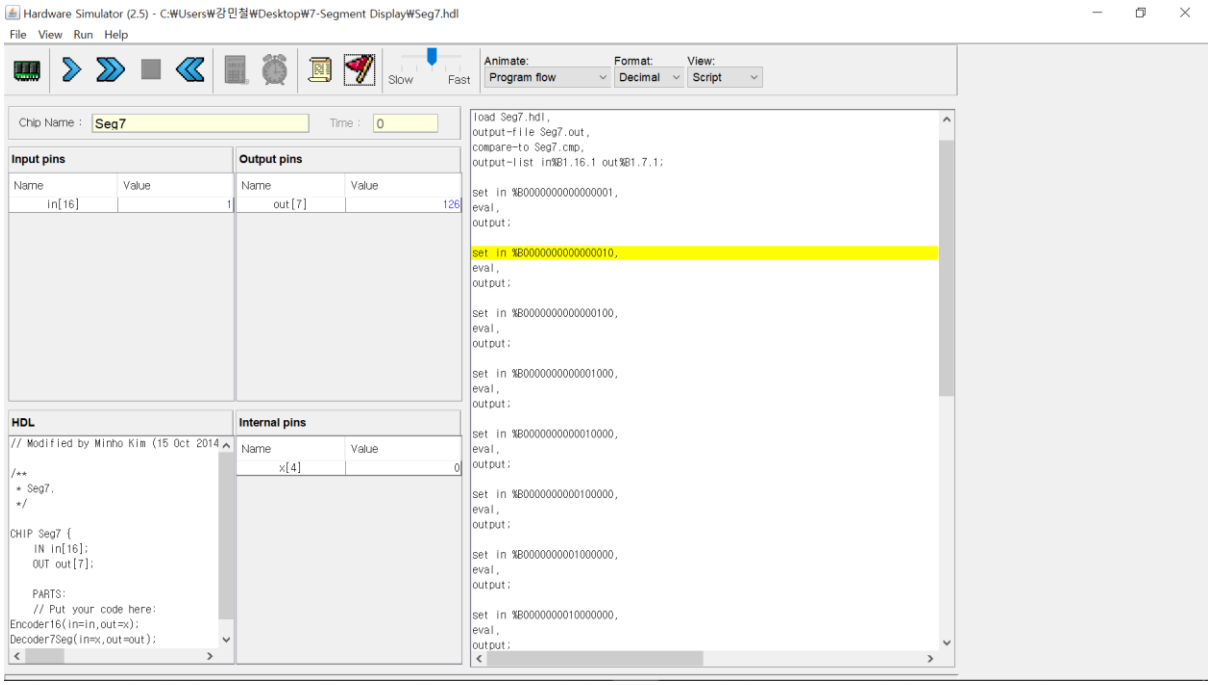
<1. 구현목표>에서 제시한 표의 output 부분, 즉

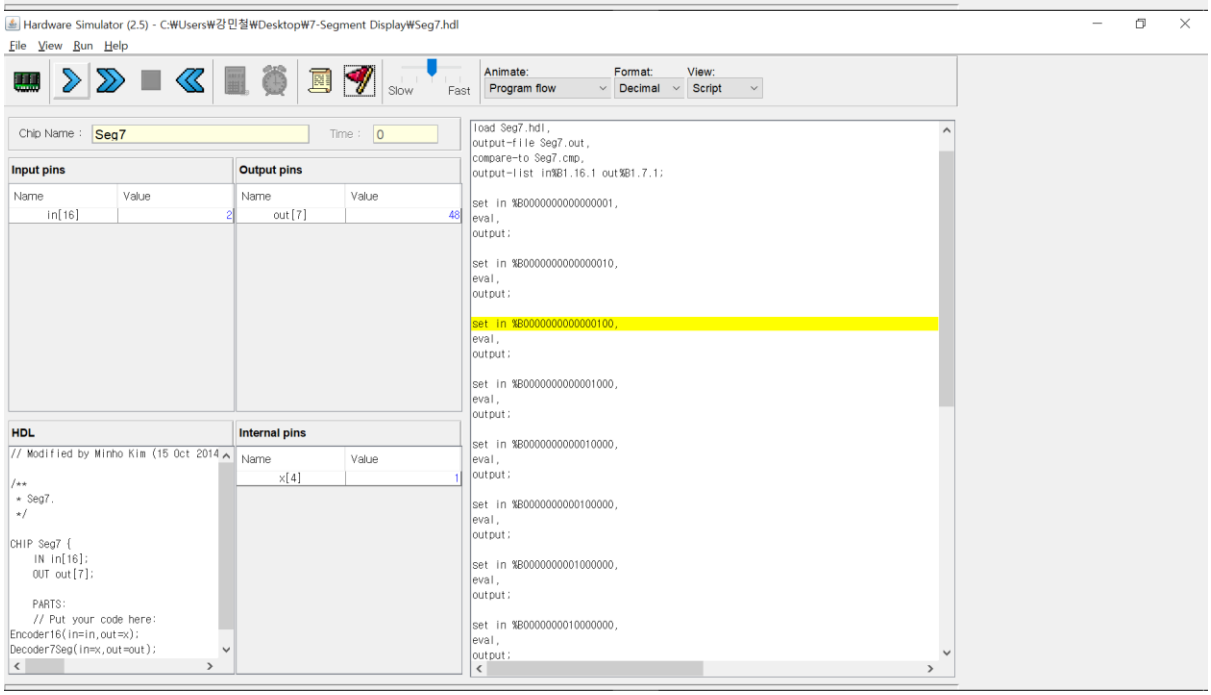
in	out
0000000000000001	1111110
0000000000000010	0110000
0000000000000100	1101101
0000000000001000	1111001
0000000000010000	0110011
0000000000100000	1011011
0000000001000000	1011111
0000000010000000	1110000
0000000100000000	1111111
0000001000000000	1110011
0000010000000000	1110111
0000100000000000	0011111
0001000000000000	1001110
0010000000000000	0111101
0100000000000000	1001111
1000000000000000	1000111

