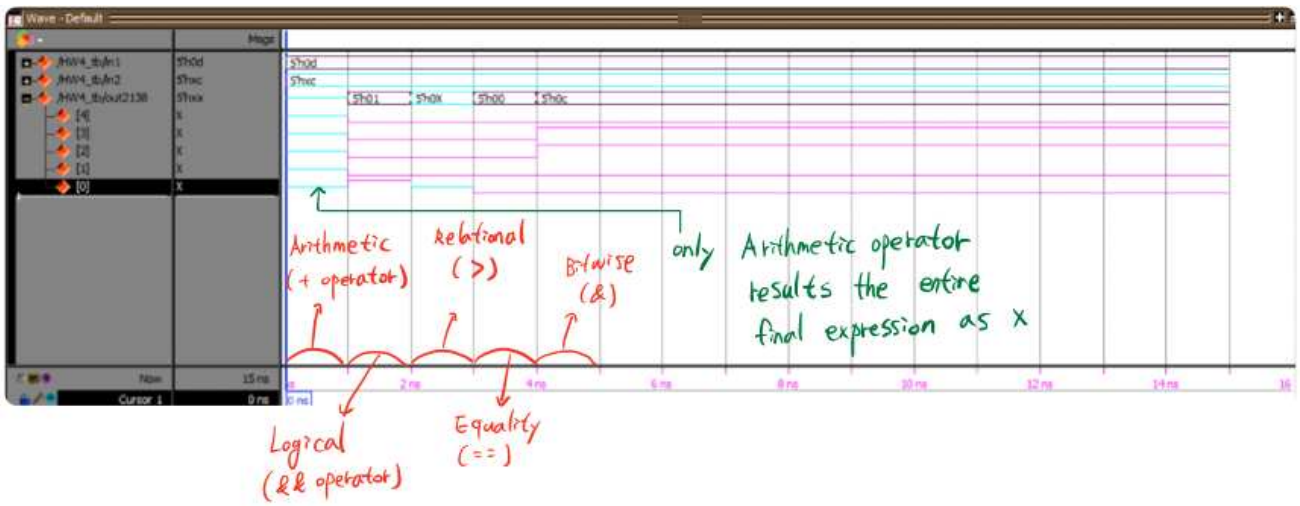
2015112138 Sejin Kwon

★number 1



★number 2

① : ln1 && ln2

ln1: xxxxx = x
&& ln2: 11110 = 1
result: 0000x

② ln1 || ln2

ln1: xxxxx ⇒x
|| ln2: 11110 ⇒1
result: 00001

③ ln1 & ln2

ln1: xxxxx
& ln2: 11110
result: xxxx0

④ ln1 | ln2

ln1: xxxxx
| ln2: 11110
result: 1111x

① : ln1 && ln2

ln1: 10110 = 1
&& ln2: 11110 = 1
00001

② ln1 || ln2

ln1: 10110 = 1
|| ln2: 11110 = 1
00001

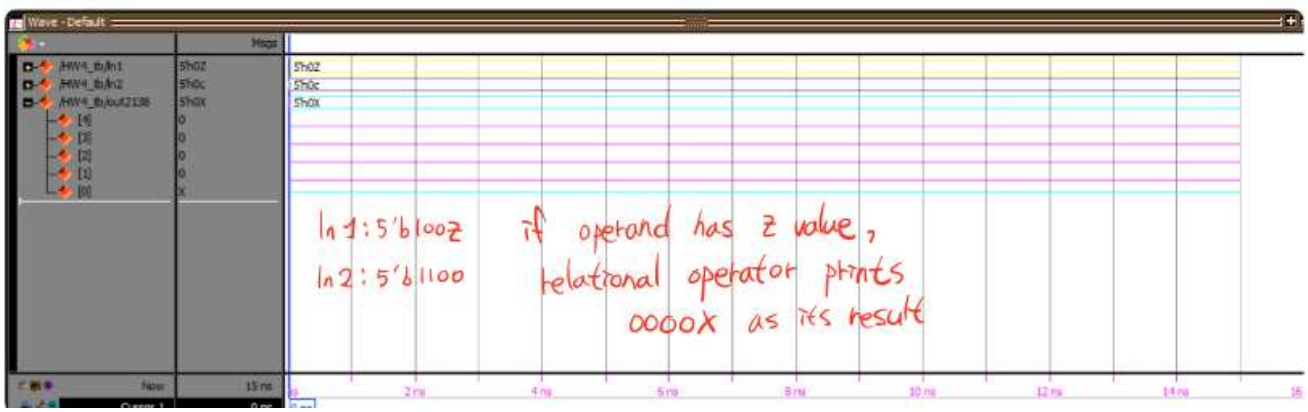
③ ln1 & ln2

ln1: 10110
& ln2: 11110
10110

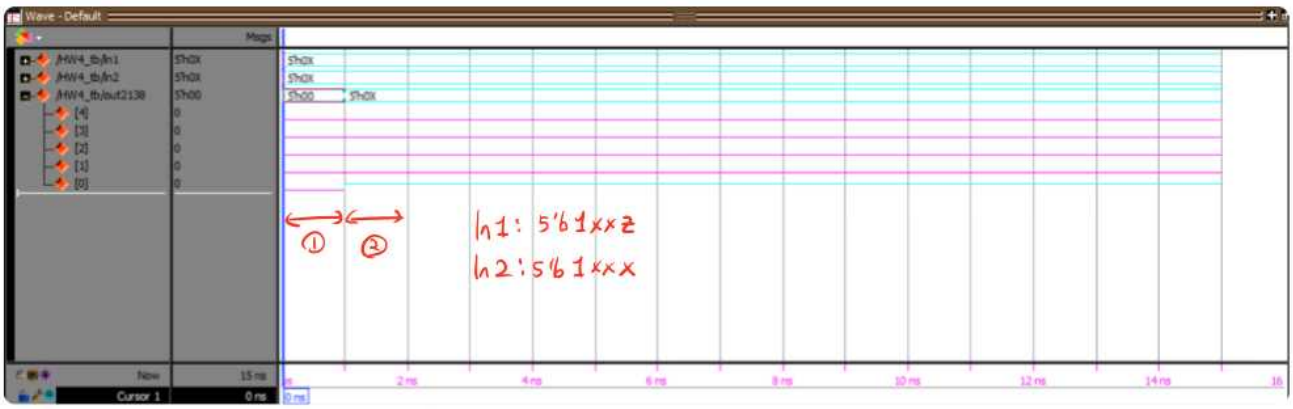
④ ln1 | ln2

ln1: 10110
| ln2: 11110
11110

★number 3



★number 4



① ln1 == ln2

ln1: 01xxz

ln2: 01xxx

00000

=> different

② ln1 == ln2

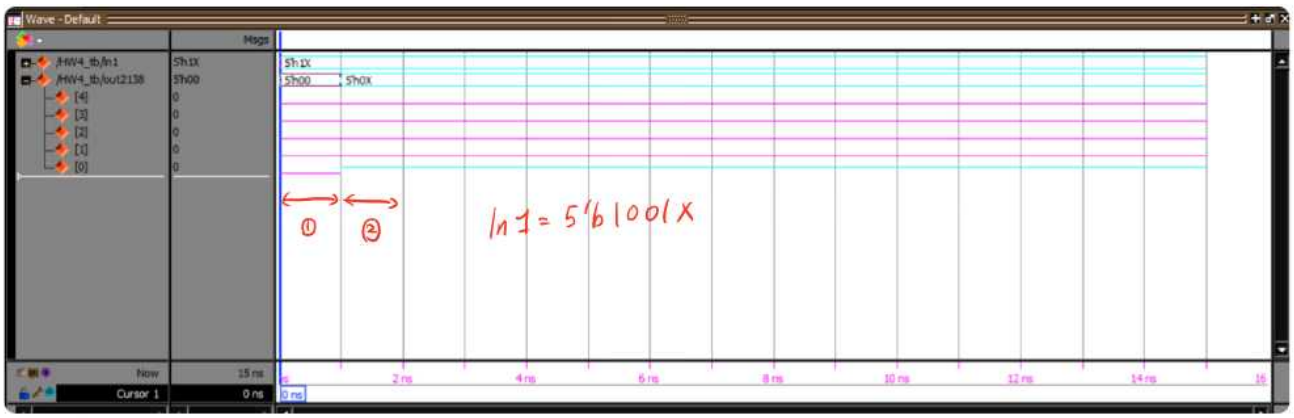
ln1: 01xxz

ln2: 01xxx

0000X

=> unknown

★number 5



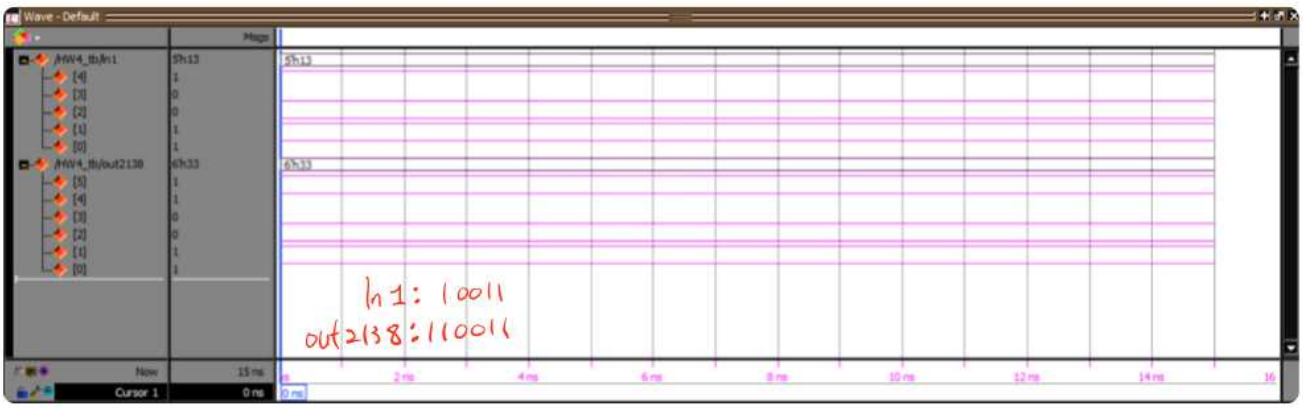
① &ln1

② ^ln1

&ln1 = 1&0&0&1&x = 00000

^ln1 = 1^0^0^1^x = 0000x

★number 6



To make 10011 → 110011,

out2138 = {ln1[0], ln1} = 110011

1 + 10011

source code

★number 1

```
module HW4_tb();
        reg [4:0] ln1,ln2;
        reg [4:0] out2138;
        initial begin
                ln1 = 5'b1101;
