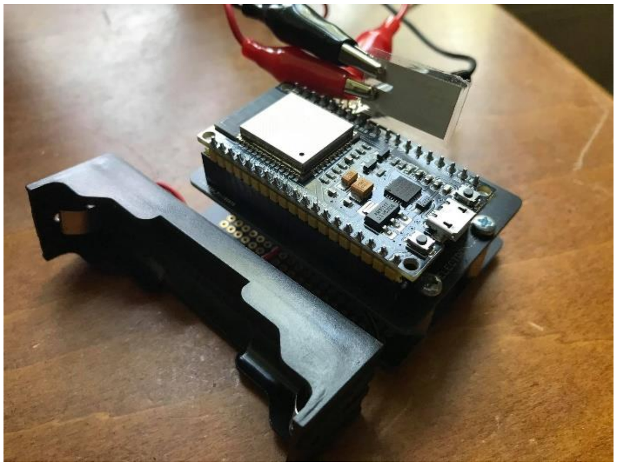
Wireless data transmission from a piezoelectric film sensor using Bluetooth technology



To be presented to

Prof. Sampada Bodkhe

Author:

Michael Flandez

October 21, 2021

Contents

1. Introduction 5

2. Current state 5

2.1 Applied technology 5

2.2 Operation 6

2.3 Bill of materials 6

3. Going wireless 7

3.1 General layout or configuration for a wireless implementation 7

3.2 Sensor 8

3.3 Signal characteristics 8

3.4 Signal conditioning 9

3.4.1 Laplace transform and S 11

3.5 Filter and amplifier 11

3.6 Analog to digital converter 12

3.7 Bluetooth transmission 13

3.8 Reception and plotting 13

3.8.1 MobaXterm 14

3.9 Power 15

4. Solutions 15

4.1 Solution 1: National Instruments Equipment. 15

4.1.1 General layout 15

4.1.2 Bill of materials 16

4.2 Solution 2: Esp32 kit with external ADC. 17

4.2.1 General layout 17

4.2.2 Sensor 18

4.2.3 Signal conditioning 18

4.2.4 Analog to digital converter 18

4.2.5 Bluetooth transmission 22

4.2.6 Data reception and plotting 23

4.2.7 Power 23

4.2.8 Configuration 24

4.2.9 Preparation 26

4.2.10 Operation 26

4.2.11 Logging and plotting 27

4.2.12 Bill of materials 29

4.2.13 Important: 32

4.2.14 Future works 32

4.3 Solution 3: Arduino Nano and HC-05. 34

4.3.1 General Layout 34

4.3.2 Sensor 34

4.3.3 Signal conditioning 34

4.3.4 Analog to digital converter 34

4.3.5 Bluetooth 35

4.3.6 Data reception and plotting 36

4.3.7 Configuration 36

4.3.8 Operation 37

4.3.9 Bill of materials 38

ANNEX 1 40

Esp32 configuration 40

ADS1115 ADC configuration 40

ANNEX 2 .... **Error! Bookmark not defined.**

Datasheets 40

1. Introduction

The following report is a presentation of the research and development undertaken by electrical engineering student Michael Flandez to be presented to professor Sampada Bodhke from the Mechanical Engineering faculty of École Polytechnique de Montréal.

The goal of this work is to make a wireless system for a custom-made sensor reading the pulse off a human. It is desired that the solution uses open source and low-cost technology and materials as well as being easy to use. The characteristics that are of interest are the precision, accuracy of the measurements, the size, and the affordability of the solution.

In the first sections we will discuss the general layout of the solution given to the student, with its bill of materials and costs.

Next, a breakdown of a general solution, presenting Bluetooth technology as the main wireless element, will explain the many different parts that will take part of the full project. Afterwards, a couple of solutions with their costs will be presented, with one of them being fully implemented and experimented on.

Finally, all the appropriate documentation will be presented with the different tutorials and datasheets for all the modules used to deliver the final project.

2. Current state

2.1 Applied technology

The original solution consists of a piezo electric film sensor used to measure the pulse signal of a human. This signal is later conditioned by amplification and filtering to have a more robust signal that can be properly read by the analog digital converter which transforms the signal into readable digital data. This data can afterwards be analysed and interpreted by software.

This original basic configuration or signal flow is represented in the following image.

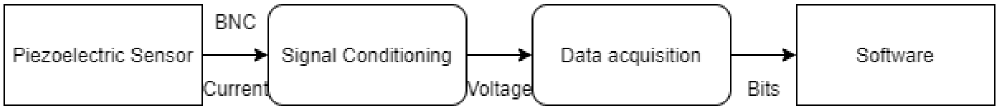


Image 1: Basic configuration for signal measurement.

2.2 Operation

The original configuration makes use of high-end electronic components from the National Instruments company that creates their own hardware and are also creators of the well-known LabVIEW engineering software as well as by TE connectivity. To begin, the commercially made sensors are connected to a BNC cable which leads to the signal conditioning stage where the signal is transformed from a current of electrons into a voltage that will be better suited for the next stage. This signal conditioning unit is made up of a charge amplifier with adjustable low pass and high pass filters. This conditioned voltage moves on to the data acquisition stage which consists of a high quality 24 bit analog to digital converter. Once converted, this data signal is received in LabVIEW where it is plotted in the software and stored as a ".txt" file for later analysis.

2.3 Bill of materials

- Commercially made sensor.
    - 7.95 CDN.
    - https://abra-electronics.com/sensors/sensors-flex-force-en/sen-09196-piezo-vibration-sensor-large-sen-09196.html
    - https://www.sparkfun.com/datasheets/Sensors/Flex/DT_Series.pdf
- BNC 50 Ω cable.
    - 4.95 CDN.
    - https://abra-electronics.com/wire-cable-accessories/wirecable/bnc-cables/bnc-110-mf-bnc-male-to-bnc-female-cable-1m.html
- TE connectivity Piezo Film Lab Pre-Amplifier.

  - 504 USD.
  - https://www.te.com/usa-en/product-CAT-PFS0015.html
- National Instruments Data Acquisition Unit.
  - NI-9218, BNC receiver to be inserted in the chassis. 2250 CDN.
  - NI cDAQ-9171, chassis. 540 CDN
  - https://www.ni.com/fr-ca/shop/hardware/products/compactdaq-chassis.html
  - https://www.ni.com/pdf/manuals/376918a_02.pdf
  - https://www.ni.com/pdf/manuals/374037b.pdf
- National Instrument LabVIEW software.
  - 660 CDN.
  - https://www.ni.com/fr-ca/shop/software/products/labview.html

Total cost floats around $ 4100 CDN. All datasheets are available online at the presented links and are also available in the shared datasheets file package.

3. Going wireless

3.1 General layout or configuration for a wireless implementation

The inclusion of wireless capabilities adds both a transmitter and a receiver to the system as shown in the following image.

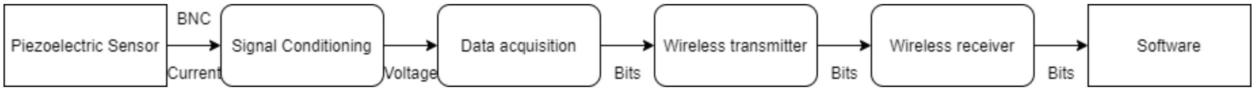


Image 2: Basic configuration including wireless modules for airway data transmission.

Image two shows one possibility of a basic configuration with wireless technology. In this case, after transformation by the data acquisition stage, the digital signal is sent to a wireless transmitter which then sends the data over the air to a paired wireless receiver. This receiver then expediates the data to the software.

The general layout is explained module by module as follows.

3.2 Sensor

The piezo electric sensor consists of a film made from a piezoelectric material. It works as a voltage source that emits a charge under mechanical loading. On each side of the sensor there is a electrode, connected to a different wire. The film also has an electrical resistance that is dependant on the material and on the physical dimensions of the film. The commercially made sensors have a resistance of approximately 5 MΩ.

3.3 Signal characteristics

When the film sensor is fully extended its charge is NULL. Once the sensor is bent, it presents a voltage potential across its terminals that can be measured. This potential depends on the intrinsic characteristics of the sensor's material. If the film is bent in one direction, it will present a positive voltage while if bent in the other direction it will present a negative voltage. Generic off the counter piezo electric sensors have voltage outputs that can fluctuate between -10 and +10 volts.

The TE piezo electric sensor used to in the original solution can be seen in the following image.

Image 3: commercially available piezoelectric film.

On the image we can see that this commercially available sensor has to terminals. Each one is connected to only one side of the film and hence are not in contact. It is possible to connect this sensor directly into the holes of a breadboard or to use alligator clips to attach to each leg.

3.4 Signal conditioning

The signal for this specific application contains a periodic wave contained between 0 Hz and 50 Hz and its amplitude will be measured experimentally to obtain the maximums and minimums. With the commercially available sensor we can easily fold the plastic film to obtain a voltage difference between both leads.