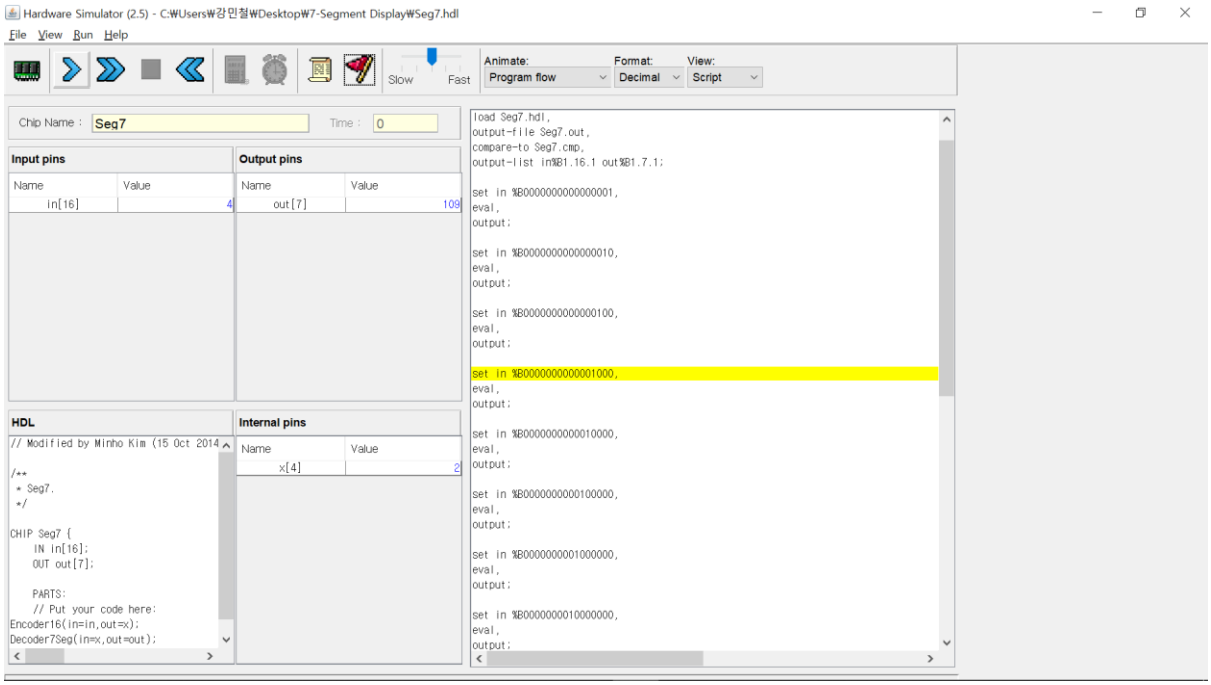
붉게 칠한 부분을 10진수로 해석해보면 각각

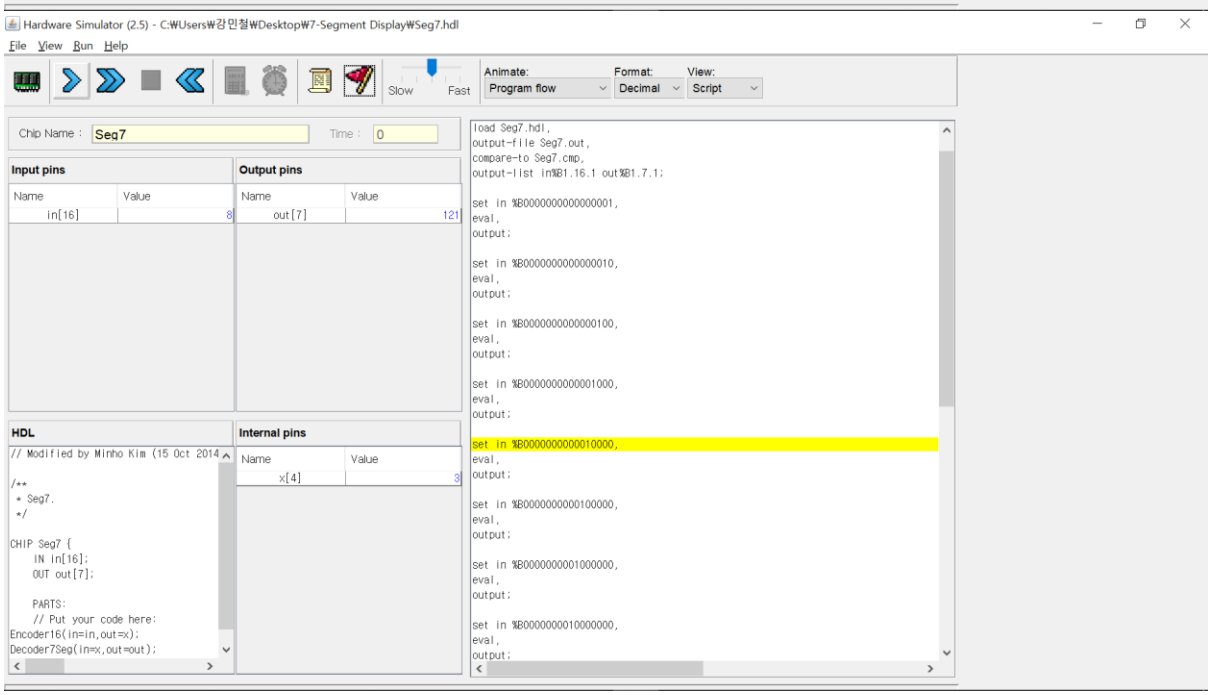
126, 48, 109, 121, 51, 91, 95, 112, 127, 115, 119, 31, 78, 61, 79, 71이 된다

