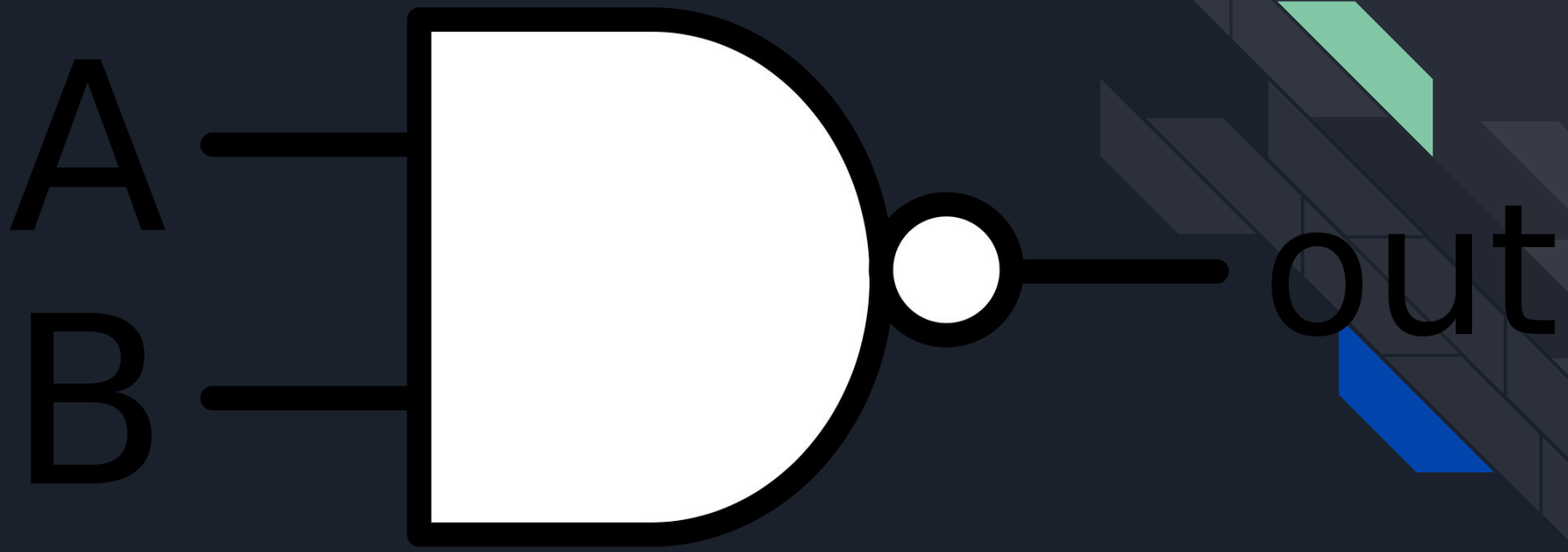
Logic gates

Meets

Logic Programming

Logic Gates ?



Logic gates

The most basic and smallest parts in computers

Very simply they are circuits that allow us to pass the current as we please making the current passing as 1 and no current as 0