In this second stage of signal conditioning there is a low-pass filter and a voltage amplifier working simultaneously. This stage receives a current signal from the sensor and outputs a voltage signal that is more suitable for the reception of the next stage.

The input voltage is fed into a TLC271 operational amplifier in an integrator configuration. When this configuration receives a voltage as an input, it can amplify it as well as filter out high frequency content. The integrator configuration is presented in the next image and is followed by the mathematical formulations extracted from it.

The negative/inverting input receives the signal coming from the sensor, and the positive input receives the polarisation voltage which is set up using a voltage divider between the main power voltage and ground. This allows to shift the continuous component of the sensors signal from 0 V to the polarisation voltage.

Image 4: TLC271 operational amplifier.

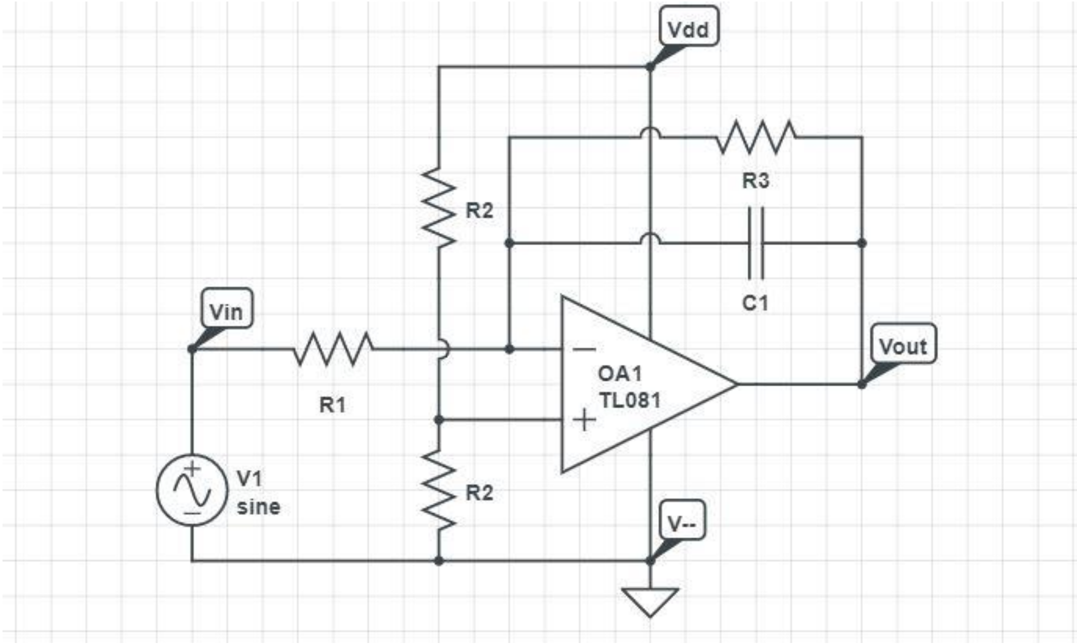


Image 5: Operational amplifier in integrator configuration.

On this image one can see the piezoelectric sensor modeled as a pure sine voltage source. It is connected to resistor R1 and then connected to the negative input on the operational amplifier. On the positive side we can see that if V_{dd} is divided by two because of the two equal resistors.

This can be calculated using the voltage divider equation:

$$V_{div} = V_{nominal} * \frac{-R2_{bottom}}{R2_{top} + R2_{bottom}} = \frac{V_{nominal}}{2} \quad (1)$$

In our case, V_{dd} will be set at V—and will be 0 V.

3.4.1 Laplace transform and S

Each component in an electrical circuit has its own resistance or impedance, a characteristic that opposes the flow of electrons. For resistors, the impedance is said linear and is equivalent to their nominal values, R_1, R_2, etc. For other components such as capacitors and inductors, their impedance is not linear, and it depends on the frequency of operation of the circuit.

$$S = j\omega \quad (2)$$

The operator S is a tool to indicate or help see in which frequency range the component operates, in other words, in what frequencies it allows for the electrons to flow freely or in what frequencies it stops the flow of electrons.

$$Impedance\ of\ C = Z_C = \frac{1}{Cs} \quad (3)$$

Where C is the capacitance value and S is frequency dependent. In this representation, if ω grows, the impedance comes closer to zero, allowing for the electrons to flow and if ω becomes smaller, then the impedance becomes bigger blocking the flow of electrons.

3.5 Filter and amplifier

As mentioned before, this configuration plays a double role of filter and gain. The general equation for this configuration in Laplace form is:

$$V_{out} = V_{in} * \left(\frac{-R_3}{R_1}\right) * \left(\frac{1}{R_f CS + 1}\right) \qquad (4)$$

Here S represents the position of the angle or the frequency of the signal. This mathematical formula tells us that if S is equal to zero, the gain of the configuration is R3 divided by R1. If S goes to infinity, the gain goes to zero. In essence this works as a low pass filter/amplifier as in, only some frequencies and not all are boosted while others are attenuated.

$$f_c = \frac{1}{2\pi C R_3} \qquad (5)$$

With this last formula it is possible to find the cut off frequency for the filter, in other words at which frequency the filter is attenuating the signal by at least 3 dB.

3.6 Analog to digital converter

The ADC or analog to digital converter will sample the amplified signal into digital form. This digital form will later be stored, analysed, plotted, and so on by the software.

ADC’s have limits to what type of signals they can read. Some of them read only positive values between a certain voltage range while others may be able to read both positive and negative values inside a range. Therefore, it is convenient to shift the DC component of the signal to a convenient value as to be able to measure entirely the signal.

Another specification for ADC’s is the sampling rate, or rather, the speed at which each sample is taken. Nyquist’s law tells us that the minimum sampling rate required to properly sample a signal is double the highest frequency present in the signal. For this application where the body pulse is measured, we are expecting a signal between 0 and 50 Hz. Hence the sampling rate must be of at least 100 Hz.

The connection is easily made by connecting the output of the amplification stage and sending it directly into an analog input of the ADC. Almost all low-cost microcontrollers have integrated ADCs.

3.7 Bluetooth transmission

Bluetooth technology is an accessible technology that allows for wireless transmission of information. It is a low-cost protocol and allows for data transmission at various speeds over short distances. What becomes most interesting for this solution is that the receiving device (computer, cell phone) can pair up with the Bluetooth transmitter without having to sacrifice connectivity to a ‘WiFi’ source. This means that we can be receiving data through the Bluetooth connection while interacting with other networks like the internet, allowing for further analysis or manipulation of devices.

To make use of this technology we can either use a microcontroller or module that has it already included or add another additional module that has it.

3.8 Reception and plotting

Most modern computer and cell phone have Bluetooth technology (transmission and reception) integrated at purchase. The host computer will be in the role of master or server and the transmitter will be in the role of slave or client. Once the devices are paired in the Bluetooth configurations, they will be open to bi-directional communication. If the proper drivers are installed, Bluetooth will open a serial COM port that will receive the packages of data coming in. This serial COM connection port can be accessed by almost any terminal or application that allows for port connections. All computers have this type of communication integrated.

On windows, it is possible to access this in the device manager under ‘ports’. If ever a driver is missing, it is possible to quickly download or update by right-clicking on the port and selecting update driver.

The information being sent by the transmitter is in the ASCII format which is a standard format making use of letters or numbers. The receiver can hold any number of applications that can read this data. It will be up to the functionalities of the local application to do as wished, for example, recording the data, plotting the data, analysing the data in real time or offline, etc.

If the computer doesn't support Bluetooth, it is possible to purchase a USB to Bluetooth Dongle like the CSR4.0 on the following image. It will be necessary to download the appropriate driver which will enable serial communication between the incoming data through the Bluetooth dongle and the computer.

Image 6: CSR4.0 USB Bluetooth dongle.

3.8.1 MobaXterm

Any software that can access the serial ports from the computer will be able to read the serial input and record it in any of many formats. One of them, MobaXterm allows for the recording of logged data in '.txt' form, which can be later analysed or plotted in other software such as Excel by Microsoft.

3.9 Power

Given the wireless nature of these technologies, the following solutions should be able to work with batteries. A rechargeable battery is used to power these solutions. The battery itself when fully charged floats a certain voltage level and diminishes as the charge is consumed. Because of this behaviour (as with all batteries), the use of a power regulator becomes necessary as it will assure a very stable voltage output assuring proper functioning of all the modules. It is also necessary that these rechargeable batteries last a couple of hours when in operation.

Image 7: LGR1865 rechargeable lithium-ion battery.

4. Solutions

Three different solutions are studied, the first one using high end, ready to use equipment from National Instruments, the second making use of low-cost electronics and a Esp32 microcontroller kit and finally another one also using low-cost Arduino technology.

