STEMKIT 4SCHOOLS

MEASURING THE SPEED OF SOUND

LESSON PLAN 1

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

This project has been funded with support from the European Commission.

Table of Contents

1. Measuring the Speed of Sound

1.1 General information

1.1.1 Short description

In this lesson plan we will conduct a scientific experiment to measure the speed of sound. For this purpose, we will make an experimental apparatus using our STEMKIT and operate it by appropriate program. Then we will collect data and analyse them to measure the speed of sound. Just like real scientists and researchers do!

1.1.2 Learning objectives

The main learning objectives of this lesson plan are:

- concept and content understanding of what is sound, sound waves, propagation of waves, speed
- designing and performing an experiment or scientific investigation with collection of data, analysis and presentation of results
- familiarizing with circuits and programs to interact with GPIO pins of Raspberry Pi
- understanding basic structures of Python programming language (namely, use in program the syntax, purpose and function of *while loop*, *if statements*, *definition of function*, etc.
- using a spreadsheet (in paper or software form) for data logging and processing basic statistics

1.1.3 Links to curriculum

The domains, subdomains, subjects/topics that this lesson plan can be linked to are:

- Physics: motion, oscillation, waves, types of waves, characteristics of waves, propagation of waves, sound, speed of sound waves, spectrum of sound waves
- Science (Physics/Chemistry/Biology/Geology): scientific method, investigation, experimentation, analysis and interpretation of results
- Computer Science/Informatics: processing unit and peripherals, interfaces, programming language and main structures, coding

- Technology: electronics, open-source hardware and software, sensors, digital signal, circuits, single board computers
- Maths/Statistics: spreadsheets and basic statistics

1.1.4 Materials required

For this lesson plan (and for each student group) besides the STEMKIT console and its Raspberry Pi we'll need:

- 8 x Male-to-Female jumper wires
- 1 x Breadboard
- 1 x HC-SR04 Ultrasonic Distance Sensor
- 1 x push button
- 4 x Resistors (between 300 and 1K Ohm). We will need 3x 1K Ohms for the distance sensor, and one 300 to 1K Ohm resistance for the push button
- A ruler or measuring tape
- A small box or object to make an obstacle

1.1.5 Duration

The duration of this lesson plan is estimated to be about 45-60 mins, i.e., one classroom hour.

1.2 Lesson plan

The lesson plan is divided in four phases, which are introduction, preparation, investigation and conclusion. As a follow-up there is also an optional exercise at the end.

1.2.1 Introduction

Sound is a vibration that propagates as an audible wave of pressure, through a transmission medium such as a gas, liquid or solid. Humans can only hear sound waves as distinct pitches when the frequency lies between about 20 Hz and 20 kHz. Sound waves above 20 kHz are known as ultrasound and is not perceptible by humans. Sound waves below 20 Hz are known as infrasound. Different animal species have varying hearing ranges.

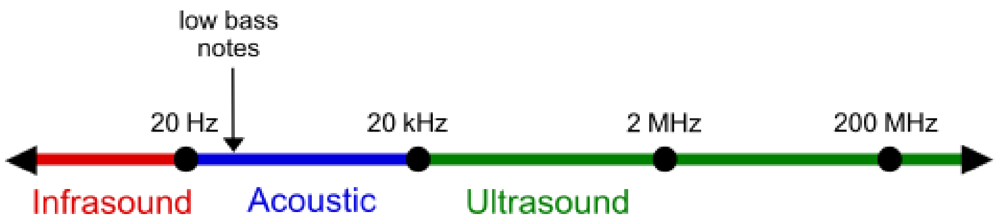

IMAGE 1. FREQUENCY RANGES CORRESPONDING TO INFRASOUND, HUMAN AUDIBLE SOUND AND ULTRASOUND

Source: https://en.wikipedia.org/wiki/Sound

Sound waves are often simplified to a description in terms of sinusoidal waves, which are characterized by: frequency, or its inverse, wavelength; amplitude, sound pressure or intensity; speed of sound; direction. In this lesson plan we will measure ourselves the speed of sound using the STEMKIT and a sensor.

