The design of an electronic system for an instrumented hip implant

Huib Achten

Delft University of Technology

The design of an electronic system for an instrumented hip implant

by

Huib Achten

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday August 25, 2023 at 09:30 AM.

Student number:	4451481	
Thesis committee:	Prof. dr. ir. A. Zadpoor,	TU Delft, Chair
	Dr. ir. N. Tümer,	TU Delft, supervisor
	Dr. ir. M. Dezyani	External committee member
	Dr. A. Hunt	TU Delft

This thesis is confidential and cannot be made public until August 25, 2025.

TU Delft

Preface

Before you lies the master thesis ”The design of an electronic system for an instrumented hip implant”. This project was written to fullfill the graduation requirements of the Msc program biomedical engineering at the delft university of technology.

During this project I learned a lot about what come looking with going from an idea to a prototype and its verification. By doing this in the health sector, which I have fully embraced during my masters, would keep me motivated for a successful ending. During this project I was supervised by Nazli Tümer. I would like to thank her for her supervision and keeping my on track also during the more difficult times, waiting for components and machines for example. Another person that have helped me a lot in completing this project is Javad Dezyani. As I had zero experience in the printed circuit board design he helped me in his free time to guide me through the process. Your enthusiasm during each meeting will always stick with me. For the practical guidance I would like to thank Sander Leeflang for delivering the needed materials and design and helping me setting up the machine. Sean Scott of the material science group helped me also in this way by letting my use their machine when ours needed to be repaired. During our monthly meetings with the whole research group Amir Zadpoor always had critical questions which would improve the results of this project so I would like to thank him for that. Lastly I would like to thank Andres Hunt for being in the thesis committee to grade my result.

Huib Achten
Delft, August 2023

Abstract

Data that can be collected with the use of instrumented implants can help in diagnosis and treatment of complications during the lifetime of total joint replacements. They can give an accurate reading of the status of the implant from within the body. In this research project an electronic system is designed to feed sensors with enough energy to be able to collect the data of interest. For the power generation a piezoelectric element is used and placed inside the neck of the implant. The communication from within the implant is achieved with a micro control unit with Bluetooth low energy. As sensors, two thermistors are used and the piezoelectric element can also be used as a force sensor. Multiple experiments were performed to investigate the power generation and power consumption of the circuit. The piezoelectric element was able to harvest 877μW on average during one gait cycle. From the data of the power management integrated circuit it was seen that there was 778mJ available every day. With this energy the circuit is able to measure data from two thermistors every minute and send it three times per day. There is also an option to use the piezoelectric element as a force sensor. The Nominal Root Mean Square Error(NRMSE) when using the element as a force sensor is 0.0314 in the best case. The end result is an instrumented hip implant which completely fits inside a customized hip implant and is able to communicate with a phone via Bluetooth Low Energy.

Contents

1
Introduction

Yearly 790,000 total knee replacements and 450,000 total hip replacement have to be performed in the United States alone [1]. As the population is getting older, it is expected that these numbers will only grow in the future with an expected value of 1.42 million and 3.41 million, respectively for total hip arthroplasty(THA) and total knee arthroplasty(TKA) in 2040 [2]. It is seen that most joint replacement can stay there for a longer period. But in 42% of all patients a THA need revision within 25 years. Studies showed that the most common causes for revision surgery were infection(25.2-36.1%), followed by asceptic loosening (16.1-21.9%) and periprosthetic fratures(9.7-13.7%) for knee implants [3, 4]. For hip revisions, the highest causes for revision surgery were mechanical failure/loosening (29.9-36.5%), dislocation or instability (14.6-24%) , periprostethic fracture (10.4-20%) and infection(7-9.9%) [5, 6]. It is also shown that when a total hip THA or TKA has to be revised this significantly increases the risk for complications namely 24.3% and 25.5% respectively. When a re-revision surgery has to be done this even increases to 35.0% and 39.1% [7].

1.1. Loosening

Loosening is one of the causes for a revision of a THA or TKA. There are two kinds of loosening that can happen, aseptic loosening and sceptic loosening.

1.1.1. Aseptic loosening

Aseptic loosening is the loosening of the implant in the absence of an infection. Causes for this can be an inadequate initial fixation, fixation loosening over time by mechanical factors or biological loss of fixation due to osteolysis [8]. The current methods to detect this loosening is with the help of plain radiography, magnetic resonance imaging(MRI), computed tomography(CT) scans, bone scintigraphy, subtraction arthrography and nuclear arthrography [9]. However these techniques lack in some cases the needed sensitivity (true positives) and specificity (true negatives) which is around 70-80%. Also the detection of the loosening of the hip is only noticed at the latter stages of loosening as the symptoms of loosening are not clear for patient and medical practitioner[10]. Because it is detected in the latter stage there is a higher chance for medical complications.

1.1.2. Sceptic loosening

Sceptic loosening is loosening caused by infection. Currently infection can be detected by blood test, radiographic imaging and histological studies [11]. However these studies are slow, tedious and non-specific to the infection site. When an infection is not treated in time this lead to loosening of the hip and has to be replaced.

So, for both causes of loosening of the implant the detection methods could be improved. In order to improve these detection methods there must be looked at other alternatives. One of the potential alternatives to these techniques is the use of smart implants, which are integrated with sensors that can communicate with the outside of the body. While these instruments can contribute to current diagnosis and treatment, they can also help in getting a deeper understanding of joint mechanics or implant performance. This information can then be used to improve the designs of future implants.

1.2. Smart implants

At this stage, there are some concepts that are implanted in the body, but most of them are just in the design phase, see [12, 13, 14, 15] for examples. A major challenge that all designs face is power requirements and long-term reliability. Other issues are the need for miniaturized and robust sensors and a secure and efficient data transfer system. Sensors are being developed and tested for these purposes, and a few examples can be seen in [16, 17, 11]. There are studies evaluating the feasibility of incorporating energy harvesters inside the implant. These energy harvesters should be able to harvest energy from the body to ensure that there is enough energy during the whole lifetime of the implant. Examples of these kinds of methods are triboelectric energy harvesting, biofuell cells and piezoelectric energy harvesting [18].

Although there is some ongoing research, to date the only in-vivo tested smart implant is presented in the research by Bergmann et al. [17]. But this design can only be used when an outer coil is placed around the leg as can be seen in figure 1.1. As the ultimate goal of the implant is to gather information throughout the lifetime of the implant it is good to look into alternatives.

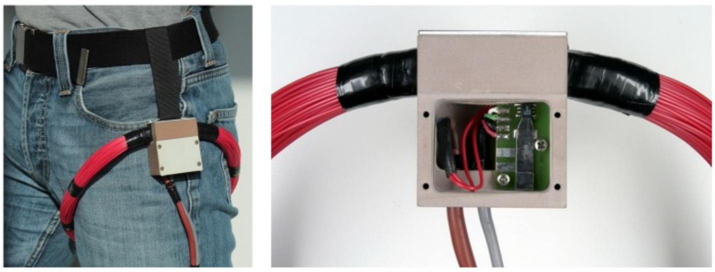

Figure 1.1: Placement of the power and communication coil around the patients's hip as presented by Bergman et al. in [17]

Other designs present only the harvesting element and it's power generation and not how the information will be send to an external device [14, 12]. Another research has designed a circuit which can be placed inside a hip implant, but it runs only on a battery so it will probably run out of power before the implant needs to be replaced [15].

1.3. Design goals and Research question

As stated, other designs show the capability to harvest energy or send it outside in a non sustainable way. The goal of this research is to design a sustainable instrumented hip implant, which is able to send information from sensors to an external device without the need for extra external hardware. For this the circuit must exist of multiple components with each of them having a certain essential function in the chain. The first component will be the power source, which will be responsible to deliver power to the controller circuit inside the implant during the whole lifetime. To regulate this power there must be a Power management integrated circuit(PMIC). As the information has to be stored or send at specific time interval or under certain conditions, there is also need for a control unit, i.e. a micro control unit(MCU). And lastly there has to be data to be read and send to an external device so specific sensors are needed. An block diagram of all the elements that are needed for a sustainable smart implant be seen in figure 1.2.

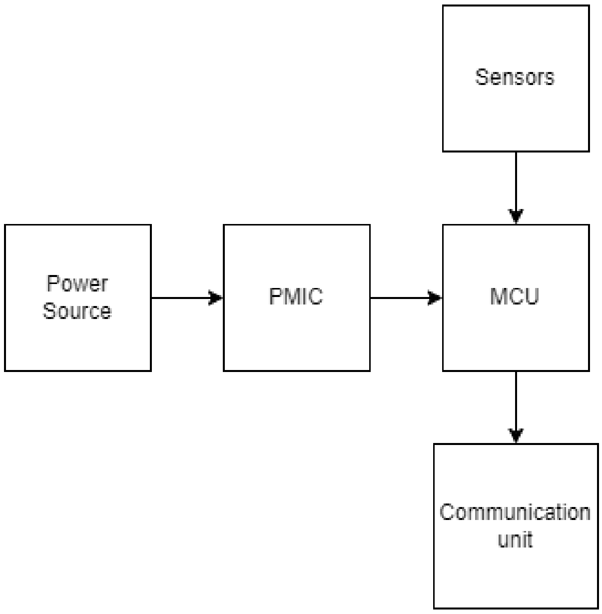

Figure 1.2: Block diagram of total system.(PMIC:Power management integrated circuit, MCU:Micro control unit)

To investigate if it is possible to design an instrumented hip implant the following research question has been formulated:

Is it possible to design an electronic system where the power generation and power consumption are in balance, fits inside an orthopedic hip implant and is able to send sensor information via wireless communication to an external device?

In order to help answering the research question the following sub questions also need to be answered:

1. How much energy can be harvested from the chosen Piezoelectric Element placed inside the hip implant?
2. What is the influence of different walking frequencies on the amount of energy harvested?
3. How much energy does the designed circuit consume during measuring of information and during sending of that information?
4. What will be the minimum footprint of the electronic circuit?
5. Is it possible to send the sensor information via Bluetooth to a mobile phone and visualize it?

As can be seen from the research questions the main goal is to verify if there is enough energy available to be able to read and send data to the outside of the implant. This project does not cover which sensors there must be used or the placement of these sensors.

2

Methodology

As described in the introduction the research goal was to design and evaluate an electronic system for an instrumented hip implant. To succeed in this process there were multiple experiments done which all will be explained in this chapter. For a better overview of the workflow of this research project, a schematic overview can be seen in figure 2.1.