                ln2 = 5'bx1100;
                //Arithmetic
                assign out2138 = ln1+ln2;
                //Logical
                #1 assign out2138 = ln1&&ln2;
                //Relational
                #1 assign out2138 = ln1>ln2;
                //Equality
                #1 assign out2138 = ln1==ln2;
                //Bitwise
                #1 assign out2138 = ln1&ln2;
        end
endmodule
```

★number 2

```
module HW4_tb();
        reg [4:0] ln1,ln2;
        reg [4:0] out2138;
        initial begin
                ln1 = 5'bx; // xxxxx
                ln2 = -5'b10; // -2
                assign out2138 = ln1&&ln2; //  => 0000x
                #1 assign out2138 = ln1||ln2; //  => 00001
                #1 assign out2138 = ln1&ln2; // => xxxx0
                #1 assign out2138 = ln1|ln2; // => 1111x

                #3 ln1 = -5'b1010; //-10
                assign out2138 = ln1&&ln2; //  => 00001
                #1 assign out2138 = ln1||ln2; // => 00001
                #1 assign out2138 = ln1&ln2; // => 10110
                #1 assign out2138 = ln1|ln2; // => 11110
        end
endmodule
```

★number 3

```
module HW4_tb();
        reg [4:0] ln1,ln2;
        reg [4:0] out2138;
        initial begin
                ln1 = 5'b100z;
                ln2 = 5'b1100;
                //Arithmetic
                assign out2138 = ln1<ln2;
        end
endmodule
```

★number 4

```
module HW4_tb();
        reg [4:0] ln1,ln2;
        reg [4:0] out2138;
        initial begin
                ln1 = 5'b1xxz;
                ln2 = 5'b1xxx;
                //Arithmetic
                assign out2138 = ln1===ln2; // => 00000
                #1 assign out2138 = ln1 == ln2; // => 0000x
        end
endmodule
```

★number 5

```
module HW4_tb();
        reg [4:0] ln1;
        reg [4:0] out2138;
        initial begin
                ln1 = 5'b1001x;
                //Arithmetic
                assign out2138 = &ln1; // => 00000
                #1 assign out2138 = ^ln1; // => 0000x
        end
endmodule
```

★number 6

```
module HW4_tb();
        reg [4:0] ln1;
        reg [5:0] out2138;
        initial begin
                ln1 = 5'b10011;
                //Arithmetic
                assign out2138 = {ln1[0],ln1};
        end
endmodule
```