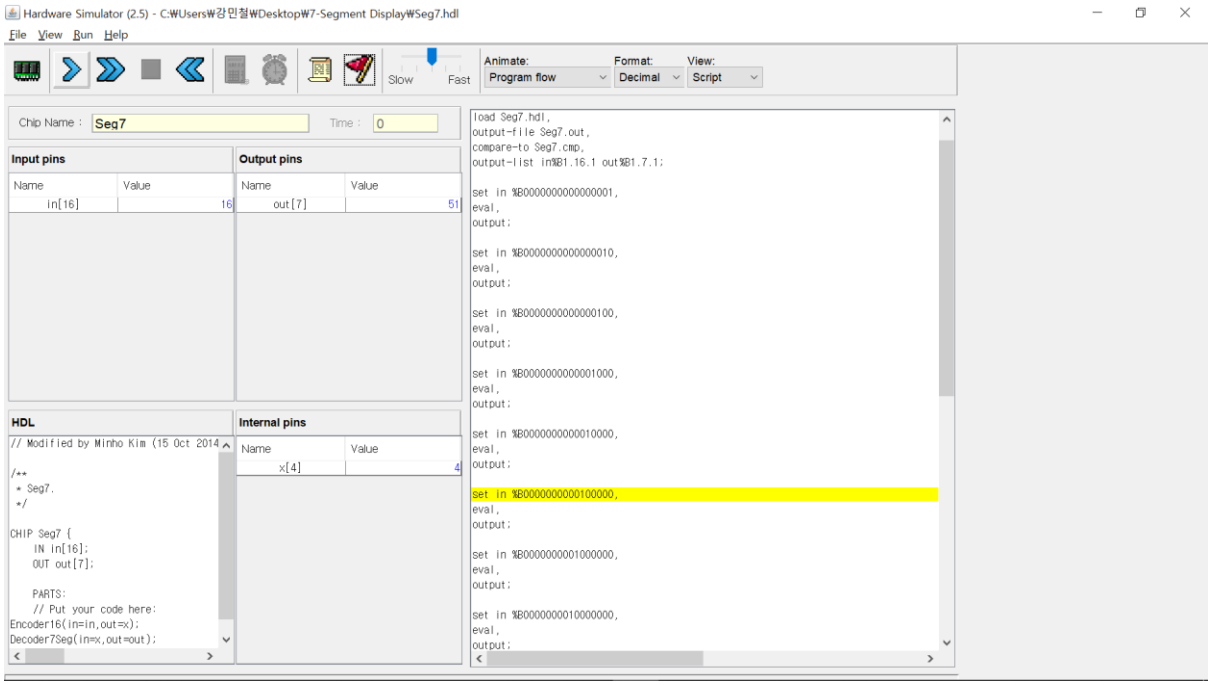
최종적으로 완성된 Seg7.hdl 를 Seg.tst로 Hardware Simulator로 돌려보면 결과는 아래와 같다

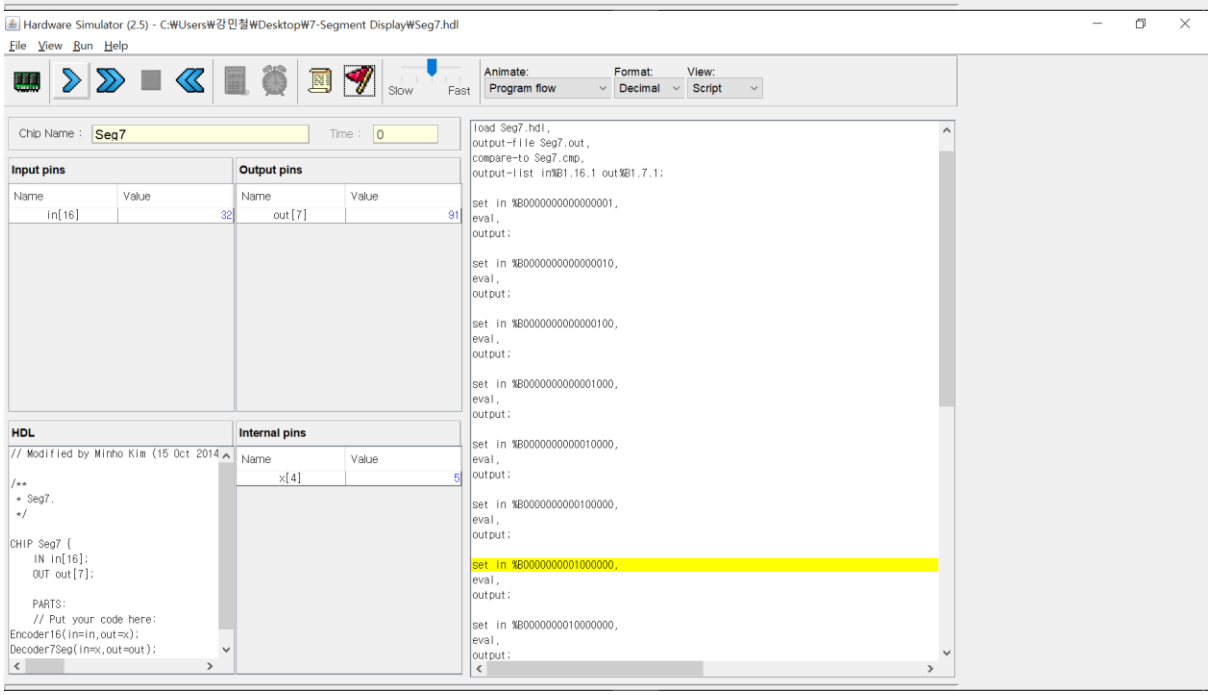

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
1
Output pins
Name
Value
out[7]
126
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
0
load Seg7.hdl,
output-file Seg7.out,
compare-to Seg7.cmp,
output-list in%B1.16.1 out%B1.7.1;
set in %B0000000000000001,
eval,
output;
set in %B0000000000000010,
eval,
output;
set in %B0000000000000100,
eval,
output;
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;