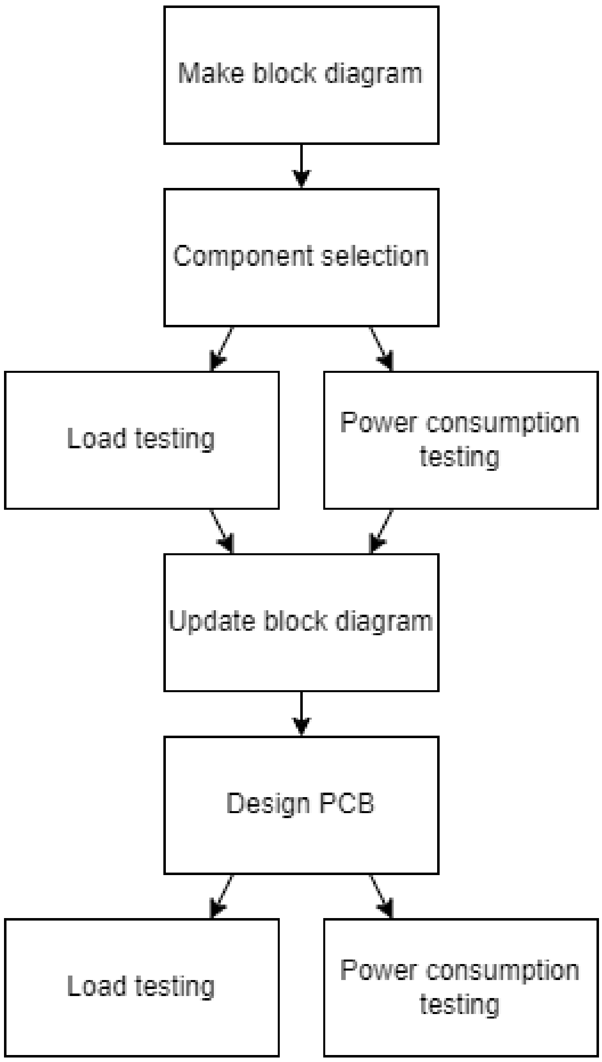

Figure 2.1: Overview and order of all experiments done during this research project

2.1. Implant design

During this research the circuit will be designed to fit inside a customized total hip implant. The customized total hip implant that will be used is a 3D printed tot hip implant made available by the section Biomaterials&Tissue Biomechanics(BMechE,3mE,TUDelft) The design can be seen in figure 2.2. It has a length of 16.4 cm and a width of 5.1 cm.

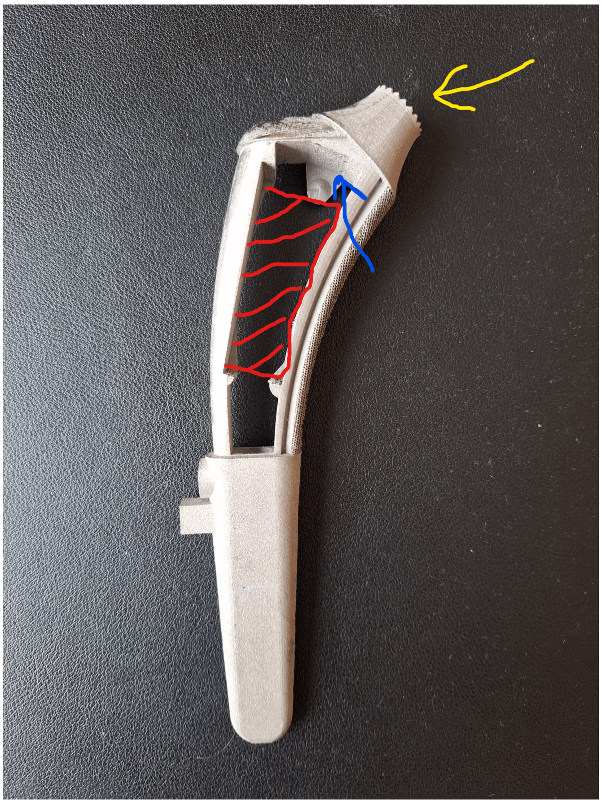

Figure 2.2: Hip implant that will be used during this research project

As can be seen in figure 2.2 the space in which the circuit has to fit is made in the upper part of the implant marked by the red area. The PE can be placed in the neck of the implant indicated by the yellow arrow and connected to the circuit via wires that can go through two wire holes which are indicated by the blue arrow. If it is necessary to guide wires from the top to the bottom this can be done through a rail which is made along the curvature of the implant. To ensure that the circuit will be hermetically sealed off the the outer part can be clamped on. This can be seen in figure 2.3.

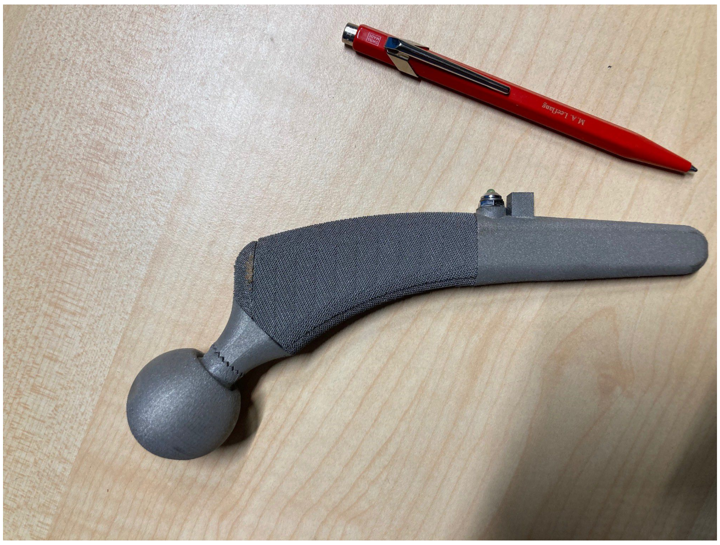

Figure 2.3: Hip implant when it's closed

2.2. Component selection

The first step of the design process is to select suitable components to fill the functions in the block diagram presented in figure 1.2 in the introduction. In this section each block will be explained with it's requirements and from these requirements a suitable component will be selected.

2.2.1. Sensors

The sensors are selected based on another ongoing master thesis project at the research group of BTB(Biomaterials & Tissue Biomechanics). Specifically, thermistors as temperature sensors were chosen. These sensors are used to examine the temperature increase near the head of the implant during walking. The thermistors used are the B57541G1103F000 from EPCOS-TDK electronics and can be seen in figure 2.4. As this research project focuses on whether or not it will be possible to send the sensor information from the implant to a communication device, it is chosen to use these thermistors as sensors during the whole project.

Figure 2.4: Thermistor used in this project

2.2.2. Power Source

For the energy harvesting element there is chosen for a piezoelectric element(PE). There is chosen for a PE because triboelectric harvesting is difficult with the geometry of the hip joint and biofuell cells are not able to produce enough energy at this moment [19]. Piezoelectric material is a material that deforms when an electric charge is applied, but also when a force is applied the material will deform and generate a charge at it's poles. For this research the latter principle is of interest as when a circuit is connected to these poles the build up charge can be harvested and used as energy for a circuit. There is charge build up at the pole because piezoelectric material consists of a asymmetric crystal structure. An overview of how it works can be seen in figure 2.5.

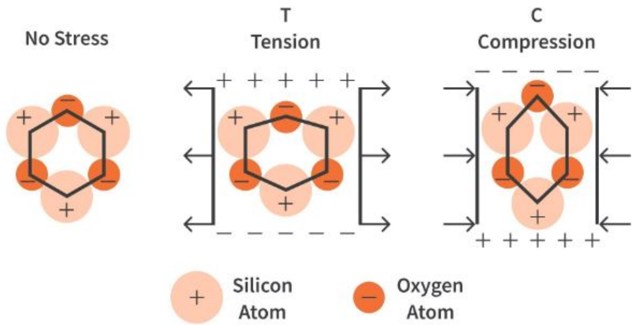

Figure 2.5: A piezoelectric material in neutral position, under tension and under compression[20]

The piezoelectric harvester is the energy source of the circuit. This element must be able to generate enough energy to be able to do measurements and sending data at certain time intervals. The first prototype was made so a round piezoelectric actuator from Thorlabs, the PH24SRZW [21], could fit inside the neck of the implant. It has the benefit that it is hermetically sealed so it will be protected from the interaction with the body. However, this element also has a downside, as the maximum force the PE is able to withstand is only 900N. Forces during a normal gait cycle can go up to 1747N for a person with a bodyweight of 75 Kg and up to 2707N for a bodyweight of 100Kg [22]. This means that the forces acting on the hip during a normal walking cycle will be too high. After a more elaborate search for another PE it had to be concluded that this was currently the best option to start with. This was because either the other PEs couldn't handle the force or the implant used had to be altered a lot. That is why there was chosen to still use the piezoelectric element by Thorlabs. This was done because it was already available and still an indication could be made on how many energy there would be available. Also the interaction between the elements can be analysed. The PE can be seen in figure 2.6. The capacitance(C) of this element is $1200nF$ and has a d_{33} of $6 * 10^4 pC/N$ which is the charge(Q) build up in the element per difference in force(ΔF) applied. From formula 2.1 and 2.2 then follows that the maximum voltage will be 45 Volts if 900N will be placed on the element.

$$Q = d_{33} * \Delta F \tag{2.1}$$

$$V = \frac{Q}{C} \tag{2.2}$$

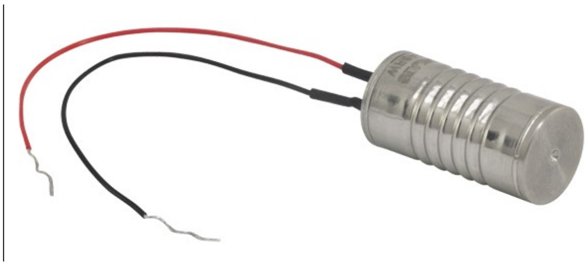

Figure 2.6: Piezoelectric element by Thorlabs, the PH24SRZW

Table 2.1: Specifications of the Piezoelecric element by Thorlabs [21]

Capacitance	$1200nF \pm 15\%$
d_{33}	$6 * 10^4 pC/N$
Blocking Force	$900N$
Outer Diameter	$12.0 \pm 0.1mm$
Outer Length	$24.2 \pm 0.1mm$

2.2.3. Power Management Integrated Circuit(PMIC)

The PMIC needs to be able to receive the energy from the piezoelectric element and feed it to the MCU. Depending on how much energy can be harvested this can be done either by directly feeding the MCU with the correct voltage or first storing the energy and deliver it when there is enough available. To be able to deliver it to the MCU it must change the AC voltage from the piezoelectric element into a steady DC voltage (most MCUs work around 3.3V). A module was found that is able to do these things. For the PMIC, an energy harvester module is chosen from the company Advanced Linear Devices, the EH300. This is a module that transforms the high instantaneous voltage from the piezoelectric element into a lower voltage and stores this in a capacitor of $6600\mu F$. When enough energy is stored in the capacitor, meaning that the voltage reaches a certain threshold, the energy is released until a lower voltage limit is reached. For the EH300, these values are 3.6 and 1.8 Volts respectively. In figure 2.7 the voltage inside the capacitor of the EH300 can be seen in the top graph. The middle graphs show the ready signal which controls whether the EH300 delivers energy or not. In the bottom graph it can be seen how it looks like when the EH300 is delivering power.

