



Cal Poly
Pomona

Advanced Computer Architecture (CS 5250.01)

Project # 1

Talha Ahmed

018960110

Submitted to: Professor Nima Davarpanah

Dated: Friday, 30 January 2026

Table of Contents

Solution	4
NOT Gate	4
Code	4
Result	4
AND Gate	4
Code	4
Result	5
OR Gate	5
Code	5
Result	6
XOR Gate	6
Code	6
Result	7
MUX Gate	7
Code	7
Result	8
DMUX Gate	8
Code	8
Result	9
NOT16 Gate	9
Code	9
Result	10
AND16 Gate	10
Code	10
Result	11
OR16 Gate	11
Code	11
Result	12
MUX16 Gate	13
Code	13
Result	14
OR8WAY Gate	14
Code	14
Result	15
MUX4WAY16 Gate	15
Code	15
Result	16
MUX8WAY16 Gate	16
Code	16

Result	17
DMUX4WAY Gate	17
Code	17
Result	18
DMUX8WAY Gate	18
Code	18
Result	19

Solution

NOT Gate

Code

```
CHIP Not {  
    IN in;  
    OUT out;  
  
    PARTS:  
    Nand(a=in, b=in, out=out);  
}
```

Result

The screenshot shows the NAND2Tetris Hardware Simulator interface. The top bar displays "NAND2Tetris / Hardware Simulator" and various icons. Below the bar, the "HDL" tab is active, showing the code for the "Not" chip. The code is as follows:

```
1 CHIP Not {  
2     IN in;  
3     OUT out;  
4  
5     PARTS:  
6     Nand(a=in, b=in, out=out);  
7 }
```

To the right of the code editor, the "Chip Not" panel shows the input and output pins. The "Input pins" section shows "in" with a value of 1. The "Output pins" section shows "out" with a value of 0. Below the code editor, the "Test" tab is active, showing a test script with the following table:

in	out
0	1
1	0

At the bottom of the simulator, a green status bar indicates: "Simulation successful: The output file is identical to the compare file".

AND Gate

Code

```
CHIP And {  
    IN a, b;  
    OUT out;
```


```

PARTS:
  Nand(a=a, b=b, out=w1);
  Nand(a=w1, b=w1, out=out);
}

```

Result

NAND2Tetris / Hardware Simulator

HDL ☒ Built-in Project 1 And  Chip And Eval Reset Clock: 0

```





1 CHIP And {
2   IN a, b;
3   OUT out;
4
5   PARTS:
6     Nand(a=a, b=b, out=w1);
7     Nand(a=w1, b=w1, out=out);
8 }

```

Input pins
a 1
b 1

Output pins
out 1

Internal pins
w1 0

Test     Slow Fast

Test Script Compare File Output File Diff Table

```

1 | a | b | out |
2 | 0 | 0 | 0 |
3 | 0 | 1 | 0 |
4 | 1 | 0 | 0 |
5 | 1 | 1 | 1 |

```

Simulation successful: The output file is identical to the compare file

OR Gate

Code

```

CHIP Or {
  IN a, b;
  OUT out;

  PARTS:
    Nand(a=a, b=a, out=notA);
    Nand(a=b, b=b, out=notB);
    Nand(a=notA, b=notB, out=out);
}

```

Result

NAND2Tetris / Hardware Simulator

HDL Built-in Project 1 Or

```
1 CHIP Or {
2   IN a, b;
3   OUT out;
4
5   PARTS:
6     Nand(a=a, b=a, out=notA);
7     Nand(a=b, b=b, out=notB);
8     Nand(a=notA, b=notB, out=out);
9 }
```

Chip Or Eval Reset Clock: 0

Input pins

a 1

b 1

Output pins

out 1

Internal pins

notA 0

notB 0

Test Slow Fast

Test Script Compare File Output File Diff Table

```
1 | a | b | out |
2 | 0 | 0 | 0 |
3 | 0 | 1 | 1 |
4 | 1 | 0 | 1 |
5 | 1 | 1 | 1 |
```