Each solution presents the general layout, electrical schematics, and bill of materials for easy assembly.

4.1 Solution 1: National Instruments Equipment.

4.1.1 General layout

We will not go in deep in explaining this solution as it is very much the same as the original solution except for the signal conditioning unit is connected to a wireless transmitter.

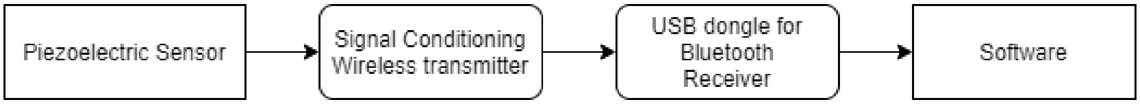

Piezoelectric Sensor
Signal Conditioning Wireless transmitter
USB dongle for Bluetooth Receiver
Software

Image 8: Layout of the National Instrument wireless solution.

In this solution the tools selected are from the National Instruments hardware and software catalog. To begin with, we have the sensor connected by a BNC cable to a portable signal conditioning unit that is itself connected to a wireless transmitter. The signal is received by a dongle receiver connected to the computer and used by the LabVIEW software. Both the signal conditioner and wireless (Wi-Fi) system are portable (battery powered) and made by National Instruments and of very high cost. This solution counts with state-of-the-art technology and years and years of experience from the development team at National Instruments.

4.1.2 Bill of materials

- Commercially made sensor.
    - 7.95 CDN.
    - https://abra-electronics.com/sensors/sensors-flex-force-en/sen-09196-piezo-vibration-sensor-large-sen-09196.html
    - https://www.sparkfun.com/datasheets/Sensors/Flex/DT_Series.pdf
- BNC 50 Ω cable.
    - 4.95 CDN.
    - https://abra-electronics.com/wire-cable-accessories/wirecable/bnc-cables/bnc-110-mf-bnc-male-to-bnc-female-cable-1m.html
- TE connectivity Piezo Film Lab Pre-Amplifier.
    - 504 USD.
    - https://www.te.com/usa-en/product-CAT-PFS0015.html
- National Instruments Data Acquisition Unit.
    - NI-9218, BNC receiver to be inserted in the chassis. 2250 CDN.
    - NI cDAQ-9191, chassis with WI-FI capabilities. 750 CDN
    - https://www.ni.com/fr-ca/shop/hardware/products/compactdaq-chassis.html

    - https://www.ni.com/pdf/manuals/376918a_02.pdf
    - https://www.ni.com/pdf/manuals/374048c.pdf
- National Instrument LabVIEW software.
    - 660 CDN.
    - https://www.ni.com/fr-ca/shop/software/products/labview.html

Total cost floats around $ 4300 CDN. All datasheets are available online at the presented links and are also available in the shared datasheets file package.

What refrains us from this solution is its extremely high cost.

4.2 Solution 2: Esp32 kit with external ADC.

4.2.1 General layout

Solution 2 is built around a Esp32 microcontroller mounted on a DEV DOIT KIT which holds the Esp32 chip itself and has pins or legs to access all analog and digital inputs and outputs. The chip includes Wi-fi and Bluetooth technology which can be used thanks to the provided Esp32 libraries.

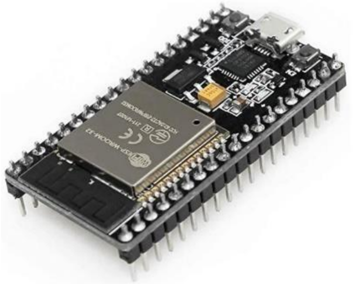


Image 9: Esp32 Wroom Kit.

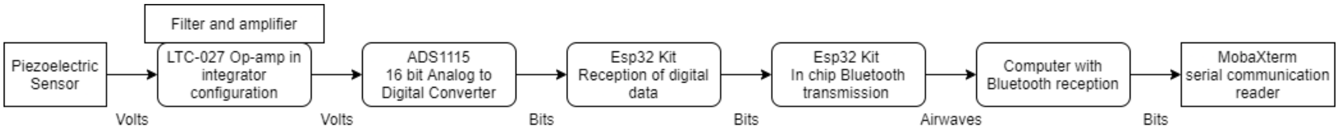

Filter and amplifier
Piezoelectric Sensor
LTC-027 Op-amp in integrator configuration
ADS1115 16 bit Analog to Digital Converter
Esp32 Kit Reception of digital data
Esp32 Kit In chip Bluetooth transmission
Computer with Bluetooth reception
MobaXterm serial communication reader
Volts
Volts
Bits
Bits
Airwaves
Bits

Image 10: General layout of the Esp32 Kit solution.

The piezoelectric sensor produces a signal that is amplified and filtered simultaneously by the TLC271 integrator configuration. Next the ADS1115 converter converts the analog signal into digital data that is sent directly into the Esp32 kit which forwards it to the integrated Bluetooth on the chip. Finally, the transmitted signal is read through serial communication by MobaXterm inside the computer as it receives the data.

4.2.2 Sensor

Either the commercial sensor or the attachable custom made one. Each sensor has its own electrical resistance. The commercial one is around 5 MΩ while the custom made one is around 1 MΩ.

4.2.3 Signal conditioning

Operational amplifier in the integrator configuration. A resistor is added to the output of the sensor and applied to the negative/inverting input of the operational amplifier. The positive input will hold the polarisation tension, the equivalent of 2.5 V. This signal will be suitable for the next part.

4.2.4 Analog to digital converter

The Esp32 microcontroller has a 12-bit ADC which can receive signals between 0 V and 3.3 V. 12 bits means that we can measure values between 0 and 4095. This seems more interesting than the Arduino ADCs because it will allow for a better precision. For example, if we take a 5 V signal and read it with a 10-bit converter, it will be possible to read 2^10 values which are 1024 different values from 0 V to 5 V.

$$\frac{5V}{1024} = 4.8\, mV\; precision$$

So, the smallest difference that we can find between one measured value and the next is 4.8 mV. Now with a 12-bit converter, we will have 2^12 values which are 4096 values.

$$\frac{5V}{4096} = 1.2\, mV\ precision$$

We see an improvement in the value of precision that we can measure. The problem with the Esp32 is that it is known for having very inaccurate ADC's which renders it useless for this type of application.

To remedy this situation, and external ADC is added to the configuration. The piece added is a the Adafruit ADS1115 which allows for up to 4 channels at 16 bits of bit depth. The first bit is used for the sign of the input signal, which leaves 15 bits of bit depth. This will give a higher precision because of the 32768 possible measurement values as well as better accuracy.

$$\frac{5V}{32768} = 0.15\, mV\ precision$$

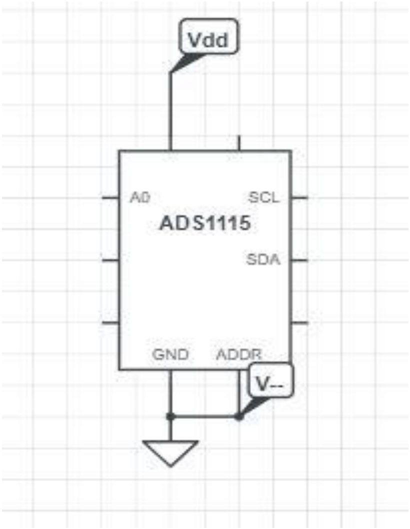


Image 11: Electrical layout for the ADS1115.

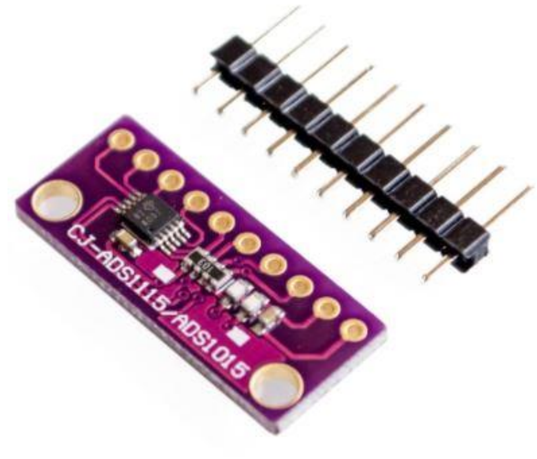
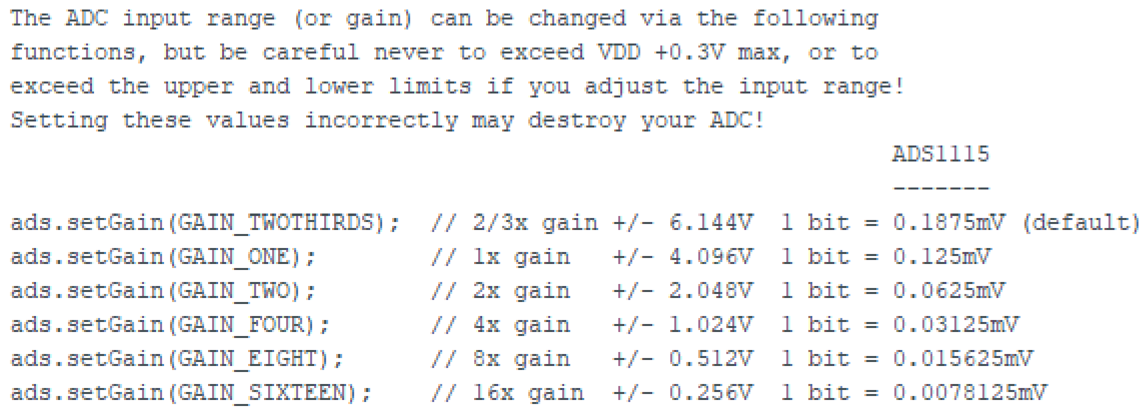


Image 12: ADS1115 module.