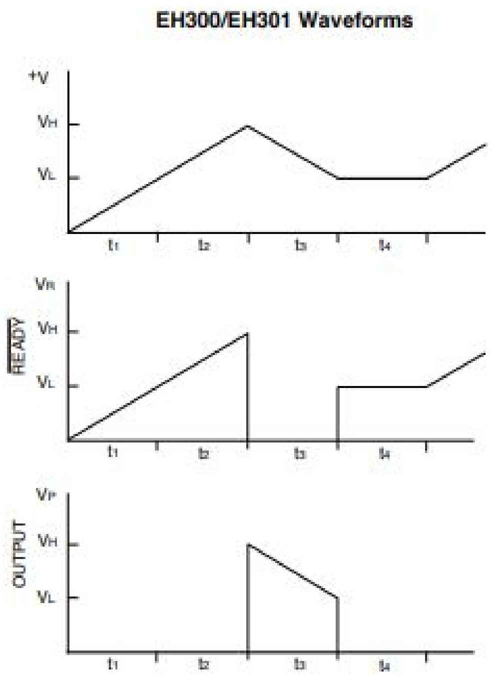

Figure 2.7: Waveforms at the output of the EH300

The main reason why this PMIC has the preference above other voltage regulators and power management integrated circuits is because it can handle voltages up to 500 volt while most common voltage regulators have a maximum of 20 volts. As from the calculation in section 2.2.2, it is expected that the voltage of the piezoelectric element can go up to 45 volts. Therefore, this module can handle the voltage input of the piezoelectric element. Next to the ability to handle high voltages, it accepts input current from 200nA to 400mA which can be delivered intermittend and irregular, also it can handle varying source impedances[23].

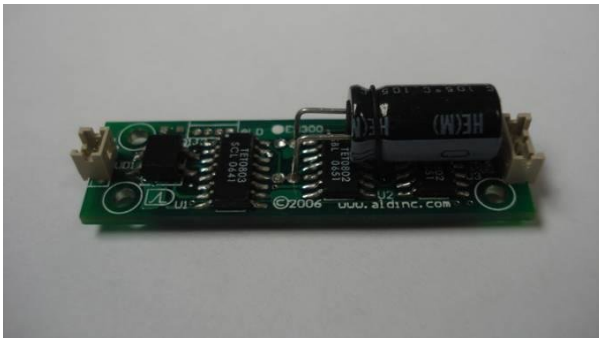

Figure 2.8: EH300 by advanced linear devices

2.2.4. Micro Control Unit(MCU)

The MCU is the brain of the circuit, it need to be able to receive the sensor inputs and send it to the outside with a wireless connection. Also it must be a low power MCU as the amount of power available will be limited. There are multiple good options on the market and after an elaborate search the first choice was to use the Renesas RE01B. Since, it is a MCU and PMIC in one with a special energy harvester function. Unfortunately they revoked support at the start of this research project due to capacity problems. Therefore, an alternative had to be found. A relatively simple and widely available MCU was chosen this time, the ESP32S3 from Espressif. The advantage of a more widely used MCU is that it is easier to find solutions for problems there might raise. Another advantage is that it has multiple modules with integrated antennas, which is a plus for the final design. To program the MCU and test the functionalities, like reading out the sensors and sending it to an external device, the ESP32-S3-DevKitC is used, see figure 2.9. This development kit makes use of the ESP32S3WROOM1 module and has 20 General Purpose Input/Output(GPIO) pins with analog to digital converter(ADC) which are needed to use analog sensors. The module comes with an integrated antenna so information can be send through Bluetooth Low Energy(BLE).

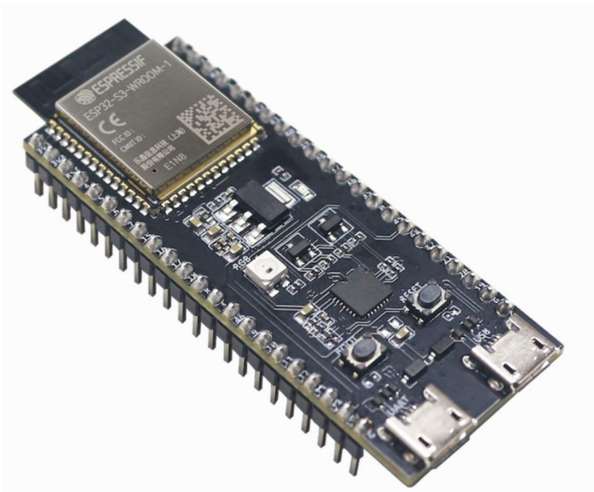

Figure 2.9: ESP32S3 dev kit

All elements are filled into the block diagram and can be seen in figure 2.10.

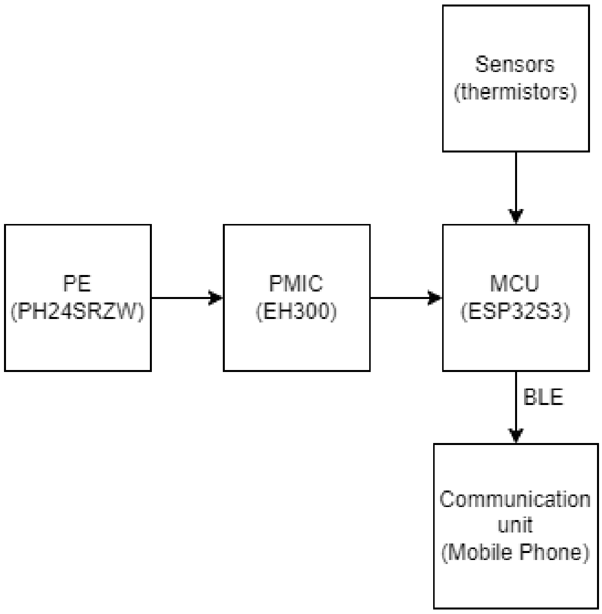

Figure 2.10: Block diagram with each component filled in it's corresponding block. (PE:Piezoelectric element, PMIC: Power management integrated circuit, MCU:Micro control unit)

2.3. Load testing of the piezo-element with attached modules

To test how much power and energy there will be available from the piezoelectric element during walking the first test consisted of a physiological load test. During this test, the Piezoelectric element was loaded under a loading scenario mimicking normal walking gait cycle. Parameters including the load resistance and gait frequency were changed to be able to answer the following questions:

1. How much energy can there be harvested during loading of the hip implant?
2. What will be the influence of the load resistance on the delivered power?
3. What will be the influence of the frequency of the loading cycle on the delivered power?

To answer the first question the output voltage of the PE over a known load needs to be measured. Then working with ohm's law (formula 2.3) and the formula for electric power(formula 2.4) the power can be calculated using formula 2.5.

$$I = \frac{V}{R} \tag{2.3}$$

$$P = V * I \tag{2.4}$$

$$P = \frac{V^2}{R} \tag{2.5}$$

Where V is the voltage over the Load resistor in Volts, R is the value of the resistor in Ohm, I is the current in Ampere and P is the Power in Watt. The average power during one gait cycle is calculated by using the following formula:

$$P_{av} = \frac{1}{Ts} * \sum_{s=0}^{Ts} P_s \tag{2.6}$$

Where Ts is the number of total samples in one gait cycle and P_s the power at that specific sample.

To determine how much energy is harvested the voltage in the capacitor and output voltage of the EH300 needs to be known. From the voltage in the capacitor, it can be determined how much energy there is inside of the capacitor by using formula 2.7.

$$E = \frac{1}{2} * C * V^2 \tag{2.7}$$

Where C is the capacitance of the storage capacitor on the EH300, V is the voltage across the capacitor of the EH300 and E is the energy stored in that capacitor.

From the voltage at the output of the EH300 it can be determined whether the EH300 is delivering power or not. When it is delivering power, the amount of energy dissipated over the load resistor can be determined by using formula 2.5 and multiplying this with the amount of seconds it is supplying this power as $E = P * t$.

For the measurements, the piezoelectric element was placed in the neck of an adjusted part of a hip implant. This part of the hip was loaded by a load controlled movement which simulates the gait cycle of normal walking conditions. To mimic the loading cycle, the ELECTROPULS® |E10000 LINEAR-TORSION ALL-ELECTRIC DYNAMIC TEST INSTRUMENT by Instron was used. This machine is able to use load controlled testing based on input data. To do that the machine first needs to determine the stiffness by generating the load-displacement curve of the specimen. With the stiffness the machine can link displacement to load. With the known stiffness, the machine will also measure the load output during the test by a load cell. This load output is then compared with the load input of the machine and will adjust accordingly. To not have a very high stiffness of the setup and to mimic displacement from a hip inside the human body the specimen is placed on a sawbone block, see figure 2.11.