Simulation successful: The output file is identical to the compare file

XOR Gate

Code

```
CHIP Xor {
  IN a, b;
  OUT out;

  PARTS:
    Not(in=a, out=notA);
    Not(in=b, out=notB);
    And(a=notA, b=b, out=w1);
    And(a=a, b=notB, out=w2);
    Or(a=w1, b=w2, out=out);
}
```

Result

NAND2Tetris / Hardware Simulator

HDL Built-in Project 1 Xor

```
1 CHIP Xor {
2   IN a, b;
3   OUT out;
4
5   PARTS:
6     Not(in=a, out=notA);
7     Not(in=b, out=notB);
8     And(a=notA, b=b, out=w1);
9     And(a=a, b=notB, out=w2);
10    Or(a=w1, b=w2, out=out);
11 }
```

Chip Xor Eval Reset Clock: 0

Input pins

a 1

b 1

Output pins

out 0

Internal pins

notA 0

notB 0

w1 0

w2 0

Test Slow Fast

Test Script Compare File Output File Diff Table

```
1 | a | b | out |
2 | 0 | 0 | 0 |
3 | 0 | 1 | 1 |
4 | 1 | 0 | 1 |
5 | 1 | 1 | 0 |
```

Simulation successful: The output file is identical to the compare file

MUX Gate

Code

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

  PARTS:
    And(a=b, b=sel, out=andBSel);
    Not(in=sel, out=notSel);
    And(a=a, b=notSel, out=andASel);
    Or(a=andASel, b=andBSel, out=out);
}
```

Result

NAND2Tetris / Hardware Simulator

HDL Built-in Project 1 Mux

```
1 CHIP Mux {
2   IN a, b, sel;
3   OUT out;
4
5   PARTS:
6     And(a=b, b=sel, out=andBSel);
7     Not(in=sel, out=notSel);
8     And(a=a, b=notSel, out=andASel);
9     Or(a=andASel, b=andBSel, out=out);
10 }
```

Chip Mux Eval Reset Clock: 0

Input pins

a 1

b 1

sel 1

Output pins

out 1

Internal pins

andBSel 1

notSel 0

andASel 0

Test Slow Fast

Test Script Compare File Output File Diff Table

```
1 | a | b | sel | out |
2 | 0 | 0 | 0 | 0 |
3 | 0 | 0 | 1 | 0 |
4 | 0 | 1 | 0 | 0 |
5 | 0 | 1 | 1 | 1 |
6 | 1 | 0 | 0 | 1 |
7 | 1 | 0 | 1 | 0 |
8 | 1 | 1 | 0 | 1 |
9 | 1 | 1 | 1 | 1 |
```

Simulation successful: The output file is identical to the compare file

DMUX Gate

Code

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

  PARTS:
    Not(in=sel, out=notSel);
    And(a=in, b=notSel, out=a);
    And(a=in, b=sel, out=b);
}
```


Result

NAND2Tetris / Hardware Simulator

HDL Built-in Project 1 DMux

```
1 CHIP DMux {
2   IN in, sel;
3   OUT a, b;
4
5   PARTS:
6     Not(in=sel, out=notSel);
7     And(a=in, b=notSel, out=a);
8     And(a=in, b=sel, out=b);
9 }
```

Chip DMux Eval Reset Clock: 0

Input pins
in 1
sel 1

Output pins
a 0
b 1

Internal pins
notSel 0

Test Slow Fast

Test Script Compare File Output File Diff Table

1	in sel a b
2	0 0 0 0
3	0 1 0 0
4	1 0 1 0
5	1 1 0 1

Simulation successful: The output file is identical to the compare file

NOT16 Gate

Code

```
CHIP Not16 {
  IN in[16];
  OUT out[16];

  PARTS:
    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]);
    Not(in=in[5], out=out[5]);
    Not(in=in[6], out=out[6]);
    Not(in=in[7], out=out[7]);
    Not(in=in[8], out=out[8]);
    Not(in=in[9], out=out[9]);
    Not(in=in[10], out=out[10]);
    Not(in=in[11], out=out[11]);
```

```

    Not(in=in[12], out=out[12]);
    Not(in=in[13], out=out[13]);
    Not(in=in[14], out=out[14]);
    Not(in=in[15], out=out[15]);
}