The ADS1115 also features an internal amplifier which allows to amplify the signal even more. One thing to consider is that the measurable voltage range gets smaller as the gain is raised and vice versa while the precision becomes higher.

The following image explains the different settings:



```
The ADC input range (or gain) can be changed via the following
functions, but be careful never to exceed VDD +0.3V max, or to
exceed the upper and lower limits if you adjust the input range!
Setting these values incorrectly may destroy your ADC!
                                                               ADS1115
                                                               -------
ads.setGain(GAIN_TWOTHIRDS);  // 2/3x gain +/- 6.144V  1 bit = 0.1875mV (default)
ads.setGain(GAIN_ONE);        // 1x gain   +/- 4.096V  1 bit = 0.125mV
ads.setGain(GAIN_TWO);        // 2x gain   +/- 2.048V  1 bit = 0.0625mV
ads.setGain(GAIN_FOUR);       // 4x gain   +/- 1.024V  1 bit = 0.03125mV
ads.setGain(GAIN_EIGHT);      // 8x gain   +/- 0.512V  1 bit = 0.015625mV
ads.setGain(GAIN_SIXTEEN);    // 16x gain  +/- 0.256V  1 bit = 0.0078125mV
```

Image 13: Programming of internal gain on the ADS1115.

On default mode, the gain is set to two thirds, allowing an input between 0 and 6.144 V with each bit representing 0.1875 mV. On gain equal to one, the maximum input is 4.096 V giving 0.125mV per bit. On the last setting of gain equal to sixteen, the maximum input of voltage is 0.256 V with a precision of 0.0078125 mV per bit.

This converter receives the signal of interest on one of its inputs and sends the digital information directly to the Esp32 kit, completely skipping the ADCs of the Esp32.

The A0 pin on the ADS1115 is to receive the signal coming from the signal conditioning stage. The SCL and SDA pins from the ADS1115 will be connected to GPIO 22 and GPIO 23 on the Esp32 kit respectively. The ADDR pin will be connected to GND in this case.

If ever it is desired to change the internal gain through Arduino, it may become necessary to change the polarisation voltage on the positive / noninverting input of the operational amplifier as to offset the DC voltage and have the full representation of the signal on the analog input. It is possible to change the polarisation voltage by changing the values of the resistors in the voltage divider/noninverting input of the TLC271 operational amplifier.

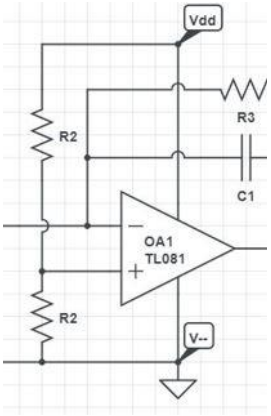


Image 14: Operational amplifier with a voltage divider between V_{dd} and V_{--}.

This image shows the voltage divider between the V_{dd} and V–nodes using R_2 as resistors. In this application, the node that will be entering the (+) noninverting input of the TLC271 will have a value of 2.5 V. This will be the offset voltage.

Now if desired, by following the voltage divider formula (1) in section 3.4, a different polarisation voltage can be enabled.

For example, if we want to have a polarisation of 1.5 V, we can pick two resistors that would fit the formula with the top resistor being 10 kΩ. We would then need a bottom 4.286 kΩ resistor.

$$1.5 = 5 * \frac{R2_{bottom}}{10k\Omega + R2_{bottom}}$$

$$4.286\ k\Omega = \ R2_{bottom}$$

Section 4.2.14 shows a table containing several resistor combinations values to match the internal gain of the Arduino program.

4.2.5 Bluetooth transmission

The Arduino program that is loaded on the Esp32 (ANNEX 2) reads the data coming from the ADS1115 thanks to a Adafruit proprietary library which must be downloaded using the Arduino IDE library manager. Every library is a collection of specific program functions or routines that are custom made for the hardware that the main program can make use of. The data is read by the program and makes use of the integrated Bluetooth technology to transmit over the air. This information is sent as serial communication over the air.

For every microcontroller kit (Esp32), as well as programmable add-ons like the ADS1115 one must download the corresponding libraries for its correct functioning. All these are done through the Arduino IDE.

This data will be streamed continuously which comes with an expense of more power. Around 100 mA of current will be drawn from the batteries for proper operation. In this type of operation, we can expect around 11 hours of operation.

4.2.6 Data reception and plotting

The Arduino program that is loaded will print onto the Bluetooth serial a line with three members. First time in milliseconds, second a comma and third the voltage value measured. These will be read in a line in MobaXterm in a column which can be saved and exported as a '.txt' file. This software can be downloaded for free while a paid version is also available.

4.2.7 Power

The Esp32 kit works at 3.3 V although it can receive 5 V at a certain pin where it regulates it down to 3.3 V. The operational amplifier can operate from 3 V and up, although it is preferable to have it working at 5 V for uniformity of microcontrollers in general. For the ADS1115 ADC it is preferable to have it connected to 5 V as to allow bigger voltages at its inputs.

The battery is placed in a battery holder that is connected to a power regulator. The power regulator used is the U1V10F5 5V tension regulator. It can accept voltages ranging from 1 V up to 5.5 V at its input, delivering a constant 5 V at the output, with up to 1 A and 4% precision.

Image 15: U1V10F5 5V power regulator.

The battery will provide up to 4.2 V (when fully charged) to the power regulator, which will then provide 5 V to the Esp32, the operational amplifier, the ADS115 ADC and to the sensor.

4.2.8 Configuration

The general layout is as follows:

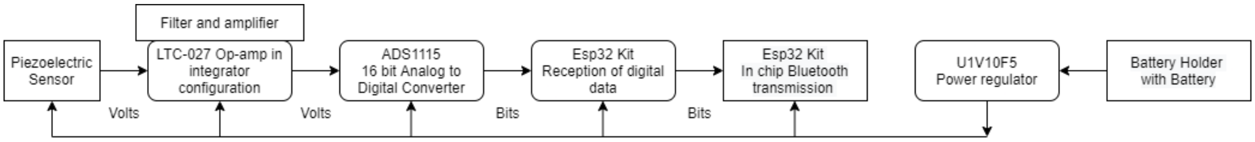

Filter and amplifier
Piezoelectric Sensor
LTC-027 Op-amp in integrator configuration
ADS1115 16 bit Analog to Digital Converter
Esp32 Kit Reception of digital data
Esp32 Kit In chip Bluetooth transmission
U1V10F5 Power regulator
Battery Holder with Battery
Volts
Volts
Bits
Bits

Image 16: General layout with components as blocks.

The general electrical layout is shown here:

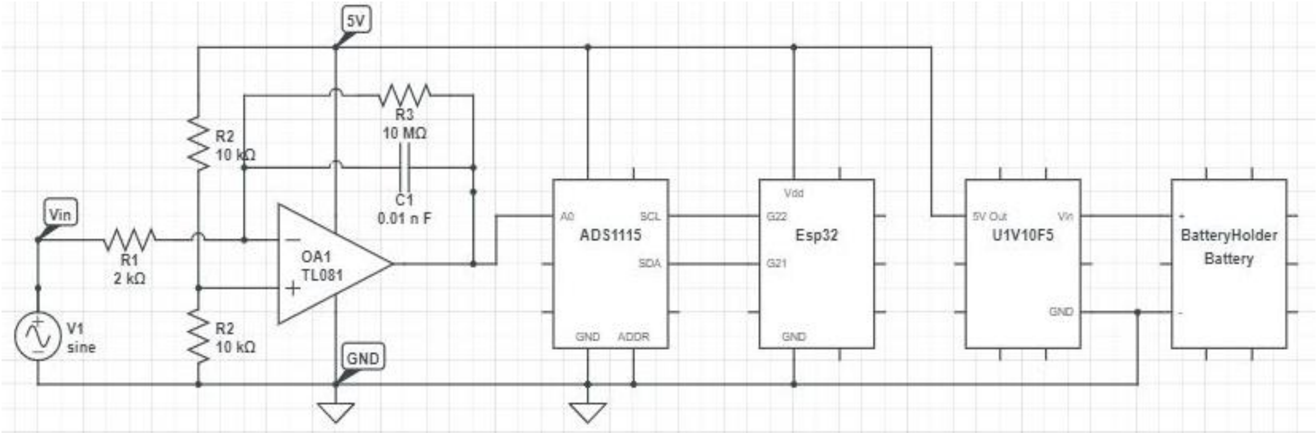


Image 17: Complete electrical layout with all components.

