Raspberry Pi Light Sensor Nightlight

Tutorial by Andrew Oakley
Public Domain 15 Sept 2018 www.cotswoldjam.org

The components

The LED has a short leg and a long leg. If you feel around the rim, you'll also find a flat edge. The short leg and flat edge always connect to negative (ground).

LEDs always need to be connected with a resistor. If you connect them without a resistor, they will burn out and probably won't work again.

The resistor can be connected any way around. Lower value (lower ohm) resistors will allow more current through and will make the LED brighter; higher values will let less current through and make it dimmer. We're using a 220 ohm resistor but anything between 220-470 ohms will work fine.

The Light Dependent Resistor (LDR), also known as a photocell, or photoresistor, can be connected any way around. They let through more current when there is more light. We're using one that is rated between 2 and 20,000 ohms but most photocells you find in hobby kits will work fine.

The capacitor has a short and a long leg, and a stripe on one side. Like LEDs, they must be connected with the short leg and stripe to negative (ground). We're using a 1 microfarad capacitor and you should too.

A capacitor stores a small amount of charge, then when it is full it lets all the charge go. If you connect it to a stronge charge, it lets the charge go faster. If you connect it to a weak charge, it takes longer to let the charge go.

This micro breadboard has 25 connection points. The columns are numbered from 0 to 4 (some have the middle columns 1, 2 and 3 marked). All points on a column are connected together, but the different columns are not connected. You can use a larger breadboard if you wish.

It does not matter what colour wires you use. The author of this tutorial is colour-blind. What matters is that they connect to the right places.

Wiring up the light sensor to the breadboard
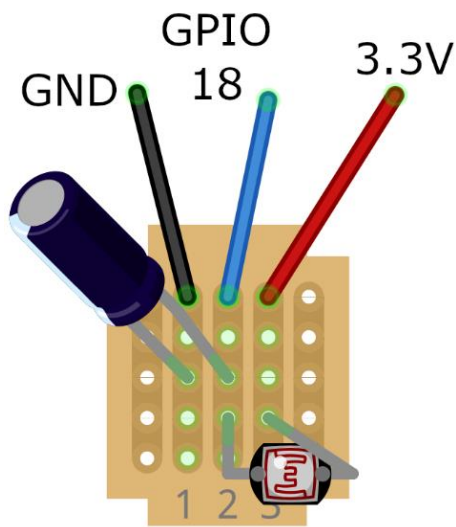
The capacitor has a short leg and a long leg. The side with the short leg also has a stripe. This is the negative (ground) side.

Place the capacitor so negative is on column 1 and the other leg is on column 2.

The light dependent resistor (LDR) needs to be connected to columns 2 and 3. It doesn't matter which way round.

Use the 20cm male-to-female jumpers to connect:
- Column 1 to ground
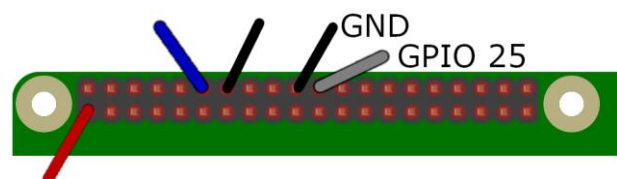- Column 2 to GPIO 18
- Column 3 to 3.3 volts
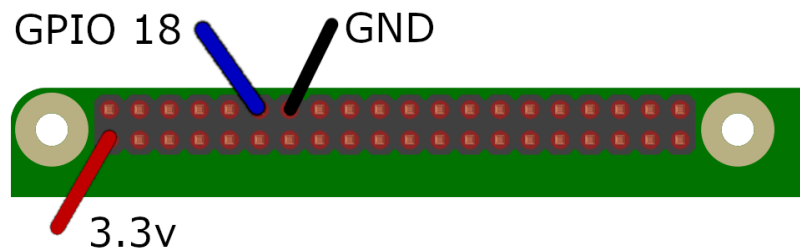
Wiring up the LED

Connect the resistor between the two 10cm female-to--female jumper wires, and attach these to the long leg (positive) of the LED. Connect this to GPIO 25.

Connect the 20cm female-to-female jumper wire to the short leg (negative) of the LED, then connect the other end of this wire to ground.

A program for the light sensor

From the Raspberry menu, select Programming, Python 3 (IDLE).