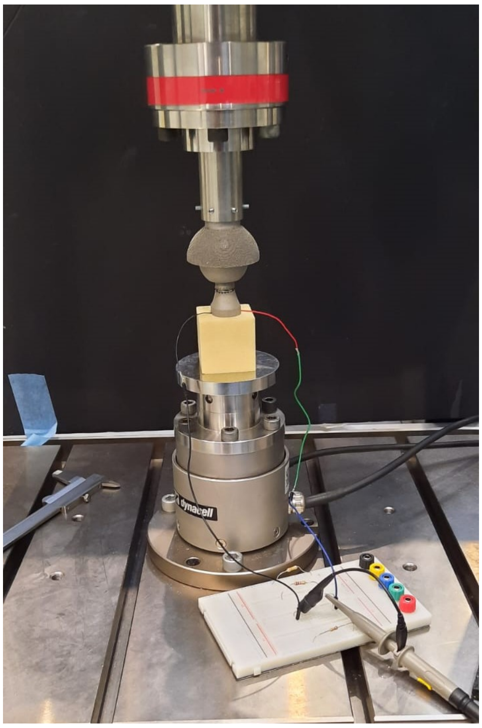

Figure 2.11: Piezoelectric inside the neck of the implant

The gait cycle frequency and forces are based on load data available from Orthoload [22]. This load data is subjected to a scaling factor to the magnitude of the applied load, so that is doesn't exceed the limit that can be carried by the PE element, i.e. 900N. The final loading pattern can be seen in figure 2.12. The input values for the machine can be found in table A.1 in appendix A

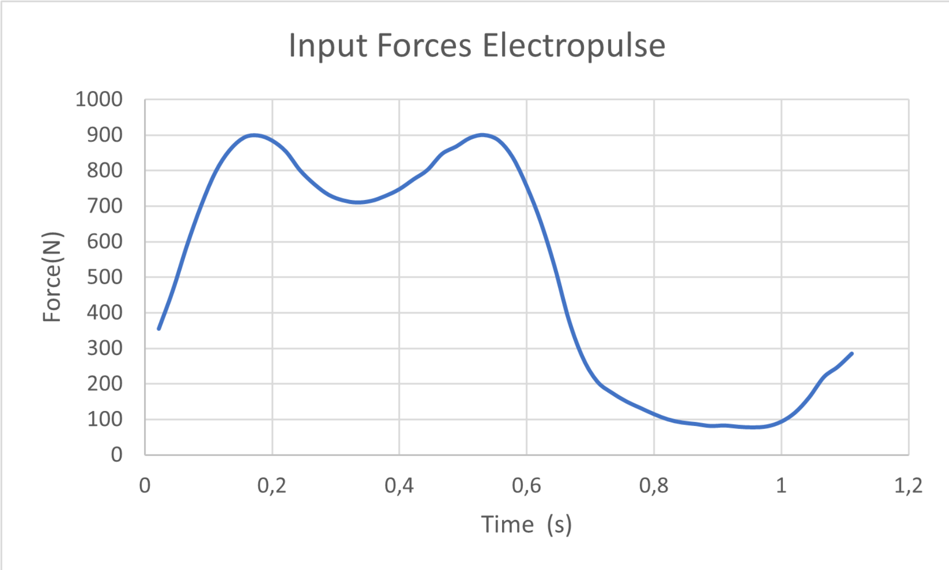

Figure 2.12: Input forces for the loading machine

The voltage was measured by probes connected to an oscilloscope. For this, the Rigol DS1054Z was used, which has four measuring channels and can measure and save data up to 600 seconds. This time period was long enough to measure the amount of energy harvested.

The loading test consisted of two experiments with different configurations. For the first experiment the PE was only connected to a load resistor, here the goal was to achieve load matching for maximum power transfer during one gait cycle. After that, the EH300 was connected between the piezoelectric element and a load resistor. As said before, the voltage before and after the energy harvester was measured to be able to determine the amount of energy harvested during a period of 500 seconds. The block diagram of the measurement setup can be seen in figure 2.13 and in figure 2.14 the experimental setup can be seen. A clearer representation of how all connections were made and where the probes were connected can be found in figure A.1 and figure A.2 in appendix A.

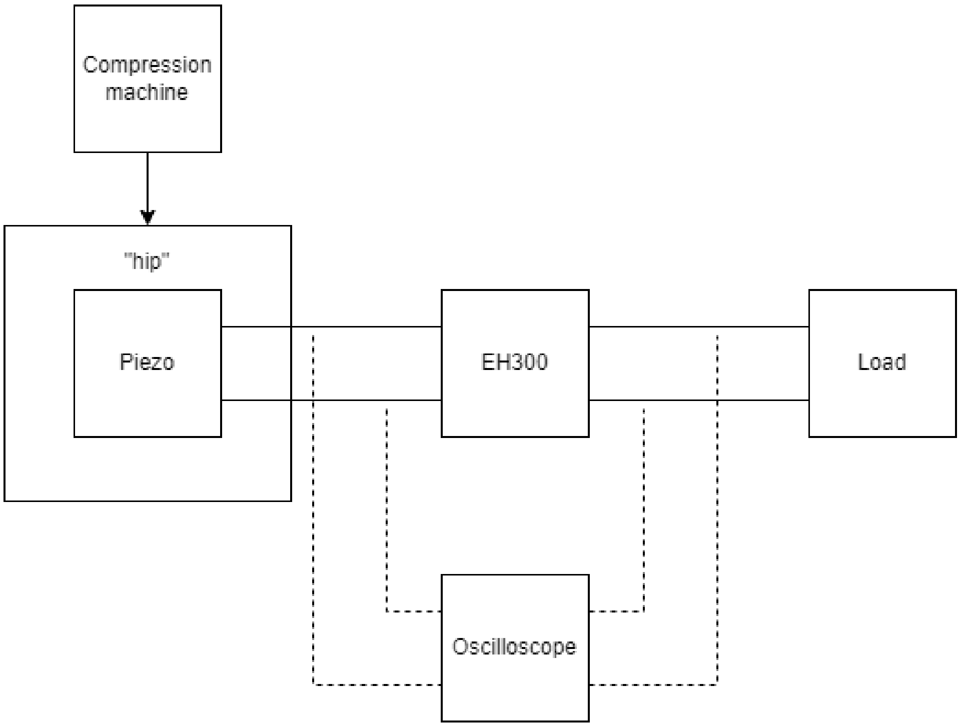

Figure 2.13: Block diagram of the measurements

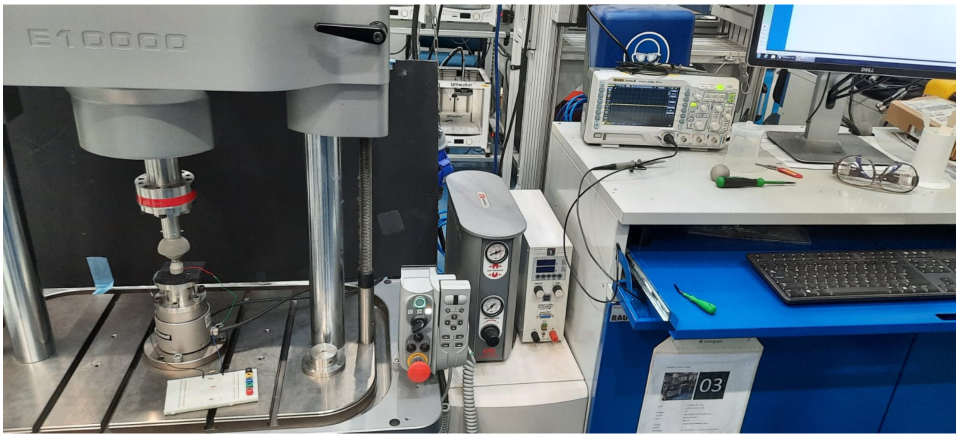

Figure 2.14: Experimental setup to for the load testing

The tests were performed in threefold with different values for resistors, the loading frequency and with-/without the EH300. All load resistors were tested at 0.9 Hz, which is the frequency for a normal walking cycle, and with and without the EH300. And from the information of the average power transferred during one gaitcycle without the EH300, the two resistors with the best power transfer were chosen to do also the tests at 0.45Hz and 1.8Hz. This was done to see what the influence was of different loading frequencies. It is expected to show a change as the impedance of a capacitor is frequency dependant. For an ideal sinus input voltage, the impedance of a capacitor can be calculated by formula 2.8. This would mean that by doubling the frequency the impedance would go down with a factor two. Considering that the loading input is not an ideal sinus and the PE is not an ideal capacitor, it is expected that the output impedance of the PE will change slightly different.

$$Z = \frac{1}{j\omega C} \tag{2.8}$$

Where Z is the impedance in Ohm, ω is the $2 * \pi * frequency$ and C is the capacitance of the PE.

A full overview of all the different conditions that were used can be seen in table 2.2.

Table 2.2: Resistors connected to the PE with the frequency used for the loading cycle

Load resistance(Ohm)	Frequency(Hz)
218	0.9
9830	0.9
163300	0.9
163300	1.8
163300	0.45
326000	0.9
326000	1.8
326000	0.45
663000	0.9
975000	0.9

The results of the experiments were post-processed using python. To see what the influence of the load resistor was, the average power during one gait cycle was calculated. The results were plotted in a load vs power curve. For the two loads with the highest average power during one gait cycle there was also looked into the average power during one gait cycle at half(i.e. 0.45Hz) and double(i.e. 1.8Hz) the frequency. For the amount of energy harvested, there was looked at the energy stored at the beginning of the test and at the end, with energy consumption by the load resistor taken into account.

These values were then compared with the results of loading the element with different frequencies to see how that influences the power. Also the amount of energy harvested in the experiment was compared with the theoretical value. The theoretical energy can be determined by the following steps. The piezo-element has a capacitance of $1200nF$ and a d_{33} of $6 * 10^4 pC/N$ from formula 2.2 was seen that the Voltage would be 45V at 900N. The amount of energy harvested will than be calculated with formula 2.9. Here it is important to see that the amount of energy harvested is dependant on the difference in Force. The forces are taken the same as in the experiment shown in figure 2.12. For the delta F the force is taken between each top as marked with A to D in figure 2.15, so the force difference between D-A,A-B,B-C and C-D filled in the formula and summed together. This results in $2,13mJ$ per gait cycle.

$$E = \frac{1}{2} * C * (\frac{d_{33} * \Delta F}{C})^2 \tag{2.9}$$

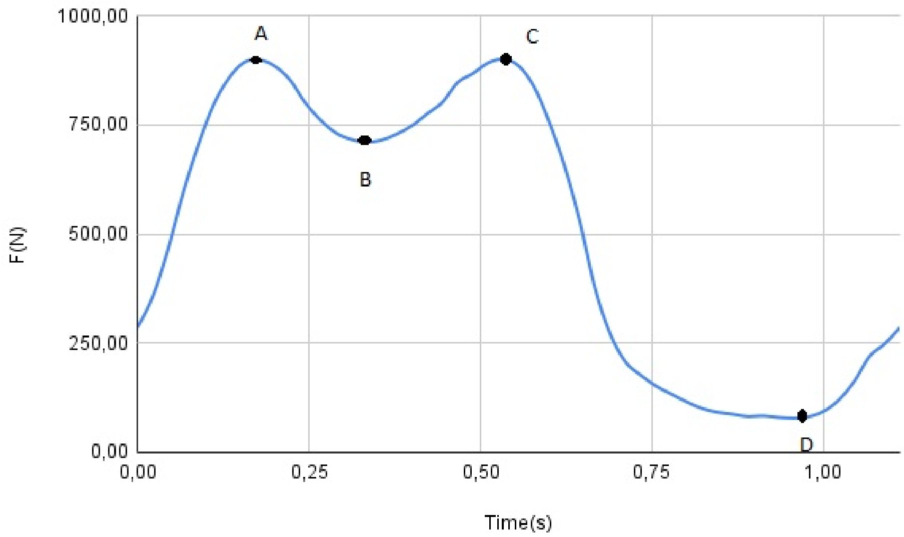

Figure 2.15: Tops used to calculate theoretical energy harvested per gait cycle.