```

Result

NAND2Tetris / Hardware Simulator

HDL ☒ Builtin Project 1 Not16

Chip Not16 Eval Reset Clock: 0

```

1 CHIP Not16 {
2   IN in[16];
3   OUT out[16];
4
5   PARTS:
6     Not(in=in[0], out=out[0]);
7     Not(in=in[1], out=out[1]);
8     Not(in=in[2], out=out[2]);
9     Not(in=in[3], out=out[3]);
10    Not(in=in[4], out=out[4]);
11    Not(in=in[5], out=out[5]);
12    Not(in=in[6], out=out[6]);
13    Not(in=in[7], out=out[7]);
14    Not(in=in[8], out=out[8]);
15    Not(in=in[9], out=out[9]);
16    Not(in=in[10], out=out[10]);

```

Input pins
in 0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 0 dec

Output pins
out 1 1 1 0 1 1 0 1 1 1 0 0 1 0 1 1 dec

Test Slow Fast

Test Script	Compare File	Output File	Diff Table
1	in	out	
2	0000000000000000	1111111111111111	
3	1111111111111111	0000000000000000	
4	1010101010101010	0101010101010101	
5	0011110011000011	1100001100111100	
6	0001001000110100	1110110111001011	

Simulation successful: The output file is identical to the compare file

AND16 Gate

Code

```

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

    PARTS:
        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]);
        And(a=a[5], b=b[5], out=out[5]);

```

```

And(a=a[6], b=b[6], out=out[6]);
And(a=a[7], b=b[7], out=out[7]);
And(a=a[8], b=b[8], out=out[8]);
And(a=a[9], b=b[9], out=out[9]);
And(a=a[10], b=b[10], out=out[10]);
And(a=a[11], b=b[11], out=out[11]);
And(a=a[12], b=b[12], out=out[12]);
And(a=a[13], b=b[13], out=out[13]);
And(a=a[14], b=b[14], out=out[14]);
And(a=a[15], b=b[15], out=out[15]);
}

```

Result

NAND2Tetris / Hardware Simulator

HDL ☒ Builtin Project 1 And16

Chip And16 Eval Reset Clock: 0

```

1 CHIP And16 {
2   IN a[16], b[16];
3   OUT out[16];
4
5   PARTS:
6     And(a=a[0], b=b[0], out=out[0]);
7     And(a=a[1], b=b[1], out=out[1]);
8     And(a=a[2], b=b[2], out=out[2]);
9     And(a=a[3], b=b[3], out=out[3]);
10    And(a=a[4], b=b[4], out=out[4]);
11    And(a=a[5], b=b[5], out=out[5]);
12    And(a=a[6], b=b[6], out=out[6]);
13    And(a=a[7], b=b[7], out=out[7]);
14    And(a=a[8], b=b[8], out=out[8]);
15    And(a=a[9], b=b[9], out=out[9]);
16    And(a=a[10], b=b[10], out=out[10]);

```

Input pins

a 0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 0 dec

b 1 0 0 1 1 0 0 0 0 0 1 1 1 0 1 1 0 dec

Output pins

out 0 0 0 1 0 0 0 0 0 0 0 1 1 0 1 0 0 dec

Test Slow Fast

Test Script	Compare File	Output File	Diff Table
1	a	b	out
2	0000000000000000	0000000000000000	0000000000000000
3	0000000000000000	1111111111111111	0000000000000000
4	1111111111111111	1111111111111111	1111111111111111
5	1010101010101010	0101010101010101	0000000000000000
6	0011110011000011	0000111111110000	0000110011000000
7	0001001000110100	100110000110110	000100000110100