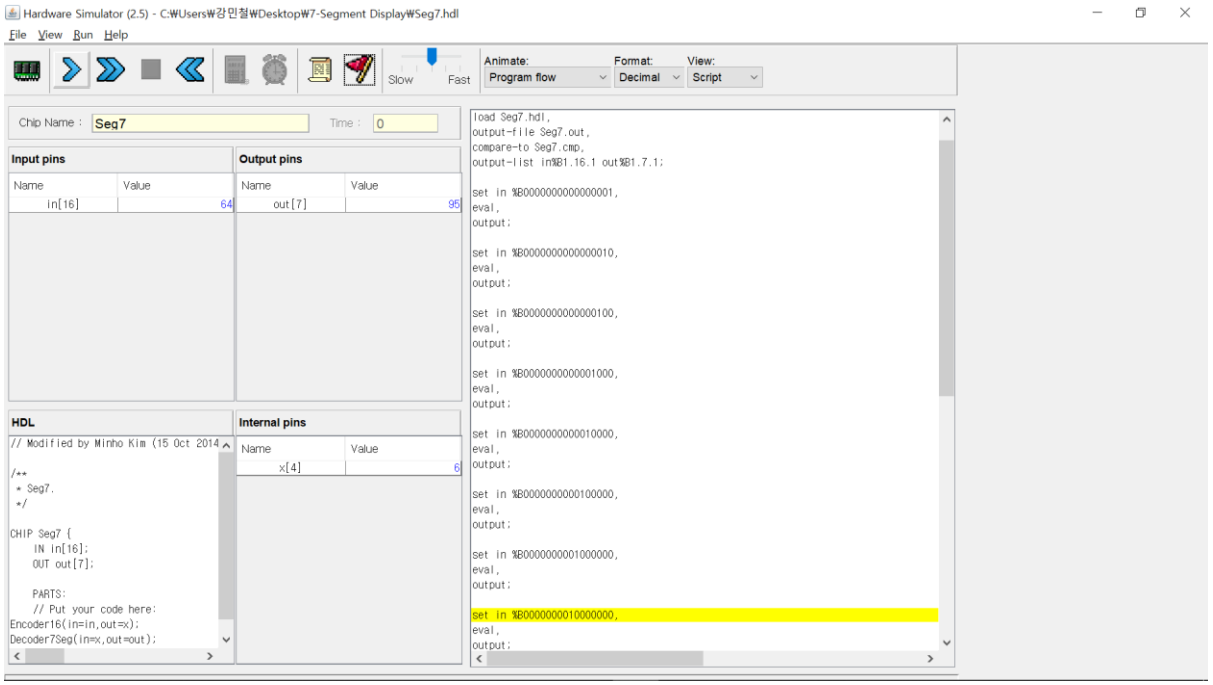

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
2
Output pins
Name
Value
out[7]
48
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
1
load Seg7.hdl,
output-file Seg7.out,
compare-to Seg7.cmp,
output-list in%B1.16.1 out%B1.7.1;
set in %B0000000000000001,
eval,
output;
set in %B0000000000000010,
eval,
output;
set in %B0000000000000100,
eval,
output;
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;

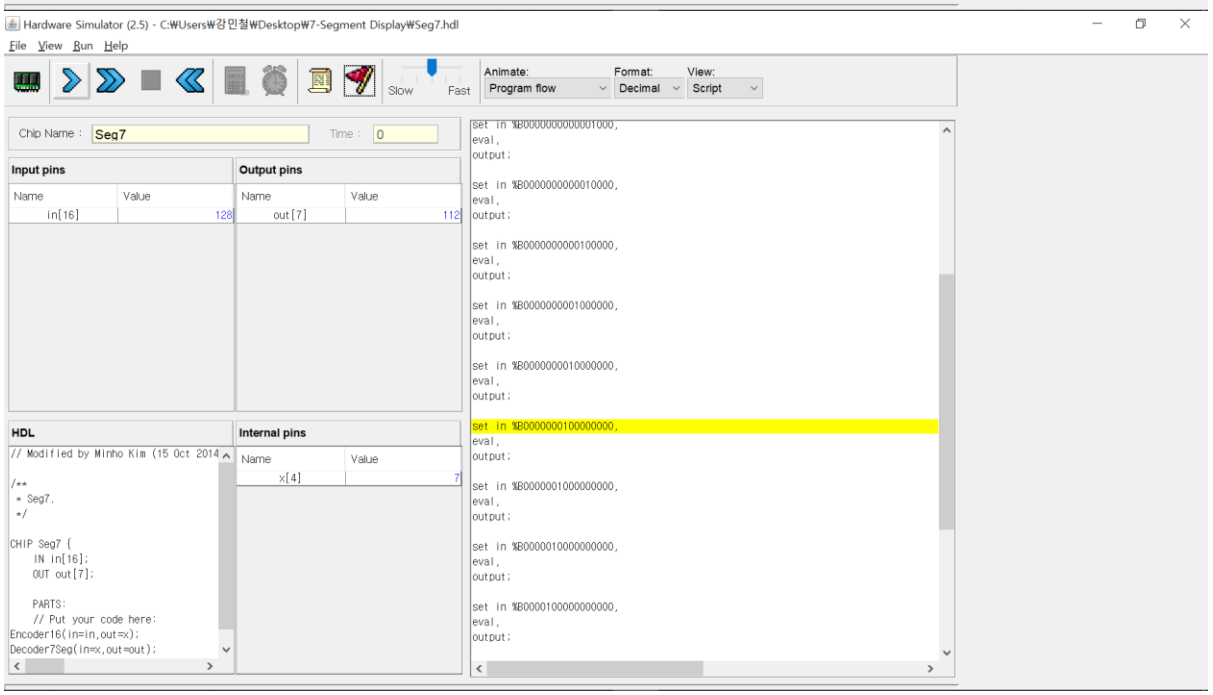

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
4
Output pins
Name
Value
out[7]
109
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
2
load Seg7.hdl,
output-file Seg7.out,
compare-to Seg7.cmp,
output-list in%B1.16.1 out%B1.7.1;
set in %B0000000000000001,
eval,
output;
set in %B0000000000000010,
eval,
output;
set in %B0000000000000100,
eval,
output;
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;

