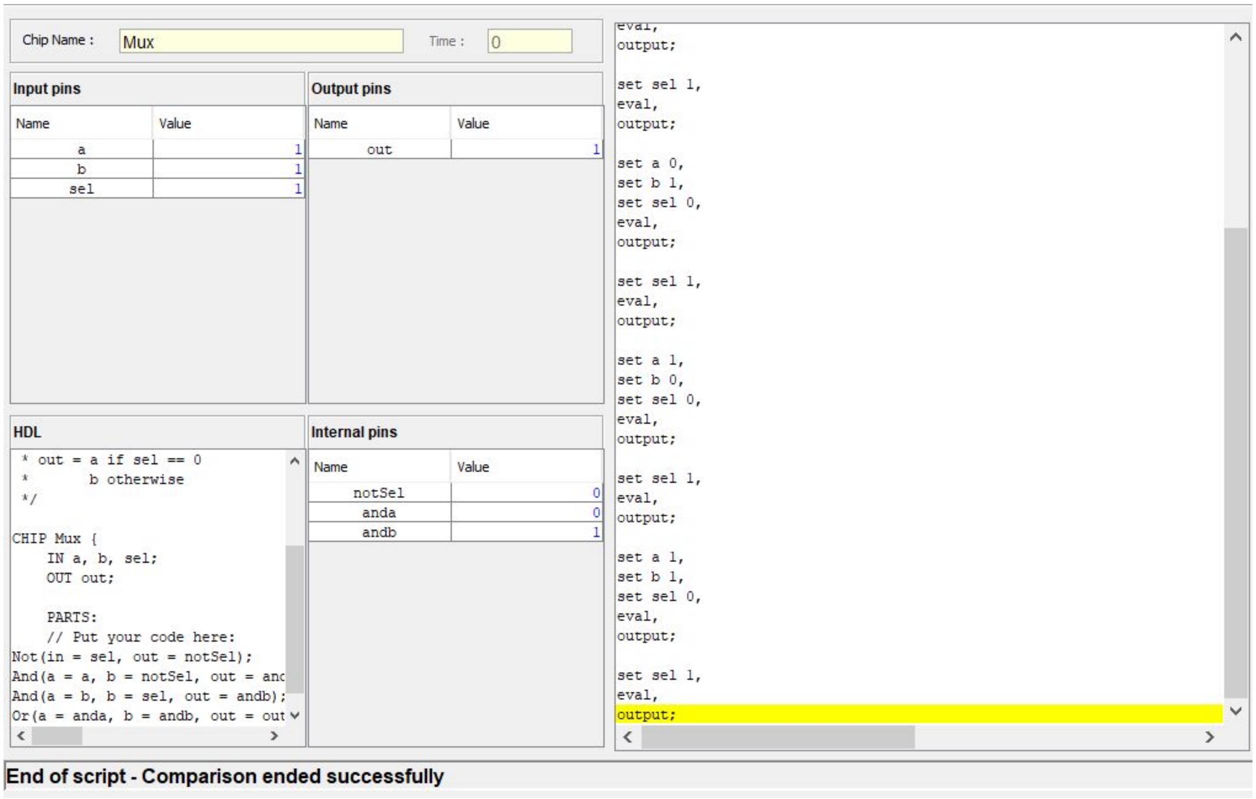
Mux



Mux

0

Input pins

Name	Value
a	1
b	1
sel	1

Output pins

Name	Value
out	1

HDL

```
 * out = a if sel == 0
 *       b otherwise
 */

CHIP Mux {
    IN a, b, sel;
    OUT out;

    PARTS:
    // Put your code here:
Not(in = sel, out = notSel);
And(a = a, b = notSel, out = anc
And(a = b, b = sel, out = andb);
Or(a = anda, b = andb, out = out
```