2.3.1. Piezo element as force sensor

To determine how well the piezoelectric element would behave as a force sensor, the voltage output of the piezoelectric element over the resistance will be compared to the actual load input from the Electropuls machine. As seen from the d_{33} of the piezoelectric element the relationship between the voltage output and the force input is linear so the translation formula will have the following form:

$$Force(N) = Data * Coefficient + offset \tag{2.10}$$

For the determination of the Coefficient and the offset a linear regression model was used in python, taking the voltage from the PE as the input and the measured force as the outcome. After the translation was done from voltage to force the accuracy was determined by using the Nominal Root Mean Square Error(NRMSE) for each gait cycle and a standard deviation(σ) over all gait cycles. The NRMSE can be calculated via formula 2.11 and the standard deviation using formula 2.12.

$$RMSE = \frac{\sqrt{(X_i - Y_i)^2}}{Mean(Y)} \tag{2.11}$$

Where X_i is the measured value of the sample and Y_i is the actual value of the sample and Mean(Y) is the mean of all the actual values.

$$SD = \sqrt{\frac{\sum(x_i - \mu)^2}{N}} \tag{2.12}$$

Where x_i is the value of the sample and μ is the mean of the samples in the set, and N the number of samples in the set.

2.4. Programming and power testing MCU

The MCU is the ”mind” of the circuit, it controls what will happen during operation. To make clear how the MCU must behave during operation, it has to be programmed. This was done using Arduino IDE, which is a very approachable Integrated development environment(IDE).

The tasks the MCU has to do are: 1) take measurements with ADC, 2) save the data temporarily and 3) send the data via Bluetooth. These tasks are done in a certain order depending on some variables. First the data is read from the sensors and saved in the memory storage called RTC memory, it is stored there because this memory stays active during deep sleep. After the measurements the MCU checks if the maximum data points have been reached. If this is not the case the MCU goes to sleep and starts back at the beginning, which is to read data. If the maximum data points are reached, the

MCU will start Bluetooth and makes a connection with a receiving device, for example, a mobile phone. After all the data has been send the MCU deletes the data from the RTC memory and goes to step 1 again. A schematic representation of this workflow can be seen in figure 2.16.

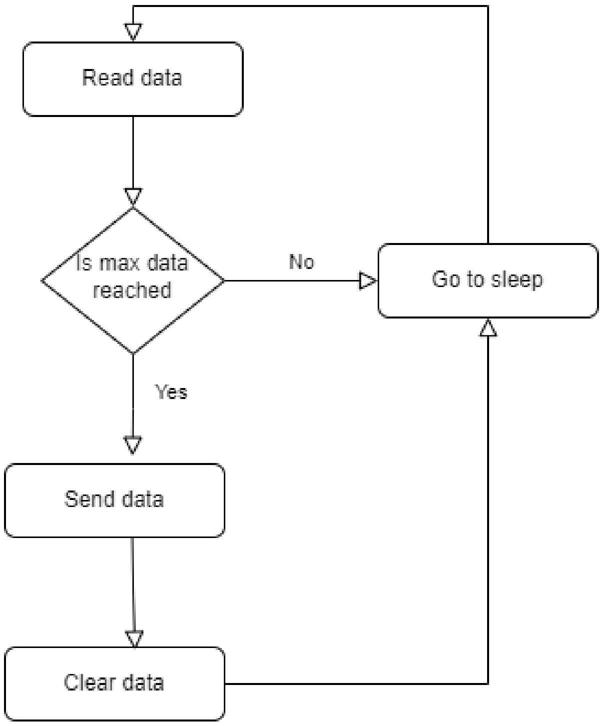

Figure 2.16: Data flow of the sensors controlled by the MCU

When the MCU is programmed correctly the power consumption needs to be known so an estimation can be made of how many times data can be read and send to the user. To test the influences of the number of sensors and the amount of data sent on the power consumption, two sorts of experiments were performed. One experiment examined what the influence would be of adding sensors(thermistors in this case) to the circuit. The other test examined whether the amount of data contributes to a higher energy consumption or not.
The parameters that are investigated separately from each other can be seen in table 2.3.

Number of sensors	Amount of data
2	50
4	250
6	500

Table 2.3: Overview of the experimental conditions

For both tests, the power was delivered by a voltage source(Tektronix PWS2185) with a steady DC voltage of 3.3 volts. The current was determined by measuring the voltage over a shunt resistor of 2 Ohm in front of the MCU with an oscilloscope(Tektronix 2022B). With this voltage drop the power consumption was calculated by formula 2.13, where V_R is the voltage over the shunt resistor, R is the resistance of the shunt resistor and V_{in} is the output voltage from the voltage source.

$$P = (V_{in} - V_R) * \frac{V_R}{R} \tag{2.13}$$

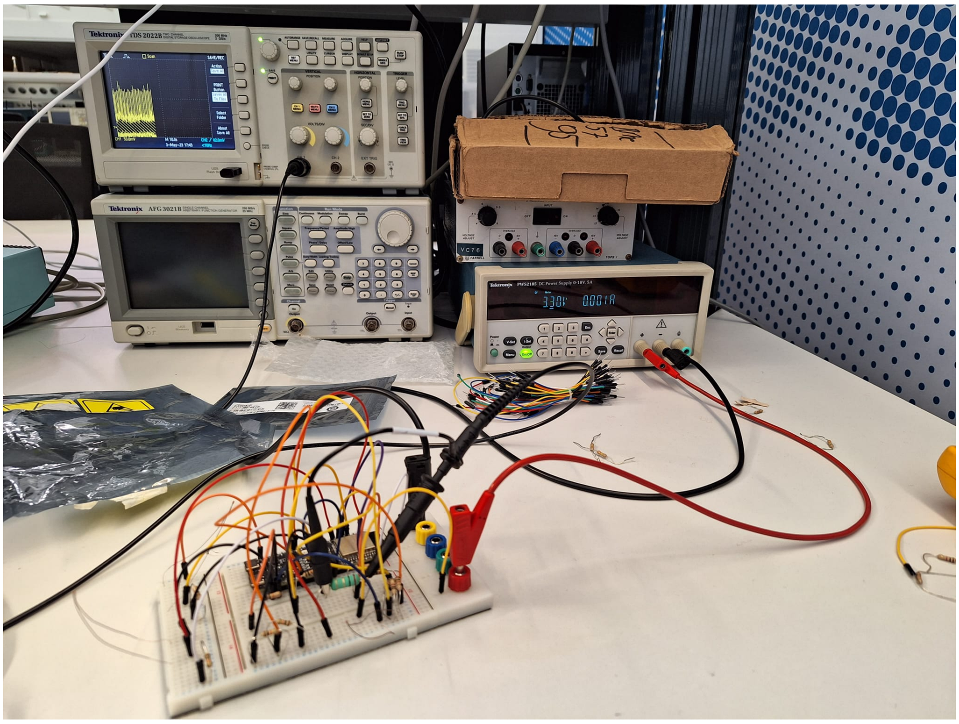

Figure 2.17: Experimental setup to measure power consumption

The first test was done by reading data every second while going to deep sleep in between. This action was continued for 100 seconds. The average power consumption per data read was determined by measuring the amount of power consumed during data reading and setting the power when in deep sleep to zero. This was done because the current in deep sleep was too low to accurately measure the power consumption and only showing noise. For the power consumption during deep sleep the value from the datasheet will be used, which is $7\mu A$, resulting in a power consumption of $23\mu W$. For the second test data was read from the sensors until the maximum set datapoints are reached. After that the MCU waited for 2 seconds and sent all the data to a phone. After all data was sent the MCU waited again for 2 seconds before measuring data again. In this way the influence of sending the data should be able to be seen as a peak in energy consumption every 4 seconds. This would continue for 100 seconds. An overview of the experimental setup can be seen in figure 2.18. A more in dept overview on where and how to connect all wires can be found in figure A.3 in appendix A. Also the codes used for these experiments can be found in appendix C.

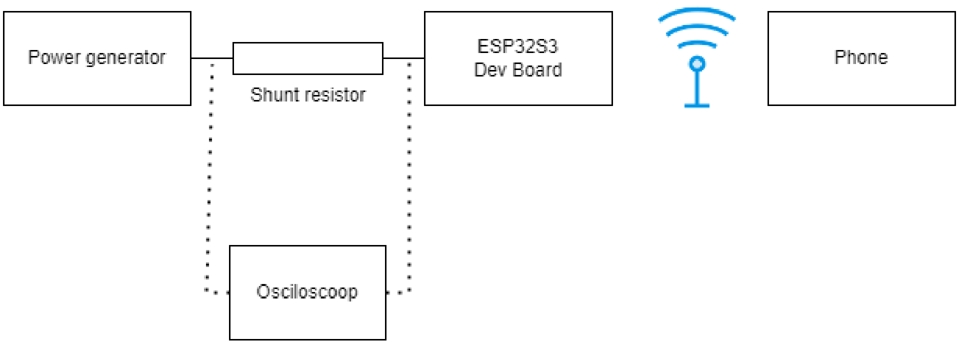

Figure 2.18: Experimental setup to measure power consumption during data reading and sending

As stated before the communication between the MCU and a phone was done via Bluetooth Low energy(BLE). To visualize the data that was sent from the MCU an app called Phyphox was used. This is an environment where the data from a bluetooth device can be received and customized graphs can be made to show the data received. Also this can be used to send data from the phone to the MCU with an edit button. An example of how this looks can be seen in figure 2.19.

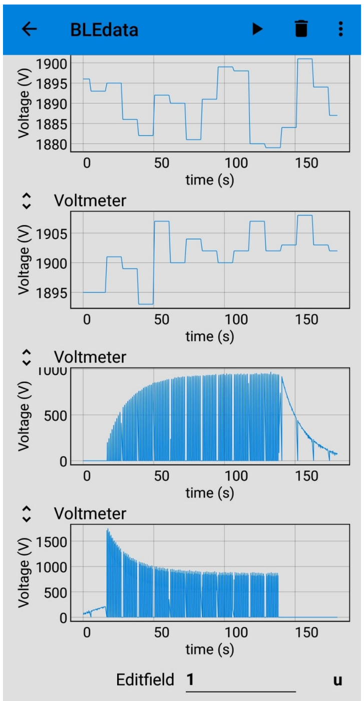

Figure 2.19: Example of how it would look like on your phone when using phyphox

2.5. Design of the printed circuit board(PCB)

After the data is processed the next step was to adjust the circuit according to the outcome of the first two experiments, i.e. the loading test and power consumption test. With these adjustments to the block diagram (addition of extra block), a circuit design was made. There are multiple design criteria but the most important one was that it should fit inside the hip implant, other criteria were that it should have low energy consumption, be able to use the piezoelectric element as a force sensor and can read out multiple sensors. For the circuit a PCB was designed and made using KICAD, which is a free software for electronic design automation. The steps for designing the PCB are:

1. Update the block diagram from figure 2.10 according to the results from the loading test
2. Translate the block diagram into an electronic circuit schematic
3. Select the right components for each of the elements in the schematic
4. From the component selection input the footprints into KICAD
5. Make the outline of the PCB according to the size constraint from the hip implant
6. Place the footprints inside the schematic
7. Connect all components on the PCB by routing of the PCB

If possible it is good to also consider some testing points for the most important nets. These can then be used to see if the system behaves the way it should be or for debugging. After the PCB is designed the PCB is ordered and assembled at the company PCBWay.