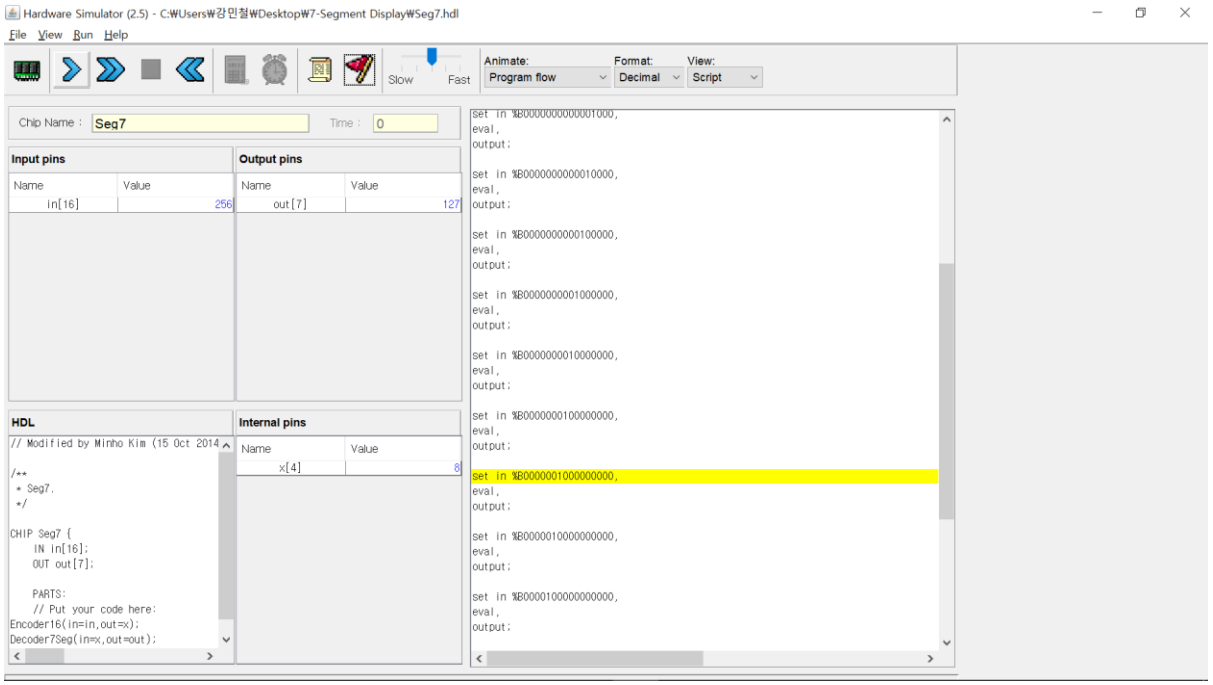

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
8
Output pins
Name
Value
out[7]
121
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
3
load Seg7.hdl,
output-file Seg7.out,
compare-to Seg7.cmp,
output-list in%B1.16.1 out%B1.7.1;
set in %B0000000000000001,
eval,
output;
set in %B0000000000000010,
eval,
output;
set in %B0000000000000100,
eval,
output;
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;

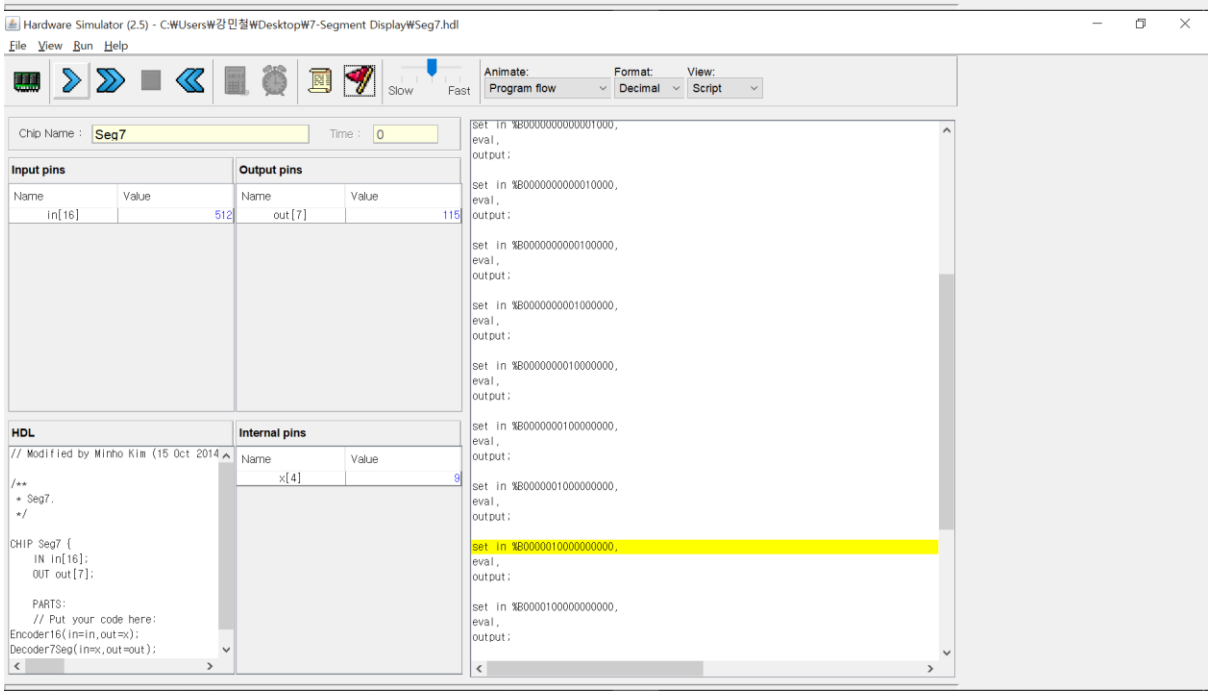

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
16
Output pins
Name
Value
out[7]
51
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
4
load Seg7.hdl,
output-file Seg7.out,
compare-to Seg7.cmp,
output-list in%B1.16.1 out%B1.7.1;
set in %B0000000000000001,
eval,
output;
set in %B0000000000000010,
eval,
output;
set in %B0000000000000100,
eval,
output;
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;