Examples

01 The NAND gates the first building block in a computer and what most circuit is made of today

02 NOT gates a gate that negate the current state

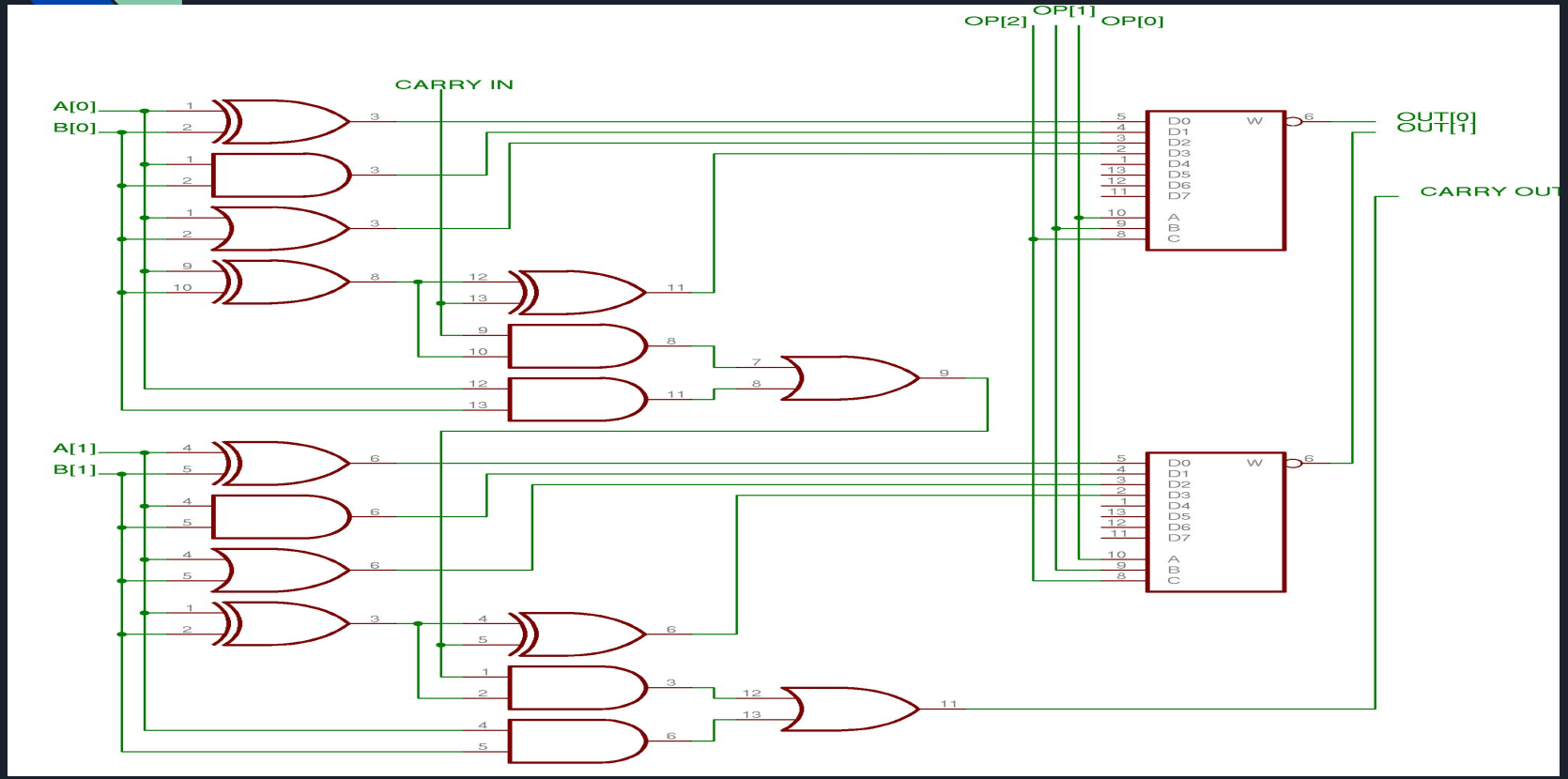
03 AND gate and OR gate

Why these things ?

Logic programming ?

What makes it special ?

