PROGRAMMING RESOURCES - ACTIVITY SHEET 5

POTENTIOMETER

#R1AS05

What is it?

In this activity sheet, we will learn about potentiometer by programming the board to adjust the brightness of a LED by turning a knob.

Available on

Pre-requisites

- R1AS01 - Blink a LED
- R1AS02 - Breadboarding
- R1AS04 - Basic Light Sensor

Duration

20 minutes

Level of difficulty

Intermediate

Material

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

LEARNING OBJECTIVES

- Wire external components to the board
- Read an analog input using a potentiometer
- Use an analog input to write an analog output

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

POTENTIOMETER

A potentiometer is a **three-terminal resistor** with a sliding or rotating contact that forms an adjustable voltage divider. If only two terminals are used, one end and the wiper, it acts as a variable resistor or rheostat. The measuring instrument called a potentiometer is essentially a **voltage divider** used for measuring electric potential (voltage); the component is an implementation of the same principle, hence its name.

Potentiometers are commonly used to **control electrical devices** such as volume controls on audio equipment. Potentiometers operated by a mechanism can be used as position transducers, for example, in a joystick. Potentiometers are rarely used to directly control significant power (more than a watt) since the power dissipated in the potentiometer would be comparable to the power in the controlled load.
Resource: https://en.wikipedia.org/wiki/Potentiometer

STEP 1 - MAKE IT

Wire the potentiometer
Connect the left prong of the potentiometer to **GND**. The right prong should be connected to **3.3V**. Wire the middle one to **A0**.

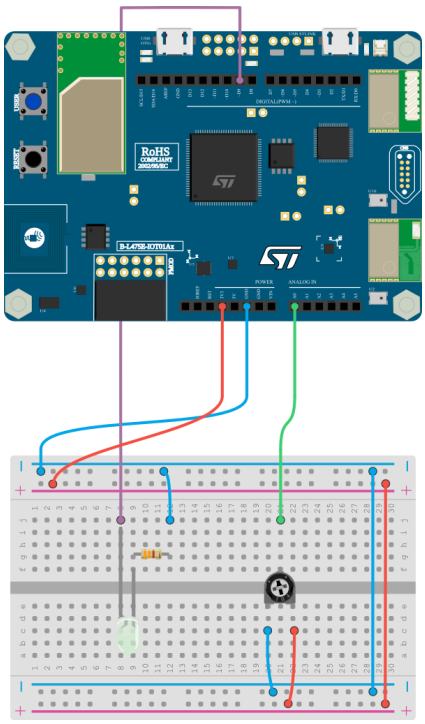
Wire the LED
Connect the **anode** (**+**) of the LED on **D9**. Connect the LED's **cathode** (**-**) to a resistor (330 ohms). Then, connect the unconnected side of the resistor to **GND**.

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.

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

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 board is ready!

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.



Wiring the potentiometer and LED

POTENTIOMETER

STEP 2 - CODE IT

```
forever(function () {
    pins.D9.analogWrite(pins.A0.analogRead())
})
```

How does it work?

The code consists of three elements:

- a forever block;
- an analogRead block;
- an analogWrite block.

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

The analogRead block is used to get the value of the potentiometer on pin A0. This value is an integer number between 0 and 1023. Turning the knob changes the value.

The potentiometer acts as an adjustable voltage divider. By changing the position of the knob, you change the voltage applied on A0. The more you turn it to the left, the more the voltage will be close to 0V. The more you turn it to the right, the more the voltage will be close to 3.3V.

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 0V and 3.3V.

The analogWrite block is used to light up the LED on D9. By using analogWrite, the board is able to limit the voltage to a certain value to make the LED shine dimmer or brighter. The brightness is set by the value of analogRead on pin A0: the higher the value, the brighter the LED.

By using pin D9, we are able to write an analogue value through a digital pin to the board. Pin D9, like a few other pins on the board, supports Pulse Width Modulation or PWM. This technique uses on-off patterns to simulate different voltages and thus different analogue signals. The value passed to analogWrite should be between 0 and 255. 0 stand for a 0V voltage and 255 for 3.3V.

As you will see by using this program, you will not use the full range of the potentiometer. You can transform the range of value of the potentiometer(0...1023) into the range of the PWM (0...255) with the map function.

POTENTIOMETER

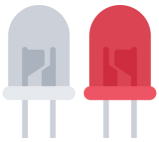
STEP 3 - IMPROVE IT

By using the map function, try to use **the full range of the potentiometer**. You can define two variables to be more expressive and separate reading, transforming and writing on a specific statement.

1

Add another LED and inverse the value of the potentiometer, so that the second LED dims as the first one lights up.

2

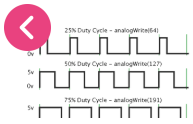


Use the potentiometer to **control the pitch of a buzzer**. Use a potentiometer to **control the position of a servo.**

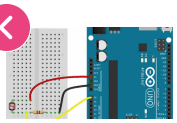
3

GOING FURTHER

Pulse Width Modulation - Arduino tutorial on the use of analogue output (PWM) to fade an LED.
https://www.arduino.cc/en/Tutorial/Foundations/PWM



Voltage Dividers - Discover how voltage dividers behave in the real world.
https://learn.sparkfun.com/tutorials/voltage-dividers



Arduino pong game on 24x16 matrix with MAX7219 - Build a small pong console.
https://www.youtube.com/watch?v=dK9F5AJM2XI

Potentiometer Game - Control a game's avatar using a potentiometer.
https://www.hackster.io/matejadjukic03/potentiometer-game-05ee93?f=1#

Explore other activity sheets

R1AS11 - Make a very readable thermometer

R1AS15 - Collecting data