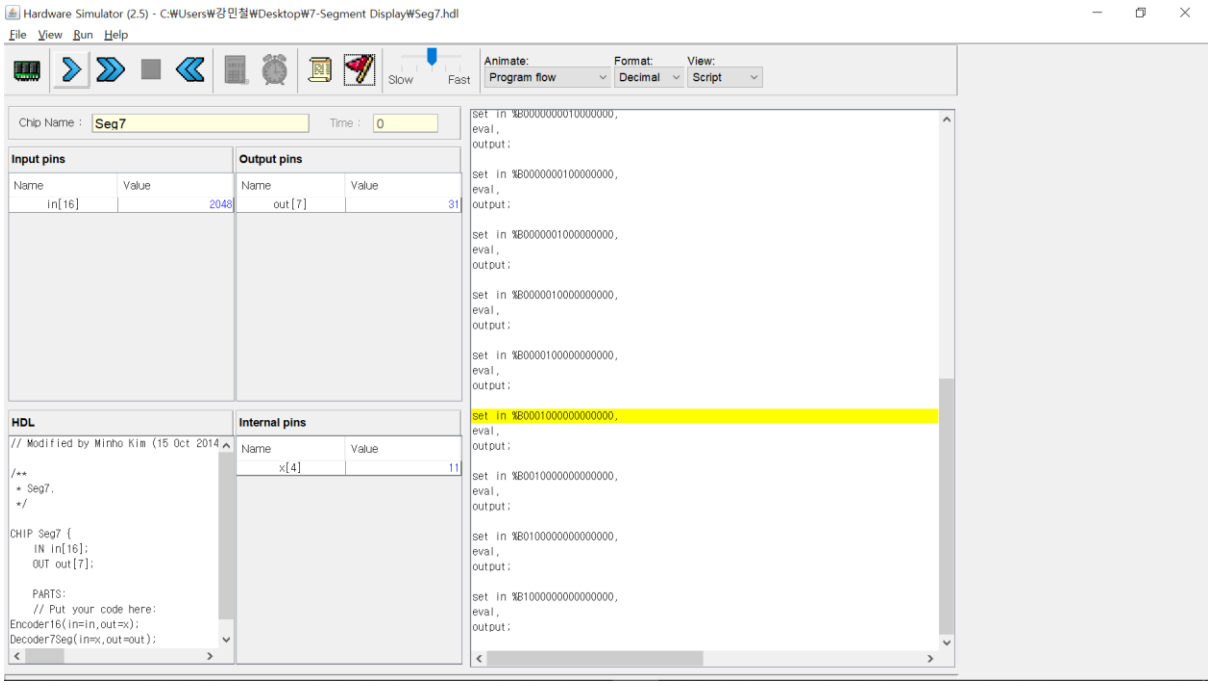

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
32
Output pins
Name
Value
out[7]
91
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
5
load Seg7.hdl,
output-file Seg7.out,
compare-to Seg7.cmp,
output-list in%B1.16.1 out%B1.7.1;
set in %B0000000000000001,
eval,
output;
set in %B0000000000000010,
eval,
output;
set in %B0000000000000100,
eval,
output;
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;

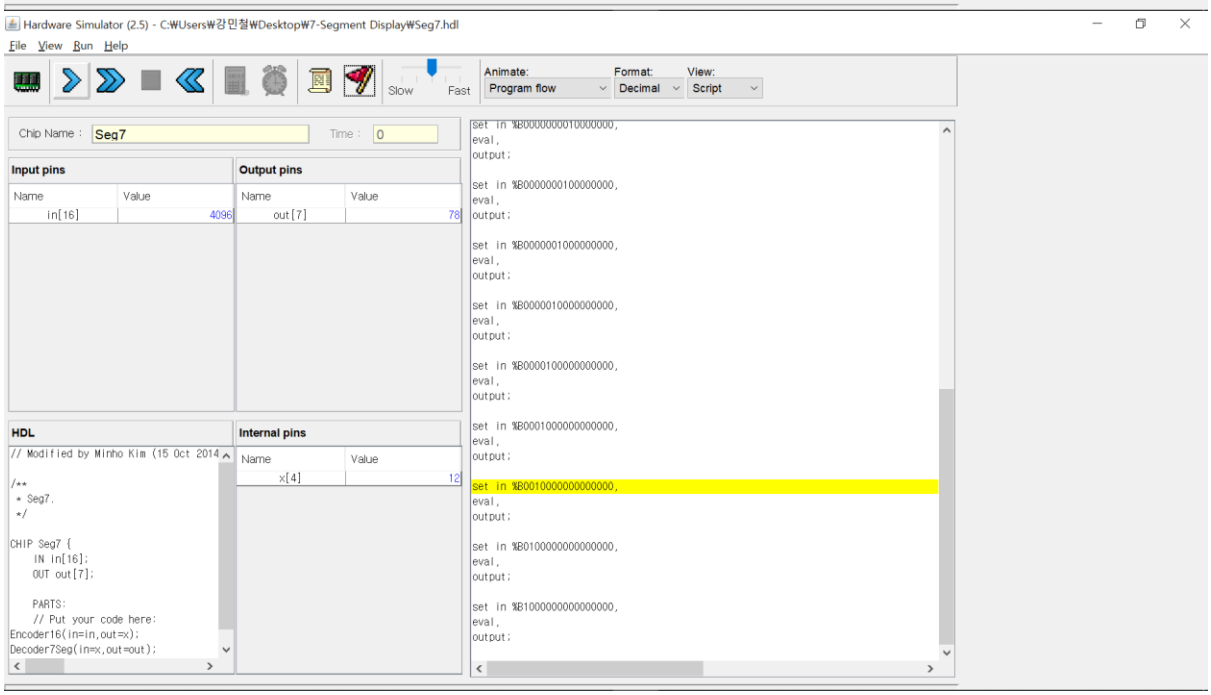

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
64
Output pins
Name
Value
out[7]
95
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
6
load Seg7.hdl,
output-file Seg7.out,
compare-to Seg7.cmp,
output-list in%B1.16.1 out%B1.7.1;
set in %B0000000000000001,
eval,
output;
set in %B0000000000000010,
eval,
output;
set in %B0000000000000100,
eval,
output;
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;