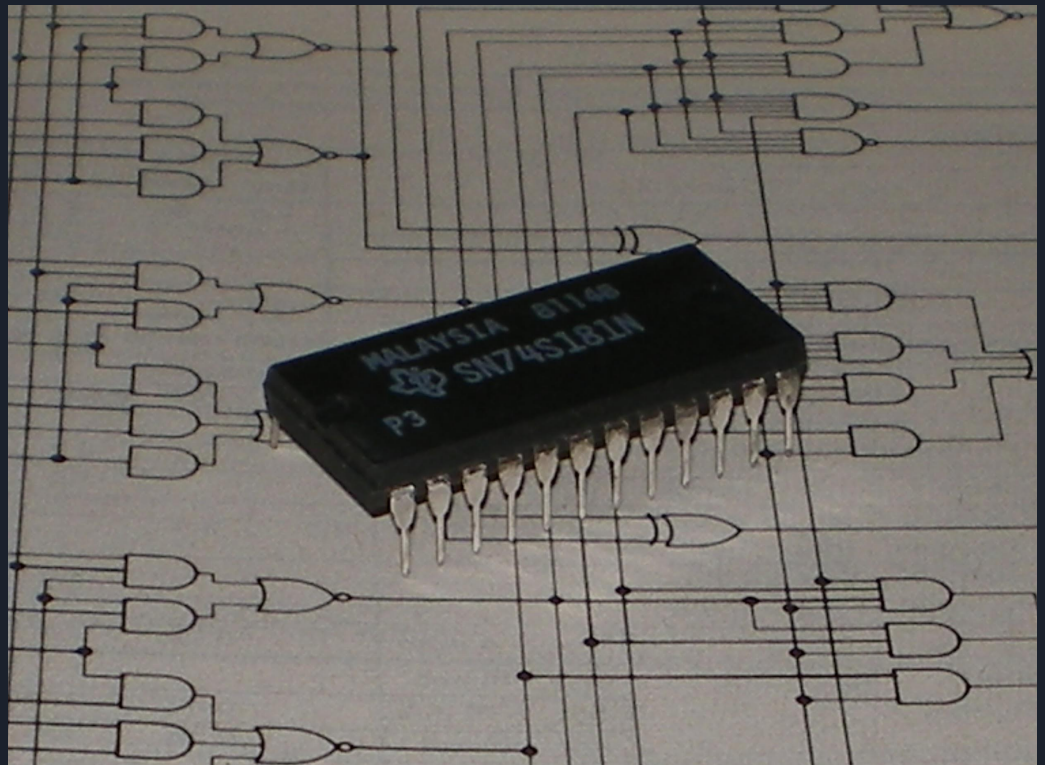
What are we making ?



