MORSE CODE

#R1AS06

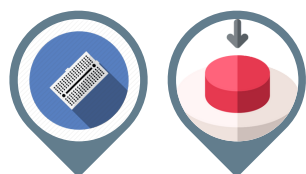## What is it?

Morse code is a method used in telecommunication to encode text characters as standardized sequences of two different signal durations, called dots and dashes.

Resource: https://en.wikipedia.org/wiki/Telecommunication

Available on

Pre-requisites

- R1AS02 - Breadboarding
- R1AS03 - Buttons and LED Display

Duration

30 minutes

Level of difficulty

Advanced

Material

- 1 Programming board "***STM32 IoT Node Board***"
- Micro-B USB Cable
- 1 Breadboard
- 1 piezo buzzer or a speaker
- 2 buttons
- Jumper wires

LEARNING OBJECTIVES

- Wire and use the passive buzzer
- Communicate with Morse code

MORSE CODE

From microwaves to game shows, buzzers are all around us and can help to signal something with a beeping noise. To emit sound (or noise), the buzzer contains a thin membrane (made of quartz), which vibrates to a given frequency (between 20Hz and 20,000Hz, which are the listenable frequencies).
Resource: https://en.wikipedia.org/wiki/Buzzer

In this activity sheet, you will attach some buttons and a buzzer to the board and learn to communicate with **morse code**!
Resource: https://en.wikipedia.org/wiki/Morse_code

STEP 1 - MAKE IT

1

Wire buzzer
In theory, a buzzer is not polarized (it means that there is no "+" nor "-"), but you often have a pair of wires black/red or signs ("+" and/or "-") on the device. If you are in this configuration, attach the lead on the "+" side of the buzzer to pin **A2** and the other one to pin **GND**. If there is no colour or indication, just plug one wire on pin **A2** and the other one on pin **GND.**

2

Wire buttons
Connect one side of each button to the pin **GND** on the board. Then attach the other sides, on pin **A0** (button 1), and pin **A1** (button 2).

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.

Wiring the buzzer and buttons

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.
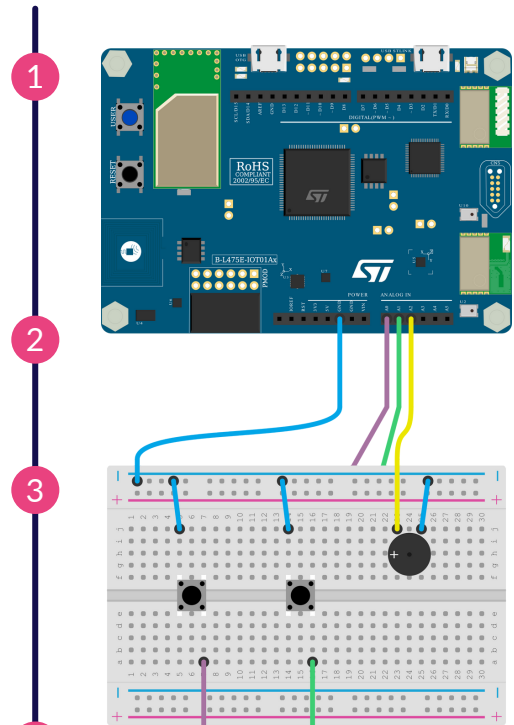*Resource: makecode.lets-steam.eu*

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 work is ready!

6

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.

STEP 2 – CODE IT

```
// Send short signal
input.buttonA0.onEvent(ButtonEvent.Click, function () {
    music.playTone(440, 100)
})

// Send long signal
input.buttonA1.onEvent(ButtonEvent.Click, function () {
    music.playTone(440, 300)
    })
```

How does it work?

The code is really simple! You can see the two functions onEvent to detect when a button is pressed.

Then we simply used the music.playTone function, with 2 parameters:
- **440**: the frequency we want to play
- **100** or **300**: the duration of the tone in milliseconds (1 second = 1,000 milliseconds)

Now you have understood the basics, we will send a morse message!

Signalling morse code

Morse code is a method of communication that encodes characters as a sequence of **2 different signal durations** known as **dots** and **dashes**.

A **dot** is a **short signal** while a **dash** is a **longer signal**. By combining multiple sequences, you can convey a message consisting of several words. Morse code can be signalled in various ways: by using a (flash)light, a radio, or with a board like the one you have!

The figure on the right gives you an overview of how to signal each letter in morse. Try to send "**SOS**" to someone!

International Morse Code

1. The length of a dot is one unit.
2. A dash is three units.
3. The space between parts of the same letter is one unit.
4. The space between letters is three units.
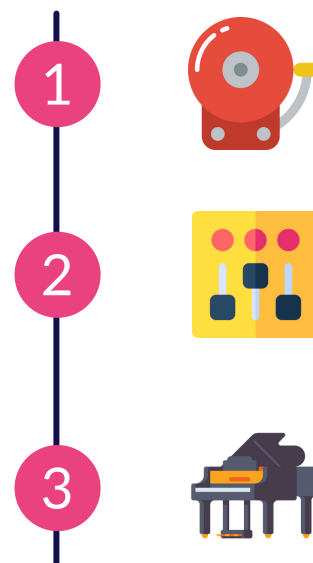5. The space between words is seven units.

STEP 3 - IMPROVE IT

To help hearing-impaired people, **add a LED to indicate when the buzzer is going on.**

1

You can try to make your **preferred music by playing several tones** when you click on the button.

2

Add more buttons and try to play a **simple melody**.

3

GOING FURTHER

Morse Code - Learn more about morse code history, representations, timing, speeds and learning methods. *https://en.wikipedia.org/wiki/Morse_code*

Buzzer Basics - Technologies, Tones, and Drive Circuits. *https://www.cuidevices.com/blog/buzzer-basics-technologies-tones-and-driving-circuits*

Sound - Discover acoustics basis, physics and perception of sounds. *https://en.wikipedia.org/wiki/Sound*

Clothespin Piano with micro:bit - Read an analogue signal (0 through 1023) from the pin. *https://browndoggadgets.dozuki.com/Guide/Clothespin+Piano+with+micro:bit/302*
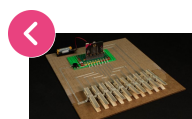
Explore other activity sheets

R1AS07 - Make a theremin with the distance sensor