Simulation successful: The output file is identical to the compare file

OR16 Gate

Code

```

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

  PARTS:

```

```

Or(a=a[0], b=b[0], out=out[0]);
Or(a=a[1], b=b[1], out=out[1]);
Or(a=a[2], b=b[2], out=out[2]);
Or(a=a[3], b=b[3], out=out[3]);
Or(a=a[4], b=b[4], out=out[4]);
Or(a=a[5], b=b[5], out=out[5]);
Or(a=a[6], b=b[6], out=out[6]);
Or(a=a[7], b=b[7], out=out[7]);
Or(a=a[8], b=b[8], out=out[8]);
Or(a=a[9], b=b[9], out=out[9]);
Or(a=a[10], b=b[10], out=out[10]);
Or(a=a[11], b=b[11], out=out[11]);
Or(a=a[12], b=b[12], out=out[12]);
Or(a=a[13], b=b[13], out=out[13]);
Or(a=a[14], b=b[14], out=out[14]);
Or(a=a[15], b=b[15], out=out[15]);
}

```

Result

NAND2Tetris / Hardware Simulator

HDL Built-in Project 1 Or16 Chip Or16 Eval Reset Clock: 0

```

1 CHIP Or16 {
2   IN a[16], b[16];
3   OUT out[16];
4
5   PARTS:
6   Or(a=a[0], b=b[0], out=out[0]);
7   Or(a=a[1], b=b[1], out=out[1]);
8   Or(a=a[2], b=b[2], out=out[2]);
9   Or(a=a[3], b=b[3], out=out[3]);
10  Or(a=a[4], b=b[4], out=out[4]);
11  Or(a=a[5], b=b[5], out=out[5]);
12  Or(a=a[6], b=b[6], out=out[6]);
13  Or(a=a[7], b=b[7], out=out[7]);
14  Or(a=a[8], b=b[8], out=out[8]);
15  Or(a=a[9], b=b[9], out=out[9]);
16  Or(a=a[10], b=b[10], out=out[10]);

```

Input pins

a 0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 0 dec

b 1 0 0 1 1 0 0 0 0 1 1 1 0 1 1 0 dec

Output pins

out 1 0 0 1 1 0 1 0 0 1 1 1 0 1 1 0 dec

Test Slow Fast

Test Script	Compare File	Output File	Diff Table
1	a	b	out
2	0000000000000000	0000000000000000	0000000000000000
3	0000000000000000	1111111111111111	1111111111111111
4	1111111111111111	1111111111111111	1111111111111111
5	1010101010101010	0101010101010101	1111111111111111
6	0011110011000011	0000111111110000	0011111111110011
7	0001001000110100	100110000110110	1001101001110110

Simulation successful: The output file is identical to the compare file

MUX16 Gate

Code

```
CHIP Mux16 {  
    IN a[16], b[16], sel;  
    OUT out[16];  
  
    PARTS:  
    Mux(a=a[0], b=b[0], sel=sel, out=out[0]);  
    Mux(a=a[1], b=b[1], sel=sel, out=out[1]);  
    Mux(a=a[2], b=b[2], sel=sel, out=out[2]);  
    Mux(a=a[3], b=b[3], sel=sel, out=out[3]);  
    Mux(a=a[4], b=b[4], sel=sel, out=out[4]);  
    Mux(a=a[5], b=b[5], sel=sel, out=out[5]);  
    Mux(a=a[6], b=b[6], sel=sel, out=out[6]);  
    Mux(a=a[7], b=b[7], sel=sel, out=out[7]);  
    Mux(a=a[8], b=b[8], sel=sel, out=out[8]);  
    Mux(a=a[9], b=b[9], sel=sel, out=out[9]);  
    Mux(a=a[10], b=b[10], sel=sel, out=out[10]);  
    Mux(a=a[11], b=b[11], sel=sel, out=out[11]);  
    Mux(a=a[12], b=b[12], sel=sel, out=out[12]);  
    Mux(a=a[13], b=b[13], sel=sel, out=out[13]);  
    Mux(a=a[14], b=b[14], sel=sel, out=out[14]);  
    Mux(a=a[15], b=b[15], sel=sel, out=out[15]);  
}
```