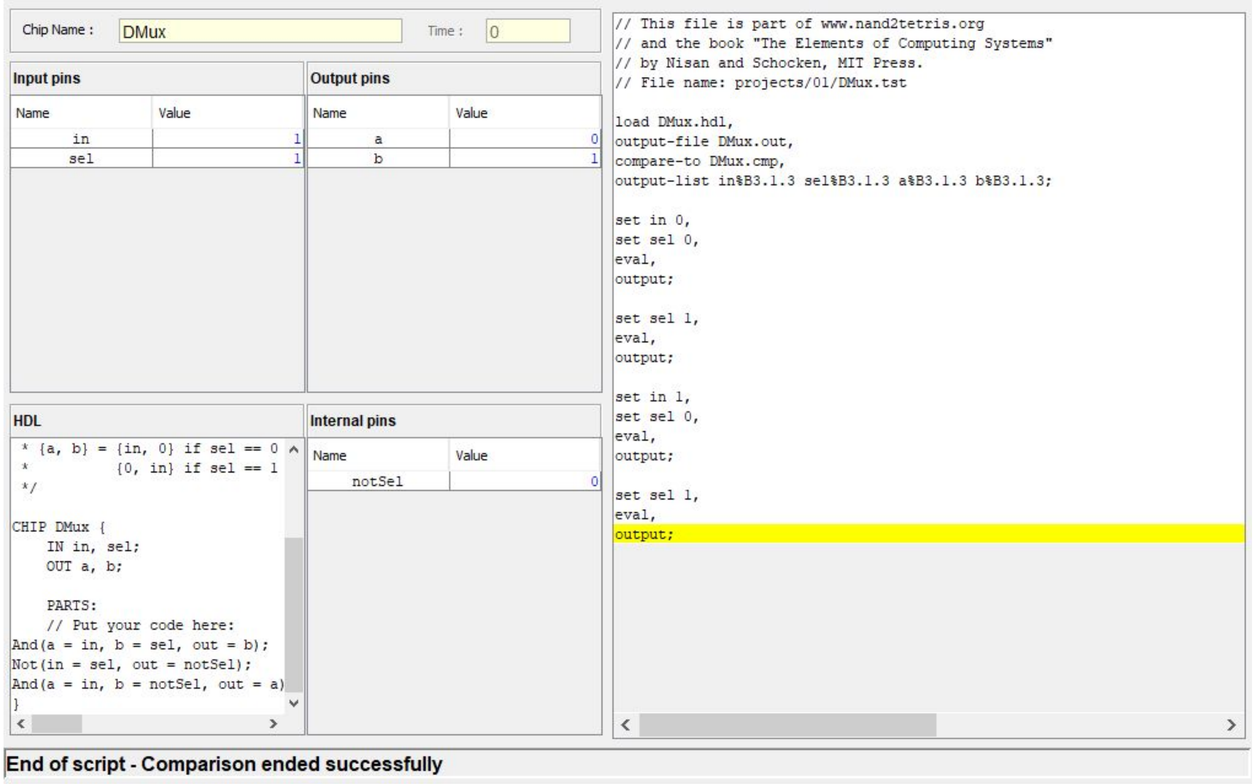
Internal pins

Name	Value
notSel	0
anda	0
andb	1

```
eval,
output;

set sel 1,
eval,
output;

set a 0,
set b 1,
set sel 0,
eval,
output;

set sel 1,
eval,
output;

set a 1,
set b 0,
set sel 0,
eval,
output;

set sel 1,
eval,
output;

set a 1,
set b 1,
set sel 0,
eval,
output;

set sel 1,
eval,
output;
```

End of script - Comparison ended successfully

DMux



Chip Name : DMux   Time : 0

Input pins

Name	Value
in	1
sel	1

Output pins

Name	Value
a	0
b	1

HDL

```
 * {a, b} = {in, 0} if sel == 0
 *          {0, in} if sel == 1
 */

CHIP DMux {
    IN in, sel;
    OUT a, b;

    PARTS:
    // Put your code here:
And(a = in, b = sel, out = b);
Not(in = sel, out = notSel);
And(a = in, b = notSel, out = a)
}
```

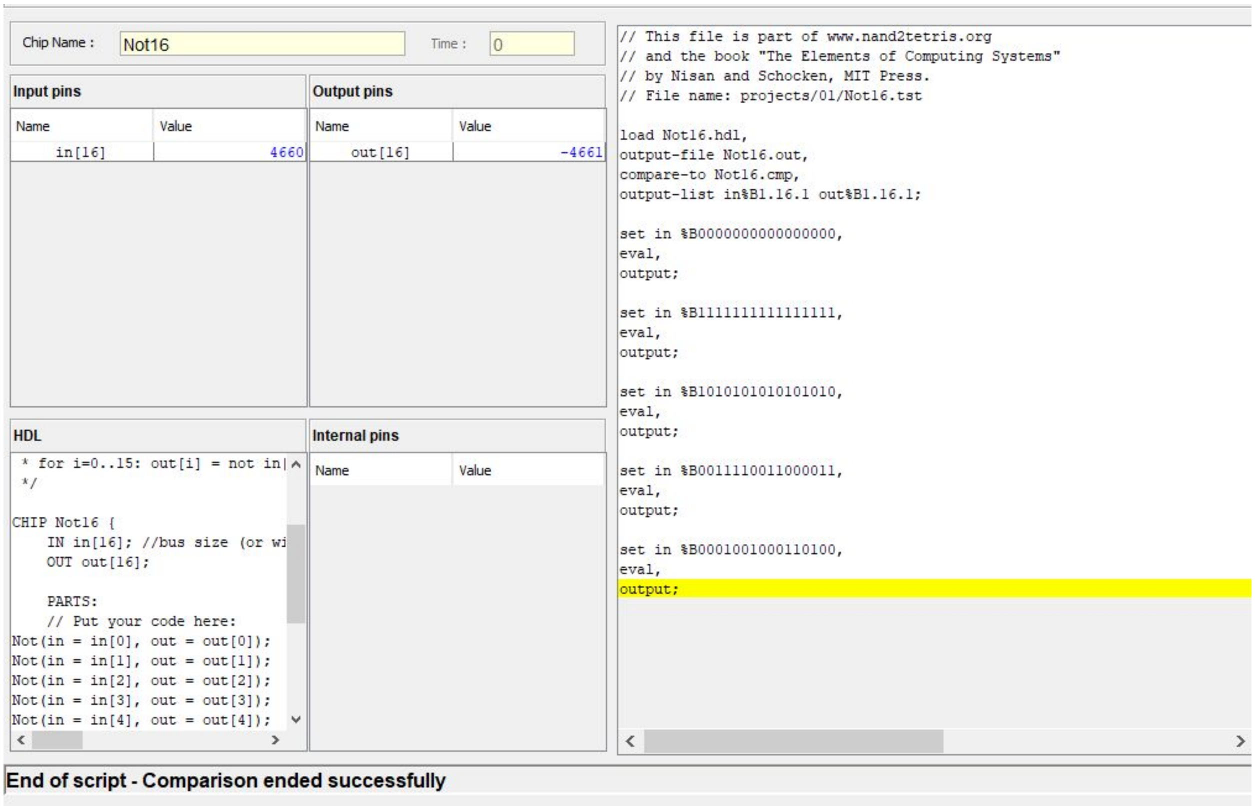
Internal pins

Name	Value
notSel	0

```
// This file is part of www.nand2tetris.org
// and the book "The Elements of Computing Systems"
// by Nisan and Schocken, MIT Press.
// File name: projects/01/DMux.tst

load DMux.hdl,
output-file DMux.out,
compare-to DMux.cmp,
output-list in%B3.1.3 sel%B3.1.3 a%B3.1.3 b%B3.1.3;

set in 0,
set sel 0,
eval,
output;

set sel 1,
eval,
output;

set in 1,
set sel 0,
eval,
output;

set sel 1,
eval,
output;
```