From left to right we see the sensor modeled as a voltage source, 2 equal value resistors which polarise the operational amplifier, the operational amplifier in the integrator low pass filter/amplifier configuration, the external ADS1115 ADC, the Esp32 Kit (with Bluetooth included). To the right of these modules are the power regulator and battery holder/battery.

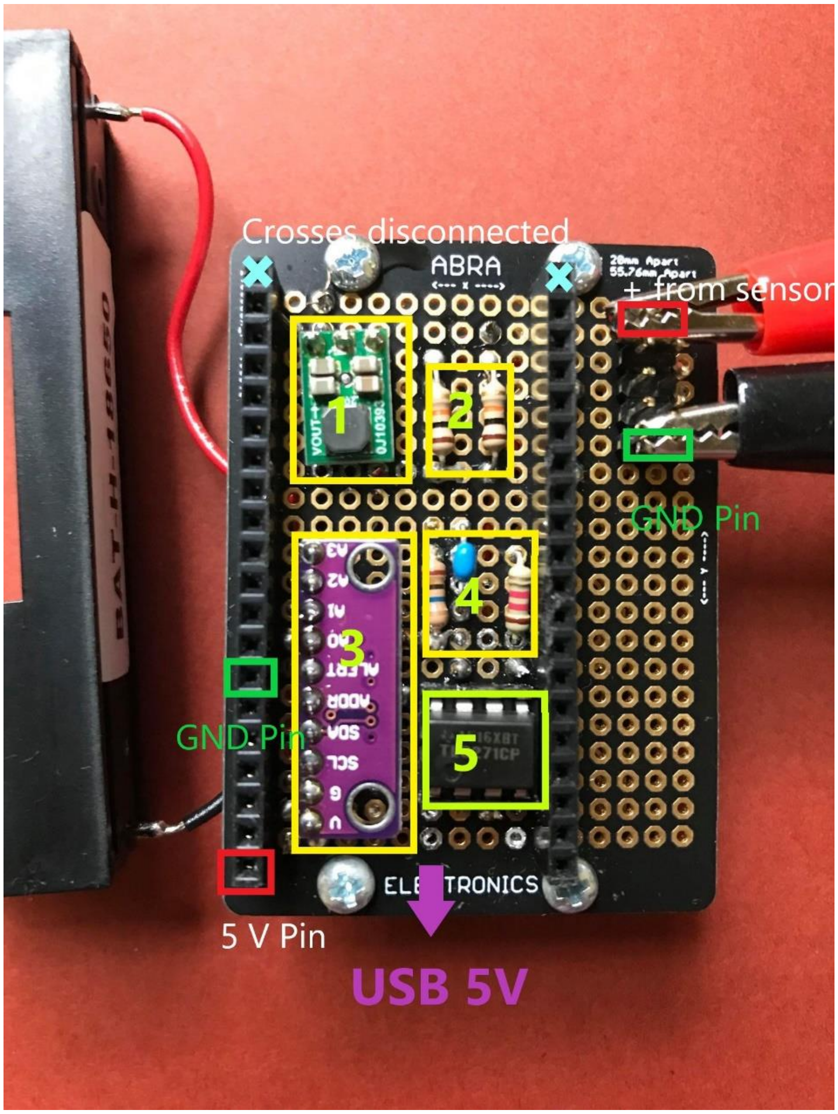


Image 18: Image of components placed and soldered on the protoboard.

On this image, we can see all the components placed on the protoboard with exception of the Esp32 microcontroller, which is to be placed on top, inserted into the female headers. We first see two 20-hole female headers running vertically. The Esp32 only has 19 pins per side, so it is to be placed as to have the USB connector facing the word 'electronics' and having the 'V5' pin inserted into the hole with the red square around named 5 V pin. The two holes with crosses over them are not to be used. It is important to properly connect the microcontroller correctly. On the top right side, we see a 2 x 5 male header. The top 2 pins are normalized to 5V on the bottom side of the protoboard and should be connected to one of the sides of the sensor. The bottom 2 pins are normalized to the inverting input of the op amp on the bottom side of the protoboard and should be connected to the other side of the sensor.

Next, we see 5 different object boxes. Box 1 is the power regulator, box 2 is has the two resistors providing the DC offset. Box 3 is the ADS1115 ADC. Box 4 has the feedback resistor and capacitor on the left and the input resistor on the right. Box 5 has the TLC027 operational amplifier.

4.2.9 Preparation

Once all the pieces have been attached and soldered, we must program the Esp32 Kit with the following program: *Arduino_program_for_Esp32*. The code for the program is presented in the the datasheets package as a PDF file and as an *Arduino INO* file. First, we must make sure that the appropriate libraries are installed. Here we are speaking of the <Wire.h>, <Adafruit_ADS1115.h> and "BluetoothSerial.h" libraries. To include these, inside the Arduino IDE, it is necessary to configure the Esp32 Kit as described in the tutorial presented at the first link in the ANNEX 1 as well as the ADS1115 ADC presented at the second link in ANNEX 1.

4.2.10 Operation

To begin with, we must power the Esp32 kit and all the other modules. This is done by either plugging in the USB connector on the Esp32 kit to a computer or installing the rechargeable Lithium-Ion Battery but <u>*never both at the same time, as this will overcharge the microcontroller and burn it.*</u>

Next it is necessary to pair the module with the computer in the Bluetooth configuration options. Once the pairing is complete, we can find which COM serial port is being used by the Bluetooth connection, this is found in the configuration panel in Windows. There one must look at the External Devices and choose the corresponding one to the Esp32 module. In its settings menu, under the material tab we can find the COM port number.

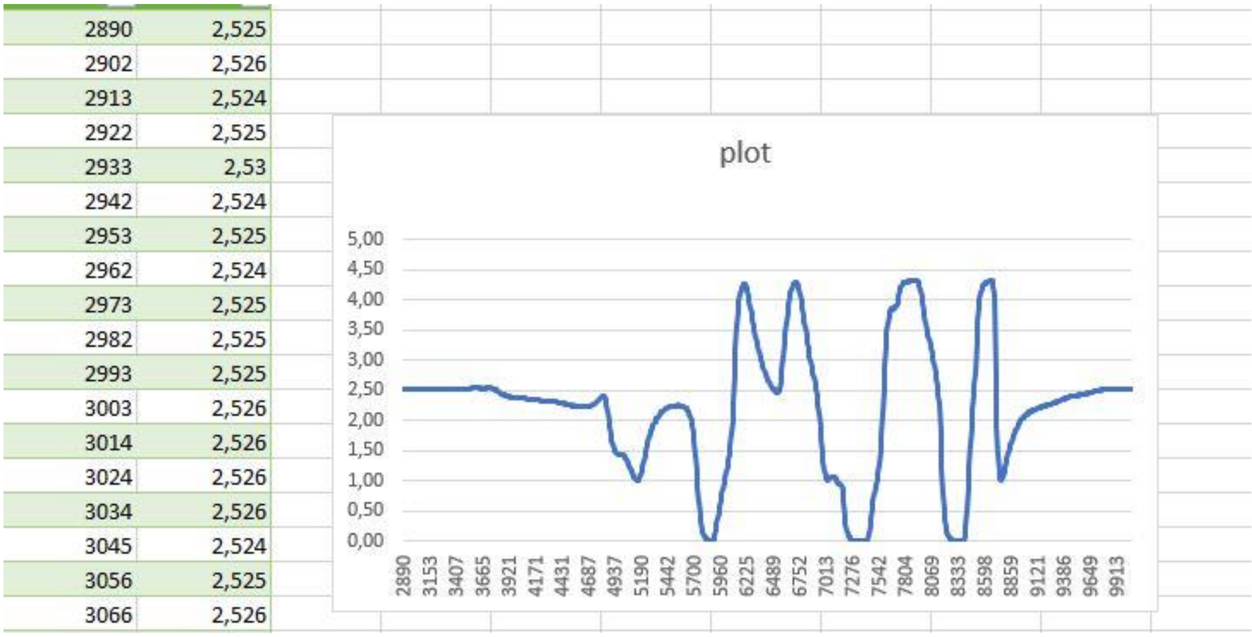
This COM port number will be used to configure the terminal in MobaXterm. To do this, we open MobaXterm and go into the session's menu at the top, there we pick New Sessions. In the new pop-up window, we select Serial from all the options. Once in Serial, under serial port we find the corresponding COM port number and we select the correct baud rate which is 115200 bps. There we press ‘ok’, and the session should be open and ready to read the incoming data.