2.5.1. Design before testing

Before the final design of the PCB was made there were a few things that could already be designed. This pre testing design will be explained here and possible changes are presented in section 3.4.
The central part of the schematic will be the MCU. This will create a good overview of where to connect all the parts. For the MCU there was chosen for a module with integrated antenna. There was chosen to go with the module and not solely the chips as antenna design is very difficult for custom PCB. As the design options are very size limited there was chosen for the smallest ESP32S3 module, the ESP32-S3-MINI-1-N8 module. During the determination of where to connect the other parts the

pin layout of the module can be seen on pages 10-12 of the datasheet [24]. At the power input pin there are two decoupling capacitors added, one of $10\mu F$ and one of $100nF$ this is done to ensure that a steady voltage is supplied to the MCU. For these to work optimal they have to be placed as close as possible to the voltage supply.

As the temperature sensors are thermistors, there was chosen for a connection from an output pin to the thermistor to an input pin with an reference resistor to ground. In this way the difference in voltage after the thermistor can be linked to the resistance of the thermistor which is related to the temperature. There was chosen to let power flow from an output and not the power source to make sure there is only power lost through the sensors when a measurement is done. This input pin has to have an ADC connection as the voltage input is analog.

To use the PE as a force sensor the voltage output of the PE needs to be lowered, as there was calculated that the maximum voltage of the PE can be 45 Volts and the maximum input voltage of the MCU is only 3 volts. To lower this voltage from 45 volts to 3 volts, a voltage divider needs to be added. A simple voltage divider as shown in figure 2.20 will be used. How much voltage will be at the input of the chip can be calculated by using formula 2.14. So if R1 would be 14KOhm and R2 1KOhm the maximum voltage would be 3 volts at the input of the chip. In section 3.4 the final values for R1 and R2 are chosen based on the results of the loading test.

$$V_{Chip} = V_{PE} * \frac{R_2}{R_1 + R2} \tag{2.14}$$

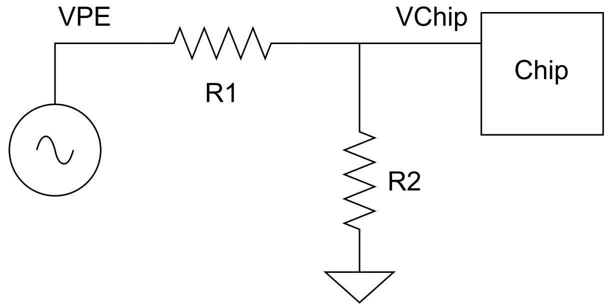

Figure 2.20: Voltage division circuit to reduce the voltage of the PE by a certain factor

The voltage output of the EH300 works as a voltage controlled switch which will close when there is 3.6 Volts and opening when there is 1.8 Volts. This will mean that the voltage output will not be steady so there was a need to regulate the voltage output of the EH300. For this a voltage regulator was added between the output of the EH300 and the MCU. At the output there will also be a decoupling capacitor of $10\mu F$. Between the voltage regulator and the battery there is also a diode added to prevent the battery from supplying power to the output of the voltage regulator which could harm the device. A $1KOhm$ resistor is added to limit the current to protect the diode.

For this design there is chosen for the possibility to program the device after assembly. In this way multiple programs can be used to verify the functionalities of the PCBs separately. Also, there will be no need to finalize the program so it can be altered between testing. To program the PCBs there is a need for four connection points, digital receiver(RXD), digital transmitter(TXD), BOOT and Reset(RST). To power the chip during programming the battery connection points can be used. To program the PCB the RXD and the TXD connection points have to be connected to the corresponding channels on the development board. The BOOT must be connected to ground while the RST is triggered to ground for a short period. After uploading is completed, BOOT needs to be made floating and RST again pulled to ground for a short period. An overview of the connections can be seen in figure B.2 in appendix B. In later configurations the MCUs can be ordered pre programmed making these four connection points redundant.

2.5.2. Placement inside the hip

A requirement of the PCB design was that it had to fit inside the hip implant. As the EH300 takes already a lot of space it was decided that the PCB will be stacked on top of the EH300 circuit. This means that the capacitor of the EH300 has to be replaced inside the hip. The outer dimensions of the EH300 are used as reference points for the design of the PCB. This was done because the implant was already designed to make sure the EH300 fitted inside. Unfortunately the ESP32-S3-MINI-1-N8 is a bit wider as the EH300. To verify whether the design for the PCB would really fit inside the hip implant, an extruded cut was made into solidworks. In figure 2.21 this verification can be seen.

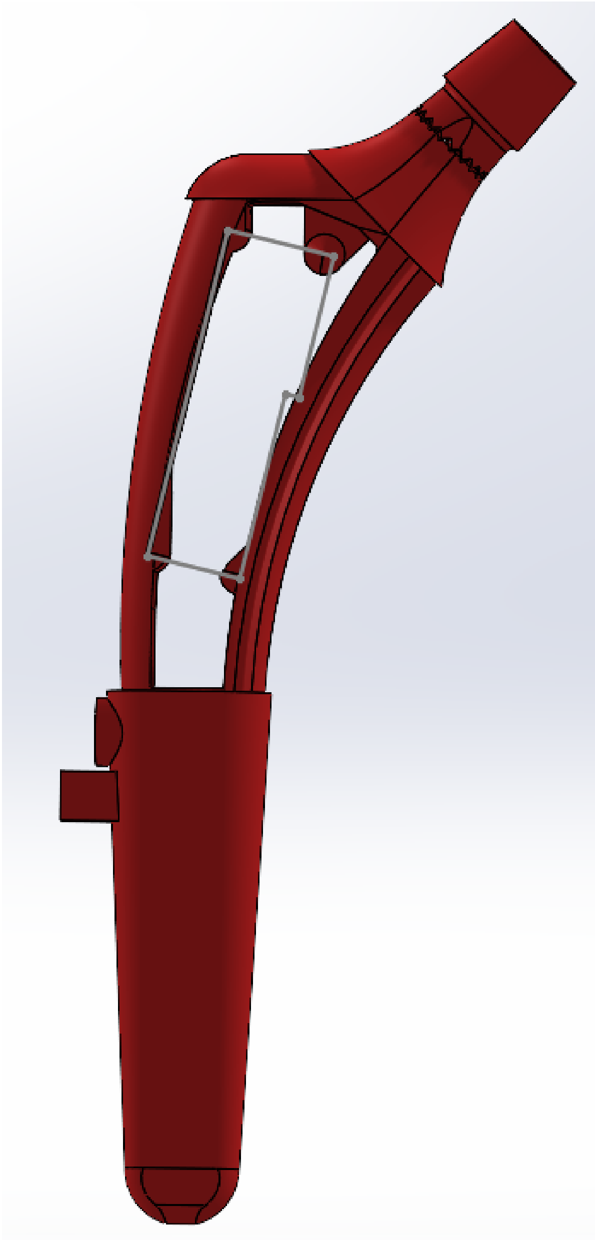

Figure 2.21: Placement of the PCB inside the hip

2.5.3. PCB footprints

The first step when starting with the PCB layout is to link each component with it's corresponding footprint. A footprint is the print that is required on the PCB to be able to successfully place the component on it during assembly. It is possible to create the footprint yourself but as KICAD has some libraries and the other footprints can by retrieved from SnapEDA, which is an online library for electronic footprint, this was not necessary.

When all footprints are linked to it's symbols they can be placed in the PCB editor. The first step was to create the edge cuts so the outline of the board can fit inside the hip. When the outline was made the components were placed on the board. During placement of the components some components are needed to be at a specific places. The MCU had to be placed at the top as there was no other place it can fit. The decoupling capacitors must be as close to the IC as possible to work effectively. That is why the decoupling capacitors of the MCU were placed at the back of the PCB.

2.6. Validating PCB

The last part was to check whether the PCB design acts like it is supposed to do. In the end there were two PCBs ordered for verification. This was done due to the fact that certain preferred components were not in stock. That is why there is spoken about two different PCBs from here, one with a relay and one without a relay as that is the difference between them. The first step was to confirm that the PCB was printed correctly and all connections are made as they were supposed to be. The next step was to do some experiments with it. The experiments done were similar to what has been done with the circuit based on the loose modules. These tests can be divided into three groups: Power generation measurements, power consumption measurements and a functionality test. The PE was again placed in only the neck of the implant and placed on the sawbone block. The PCB was placed on a breadboard so all connections could be easily made. With the information from these tests an estimation could be made in terms of how many times per day a measurement can be taken and send to the patient/practitioner and how well the PE can act as a force sensor.

2.6.1. Power generation

This experiment was the same as the first experiment, with the same loading conditions. The goal of this testing was to verify if the power generation would be similar to that of the first experimental setup. Instead of using a load resistance the "load" connected to the EH300 was the PCB. An overview for both PCB options can be seen in figure 2.22 and 2.23. A clear overview of how all connections have to be made can be found in figure A.5 in appendix A.