End of script - Comparison ended successfully

Not16



Chip Name : Not16

Time : 0

Name	Value
in[16]	4660

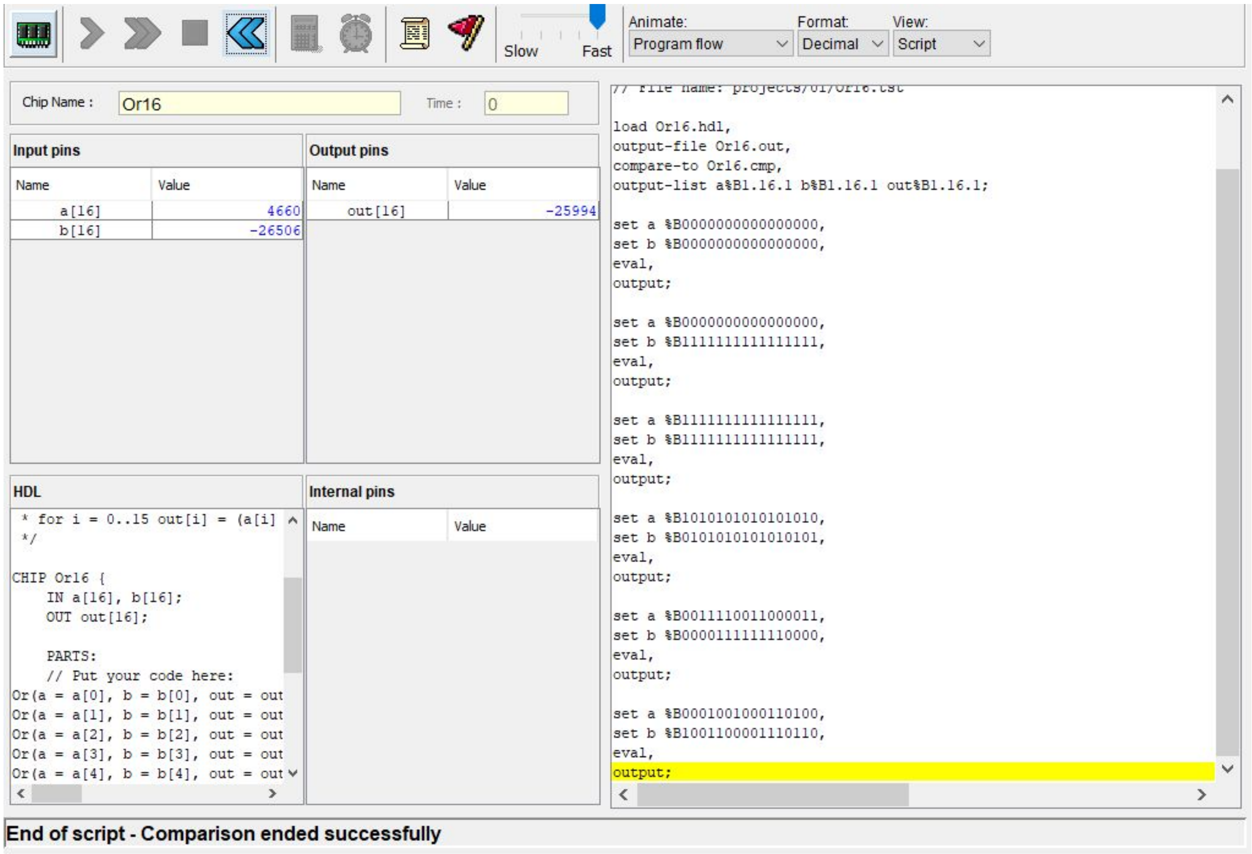
Name	Value
out[16]	-4661

```
 * for i=0..15: out[i] = not in[
 */

CHIP Not16 {
    IN in[16]; //bus size (or wi
    OUT out[16];

    PARTS:
    // Put your code here:
Not(in = in[0], out = out[0]);
Not(in = in[1], out = out[1]);
Not(in = in[2], out = out[2]);
Not(in = in[3], out = out[3]);
Not(in = in[4], out = out[4]);
```

```
// This file is part of www.nand2tetris.org
// and the book "The Elements of Computing Systems"
// by Nisan and Schocken, MIT Press.
// File name: projects/01/Not16.tst

load Not16.hdl,
output-file Not16.out,
compare-to Not16.cmp,
output-list in%B1.16.1 out%B1.16.1;

set in %B0000000000000000,
eval,
output;

set in %B1111111111111111,
eval,
output;

set in %B1010101010101010,
eval,
output;

set in %B0011110011000011,
eval,
output;

set in %B0001001000110100,
eval,
output;
```

End of script - Comparison ended successfully

Or16



Animate: Program flow
Format: Decimal
View: Script

Chip Name : Or16
Time : 0

Input pins

Name	Value
a[16]	4660
b[16]	-26506

Output pins

Name	Value
out[16]	-25994

HDL

```
 * for i = 0..15 out[i] = (a[i]
 */

CHIP Or16 {
    IN a[16], b[16];
    OUT out[16];

    PARTS:
    // Put your code here:
Or(a = a[0], b = b[0], out = out
Or(a = a[1], b = b[1], out = out
Or(a = a[2], b = b[2], out = out
Or(a = a[3], b = b[3], out = out
Or(a = a[4], b = b[4], out = out
```

