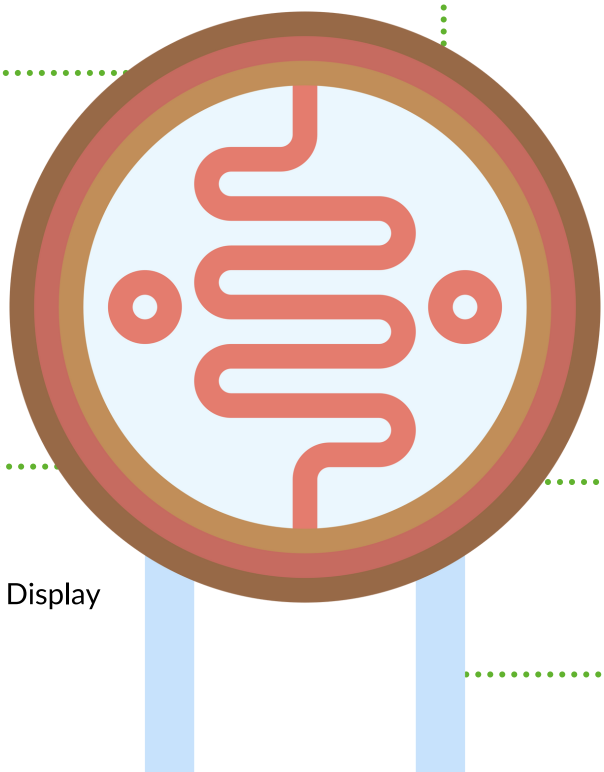
PROGRAMMING RESOURCES - ACTIVITY SHEET 4

BASIC LIGHT SENSOR

#R1AS04



What is it?

This activity sheet will approach resistors. A light-dependent resistor (LDR) is a component used to measure light levels.

Available on

Duration

25 minutes

Pre-requisites

- R1AS02 - Breadboarding
- R1AS03 - Buttons and LED Display

Level of difficulty

Intermediate

Material

- 1 Programming board "***STM32 IoT Node Board***"
- Micro-B USB Cable
- 1 set of resistors
- 1 LDR
- 1 Breadboard
- Jumper wires

LEARNING OBJECTIVES

- Create a simple light sensor with a few electronic components on a breadboard and connect it to the board
- Create a program in MakeCode that is able to measure an analogue physical quantity by means of a sensor
- Produce a plot that shows how a measured value varies over time

This activity sheet is part of the Let's STEAM project funded with support from the European Commission through the Erasmus + Strategic Partnership programme. Its content reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

BASIC LIGHT SENSOR

This activity illustrates a key feature of physical computing: the ability to measure a physical quantity using a sensor and graphically represent how this quantity varies over time. We will connect a light-dependent resistor (LDR) to the board to measure light levels. This kind of sensor is called an **analogue sensor** because we need to get an analogue characteristic of the circuit (the voltage) to get the value of the sensor.
Resource: https://www.watelectrical.com/what-are-analog-sensors-types-and-their-characteristics/