Introducing: prolog ALU

The architecture is belongs to nand2tetris

We worked up our way by building basic gates up to an actual ALU

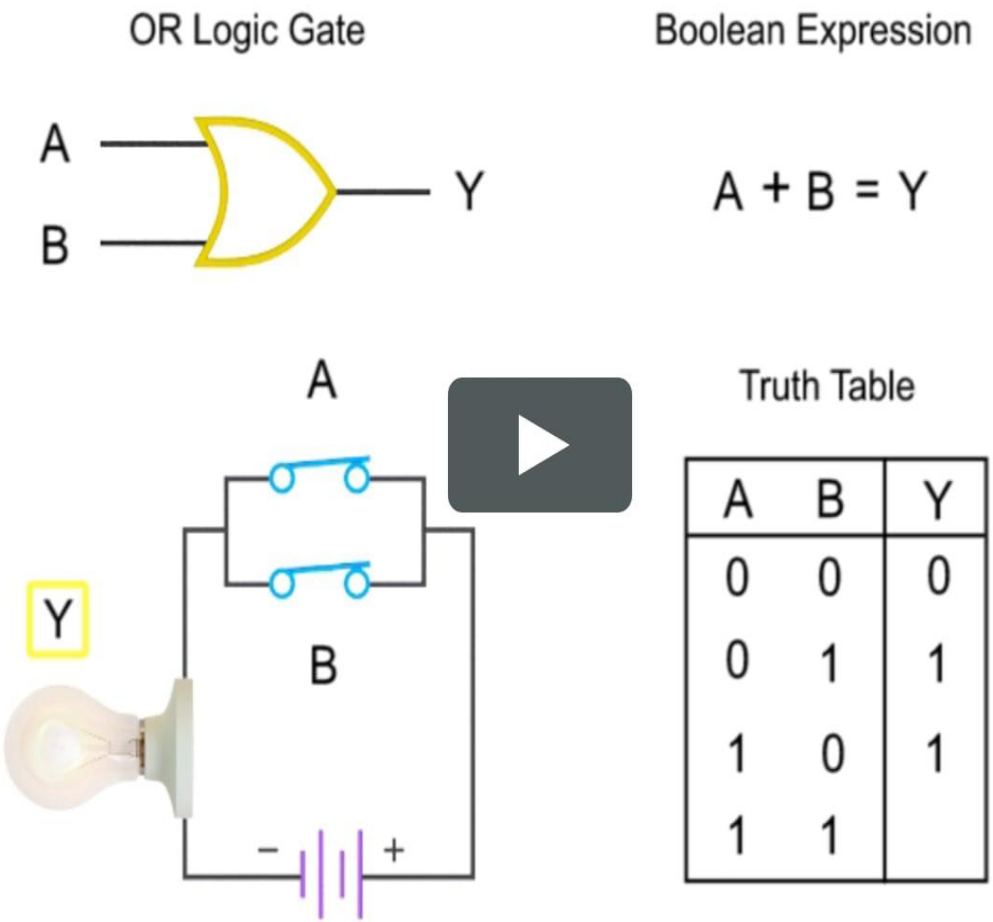


HOW we did it

The truth table becomes

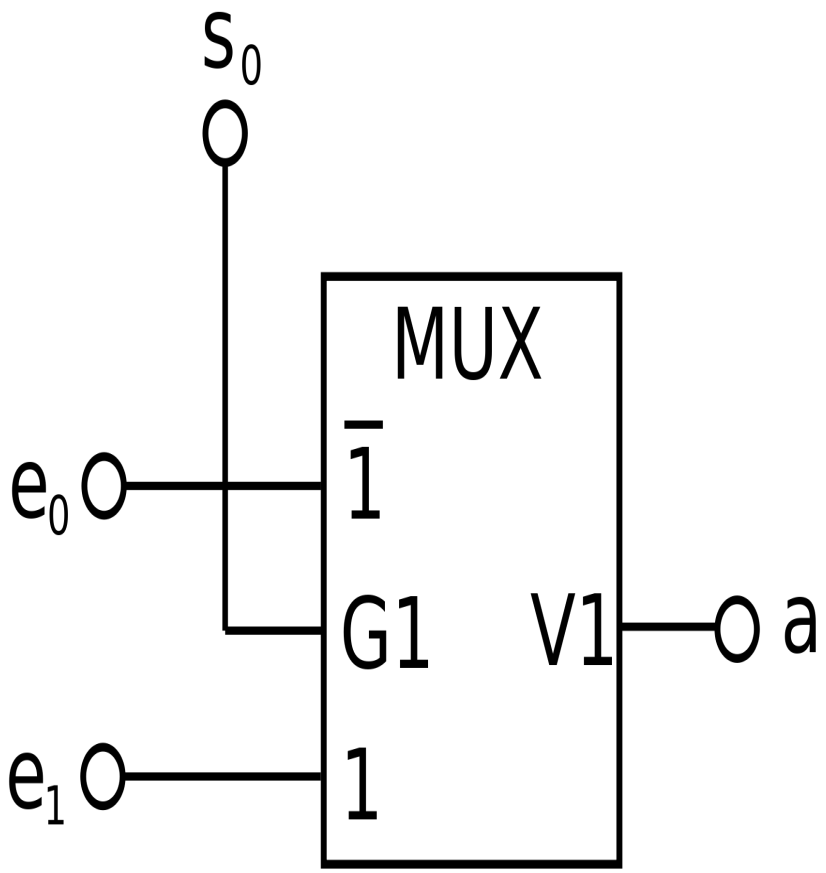
and(1,1,1).

and(_,_,0).