Result

NAND2Tetris / Hardware Simulator

HDL Built-in Project 1 Mux16

```
1 CHIP Mux16 {
2   IN a[16], b[16], sel;
3   OUT out[16];
4
5   PARTS:
6     Mux(a=a[0], b=b[0], sel=sel, out=out[0]);
7     Mux(a=a[1], b=b[1], sel=sel, out=out[1]);
8     Mux(a=a[2], b=b[2], sel=sel, out=out[2]);
9     Mux(a=a[3], b=b[3], sel=sel, out=out[3]);
10    Mux(a=a[4], b=b[4], sel=sel, out=out[4]);
11    Mux(a=a[5], b=b[5], sel=sel, out=out[5]);
12    Mux(a=a[6], b=b[6], sel=sel, out=out[6]);
13    Mux(a=a[7], b=b[7], sel=sel, out=out[7]);
14    Mux(a=a[8], b=b[8], sel=sel, out=out[8]);
15    Mux(a=a[9], b=b[9], sel=sel, out=out[9]);
16    Mux(a=a[10], b=b[10], sel=sel, out=out[10]);
17  }
```

Chip Mux16 Eval Reset Clock: 0

Input pins

a 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 dec

b 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 dec

sel 1

Output pins

out 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 dec

Test Slow Fast

Test Script	Compare File	Output File	Diff Table
1	a	b	sel out
2	0000000000000000	0000000000000000	0 0000000000000000
3	0000000000000000	0000000000000000	1 0000000000000000
4	0000000000000000	0001001000110100	0 0000000000000000
5	0000000000000000	0001001000110100	1 0001001000110100
6	1001100001110110	0000000000000000	0 1001100001110110
7	1001100001110110	0000000000000000	1 0000000000000000
8	1010101010101010	0101010101010101	0 1010101010101010
9	1010101010101010	0101010101010101	1 0101010101010101

Simulation successful: The output file is identical to the compare file

OR8WAY Gate

Code

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

  PARTS:
    Or(a=in[0], b=in[1], out=w1);
    Or(a=w1, b=in[2], out=w2);
    Or(a=w2, b=in[3], out=w3);
    Or(a=w3, b=in[4], out=w4);
    Or(a=w4, b=in[5], out=w5);
    Or(a=w5, b=in[6], out=w6);
    Or(a=w6, b=in[7], out=out);
}
```

Result

NAND2Tetris / Hardware Simulator

HDL ☒ Builtin Project 1 Or8Way

```
1 CHIP Or8Way {
2   IN in[8];
3   OUT out;
4
5   PARTS:
6     Or(a=in[0], b=in[1], out=w1);
7     Or(a=w1, b=in[2], out=w2);
8     Or(a=w2, b=in[3], out=w3);
9     Or(a=w3, b=in[4], out=w4);
10    Or(a=w4, b=in[5], out=w5);
11    Or(a=w5, b=in[6], out=w6);
12    Or(a=w6, b=in[7], out=out);
13 }
```

Chip Or8Way Eval Reset Clock: 0

Input pins
in 0 0 1 0 0 1 1 0 dec

Output pins
out 1

Internal pins
w1 1
w2 1
w3 1
w4 1
w5 1

Test ☒ ☐ ☐ Slow Fast

Test Script Compare File Output File Diff Table

	in	out
1	00000000	0
2	11111111	1
3	00010000	1
4	00000001	1
5	00100110	1

Simulation successful: The output file is identical to the compare file

MUX4WAY16 Gate

Code

```
CHIP Mux4Way16 {
  IN a[16], b[16], c[16], d[16], sel[2];
  OUT out[16];