STEP 1 - MAKE IT

Wire the photocell

1

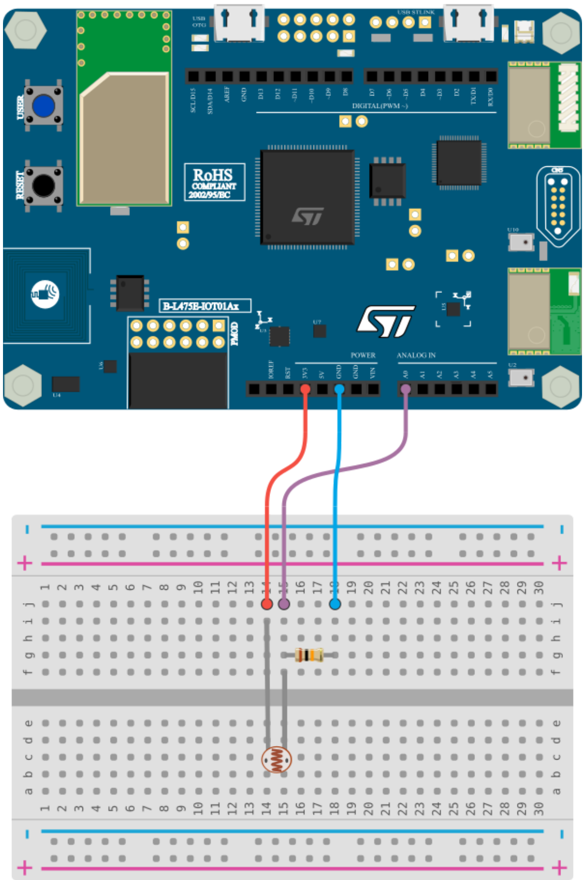
The circuit we need to assemble consists of two components: a **4.7 kΩ resistor** and a **photocell**.

The colour of the first three stripes indicates the resistance value of the component, according to a code that is known as "resistors colour code". The fourth stripe indicates that the resistance value is subject to uncertainty (tolerance) that may be either 5% (if the stripe is gold) or 10% (if the stripe is silver) of the nominal resistance value.

Light Dependent Resistors (aka. LDR, Photocell, Photoresistor and CdS Cell) is a component whose electrical resistance varies according to the intensity of the light to which the component is exposed.

The easiest way to measure a resistive sensor is to connect one end to Power and the other to a pull-down resistor to the ground. Then, the point between the fixed pulldown resistor and the variable photocell resistor is connected to the analogue input of a microcontroller. Such an arrangement forms what we call an analogue sensor. This term means that **this circuit is able to sense a physical quantity** (namely, light intensity) and transform it into a **proportional electrical quantity** (specifically, a voltage whose value is between 0 V and 3.3 V).

These two components need to be assembled on a small breadboard, as depicted in the picture aside.



Assembly of the 4.7 kΩ resistor and the photocell on the breadboard

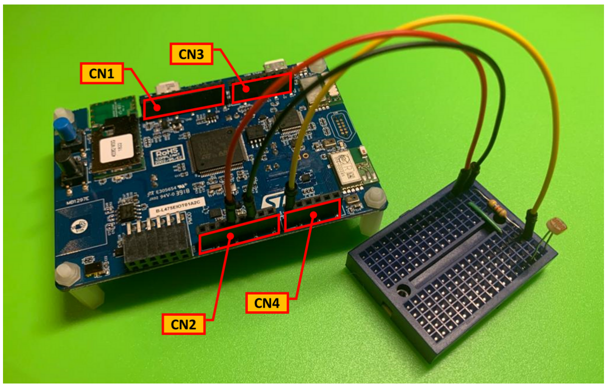
Wire the breadboard to the STM board

2

Once the breadboard has been assembled, it needs to be connected to the board. The picture shows that the board has four connectors, named **CN1**, **CN2**, **CN3** and **CN4**, respectively. Since the four connectors have different purposes, use the blue buttons located at one of the four corners of the board to properly identify the four connectors.

STEP 1 - MAKE IT

The red wire must be connected to **pin 4** of connector **CN2**, which is internally connected to a 3.3 V potential. The black wire must be connected to **pin 6** of the connector **CN2**, which is internally connected to the ground potential (**GND**). Finally, the yellow wire must be connected to **pin 1** of connector **CN4**. This pin is internally connected to the analogue input pin named **A0**.



Wiring the breadboard to the STM board

3

Connect the board to the computer

With your USB Cable, connect the board to your computer by using the **micro-USB ST-LINK connector** (on the right corner of the board). If everything is going well you should see a new drive on your computer called **DIS_L4IOT**. This drive is used to program the board just by copying a binary file.