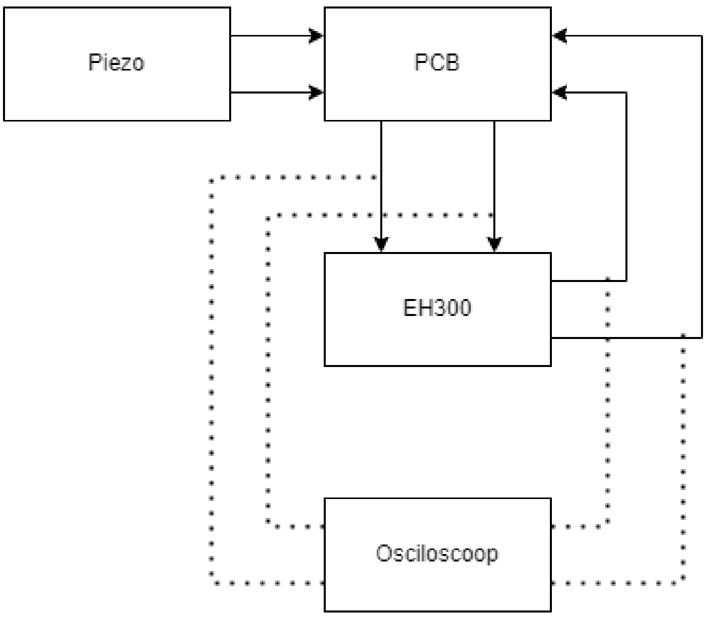

Figure 2.22: Schematic of the PCB design with relay

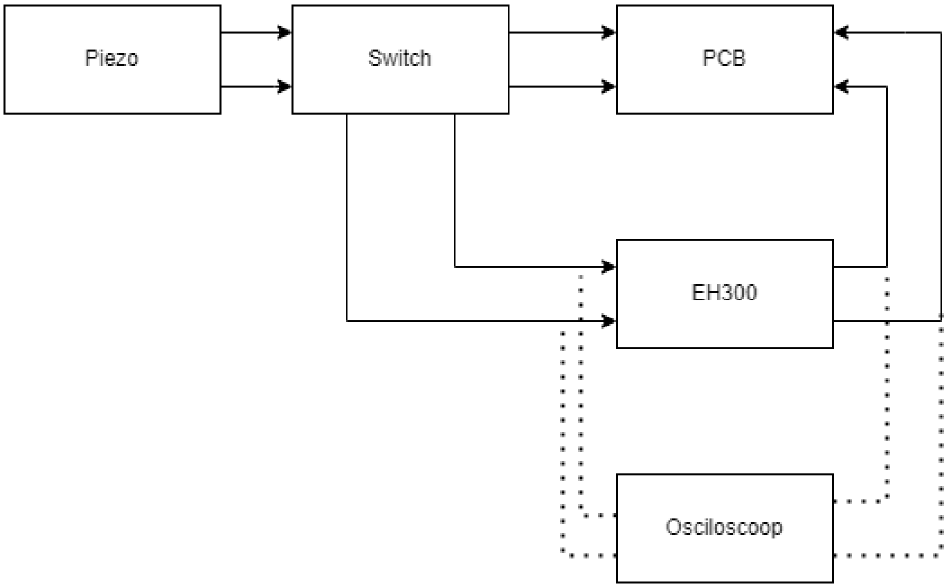

Figure 2.23: Schematic of the PCB design without relay

2.6.2. Power consumption

For this a similar test was done as stated in section 2.4. The difference now was that the power consumption of the two PCBs were expected to differ a lot due to the addition of a relay in one circuit while the other just has a manual switch. Both circuits were tested during measuring and harvesting mode. The circuits were both not loaded by the machine while being tested but the power was just supplied by a voltage generator. This was done because the consumption is independent of the generation. A full overview of how to setup the experiment with all connections can be found in figure A.4 in appendix A.

2.6.3. Functionality test

For the last test the accuracy of the PE as a force sensor was investigated and the functionality of the system as a whole. The functionalities are to receive data from the circuit and to be able to send something to the PCB. To determine the accuracy of the PE as a force sensor first the coefficient, as seen in formula 2.10, to transfer digital output to force was retrieved from measurements which were done with the gait cycle being the same as in the first loading test. This coefficient was again determined with a linear regression model. To measure how accurate this translation works in other conditions, alternate loading cycles were made. One where the magnitude was scaled down and four others were the frequency were changed compared with the normal gait cycle frequency. So in total there were five new waveforms, one with a frequency of 0.9 Hz where the maximum force was scaled down to 795N. And four where the maximum force stayed at 900N but the frequency was changed to 0.8Hz, 1.0Hz, 1.8Hz and 0.45Hz. For these last measurements there was no need to also measure the voltage but there was only looked at the data that was received on the phone. These results were then compared with the actual loading of the element. Again the overview of all the connections can be seen in figure A.4 in appendix A.

3

Results

In this chapter all the results from the experiments as described in chapter 2 will be presented.

3.1. Loading Test

The first step was to determine the stiffness of the specimen used, this was done by the Electropuls machine. The stiffness of the setup was 569,34 N/mm meaning that a force of 900 Newton compresses the setup by 1.58 mm. The input force and the achieved output loading curve from the machine can be seen in figure 3.1. It can be clearly seen that the force input and force output are almost identical.

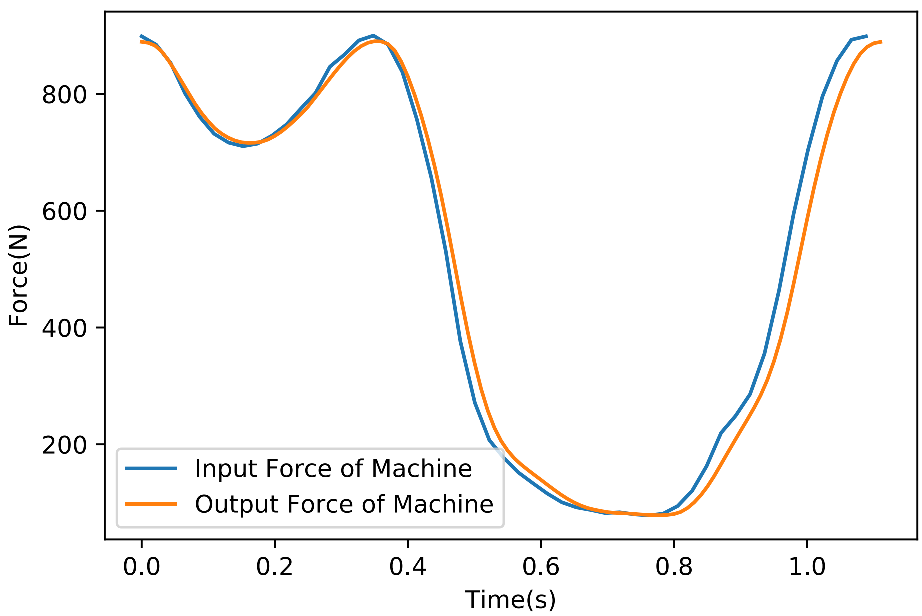

Figure 3.1: Force input and output of the electropuls machine

3.1.1. Piezoelectric element connected to resistor only

From the test where only the piezoelectric element was connected to a resistor the average power generated from one cycle could be calculated. In figure 3.2, the voltages can be seen during one gaitcycle. It can be seen that from a resistance of 163KOhm and higher, the voltage follows the same pattern as the loading cycle.

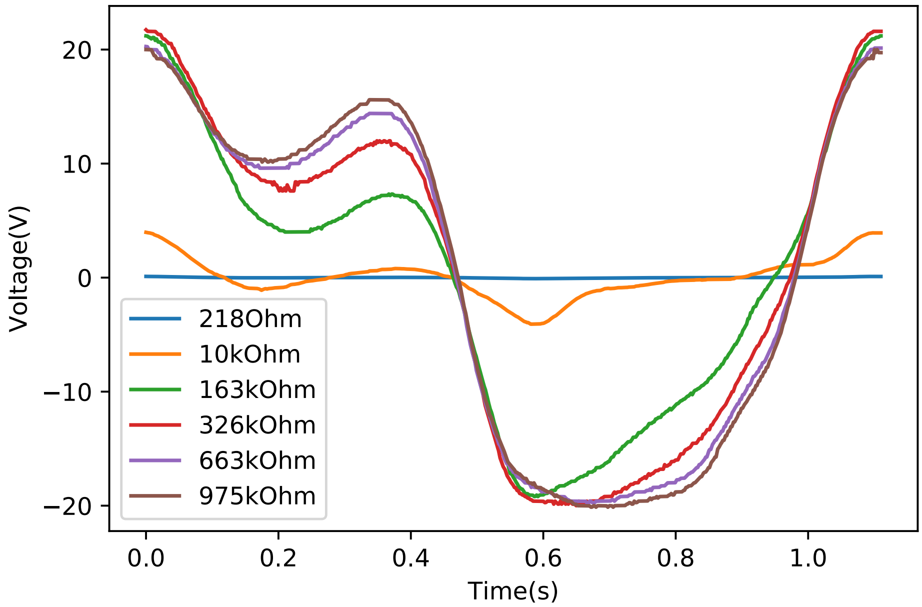

Figure 3.2: The voltage of all resistors during one gait cycle

For the lower load resistors seen in figure 3.3 and figure 3.4, there can be seen a pattern looking like the gait cycle but this charge is lost earlier which can be seen by the extra zero crossings. The reason behind this will be discussed in section 4.1.1.

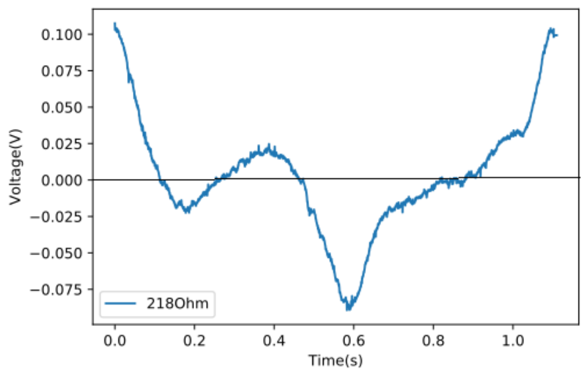

Figure 3.3: The voltage over a 218 Ohm load resistor during one gait cycle

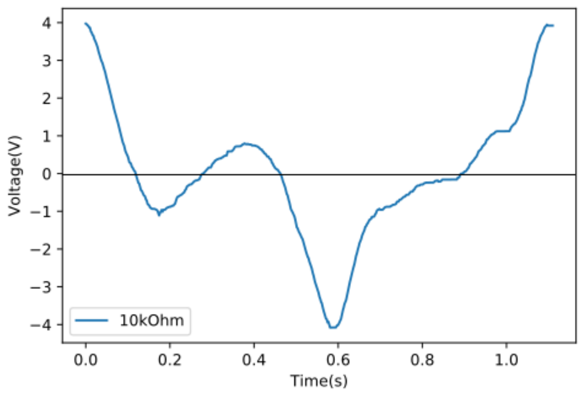

Figure 3.4: The voltage over a 10KOhm load resistor during one gait cycle

As described in section 2.3, the average power during one gait cycle is calculated by using formula 2.6. It is seen that the most power will be delivered to a load resistor of 163KOhm. The average power during one gait cycle is 877μW and a peak power of 2.75mW. The results for the other load resistors can be found in table 3.1.