  PARTS:
    Mux16(a=a, b=b, sel=sel[0], out=muxAB);
    Mux16(a=c, b=d, sel=sel[0], out=muxCD);
    Mux16(a=muxAB, b=muxCD, sel=sel[1], out=out);
}
```

Result

NAND2Tetris / Hardware Simulator

HDL ☒ Builtin Project 1 Mux4Way16

```
1 CHIP Mux4Way16 {
2   IN a[16], b[16], c[16], d[16], sel[2];
3   OUT out[16];
4
5   PARTS:
6     Mux16(a=a, b=b, sel=sel[0], out=muxAB);
7     Mux16(a=c, b=d, sel=sel[0], out=muxCD);
8     Mux16(a=muxAB, b=muxCD, sel=sel[1], out=out);
9 }
```

Chip Mux4Way16 Eval Reset Clock: 0

Input pins

a 0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 0 dec

b 1 0 0 1 1 0 0 0 0 0 1 1 1 0 1 0 dec

c 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 dec

d 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 dec

sel 1 1 dec

Output pins

out 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 dec

Internal pins

muxAB 1 0 0 1 1 0 0 0 0 0 1 1 1 0 1 0 dec

Test ☒ ☐ ☐ ☐ Slow Fast

Test Script	Compare File	Output File	Diff Table			
1	a	b	c	d	sel	out
2	0000000000000000	0000000000000000	0000000000000000	0000000000000000	00	0000000000000000
3	0000000000000000	0000000000000000	0000000000000000	0000000000000000	01	0000000000000000
4	0000000000000000	0000000000000000	0000000000000000	0000000000000000	10	0000000000000000
5	0000000000000000	0000000000000000	0000000000000000	0000000000000000	11	0000000000000000
6	0001001000110100	1001100001110110	1010101010101010	0101010101010101	00	0001001000110100
7	0001001000110100	1001100001110110	1010101010101010	0101010101010101	01	1001100001110110
8	0001001000110100	1001100001110110	1010101010101010	0101010101010101	10	1010101010101010
9	0001001000110100	1001100001110110	1010101010101010	0101010101010101	11	0101010101010101

Simulation successful: The output file is identical to the compare file

MUX8WAY16 Gate

Code

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

  PARTS:
    Mux4Way16(a=a, b=b, c=c, d=d, sel=sel[0..1], out=w1);
    Mux4Way16(a=e, b=f, c=g, d=h, sel=sel[0..1], out=w2);
    Mux16(a=w1, b=w2, sel=sel[2], out=out);
}
```


Result

NAND2Tetris / Hardware Simulator

HDL ☒ Builtin Project 1 Mux8Way16

```

1 CHIP Mux8Way16 {
2   IN a[16], b[16], c[16], d[16],
3     e[16], f[16], g[16], h[16],
4     sel[3];
5   OUT out[16];
6
7   PARTS:
8     Mux4Way16(a=a, b=b, c=c, d=d, sel=sel[0..1], out=w1);
9     Mux4Way16(a=e, b=f, c=g, d=h, sel=sel[0..1], out=w2);
10    Mux16(a=w1, b=w2, sel=sel[2], out=out);
11 }

```

Chip Mux8Way16 Eval Reset Clock: 0

Input pins

a	0 0 0 1 0 0 1 0 0 0 1 1 0 1 0 0	dec
b	0 0 1 0 0 0 1 1 0 1 0 0 0 1 0 1	dec
c	0 0 1 1 0 1 0 0 0 1 0 1 0 1 1 0	dec
d	0 1 0 0 0 1 0 1 0 1 1 0 0 1 1 1	dec
e	0 1 0 1 0 1 1 0 0 1 1 1 1 0 0 0	dec
f	0 1 1 0 0 1 1 1 1 0 0 0 1 0 0 1	dec
g	0 1 1 1 1 0 0 0 1 0 0 1 1 0 1 0	dec
h	1 0 0 0 1 0 0 1 1 0 1 0 1 0 1 1	dec

