



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

```
9 CHIP And {
10  IN a, b;
11  OUT out;
12  PARTS:
13    Nand(a = a, b = b, out = out1);
14    Nand(a = out1, b = out1, out = out);
15 }
```

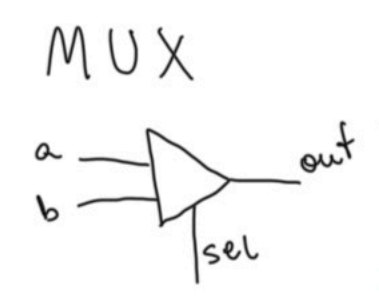
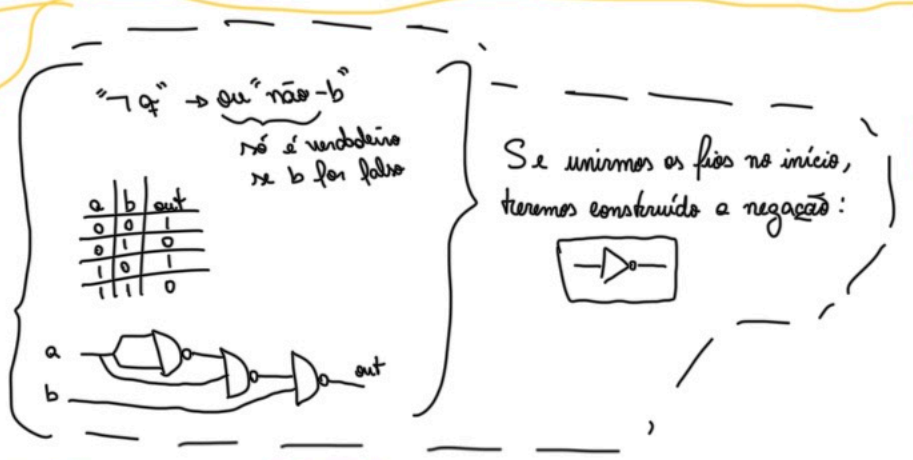
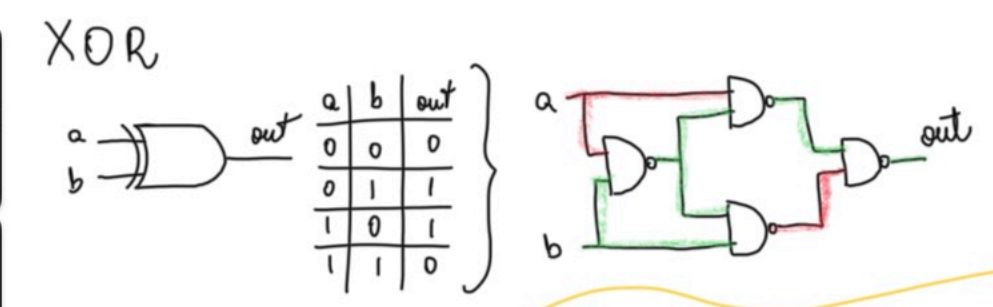
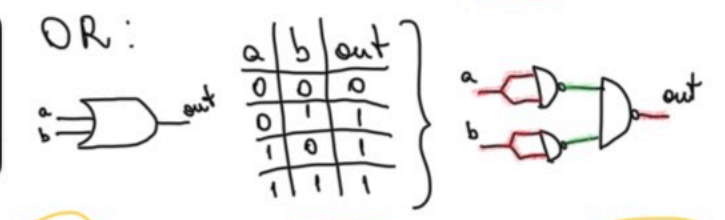
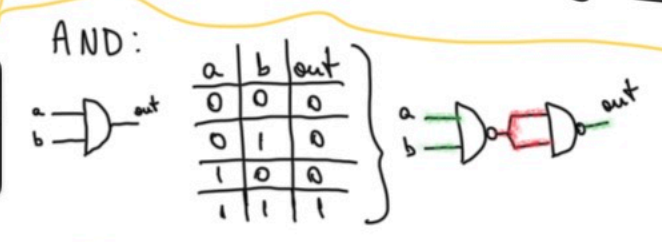
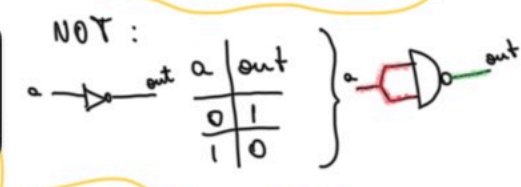
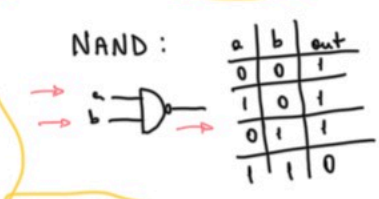
```
9 CHIP Or {
10  IN a, b;
11  OUT out;
12  PARTS:
13    Nand(a = a, b = a, out = out1);
14    Nand(a = b, b = b, out = out2);
15    Nand(a = out1, b = out2, out = out);
16 }
```

```
9 CHIP Xor {
10  IN a, b;
11  OUT out;
12  PARTS:
13    Nand(a = a, b = b, out = out0);
14    Nand(a = a, b = out0, out = out1);
15    Nand(a = out0, b = b, out = out2);
16    Nand(a = out1, b = out2, out = out);
17 }
```

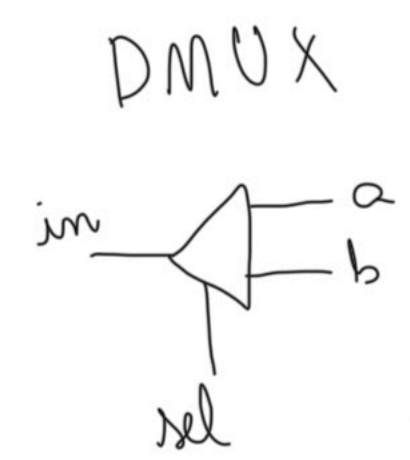
```
9 CHIP Xor {
10  IN a, b;
11  OUT out;
12  PARTS:
13    Not(in = a, out = nota);
14    Not(in = b, out = notb);
15    And(a = a, b = notb, out = w1);
16    And(a = nota, b = b, out = w2);
17    Or(a = w1, b = w2, out = out);
18 }
19 }
```

0 =

1 =



a	b	sel	out
0	0	0	0
0	1	0	0
1	0	0	1
1	1	0	1
0	0	1	0
0	1	1	1
1	0	1	0
1	1	1	1



sel	in	a	b
0	0	0	0
0	1	1	0
1	0	0	1
1	1	1	1