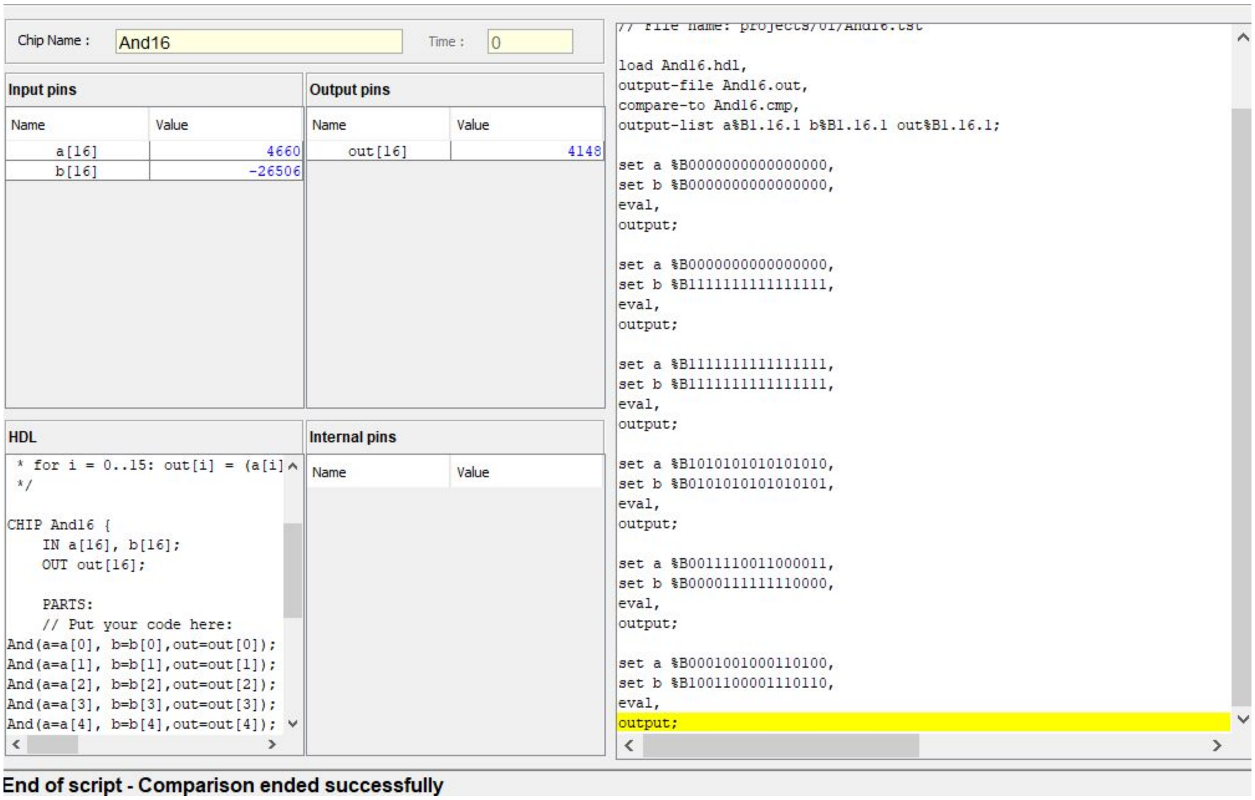
Internal pins

Name	Value

```
// File name: projects/01/Or16.tst

load Or16.hdl,
output-file Or16.out,
compare-to Or16.cmp,
output-list a%B1.16.1 b%B1.16.1 out%B1.16.1;

set a %B0000000000000000,
set b %B0000000000000000,
eval,
output;

set a %B0000000000000000,
set b %B1111111111111111,
eval,
output;

set a %B1111111111111111,
set b %B1111111111111111,
eval,
output;

set a %B1010101010101010,
set b %B0101010101010101,
eval,
output;

set a %B0011110011000011,
set b %B0000111111110000,
eval,
output;

set a %B0001001000110100,
set b %B1001100001110110,
eval,
output;
```

End of script - Comparison ended successfully

And16



Chip Name : And16   Time : 0

Input pins

Name	Value
a[16]	4660
b[16]	-26506

Output pins

Name	Value
out[16]	4148

HDL

```
 * for i = 0..15: out[i] = (a[i]
 */

CHIP And16 {
    IN a[16], b[16];
    OUT out[16];

    PARTS:
    // Put your code here:
And(a=a[0], b=b[0],out=out[0]);
And(a=a[1], b=b[1],out=out[1]);
And(a=a[2], b=b[2],out=out[2]);
And(a=a[3], b=b[3],out=out[3]);
And(a=a[4], b=b[4],out=out[4]);
```