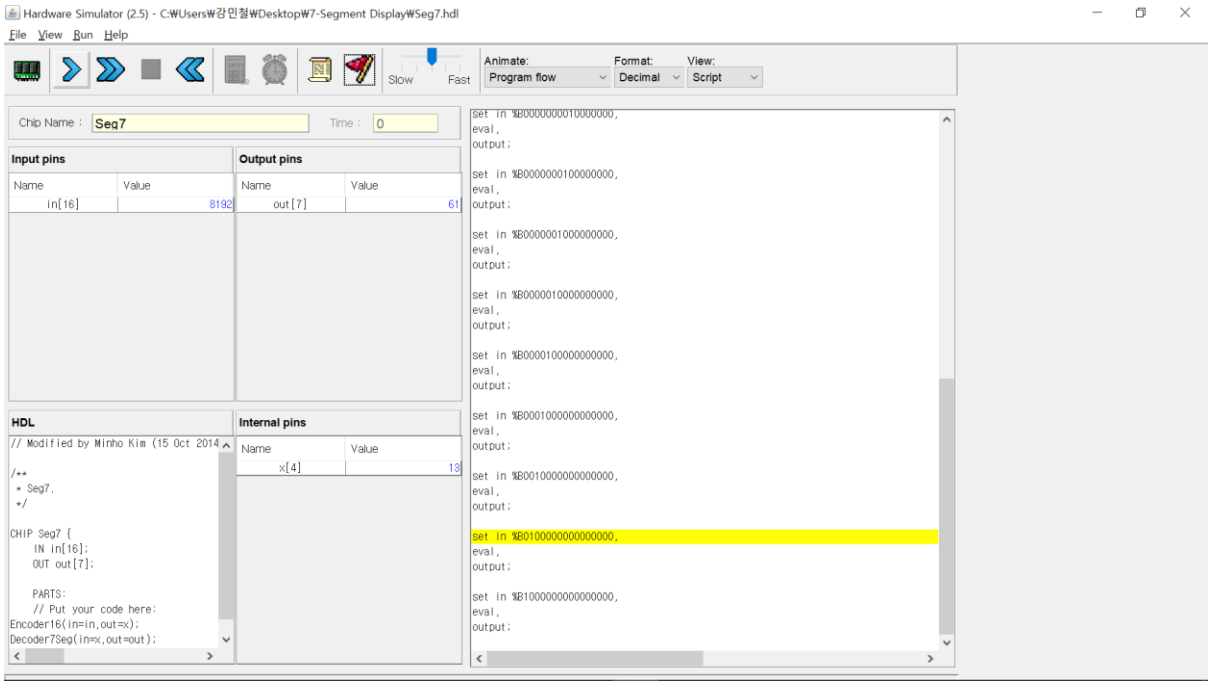

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
128
Output pins
Name
Value
out[7]
112
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
7
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;
set in %B0000000100000000,
eval,
output;
set in %B0000001000000000,
eval,
output;
set in %B0000010000000000,
eval,
output;

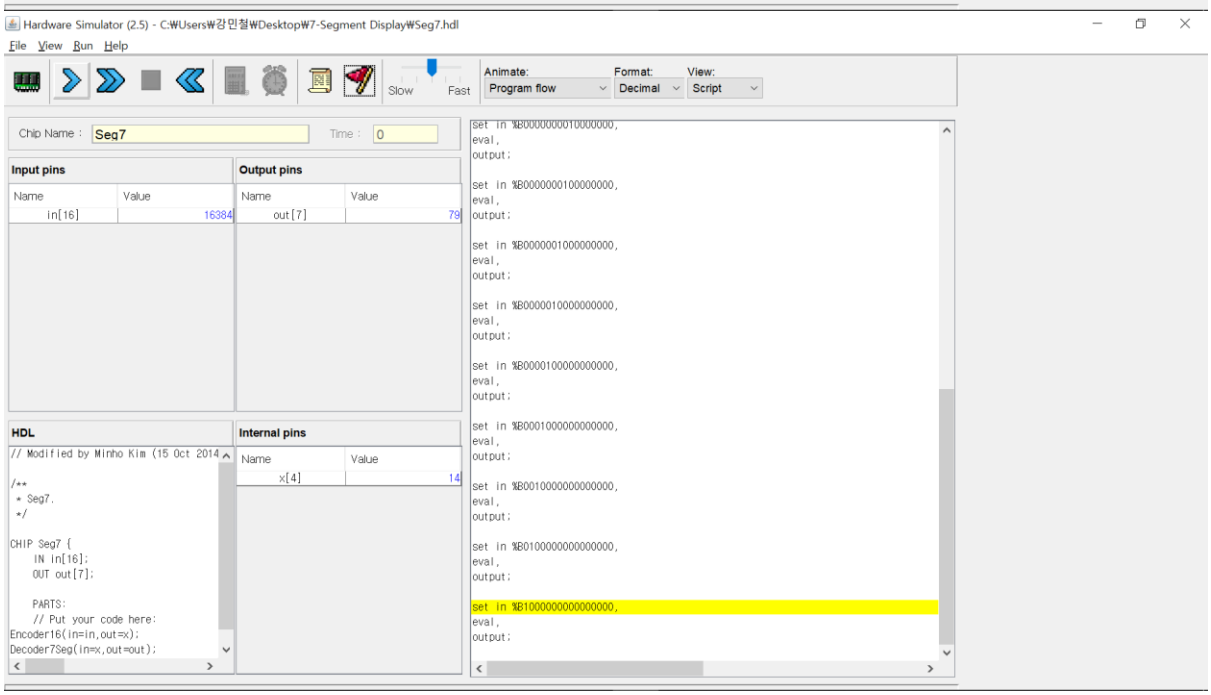

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
256
Output pins
Name
Value
out[7]
127
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
8
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;
set in %B0000000100000000,
eval,
output;
set in %B0000001000000000,
eval,
output;
set in %B0000010000000000,
eval,
output;
set in %B0000100000000000,
eval,
output;