As already said, sound that is perceptible by humans has frequencies from about 20 Hz to 20,000 Hz. In air at standard temperature and pressure, the corresponding wavelengths of sound waves range from 17 m to 17 mm.

The speed of sound is the distance travelled per unit time by a sound wave as it propagates through an elastic medium. At 20 °C, the speed of sound in air is about 343 metres per second (or 1235 km/h), or a kilometre in 2.9 sec. It depends strongly on temperature, but also varies by several metres per second, depending on which gases exist in the medium through which a soundwave is propagating.

In common everyday speech, speed of sound refers to the speed of sound waves in air. However, the speed of sound varies from substance to substance: sound travels most slowly in gases; it travels faster in liquids; and even faster in solids. For example, sound travels at 343 m/sec in air; it travels at 1480 m/sec in water (i.e., 4.3 times as fast as in air); and at 5120 m/sec in iron (i.e., about 15 times as fast as in air).

In our case we will measure the speed of sound in air using a device, namely an HC-SR04 Ultrasonic distance sensor, that emits and receives/detects ultrasound. Ultrasound is sound waves with frequencies higher than 20,000 Hz (or 20 kHz). Ultrasound is not different from "normal" (audible) sound in its physical properties, except in that humans cannot hear it. So, during our experiments and measurements we will not hear anything! Ultrasound devices operate with frequencies from 20 kHz up to several gigahertz.

The HC-SR04 Ultrasonic distance sensor consists of two ultrasonic transducers. The one acts as a transmitter which converts electrical signal into 40 KHz ultrasonic sound pulses. The receiver listens for the transmitted pulses. If it receives them, it produces an output pulse whose width can be used to determine the distance the pulse travelled. The principle of operation is depicted below.

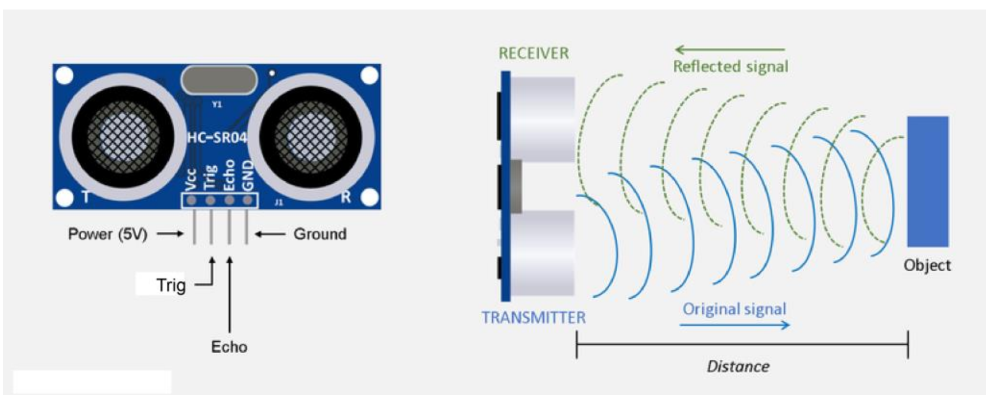

IMAGE 2. *PRINCIPLE OF OPERATION OF THE HC-SR04 ULTRASONIC SENSOR*

Source: http://osoyoo.com/2018/09/18/micro-bit-lesson-using-the-ultrasonic-module/

Thus, we can determine the speed of sound if we can place an object at known distance from the sensor and record the time it takes the ultrasonic pulse to travel this distance twice. Note that twice, because it travels from the transmitter to the obstacle object and then is reflected and it travels back to the receiver to be detected.

Now that we have acquired the basic understanding let’s proceed and start preparing our own scientific investigation.

1.2.2 Preparation

First thing we need to do is to make a circuit and connect our sensor to GPIO pins of our Raspberry Pi. Before we proceed, we turn off our Raspberry Pi and unplug it. For our circuit, we will need our sensor, a breadboard, resistors, jumper wires and a push button. The complete circuit is in the schematic diagram that follows.