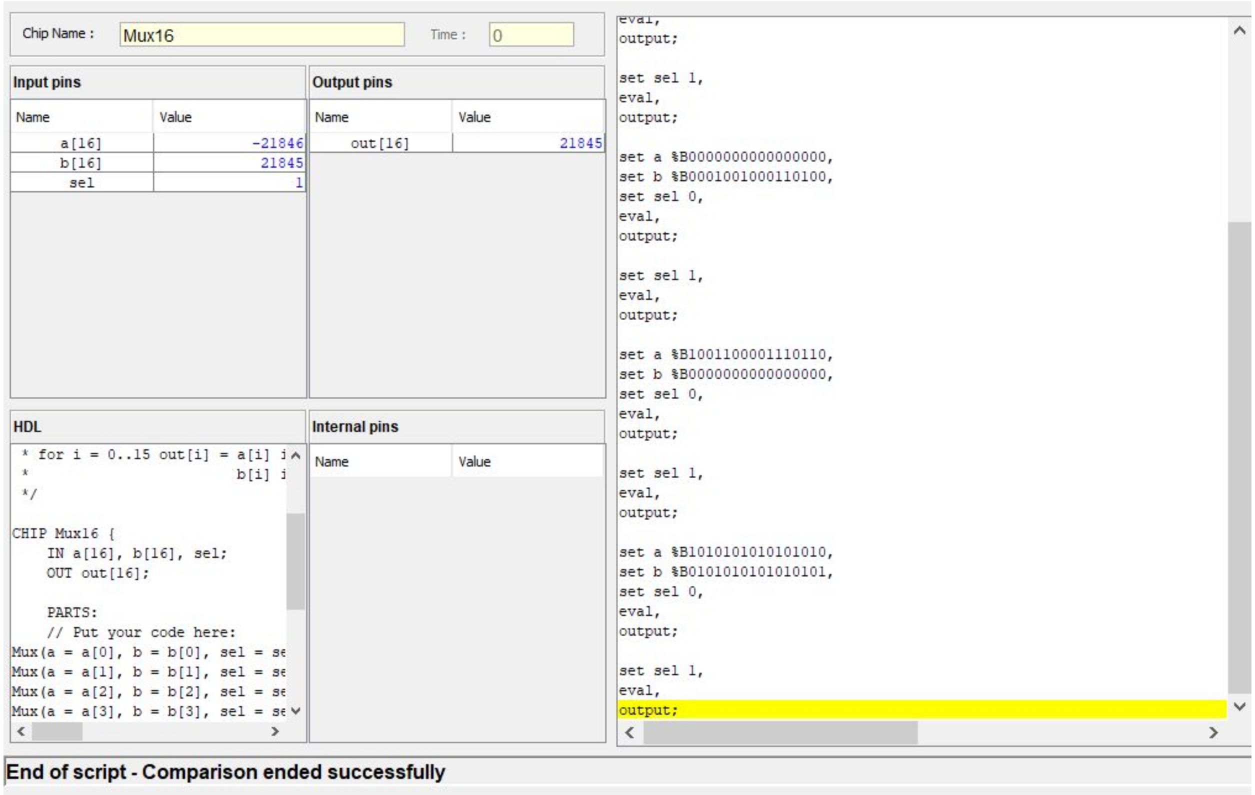
Internal pins

Name	Value

```
// File name: projects/01/And16.tst

load And16.hdl,
output-file And16.out,
compare-to And16.cmp,
output-list a%B1.16.1 b%B1.16.1 out%B1.16.1;

set a %B0000000000000000,
set b %B0000000000000000,
eval,
output;

set a %B0000000000000000,
set b %B1111111111111111,
eval,
output;

set a %B1111111111111111,
set b %B1111111111111111,
eval,
output;

set a %B1010101010101010,
set b %B0101010101010101,
eval,
output;

set a %B0011110011000011,
set b %B0000111111110000,
eval,
output;

set a %B0001001000110100,
set b %B1001100001110110,
eval,
output;
```

End of script - Comparison ended successfully

Mux16



Chip Name : Mux16 Time : 0

Input pins

Name	Value
a[16]	-21846
b[16]	21845
sel	1

Output pins

Name	Value
out[16]	21845

HDL

```
 * for i = 0..15 out[i] = a[i] i
 *                         b[i] i
 */

CHIP Mux16 {
    IN a[16], b[16], sel;
    OUT out[16];

    PARTS:
    // Put your code here:
Mux(a = a[0], b = b[0], sel = se
Mux(a = a[1], b = b[1], sel = se
Mux(a = a[2], b = b[2], sel = se
Mux(a = a[3], b = b[3], sel = se
```

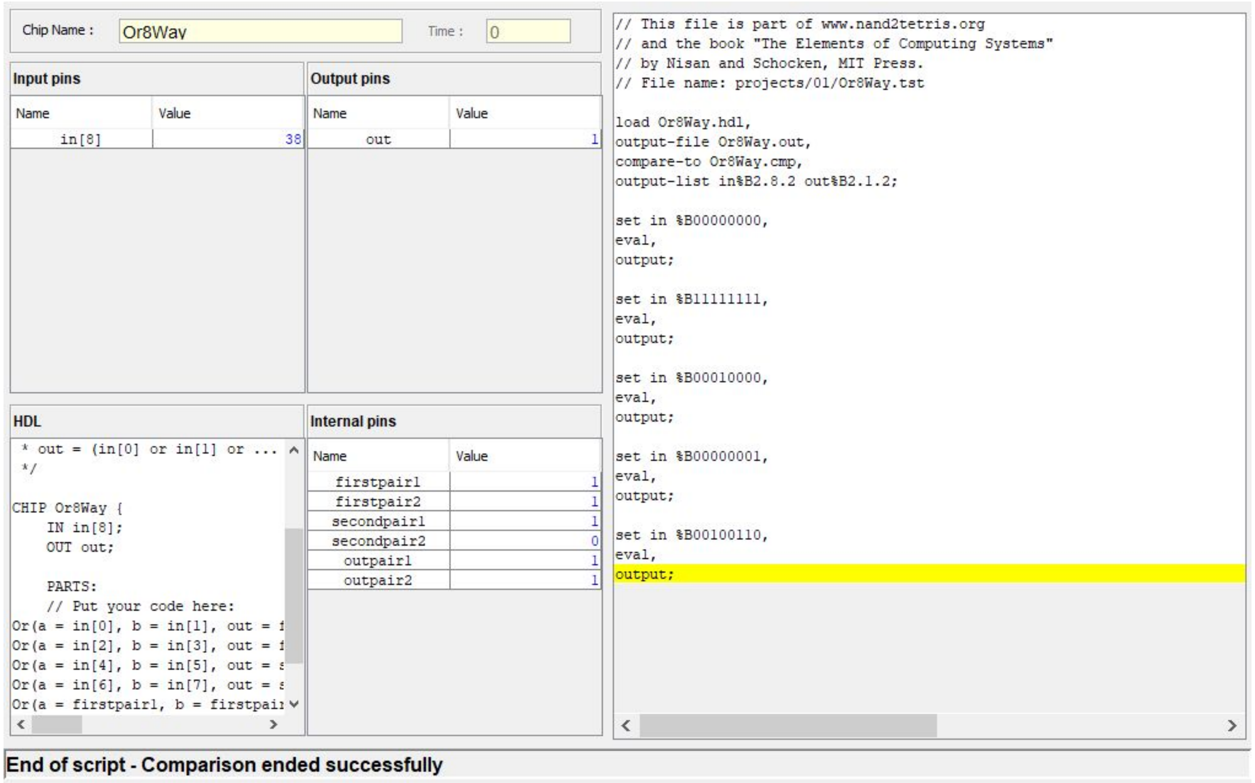
Internal pins

Name	Value

```
eval,
output;

set sel 1,
eval,
output;

set a %B0000000000000000,
set b %B0001001000110100,
set sel 0,
eval,
output;

set sel 1,
eval,
output;

set a %B1001100001110110,
set b %B0000000000000000,
set sel 0,
eval,
output;

set sel 1,
eval,
output;

set a %B1010101010101010,
set b %B0101010101010101,
set sel 0,
eval,
output;

set sel 1,
eval,
output;
```

End of script - Comparison ended successfully

Or8Way



Chip Name : Or8Way Time : 0

Input pins

Name	Value
in[8]	38

Output pins

Name	Value
out	1

HDL

```
 * out = (in[0] or in[1] or ...
 */

CHIP Or8Way {
    IN in[8];
    OUT out;

    PARTS:
    // Put your code here:
Or(a = in[0], b = in[1], out = f
Or(a = in[2], b = in[3], out = f
Or(a = in[4], b = in[5], out = s
Or(a = in[6], b = in[7], out = s
Or(a = firstpair1, b = firstpai
```