The session will be saved and accessible by the left side window next time we open MobaXterm.

4.2.11 Logging and plotting

Since the data is being read continuously in real time, the counter will be set to zero only when the reset button on the microcontroller is pressed. Once the counter has been reset, we can open a serial port terminal in MobaXterm for the corresponding COM port. The appearing terminal will then display the data, line by line at high speed. Next we press reset to stop the logging of the data. We can now right click on the terminal and select ‘save to file’, to save as a ‘.txt’ file in a convenient place on the computer.

Now we can open this data into Microsoft Excel to plot.



2890	2,525
2902	2,526
2913	2,524
2922	2,525
2933	2,53
2942	2,524
2953	2,525
2962	2,524
2973	2,525
2982	2,525
2993	2,525
3003	2,526
3014	2,526
3024	2,526
3034	2,526
3045	2,524
3056	2,525
3066	2,526

Image 19: An example of a plot from the incoming information received on MobaXterm, using Microsoft Excel. On the vertical axis is the measured voltage while on the horizontal axis is the time in milliseconds.

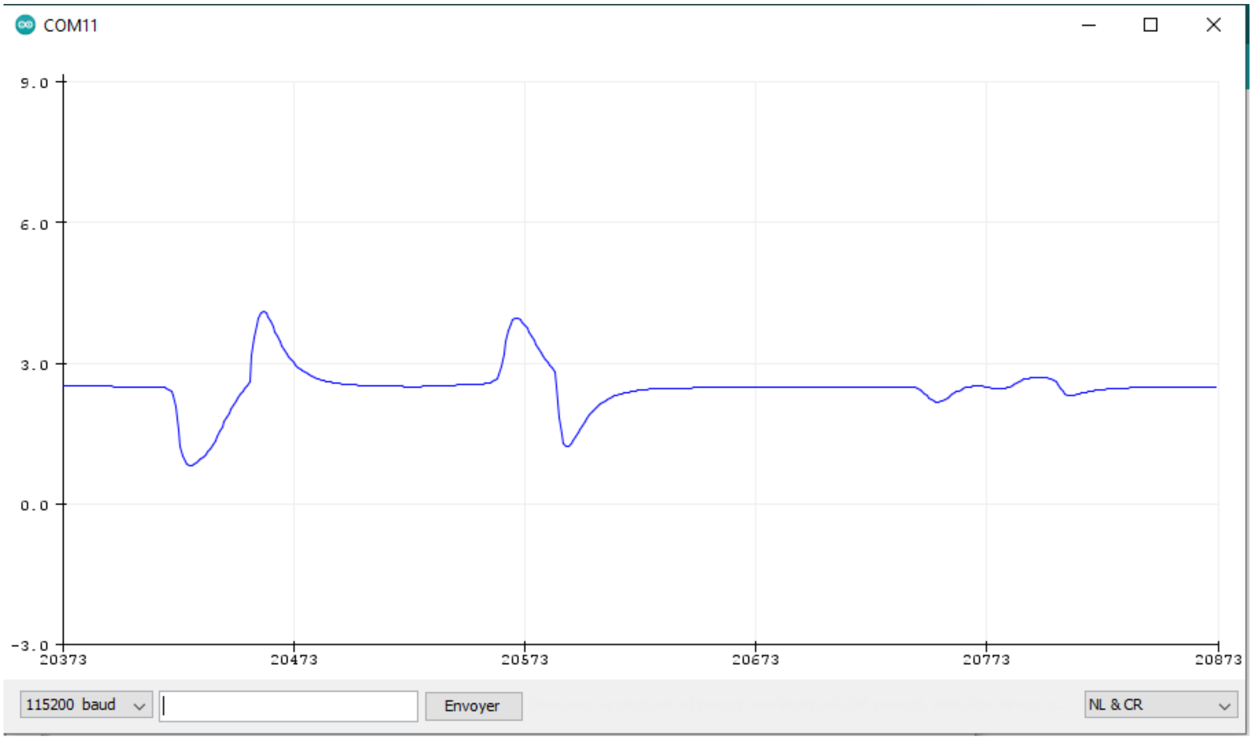
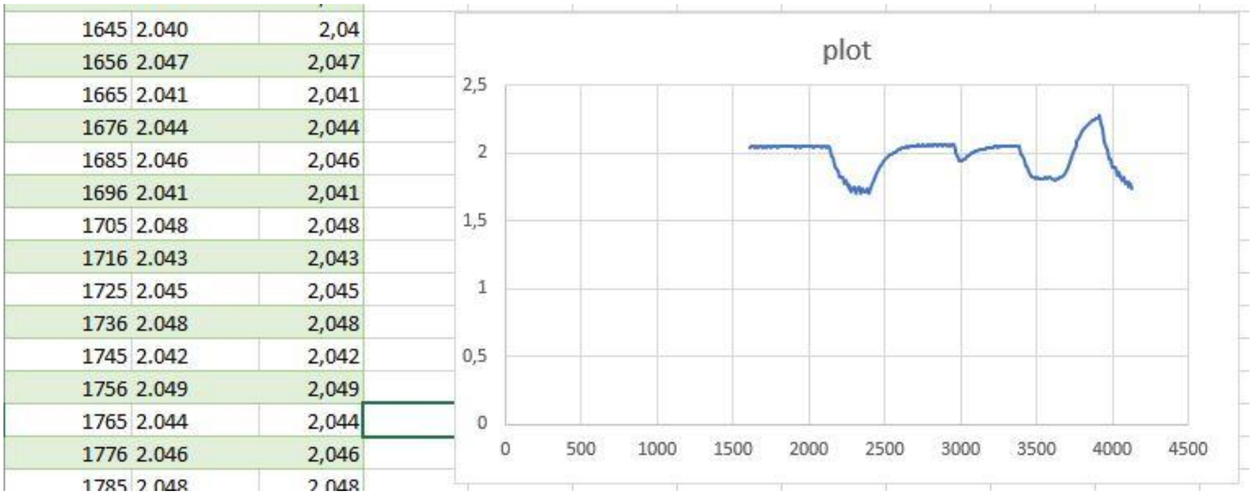


Image 20 : An example of a plot from the incoming information received using the Arduino IDE. On the vertical axis is the measured voltage while on the horizontal axis is the time in milliseconds.



1645 2.040	2,04
1656 2.047	2,047
1665 2.041	2,041
1676 2.044	2,044
1685 2.046	2,046
1696 2.041	2,041
1705 2.048	2,048
1716 2.043	2,043
1725 2.045	2,045
1736 2.048	2,048
1745 2.042	2,042
1756 2.049	2,049
1765 2.044	2,044
1776 2.046	2,046
1785 2.048	2.048

Image 21: An example of a plot from the incoming information received on MobaXterm, using Microsoft Excel. On the vertical axis is the measured voltage while on the horizontal axis is the time in milliseconds.

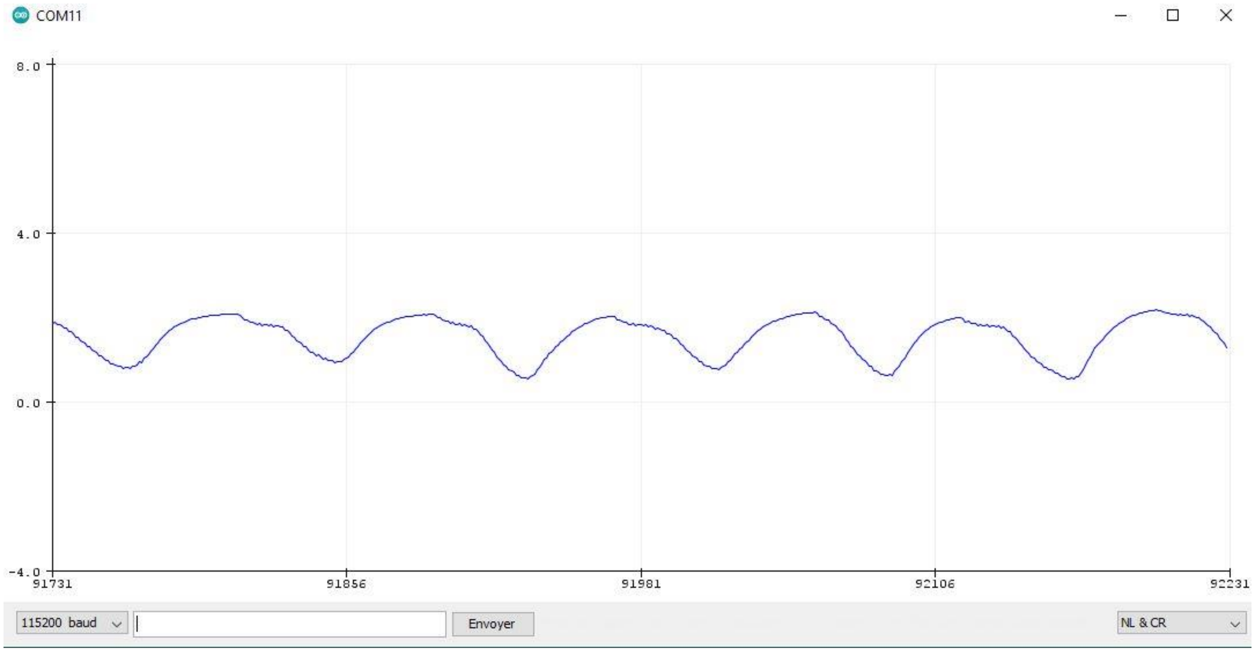


Image 22 : An example of a plot from the incoming information received using the Arduino IDE. On the vertical axis is the measured voltage while on the horizontal axis is the time in milliseconds.