4

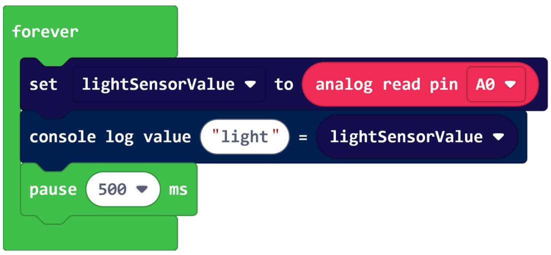
Open MakeCode

Go to the **Let's STEAM MakeCode editor.** On the home page, create a new project by clicking on the "New Project" button. Give a name to your project more expressive than "Untitled" and launch your editor.
Resource: makecode.lets-steam.eu

5

Program your board

Inside the MakeCode Javascript Editor, copy/paste the code available in the **Code It Section** below. If not already done, think of giving a name to your project and click on the "**Download**" button. Copy the Binary file on the drive **DIS_L4IOT**, wait until the board finishes blinking and your program is ready!



Full blocks enabling the program to run

6

Connect to the board console

In your MakeCode editor, click on the button "Show console Simulator" below, on the left side, the board simulation. The terminal shows then the periodic light values read by the program. This value can be exported as a CSV file by clicking on the button "export data" in the top right corner of the console.

Console on the MakeCode editor

7

Run, modify, play

Your program will automatically run each time you save it or reset your board (push the button labelled RESET). Try to understand the example and start modifying it by changing the period between two measurement sessions. You can hide the photocell with your hand to directly observe the value changing.

STEP 2 - CODE IT

```
let lightSensorValue = 0
forever(function () {
   lightSensorValue = pins.A0.analogRead()
   console.logValue("light", lightSensorValue)
   pause(500)
})
```

How does it work?

The code consists of:

- a forever block;
- a console log block;
- a pause block.

The forever block implements “a loop”, which keeps executing three basic instructions until the board is turned off.

The first block reads the value of the analogue input pin **A0** and stores it in a variable named lightSensorValue. This value is an integer number between 0 and 1023.

An analogue input pin may be used to read a value between 0 and 1023. This value is proportional to the voltage applied to the pin, which MUST be comprised between 0 V and 3.3 V (relative to GND).

The second block writes to the console terminal of the board what is obtained by reading the value of the sensor.

As soon as this instruction has been carried out, the board suspends its activity (pause) for 500 milliseconds, i.e. half a second.

Now a question naturally arises: what is the board console? How is it possible for us to read what is written to the console? The board console allows the board to simply interact with the PC connected to it through the USB cable.

STEP 3 - IMPROVE IT

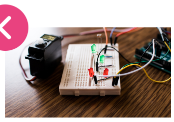
1. Use your sensor in **many light conditions** (ambient light, moonlit night, ...). How can we calibrate our sensor to be well adapted to the sensing condition? **Try several values of the pull-down resistor to see the impact.**

2. Add an LED and transform this circuit into a hand **controllable light dimmer.**

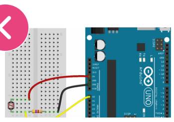
3. The actual value of the sensor is a value between 0 and 1023. **Read the value of the darkest light and the value of the brightest light** and transform the original value into a more explicit percentage value.

GOING FURTHER

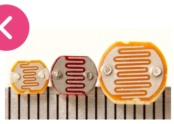
Light-dependent resistor - Learn more about photoresistors, their applications and design. *https://en.wikipedia.org/wiki/Photoresistor*



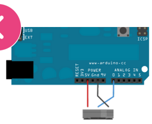
Photocell Hookup Guide - Quick primer on resistive photocells', and demonstrates how to hook them up and use them. *https://learn.sparkfun.com/tutorials/photocell-hookup-guide/all*



Photocells - Discover photocells, a resistor that changes its resistive value depending on how much light is shining onto the squiggly face. *https://learn.adafruit.com/photocells*



Analog Read Pin - Choose a pin and read an analogue signal (0 through 1023) from it. *https://makecode.microbit.org/reference/pins/analog-read-pin*



Explore other activity sheets

R1AS11 - Make a very readable thermometer

R1AS15 - Collecting data