From the IDLE menu, select File, Open. Find the python folder and double-click it to open it. Find the lightsensor folder and double-click it to open it. Select the lightsensor1.py program and click Open. You should see:

```
from gpiozero import LightSensor
from time import sleep
sensor=LightSensor(18)
while True:
  print (sensor.value)
  sleep(0.2)
```

Run this program using Run menu, Run Module. It will take a moment to start, then it should print out some numbers.

Cover the light sensor with your finger. The numbers will get lower. Enclose the breadboard in your hand so that no light gets to it at all - the numbers should go to zero.

Stop the program by holding down the CTRL key and pressing C. You can close the shell window with the X button at the top-right of the window.

The nightlight - turn the light on when it gets dark

From the IDLE menu, load the lightsensor2.py program:

```
from gpiozero import LightSensor, LED
from time import sleep
sensor=LightSensor(18)
led=LED(25)
sensor.threshold=0.1
while True:
  print (sensor.value)
  if ( sensor.light_detected ):
    led.off()
  else:
    led.on()
  sleep(0.2)
```

Run this program using Run menu, Run Module. It will take a moment to start, then it should print out some numbers, and when the numbers fall below 0.1 (dark), the LED will switch on.

Stop the program and try changing the sensor threshold to, for example, 0.2 or 0.4 or even 0.6 . See how you can set the nightlight to turn on at different levels of darkness.

```
sensor.threshold=0.4
```

What are these programs doing?

The Light Dependent Resistor is an analogue sensor - it doesn't acually produce numbers, it simply lets more current through when it is light, and less when it is dark.

The Raspberry Pi cannot read analogue sensors. The Raspberry Pi's GPIO pins are digital and binary - they can only detect high (1) or low (0) voltage.

What these programs are doing is timing how long it takes for the electrical charge to fill up the capacitor. The capacity is like a tiny rechargeable battery.

If there is a lot of light, the LDR lets lots of current through, and capacitor fills up quickly. If there is not very much light, the LDR lets only a small amount of current through, and the capacitor fills up slowly.

When the capacitor is full, it releases all of its charge suddenly. The Raspberry Pi can detect this as high (1) and it stops the timer.

The LightSensor object then reverses this time measurement to express a "brightness value".

value = 1 - (time take to fill the capacitor)

The capacitor is very sensitive, so the time taken will only be a tiny fraction of a second!

Advanced topics

Have a look at the lightsensor3.py program. In this program, we are testing for three light levels - dark, medium and light - and using the PWMLED object to set the brightness of the LED, rather than simply turning it on or off.

Finally, there is the lightsensor-old.py program. This uses a less advanced way to measure the light - instead of using the LightSensor object, it actually times how long the capacitor takes to fill up, in milliseconds, and displays this time.

5V	5V	GND	GPIO14	GPIO15	GPIO18	GND	GPIO23	GPIO24	GND	GPIO25	GPIO8	GPIO7	DNC	GND	GPIO12	GND	GPIO16	GPIO20	GPIO21
2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39
3.3V	GPIO2	GPIO3	GPIO4	GND	GPIO17	GPIO27	GPIO22	3.3V	GPIO10	GPIO9	GPIO11	GND	DNC	GPIO5	GPIO6	GPIO13	GPIO19	GPIO26	GND

Preparation

If your tutor has not prepared your Raspberry Pi and equipment, or if you don't have a tutor, you can set things up by following these instructions before you start.

Equipment

eBay is a good place to buy electronic components. You will need:

- A Raspberry Pi running the Raspbian operating system. This tutorial suits all models to date, which includes the A, A+, B, B+ and 2B.
- A solderless breadboard – we're using a tiny 25-point one, but any solderless breadboard will do.
- An LED. We're using a 5mm yellow one. It needs to be bright enough to trigger the photocell, so yellow, green or white will usually work beter than red or blue.
- A Light Dependent Resistor (LDR), also known as a photocellor photoresistor. We're using a 4mm 2-20k ohm one.
- A resistor. We used a 220 ohm one, but anything from 220-470 ohms is fine.
- Three 20cm male-to-female jumper wires
- Two 10cm female-to-female jumper wires
- One 20cm female-to-female jumper wire

Files

Please use an up-to-date installation of Raspbian Jessie. This should have GPIOZero already installed.

You can create the folder structure and download the files by going to the terminal (Menu - Accessories - Terminal) and typing:

```
cd
mkdir python
cd python
curl -O http://cotswoldjam.org/downloads/2018-09/lightsensor.zip
unzip lightsensor.zip
```

…where -O is a minus sign followed by a capital letter O (not the numeral zero).