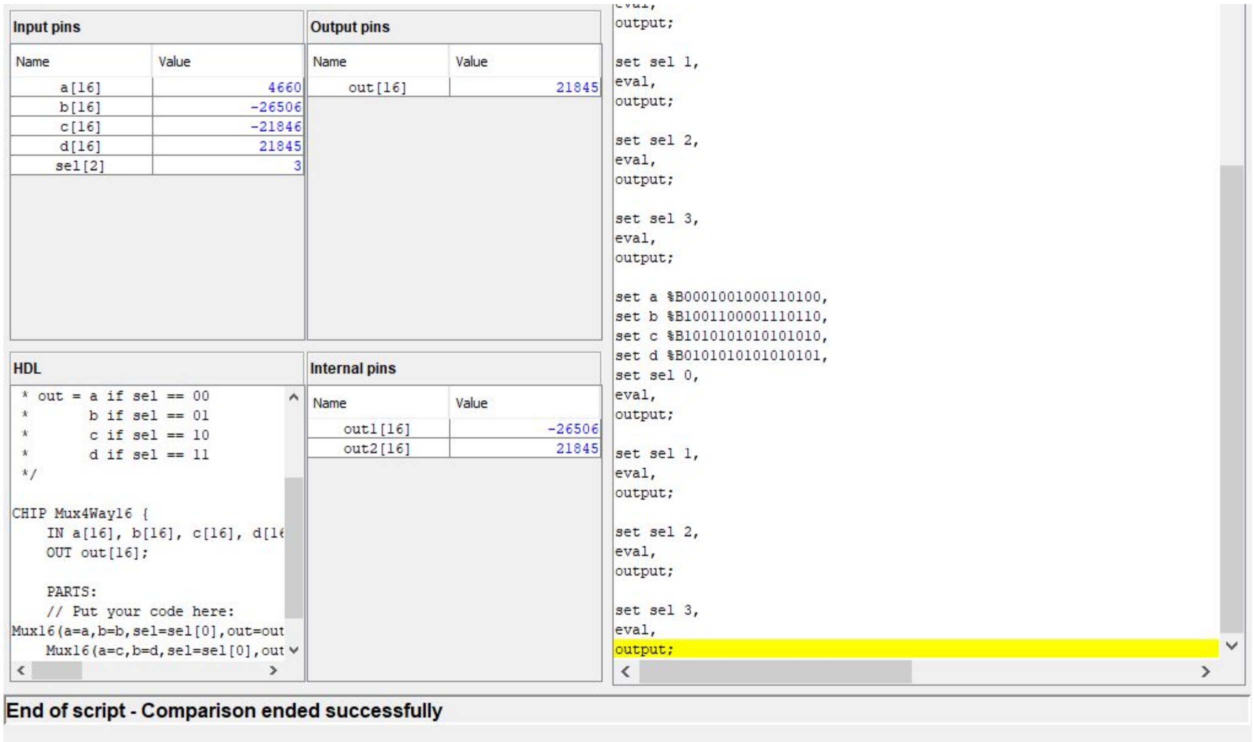
Internal pins

Name	Value
firstpair1	1
firstpair2	1
secondpair1	1
secondpair2	0
outpair1	1
outpair2	1

```
// This file is part of www.nand2tetris.org
// and the book "The Elements of Computing Systems"
// by Nisan and Schocken, MIT Press.
// File name: projects/01/Or8Way.tst

load Or8Way.hdl,
output-file Or8Way.out,
compare-to Or8Way.cmp,
output-list in%B2.8.2 out%B2.1.2;

set in %B00000000,
eval,
output;

set in %B11111111,
eval,
output;

set in %B00010000,
eval,
output;

set in %B00000001,
eval,
output;

set in %B00100110,
eval,
output;
```

End of script - Comparison ended successfully

Mux4Way16



Name	Value
a[16]	4660
b[16]	-26506
c[16]	-21846
d[16]	21845
sel[2]	3

Name	Value
out[16]	21845

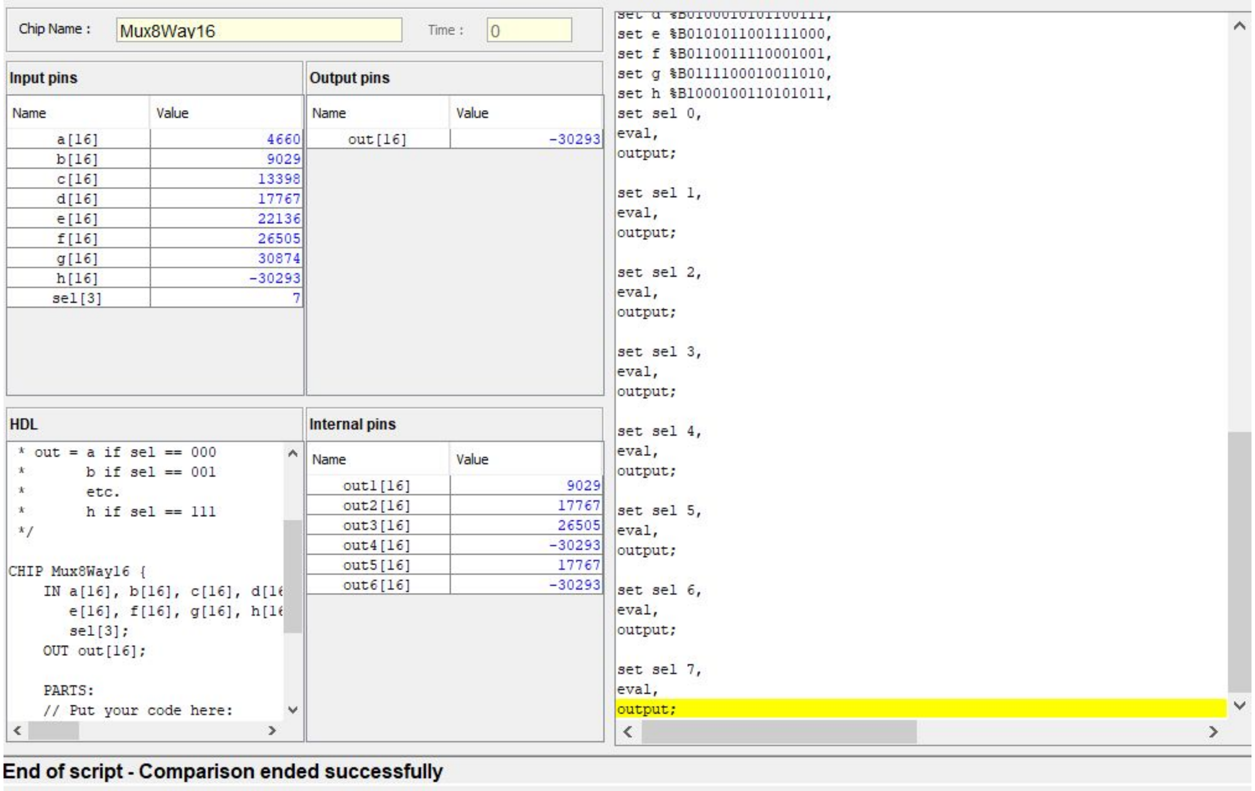
```
 * out = a if sel == 00
 *       b if sel == 01
 *       c if sel == 10
 *       d if sel == 11
 */

CHIP Mux4Way16 {
    IN a[16], b[16], c[16], d[16
    OUT out[16];

    PARTS:
    // Put your code here:
Mux16(a=a,b=b,sel=sel[0],out=out
    Mux16(a=c,b=d,sel=sel[0],out
```