The HC-SR04 ultrasonic distance sensor comes with 4 pins: power (VCC), trigger (TRIG), echo (ECHO), and ground (GND). The power pin will be connected to the Raspberry Pi's 5V pin, trigger will be assigned to a GPIO pin as output (pin 4), echo will be assigned to a GPIO pin as input (pin 18), and ground will be connected to a ground pin on the Raspberry Pi.

We also connect a push button to GPIO pin 24 which we will set up as input. The other end of the button should be connected to the 3.3V voltage pin of Raspberry Pi via a resistor. Thus, when we push the button then GPIO pin 24 will be in HIGH state, i.e., in 3.3V. We then need a program to operate in a way that whenever we push the button the sensor emits an ultrasound pulse and the program keeps timing how long it takes to receive the echo back (i.e., how much time passed while the sound waves were emitted, hit the object in front of the sensor, bounced back, and came back to the sensor).

When we are done with our circuit, we can turn on the Raspberry Pi and start our program in Python in file `lesson_plan_speed_of_sound.py`

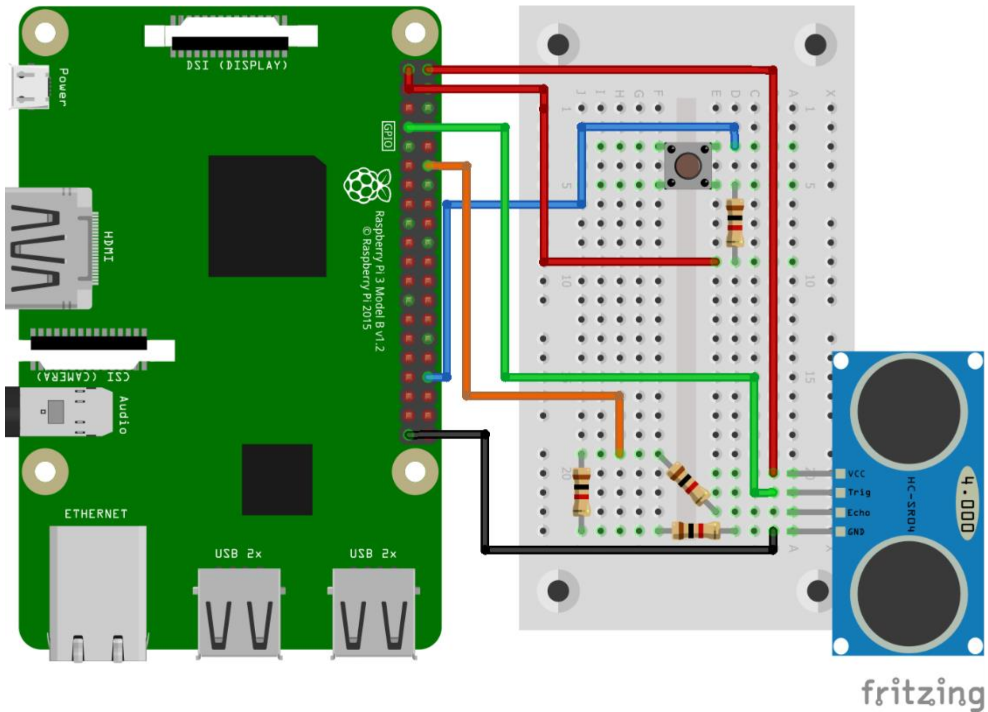

IMAGE 3. SCHEMATIC DIAGRAM OF CIRCUIT WITH DISTANCE SENSOR AND PUSH BUTTON CONNECTED TO GPIO PINS

Source: STEMKIT4Schools project

Before we start our investigation, let’s have a look into the program so we better understand how it operates.

In brief, in our program we first import the Python modules that we need, then we set up the GPIO pins for button and sensor as output or/and input accordingly. Then we put most of the code to interact with the sensor into function `get_time()`. Therein we first emit a burst of ultrasound from the sensor. After that we issue a signal out and then measure the time interval. When we finally do get an input time, we can subtract the end time from the start time and calculate the value of time elapsed.

