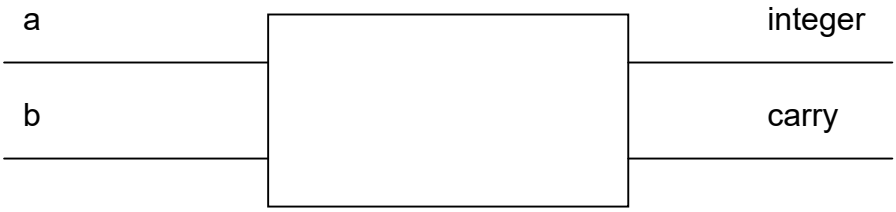
The half adder

With binary inputs at **a** and **b**, we want to be able to add two single digits. This means we need a two-digit output, since one plus one is two, and two in binary is represented as 10 [That's "one zero" **not** "ten"!!!]



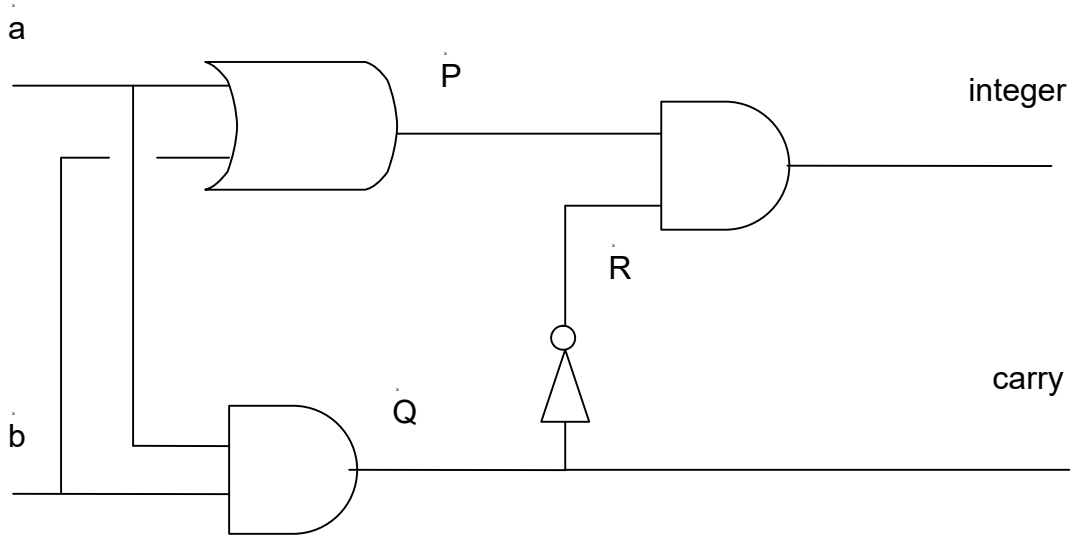
So our **truth table** for this device needs to look like this:

a	b	Carry	integer
0	0	0	0
1	0	0	1
0	1	0	1
1	1	1	0

If we know a little Boolean logic, we can see that the **carry** digit is nice and easy: It's a simple **a AND b**.

Our **integer** value is a little more complicated. It's like **a OR b** except that we want it to be zero when a and b are both true. [The technical term for this is an "exclusive OR"]. That means we want something that is true when **a OR b** are true, but **NOT** when **a AND b** are true.

When both are true, the binary value is two. Since we can only use the digits 1 and 0 in binary, then two requires us to carry a digit and represent it as 10
[Remember that's "one zero" **not** "ten"!]

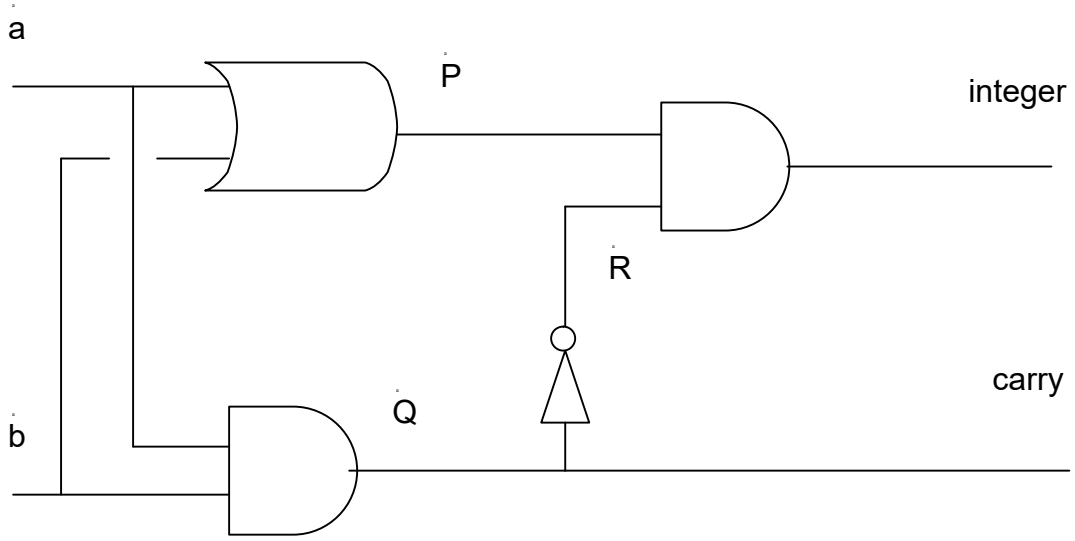


How does it work?

OK, it looks pretty enough - let's see what's going on.

Its sometimes difficult to work out what's happening in these logic circuits all in one go, so here's the trick:

I have labelled each of the points where something has "happened" - P, Q and R. Now if we write a bigger truth table, we can break this thing down into smaller parts, and really start to understand what's going on:



a	b	P [a OR b]	Q [a AND b]	R [NOT Q]	integer [P AND R]	carry (same as Q)
0	0	0	0	1	0	0
1	0	1	0	1	1	0
0	1	1	0	1	1	0
1	1	1	1	0	0	1

Follow it through a line at a time.

So the integer is **(a OR b) AND (NOT (a AND b))**

Tip:

Please familiarise yourself with this diagram and how to work it out from basic understanding of the three types of gate, **AND**, **OR** and **NOT**.

As well as often coming up in the exam (A451), this kind of logical thinking will help you enormously in your analysis, design and programming skills for units A452 (Investigation) and A453 (Practical Project)

[It's also a great way to figure out what's going on in some of the more advance computer games!]