Name	Value
out1[16]	-26506
out2[16]	21845

```
eval,
output;

set sel 1,
eval,
output;

set sel 2,
eval,
output;

set sel 3,
eval,
output;

set a %B0001001000110100,
set b %B1001100001110110,
set c %B1010101010101010,
set d %B0101010101010101,
set sel 0,
eval,
output;

set sel 1,
eval,
output;

set sel 2,
eval,
output;

set sel 3,
eval,
output;
```

End of script - Comparison ended successfully

Mux8Way16



Chip Name : Mux8Way16   Time : 0

Input pins

Name	Value
a[16]	4660
b[16]	9029
c[16]	13398
d[16]	17767
e[16]	22136
f[16]	26505
g[16]	30874
h[16]	-30293
sel[3]	7

Output pins

Name	Value
out[16]	-30293

HDL

```
 * out = a if sel == 000
 *       b if sel == 001
 *       etc.
 *       h if sel == 111
 */

CHIP Mux8Way16 {
    IN a[16], b[16], c[16], d[1
       e[16], f[16], g[16], h[1
       sel[3];
    OUT out[16];

    PARTS:
    // Put your code here:
```

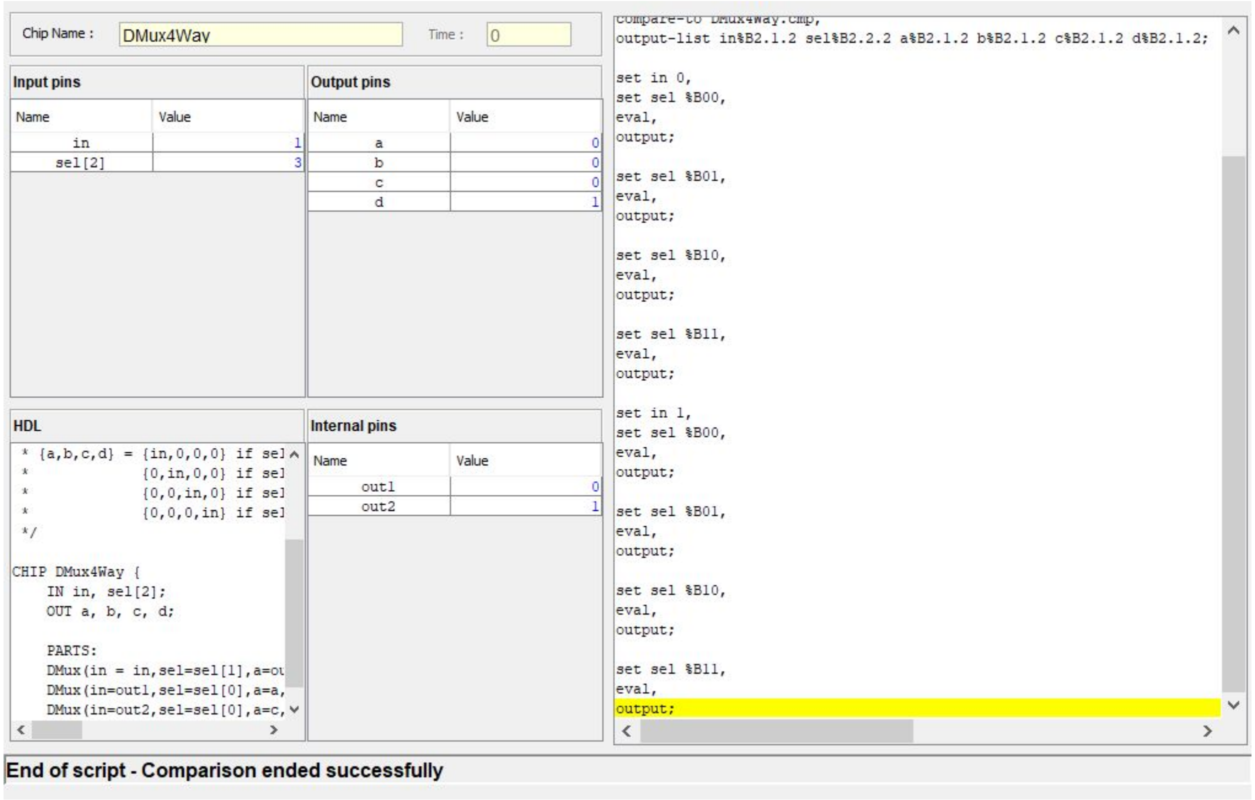
Internal pins

Name	Value
out1[16]	9029
out2[16]	17767
out3[16]	26505
out4[16]	-30293
out5[16]	17767
out6[16]	-30293

```
set d %B0100010101100111,
set e %B0101011001111000,
set f %B0110011110001001,
set g %B0111100010011010,
set h %B1000100110101011,
set sel 0,
eval,
output;

set sel 1,
eval,
output;

set sel 2,
eval,
output;

set sel 3,
eval,
output;

set sel 4,
eval,
output;

set sel 5,
eval,
output;

set sel 6,
eval,
output;

set sel 7,
eval,
output;
```