Test ☒ ☐ ☐ ☐ Slow Fast

Test Script	Compare File	Output File	Diff Table
1	a	b	c
2	0000000000000000	0000000000000000	0000000000000000
3	0000000000000000	0000000000000000	0000000000000000
4	0000000000000000	0000000000000000	0000000000000000
5	0000000000000000	0000000000000000	0000000000000000
6	0000000000000000	0000000000000000	0000000000000000
7	0000000000000000	0000000000000000	0000000000000000
8	0000000000000000	0000000000000000	0000000000000000
9	0000000000000000	0000000000000000	0000000000000000

Simulation successful: The output file is identical to the compare file

DMUX4WAY Gate

Code

```

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

  PARTS:
    DMux(in=in, sel=sel[1], a=outAB, b=outCD);
    DMux(in=outAB, sel=sel[0], a=a, b=b);
    DMux(in=outCD, sel=sel[0], a=c, b=d);
}

```

Result

NAND2Tetris / Hardware Simulator

HDL Built-in Project 1 DMux4Way

```
1 CHIP DMux4Way {
2   IN in, sel[2];
3   OUT a, b, c, d;
4
5   PARTS:
6     DMux(in=in, sel=sel[1], a=outAB, b=outCD);
7     DMux(in=outAB, sel=sel[0], a=a, b=b);
8     DMux(in=outCD, sel=sel[0], a=c, b=d);
9 }
```

Chip DMux4Way Eval Reset Clock: 0

Input pins
in 1
sel 1 1 dec

Output pins
a 0
b 0
c 0
d 1

Internal pins
outAB 0

Test Slow Fast

Test Script Compare File Output File Diff Table

	in	sel	a	b	c	d
1	0	00	0	0	0	0
2	0	01	0	0	0	0
3	0	10	0	0	0	0
4	0	11	0	0	0	0
5	1	00	1	0	0	0
6	1	01	0	1	0	0
7	1	10	0	0	1	0
8	1	11	0	0	0	1

Simulation successful: The output file is identical to the compare file

DMUX8WAY Gate

Code

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

  PARTS:
    DMux(in=in, sel=sel[2], a=outABCD, b=outEFGH);
    DMux4Way(in=outABCD, sel=sel[0..1], a=a, b=b, c=c, d=d);
    DMux4Way(in=outEFGH, sel=sel[0..1], a=e, b=f, c=g, d=h);
}
```

Result

The screenshot shows the HDLBits website interface. On the left, there's a dark-themed code editor with the following Verilog code:

```
1 CHIP DMux8Way {
2   IN in, sel[3];
3   OUT a, b, c, d, e, f, g, h;
4
5   PARTS:
6     DMux(in=in, sel=sel[2], a=outABCD, b=outEFGH);
7     DMux4Way(in=outABCD, sel=sel[0..1], a=a, b=b, c=c, d=d);
8     DMux4Way(in=outEFGH, sel=sel[0..1], a=e, b=f, c=g, d=h);
9 }
```

On the right, the "Chip DMux8Way" panel displays the input pins (in: 1, sel: 1 1 1 dec) and output pins (a through h, all 0). Below this, the "Test" section shows a table comparing the test script output with the compare file output.

Test Script	Compare File	Output File	Diff Table
1 in sel a b c d e f g h			
2 0 000 0 0 0 0 0 0 0 0			
3 0 001 0 0 0 0 0 0 0 0			
4 0 010 0 0 0 0 0 0 0 0			
5 0 011 0 0 0 0 0 0 0 0			
6 0 100 0 0 0 0 0 0 0 0			
7 0 101 0 0 0 0 0 0 0 0			
8 0 110 0 0 0 0 0 0 0 0			
9 0 111 0 0 0 0 0 0 0 0			

At the bottom, a green banner states: "Simulation successful: The output file is identical to the compare file".