4.2.12 Bill of materials

- Commercially made sensor.
  - 7.95 CDN.
  - https://abra-electronics.com/sensors/sensors-flex-force-en/sen-09196-piezo-vibration-sensor-large-sen-09196.html
  - https://www.sparkfun.com/datasheets/Sensors/Flex/DT_Series.pdf (Datasheet is the same as Solution 1)
- Esp32 Dev Kit.
  - 18.21 CDN.
  - https://abra-electronics.com/wireless/wireless-wifi-en/wifi-esp32-esp-wroom-32-wi-fi-bluetooth-ble-low-power-iot-microcontroller.html

    - https://www.espressif.com/en/products/socs/esp32
- TLC271 operational amplifier.
    - 1.22 CDN.
    - https://abra-electronics.com/ics-semiconductors/linear-series/tlc271cp-op-amp-1.7mhz-3.6v-us-dip-8.html
    - https://www.ti.com/lit/ds/symlink/tlc271.pdf?ts=1628865005528&ref_url=https%253A%252F%252Fwww.google.com%252F
- ADS1115 16-bit ADC.
    - 10.76 CDN.
    - https://abra-electronics.com/robotics-embedded-electronics/raspberry-pi-en-3/modules-en-2/am-139-ads1115-16-bit-precision-analog-to-digital-converter-i2c-breakout-board.html
    - https://www.best-microcontroller-projects.com/support-files/ads1115.pdf
- U1V10F5 5V power regulator.
    - 6.91 CDN.
    - https://www.pololu.com/product/2564
- Alligator cables.
    - 6.09 CDN.
    - https://abra-electronics.com/test-equipment-tools/test-clips-leads-banana-posts/generic-low-cost/tl-600-alligator-clip-with-pigtail-4-pack.html
- Resistors and capacitor.
    - 2 x 10 kΩ.
    - 1 x 10 MΩ.
    - 1 x 0.01 uF.
    - 1 x 2 kΩ. Depending on the impedance of the sensor.
- Battery.
    - 8.95 CDN
    - BAT-18650-T rechargeable 3.7 V, 1350 mAh, lithium-ion battery.

  - https://abra-electronics.com/batteries-holders/batteries-polymer-lithium-ion/bat-18650-t-3.7v-rechargeable-lithium-ion-cell-1350mah.html
  - 2.19 CDN
  - BAT-H 18650 battery cell.
  - https://abra-electronics.com/batteries-holders/battery-holder-18650-cells/bat-h-18650-18650-battery-holder-single-cell.html
  - 15.32 CDN.
  - BAT-CHARGER for 18650 batteries.
  - https://abra-electronics.com/batteries-holders/battery-chargers-testers/bat-charger-charger-for-18650-3.7v-li-ion-rechargeable-battery.html
- Others.
  - 3.07 CAD.
  - 2 x Prototyping perf board.
  - https://abra-electronics.com/boards/printed-circuit-boards/brd-abra-320s-single-gold-plated-protoboard.html
  - Female headers, 20 x 1 holes.
  - Male headers, 2 x 5 pin.
- Arduino IDE & librairies.
  - Open source.
  - https://www.arduino.cc/en/software
- MobaXterm.
  - Free and payable versions.
  - https://mobaxterm.mobatek.net/.
- Microsoft Excel.

Total cost floating at around $ 87 CDN. All datasheets are available online at the presented links and are also available in the shared datasheets file package.

4.2.13 Important:

To make sure that the Esp32 kit is properly placed on the female headers, the USB connector on the Esp32 kit must be facing the side of the perf board that says electronics. Also, the ‘V5’ pin on the Esp32 kit must be inserted into the corner most hole on the female header, on the same side where the word ‘electronic’ is written on the perf board.

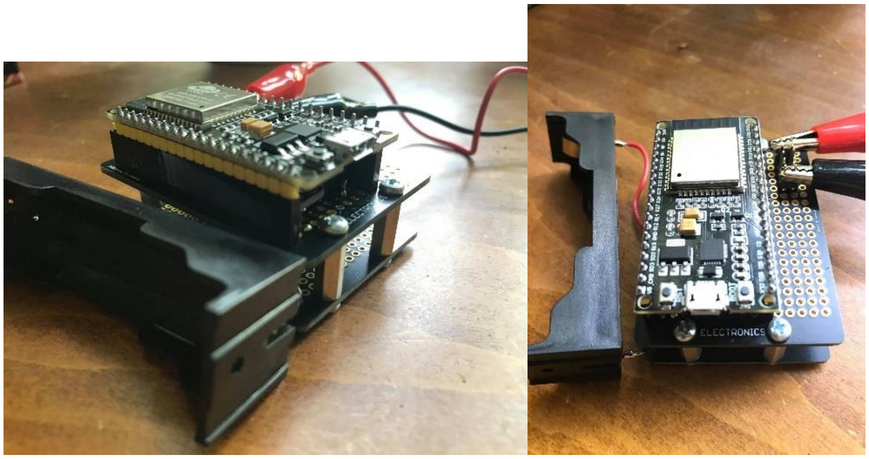


Image 23: Completed module with the Esp32 kit properly placed.

Never connect the USB cable to the Esp32 while the battery is connected. Never connect the battery to the Esp32 while the microcontroller is connected to the USB cable. Never connect the battery to the module if the Esp32 kit isn’t installed on the module. Make sure that the polarity of the battery is the correct one.

4.2.14 Future works

As explained previously in sections 3.4 (signal conditioning) and 4.2.4 (analog to digital converter), the DC offset or polarisation voltage of the noninverting input for the TLC027 operational amplifier must be adjusted to properly match the internal gain of the ADS1115 ADC that is to be programmed into the Arduino program. This is to be done once we have a better understanding of the sensitivity of the sensor being used.

Table 1: Table showing all the possible combinations.

Resistor 1 (Ω) Top	Resistor 2 (Ω) Bottom	DC offset (V)	Internal Gain (x)
10 k	10 k	5	2/3
10 k	6.66 k	2	1
10 k	2.5 k	1	2
10 k	1.11 k	0.5	4
10 k	530	0.256	8
10 k	260	0.128	16

The first column has the value of the resistor connected between V_{dd} and the noninverting input of the Op Amp. The second column holds the values for the resistor between the noninverting input of the Op Amp and GND. Third column shows the DC voltage present at the noninverting input and the fourth column indicates which gain setting must be uncommented in the Arduino program.

A few other possible modification would be to find a smaller rechargeable battery and battery holder to the project. This would depend on the final real life use of the full system because the system can operate upto 10 hours with one charge.

Another future adaptation is to add a potentiometer (adjustable resistor) at the non inverting input of the operational amplifier, to facilitate the DC offset change.

And finally, it could also be handy to include a power switch instead of having to disconnect the battery.

4.3 Solution 3: Arduino Nano and HC-05.

4.3.1 General Layout

This solution is focused on open-source technology based on the Arduino Nano. The following image outlines the general configuration.

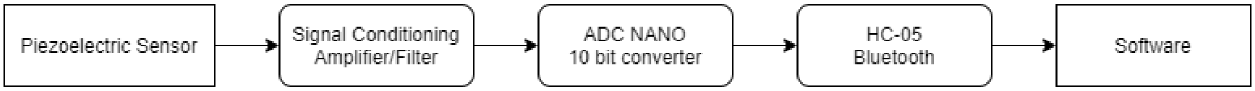

Piezoelectric Sensor
Signal Conditioning Amplifier/Filter
ADC NANO 10 bit converter
HC-05 Bluetooth
Software

Image 24: Configuration for an Arduino based solution.

4.3.2 Sensor

Either the commercial sensor or the attachable custom made one. Each sensor has its own electrical resistance. The commercial one is around 5 MΩ while the custom made one is around 1 MΩ.