And this thing

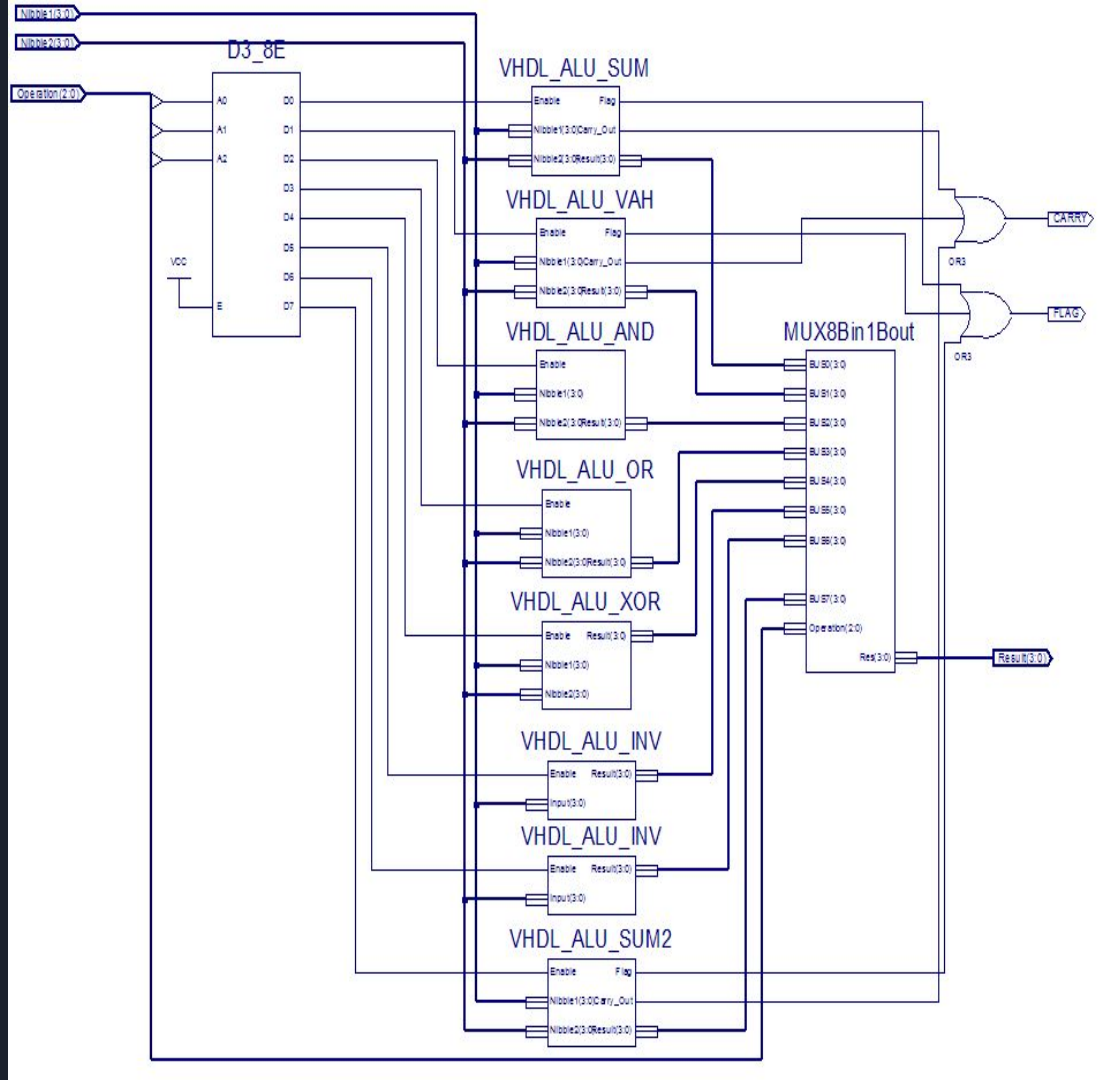
This is a mux gate whats known as if gate
It handles basic logic sequence