We call this function in a while loop that runs forever.

At the end we start a while loop that runs forever. In the loop we check if the input pin connected to button is in high state. If yes, then this means that the button is pushed and so we want to call the `get_time()` and print on screen the time value we get from the sensor. Also note that we need to put some sleep time between each call of the function, otherwise the sensor may not behave normally.

Once we have finished with the set of measurements, we can stop the program and the infinite while loop by pressing Ctrl-C.

```
################################################

# first import libraries and set gpio numbering mode
import RPi.GPIO as GPIO
import time
GPIO.setwarnings(False)
# doing this first, since we're using a while True.
GPIO.cleanup()
GPIO.setmode(GPIO.BCM)


# pin for push button
pin_button = 24
GPIO.setup(pin_button, GPIO.IN)


# define TRIG and ECHO pins and set them up as output and input
TRIG = 4
ECHO = 18
GPIO.setup(TRIG,GPIO.OUT)
GPIO.setup(ECHO,GPIO.IN)

# function to get distance value
def get_time():

    # emit a burst of ultrasound
    GPIO.output(TRIG, True)
    time.sleep(0.00001)
    GPIO.output(TRIG, False)

    # measure time interval
    while GPIO.input(ECHO) == False:
        tstart = time.time()

    while GPIO.input(ECHO) == True:
        tend = time.time()

    sig_time = tend-tstart

    #print("time(sec):", sig_time)

    return sig_time


# do this loop forever! Press Ctrl-C to stop it
# When button is pressed then measure time
```

```
while True:
    if GPIO.input(pin_button) == GPIO.HIGH:
        #print("button pushed")
        value = get_time()
        time.sleep(0.05)
        print("time(sec):", value)
        time.sleep(5.)
    else :
        #do nothing

################################################
```

1.2.3 Investigation

Now that we have familiarized ourselves with our experimental apparatus, we can start taking measurements.

Collection of data

We place the obstacle object in front of the sensor and with a ruler or measuring tape we measure the distance according to the following table. We press the button and log in the table the time measurement shown on screen. We repeat for the same distance or different distances.

TABLE 1 DATA LOG TABLE

Index	Distance of object (cm)	Time measured (sec)	Speed = 2*Distance/Time measured (cm/sec)
1	5		
2	5		
3	10		
4	10		
5	20		
6	20		
7	30		
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
…	…	…	…
			Average

Analysis of data

Once the table is filled-up with our measurements of distance and time we can transfer them in an Excel or Open/Libre spreadsheet to determine the speed of sound that corresponds in each pair of measurements. Or alternatively we continue with the paper spreadsheet/table. At the end we compute the average value and express it in cm/sec, in m/sec and in km/hr.

Presentation of results

We present our result and compare it with what other student groups found. We also compare our results with reference values of speed of sound from bibliography (e.g., At 20 °C, the speed of sound in air is about 343 metres per second or 1235 km/h). Did we get close to this reference value? If no, what could be the cause? Do we need to repeat the experimental procedure?

1.2.4 Conclusion

We have succeeded measuring the speed of sound ourselves! In this phase we recapitulate what we did and how, which were the main steps, discuss any difficulties experienced.

1.2.5 Follow-up exercise (optional)

As a follow-up to this lesson plan, we may proceed to the following exercise. Now that we have measured and know the value of the speed of sound let’s revisit the program code we used. What alterations can we make to it so that we can use our apparatus to measure directly the distance of an object?

1.3 References or Resources

Here are some useful references and additional resources related to this lesson plan.

- https://en.wikipedia.org/wiki/Sound
- https://en.wikipedia.org/wiki/Speed_of_sound
- https://randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04/
- http://osoyoo.com/2018/09/18/micro-bit-lesson-using-the-ultrasonic-module/
- https://lastminuteengineers.com/arduino-sr04-ultrasonic-sensor-tutorial/