4.3.3 Signal conditioning

Operational amplifier in the integrator configuration. A resistor is added to the output of the sensor and applied to the negative/inverting input of the operational amplifier. The positive input will hold the polarisation tension, the equivalent of 2.5 V. This signal will be suitable for the next part.

4.3.4 Analog to digital converter

For this specific solution, an Arduino NANO microcontroller is used. The Arduino NANO allows the implementation of applications programed in the Arduino Integrated Development Environment and hosts a series of analog and digital inputs and digital outputs.

Arduino NANO boards hold 10-bit ADC's which allow for 1024 values between 0 and 1023. Another important fact is that the voltage range that these ADC's can take is from 0 to 5 volts. It is then most interesting to have the output of the signal conditioning stage in this range.

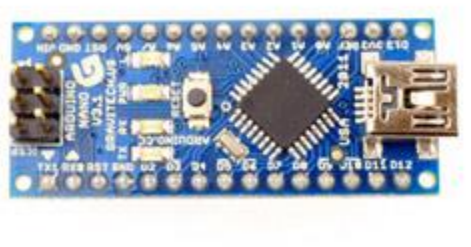


Image 25: Arduino NANO 3.1

4.3.5 Bluetooth

The Arduino NANO microcontroller does not have an integrated Bluetooth, but it is possible to add a Bluetooth module by serial connection. This means that by writing to the serial connection on the NANO, the data will be transmitted to the Bluetooth module and sent through the air. The module chosen is the HC-05 as it is low cost, easily configurable, with different transmission speeds and can work in both master and slave roles. The Arduino IDE has functions allowing to write or read to the serial connection. This serial connection is normalised to the TX and RX pins from the Arduino. These pins will be connected to the TX and RX pins on the HC-05 to form a bi-directional connection, enabling communication from the Arduino NANO/HC-05 to the Bluetooth receiver and vice versa.

Image 26: HC-05 Bluetooth Module

4.3.6 Data reception and plotting

The Arduino program that is loaded will print onto the Bluetooth serial a line with three members. First time in milliseconds, second a comma and third the voltage value measured. These will be read in a line in MobaXterm in a column which can be saved and exported as a '.txt' file.

4.3.7 Configuration

The following section describes a little more the actual configuration of the solution beginning at the sensor stage and finalizing at the logging/plotting implementation.

In the first image presented below we see that the piezo sensor at the top left is connected via a resistor to an operational amplifier in the integrator configuration, filtering out high frequencies. Afterwards the output of the filter is connected to the first analog input from the Arduino Nano microcontroller, where it is given a value between 0 and 1023, from where it is sent via serial protocol to the external Bluetooth transmitter.

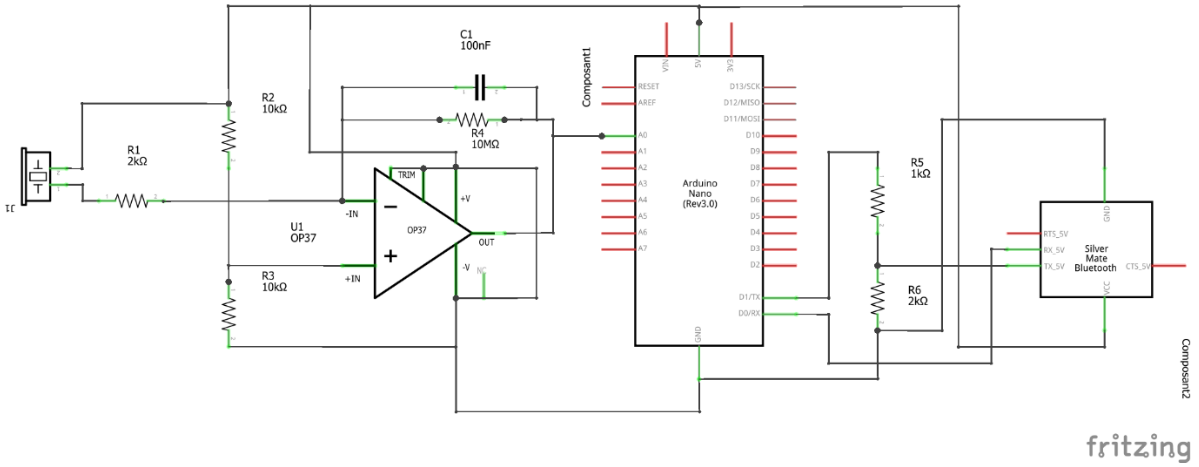


Image 27: Complete electrical layout with all components. In this solution we had considered to not add the ADS1115 or the POLOLU power regulator with the battery although it would be possible.

4.3.8 Operation

The operation of this solution works in the same manner as with the Esp32 Solution.

To being with, we must power the Arduino Nano and all the other modules. This is done by either plugging in the USB connector as this solution has not yet been tested with the alternative power supply.

Next it is necessary to pair the module with the computer in the Bluetooth configuration options. Once the pairing is complete, we can find which COM serial port is being used by the Bluetooth connection, this is found in the configuration panel in Windows. There one must look at the External Devices and choose the corresponding one to the Arduino NANO module. In its settings menu, under the material tab we can find the COM port number.

This COM port number will be used to configure the terminal in MobaXterm. To do this, we open MobaXterm and go into the session's menu at the top, there we pick New Sessions. In the new pop-up window, we select Serial from all the options. Once in Serial, under serial port we find the corresponding COM port number and we select the correct baud rate which is 115200 bps. There we press ‘ok’, and the session should be open and ready to read the incoming data.

The session will be saved and accessible by the left side window next time we open MobaXterm.

Since the data is being read continuously in real time, the counter will be set to zero only when the reset button on the microcontroller is pressed. Once the counter has been reset, we can open a serial port terminal in MobaXterm for the corresponding COM port. The appearing terminal will then display the data, line by line at high speed. Next, we press reset to stop the logging of the data. We can now right click on the terminal and select ‘save to file’, to save as a ‘.txt’ file in a convenient place on the computer.

Now we can open this data into Microsoft Excel to plot.

4.3.9 Bill of materials

- Commercially made sensor.
    - 7.95 CDN.
    - https://abra-electronics.com/sensors/sensors-flex-force-en/sen-09196-piezo-vibration-sensor-large-sen-09196.html
    - https://www.sparkfun.com/datasheets/Sensors/Flex/DT_Series.pdf
- Arduino Nano.
    - 19.45 CDN.
    - https://abra-electronics.com/robotics-embedded-electronics/arduino-boards/abranano-abra-arduino-nano-v3.0-compatible-atmega328p.html
    - https://roboromania.ro/datasheet/Arduino-Nano-roboromania.pdf
- TLC271 operational amplifier.
    - 1.22 CDN.

  - https://abra-electronics.com/ics-semiconductors/linear-series/tlc271cp-op-amp-1.7mhz-3.6v-us-dip-8.html
  - https://www.ti.com/lit/ds/symlink/tlc271.pdf?ts=1628865005528&ref_url=https%253A%252F%252Fwww.google.com%252F
- HC-05 Bluetooth module.
  - 15.31 CDN.
  - https://abra-electronics.com/wireless/wireless-bluetooth-en/hc-05-3.6v-6v-bluetooth-module-master-and-slave-mode.html
  - https://www.gme.cz/data/attachments/dsh.772-148.1.pdf
- Alligator cables.
  - 6.09 CDN.
  - https://abra-electronics.com/test-equipment-tools/test-clips-leads-banana-posts/generic-low-cost/tl-600-alligator-clip-with-pigtail-4-pack.html
- Resistors and capacitor.
  - 2 x 10 kΩ.
  - 1 x 10 MΩ.
  - 1 x 0.01 uF.
  - 1 x 2 kΩ. Depending on the impedance of the sensor.
  - 1 x 1 kΩ.
  - 1 x 2 kΩ.
- Others.
  - 3.07 CAD.
  - 2 x Prototyping perf board.
  - https://abra-electronics.com/boards/printed-circuit-boards/brd-abra-320s-single-gold-plated-protoboard.html
  - Female headers, 20 x 1 holes.
  - Male headers, 2 x 5 pin.
- Arduino IDE & librairies.
  - Open source.

    - https://www.arduino.cc/en/software
- MobaXterm.
    - Free and payable versions.
    - https://mobaxterm.mobatek.net/.
- Microsoft Excel.

This solution as is would float around $ 62 CDN. If we added the ADS1115 16-bit ADC and the POLOLU power regulator with battery, battery holder and battery charger, the total would come up to $ 106 CDN. All datasheets are available online at the presented links and are also available in the shared datasheets file package.

ANNEX 1

Esp32 configuration

https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/

ADS1115 ADC configuration

https://microcontrollerslab.com/ads1115-external-adc-with-esp32/

Both these documents can be found in the shared datasheets file package.

Datasheets

All data sheets are available in the shared datasheets package.