End of script - Comparison ended successfully

DMux4Way



Chip Name : DMux4Way

Time : 0

Input pins

Name	Value
in	1
sel[2]	3

Output pins

Name	Value
a	0
b	0
c	0
d	1

HDL

```
 * {a,b,c,d} = {in,0,0,0} if sel
 *             {0,in,0,0} if sel
 *             {0,0,in,0} if sel
 *             {0,0,0,in} if sel
 */

CHIP DMux4Way {
    IN in, sel[2];
    OUT a, b, c, d;

    PARTS:
    DMux(in = in,sel=sel[1],a=ou
    DMux(in=out1,sel=sel[0],a=a,
    DMux(in=out2,sel=sel[0],a=c,
```

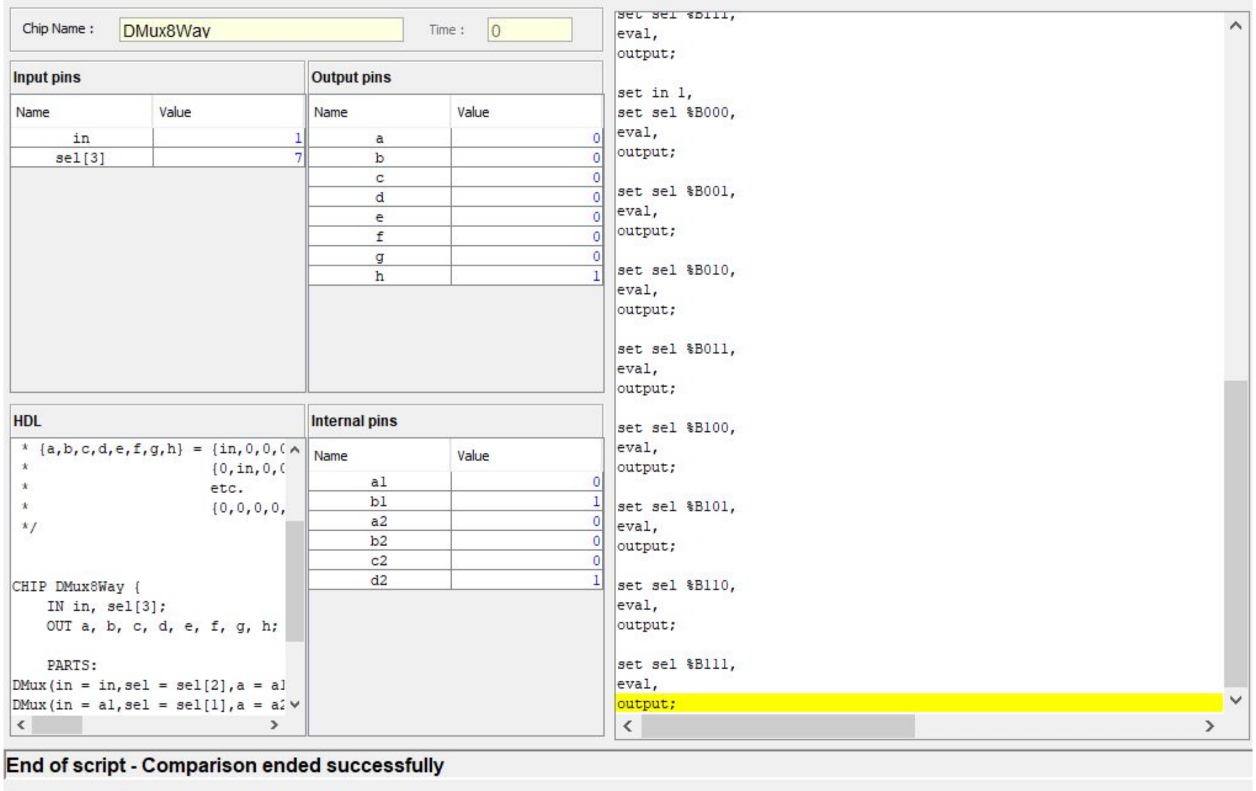
Internal pins

Name	Value
out1	0
out2	1

```
compare-to DMux4Way.cmp,
output-list in%B2.1.2 sel%B2.2.2 a%B2.1.2 b%B2.1.2 c%B2.1.2 d%B2.1.2;

set in 0,
set sel %B00,
eval,
output;

set sel %B01,
eval,
output;

set sel %B10,
eval,
output;

set sel %B11,
eval,
output;

set in 1,
set sel %B00,
eval,
output;

set sel %B01,
eval,
output;

set sel %B10,
eval,
output;

set sel %B11,
eval,
output;
```

End of script - Comparison ended successfully

DMux8Way



Chip Name : DMux8Way

Time : 0

Input pins

Name	Value
in	1
sel[3]	7

Output pins

Name	Value
a	0
b	0
c	0
d	0
e	0
f	0
g	0
h	1

HDL

```
 * {a,b,c,d,e,f,g,h} = {in,0,0,(
 *                      {0,in,0,(
 *                     etc.
 *                      {0,0,0,0,
 */


CHIP DMux8Way {
    IN in, sel[3];
    OUT a, b, c, d, e, f, g, h;

    PARTS:
DMux(in = in,sel = sel[2],a = a1
DMux(in = a1,sel = sel[1],a = a2
```

Internal pins

Name	Value
a1	0
b1	1
a2	0
b2	0
c2	0
d2	1

```
set sel %B111,
eval,
output;

set in 1,
set sel %B000,
eval,
output;

set sel %B001,
eval,
output;

set sel %B010,
eval,
output;

set sel %B011,
eval,
output;

set sel %B100,
eval,
output;

set sel %B101,
eval,
output;

set sel %B110,
eval,
output;

set sel %B111,
eval,
output;
```

End of script - Comparison ended successfully