Fianlly the ALU

By combining the

Basic gates we get a brain for a computer

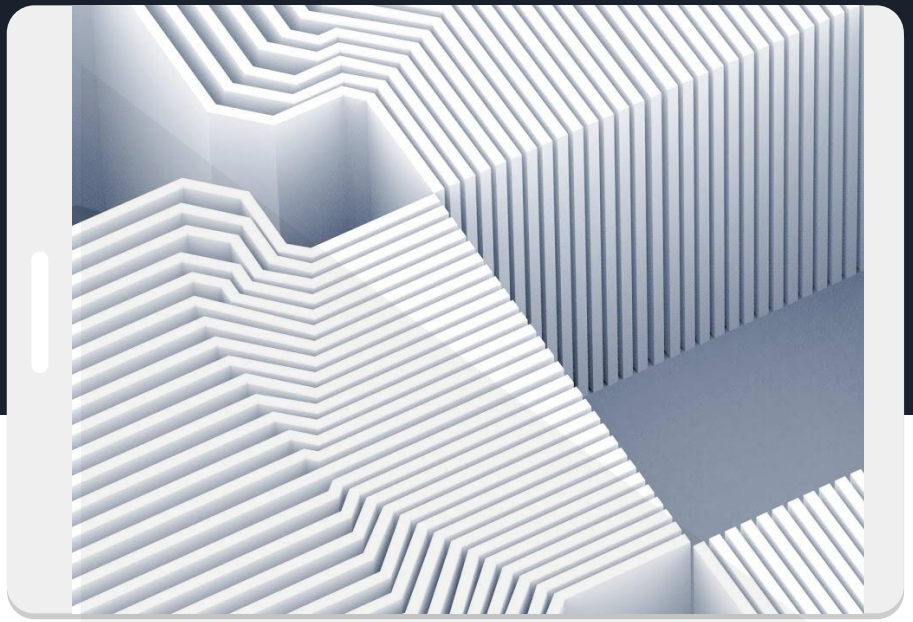


How to use it

```
x[16], y[16],  // 16-bit inputs

zx, // zero the x input?

nx, // negate the x input?

zy, // zero the y input?

ny, // negate the y input?

f,  // compute  out = x + y (if 1) or out = x & y (if 0)

no; // negate the out output?
```

OUT

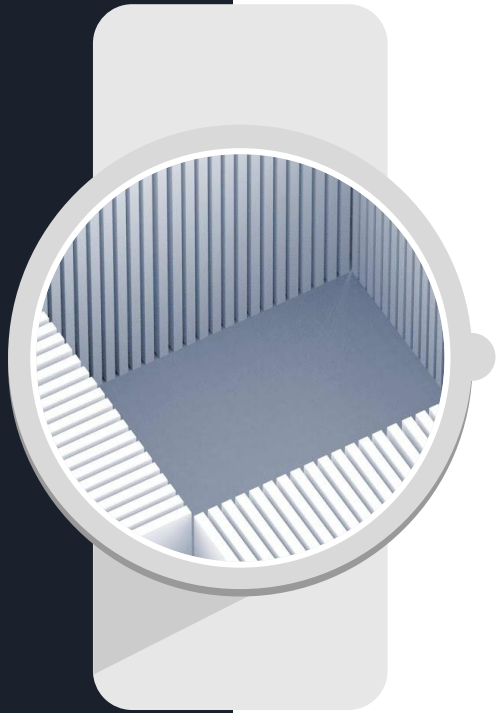
```
out[16], // 16-bit output
zr, // 1 if (out==0), 0 otherwise
ng; // 1 if (out<0),  0 otherwise
```



Using these parameters we get our own unique assembly like code

The usage

alu(A,B,ZX,NX,ZY,NY,F,NO,OUT,ZG,NG).



A,B binary numbers as the input

OUT binary output

ZX zero the A ZY is the same

NY NX negate the input

F choose to add or to and

NO negate the output

ZG if the output is 0

NG if the output is negative

Final words