



## Interface additional LEDs with neonious one

### Which pins to use

As with most modern microcontrollers, neonious one cannot drive much current per pin (3 mA). A normal-sized LED uses around 20 mA, typically requiring you to use a transistor or MOSFET in between.

neonious one has 2 special pins however, which do support 20 mA: Pins 4 and 5. With these pins you can drive LEDs directly.

These pins are open-drain only pins, however. At level 1 pin voltage is not 3.3 V, the pin is unconnected instead.

This has no implications, other that we will have to make the LED light up at level 0 and not at level 1, because only at level 0 an electronic circuit is closed.

**Step 1)** Connect: Pin -> LED -> resistor -> 3.3 V PIN

The resistor is needed to limit the voltage on the LED, and because of that also the current flowing through the LED.

Formula:  $R = (3,3 \text{ V} - V_{LED}) / I_{LED}$

Typical value for  $V_{LED}$  is 2.2 V, for  $I_{LED}$  is 20 mA. This would result into a resistor value of 55 Ohm. Take the next resistor with the next highest value you have.

---

Warning: If you connect the LED to any other pin, too much current might flow through that pin, breaking your neonious one.

Warning: If you connect the LED without the resistor, too much current might flow through the pin and the resistor, breaking your neonious one and/or your LED.

---

**Step 2)** Write code

For example, if you have an LED on pin 4 and another one on pin 5, you could write into your /src/index.js:

---

```
let gpio = require('gpio');
gpio.pins[4].setType(gpio.OUTPUT_OPENDRAIN);
gpio.pins[5].setType(gpio.OUTPUT_OPENDRAIN);

let level4 = false;
setInterval(() => {
  level4 = !level4;
  gpio.pins[4].setValue(level4);
}, 500);

let level5 = false;
setInterval(() => {
  level5 = !level5;
  gpio.pins[5].setValue(level5);
}, 700);
```

---

**Step 3)** Press run! If the LEDs do not blink, try putting the LED(s) in the other way around.