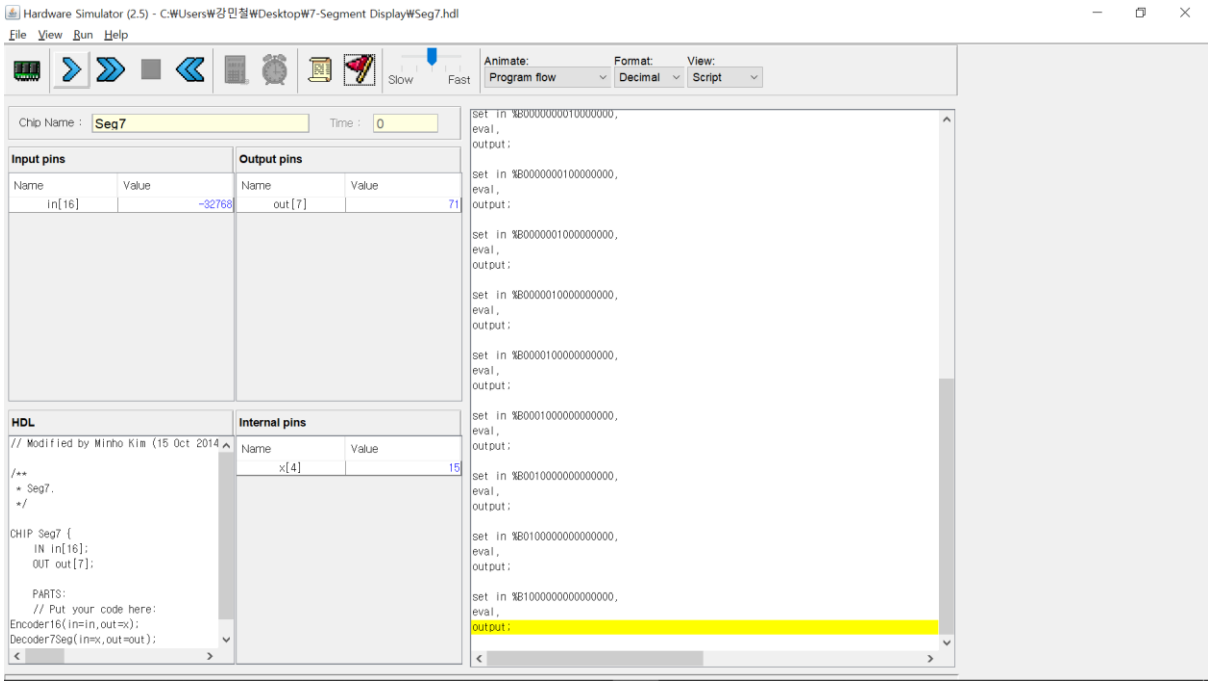

Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
512
Output pins
Name
Value
out[7]
115
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
9
set in %B0000000000001000,
eval,
output;
set in %B0000000000010000,
eval,
output;
set in %B0000000000100000,
eval,
output;
set in %B0000000001000000,
eval,
output;
set in %B0000000010000000,
eval,
output;
set in %B0000000100000000,
eval,
output;
set in %B0000001000000000,
eval,
output;
set in %B0000010000000000,
eval,
output;


Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
2048
Output pins
Name
Value
out[7]
31
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
11
set in %B0000000001000000,
eval,
output;
set in %B0000000100000000,
eval,
output;
set in %B0000001000000000,
eval,
output;
set in %B0000010000000000,
eval,
output;
set in %B0000100000000000,
eval,
output;
set in %B0001000000000000,
eval,
output;
set in %B0010000000000000,
eval,
output;
set in %B0100000000000000,
eval,
output;
set in %B1000000000000000,
eval,
output;


Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
4096
Output pins
Name
Value
out[7]
78
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
12
set in %B0000000001000000,
eval,
output;
set in %B0000000100000000,
eval,
output;
set in %B0000001000000000,
eval,
output;
set in %B0000010000000000,
eval,
output;
set in %B0000100000000000,
eval,
output;
set in %B0001000000000000,
eval,
output;
set in %B0010000000000000,
eval,
output;
set in %B0100000000000000,
eval,
output;
set in %B1000000000000000,
eval,
output;


Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
8192
Output pins
Name
Value
out[7]
61
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
13
set in %B0000000010000000,
eval,
output;
set in %B0000000100000000,
eval,
output;
set in %B0000001000000000,
eval,
output;
set in %B0000010000000000,
eval,
output;
set in %B0000100000000000,
eval,
output;
set in %B0001000000000000,
eval,
output;
set in %B0010000000000000,
eval,
output;
set in %B0100000000000000,
eval,
output;
set in %B1000000000000000,
eval,
output;


Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
16384
Output pins
Name
Value
out[7]
79
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
14
set in %B0000000010000000,
eval,
output;
set in %B0000000100000000,
eval,
output;
set in %B0000001000000000,
eval,
output;
set in %B0000010000000000,
eval,
output;
set in %B0000100000000000,
eval,
output;
set in %B0001000000000000,
eval,
output;
set in %B0010000000000000,
eval,
output;
set in %B0100000000000000,
eval,
output;
set in %B1000000000000000,
eval,
output;


Hardware Simulator (2.5) - C:₩Users₩강민철₩Desktop₩7-Segment Display₩Seg7.hdl
File View Run Help
Slow
Fast
Animate:
Program flow
Format:
Decimal
View:
Script
Chip Name :
Seg7
Time :
0
Input pins
Name
Value
in[16]
-32768
Output pins
Name
Value
out[7]
71
HDL
// Modified by Minho Kim (15 Oct 2014
/**
* Seg7.
*/
CHIP Seg7 {
IN in[16];
OUT out[7];
PARTS:
// Put your code here:
Encoder16(in=in,out=x);
Decoder7Seg(in=x,out=out);
Internal pins
Name
Value
x[4]
15
set in %B0000000010000000,
eval,
output;
set in %B0000000100000000,
eval,
output;
set in %B0000001000000000,
eval,
output;
set in %B0000010000000000,
eval,
output;
set in %B0000100000000000,
eval,
output;
set in %B0001000000000000,
eval,
output;
set in %B0010000000000000,
eval,
output;
set in %B0100000000000000,
eval,
output;
set in %B1000000000000000,
eval,
output;