Table 3.1: Delivered power of the Piezoelectric element during one gait cycle

Load Resistance($K\Omega$)	Average power(μW)	Peak power(mW)
0,218	7,18	0,0530
9,83	300,81	1,58
163,3	877,31	2,76
326	603,70	1,45
663	312,00	0,619
975	224,69	0,410

To verify whether the results from changing the value of the load resistor makes sense a power to load curve is made. In figure 3.5 this power to load curve can be seen.

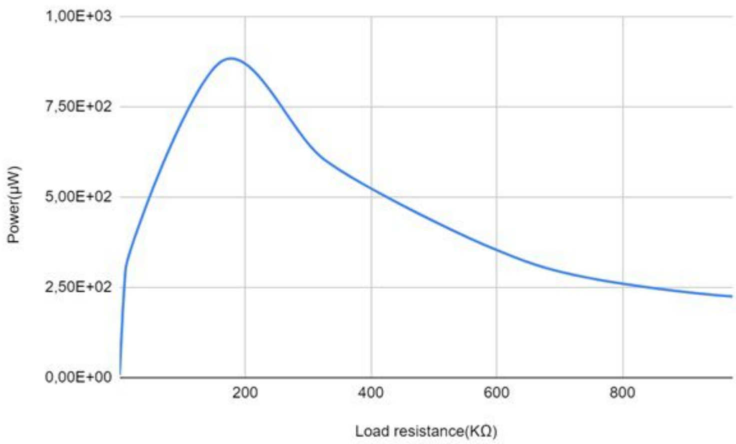

Figure 3.5: Average power vs load resistance during one gait cycle

3.1.2. Piezoelectric element connected to EH300 with load resistor

The amount of energy harvested from the loading of the PE was determined by measuring the voltage at the input of the EH300 and at the output of the EH300. From these values the amount of energy stored at the beginning and at the end can be determined, while also taking into account the energy loss at the load resistors during measurements. The results can be seen in figure 3.6 and figure 3.7.

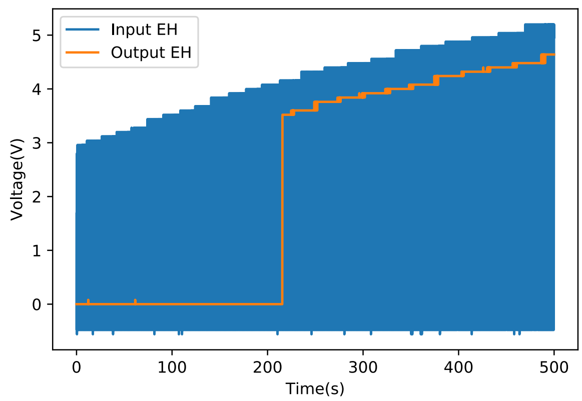

Figure 3.6: The voltage before and after the EH300 that keeps on harvesting during delivery of energy

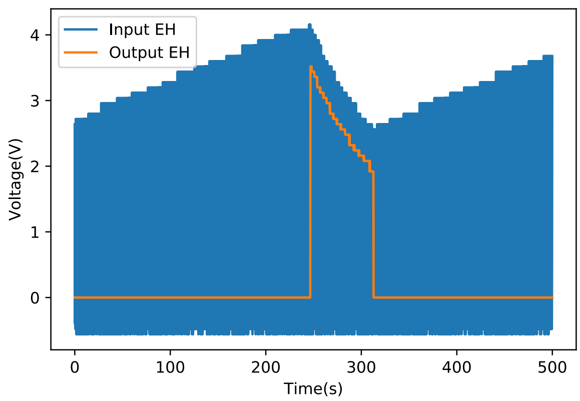

Figure 3.7: The voltage before and after the EH300 clearly seeing the delivering phase

As can be seen from figure 3.6 and figure 3.7, there were two different scenarios that would happen. In the first scenario(figure 3.6), the load resistance was so high that when delivering energy to the resistance the amount of energy harvested was higher than consumed. As the amount of energy inside the EH300 keeps rising while also delivering energy to a load resistor. The reason why the voltage at the output of the EH300 will only start delivering halfway is because a certain threshold has to be reached as described in section 2.2.3. In the second scenario(figure 3.7), the amount of energy harvested was less than consumed. As power is delivered to the load resistor until the lower threshold is reached. At that point the switch will turn off and stops delivering energy to the load resistance. The amount of energy harvested during a 500 second measurements can be seen in table 3.2.

Table 3.2: Delivered Energy in 500 seconds of harvesting for different load resistors

Load Resistance($K\Omega$)	Delivered energy(mJ)
0.218	69.94
9.83	65.07
163.3	70.24
326	67.84
975	73.41

It can be noted that there is not really a difference in amount of energy harvested when a different load resistor is connected to the output of the EH300.

3.1.3. Piezoelectric element with different frequencies

To answer the subquestion on what the influence of frequency is on the amount of energy and power transfer these test were repeated with the same input loading curve but now at a frequency of 1.8Hz and 0.45 Hz. This was only done for the two load resistors with the highest average power over 1 gait cycle, which are the 163KOhm and 326KOhm resistor as can be seen in table 3.1.

In figure 3.8 and figure 3.9 the results of the voltage over the load resistors of 163KOhm and 326KOhm can be seen. It shows clearly that the loading frequency influences the amount of voltage. It shows a decrease in power for the 0.45Hz and an increase in power for 1.8Hz.

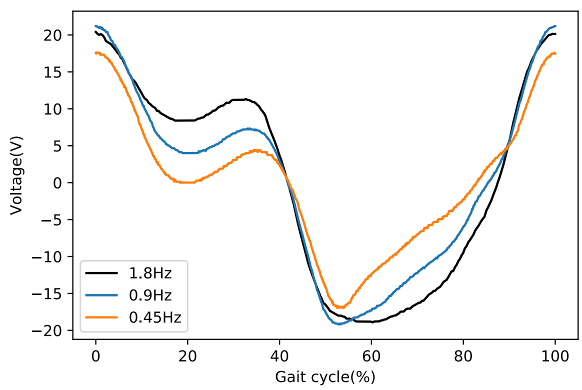

Figure 3.8: The voltage over a load resistor of 163KOhm with different loading frequencies.

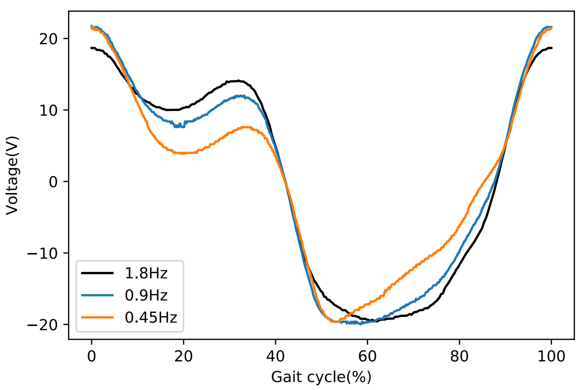

Figure 3.9: The voltage over a load resistor of 326KOhm with different loading frequencies.

This means that by increasing or decreasing the frequency at which the piezoelectric element is loaded the amount of energy harvested from it will be influenced.

Table 3.3: Average power and delivered energy for a period of 2,22 Seconds for different frequencies

Load Resistance($K\Omega$)	Frequency(Hz)	Average Power(μW)	Delivered energy(mJ)
163	1.8	1095.66	2.43
163	0.45	497.30	1.10
163	0.9	877.31	1.95
326	1.8	605.04	1.34
326	0.45	443.72	0.99
326	0.9	603.70	1.34

3.1.4. Piezoelectric element as force sensor

To see how good the piezoelectric element would behave as a force sensor, the force output of the electropuls machine is compared with the force output from the piezoelectric element. The force output of the piezoelectric element is based on the voltage output. As a linear relationship is expected and the voltage output from the PE over the load of 975kOhm looks the most on the force output (see figure 3.2), the datasets from that load resistor were used. The result of the force output from the machine compared with the force calculated from the piezoelectric element can be seen in figure 3.10. The translation from volt to force was done by getting the coefficients from a linear regression model which resulted in the following translation formula:

$$Force(N) = Voltage * 21.32605471 + 504.44436452090684 \quad (3.1)$$

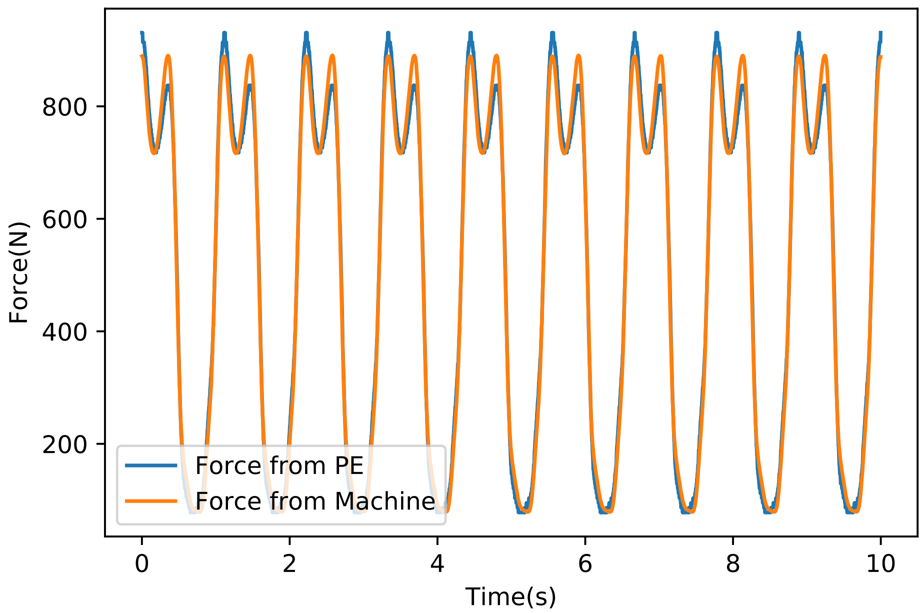

Figure 3.10: Force output from the machine and calculated by the piezo element

From figure 3.10 can be seen that the graphs overlap for most of the time except for the peaks. This comes due to the fact that there is less voltage generated at the second peak compared with the first peak. As the forces are almost the same at these peaks the model compensates this by overestimating the first peak so the second peak comes closer to the actual value. This resulted in a NRMSE of 0.0519 with a standard deviation(σ) of 0.000232, which are calculated by using formula 2.11 and formula 2.12.

A sidenote is that the piezoelectric element can't be used as a force sensor when connected to the EH300. This could be seen in the voltage input of the EH300 when the measurements were done to determine the amount of energy harvested. In figure 3.11 it can be clearly seen that the voltage doesn't follow the smooth input. Also during the measurements of the harvester the peak voltage is dependant on the amount of energy stored in the capacitor making it impossible to determine the force accurately.

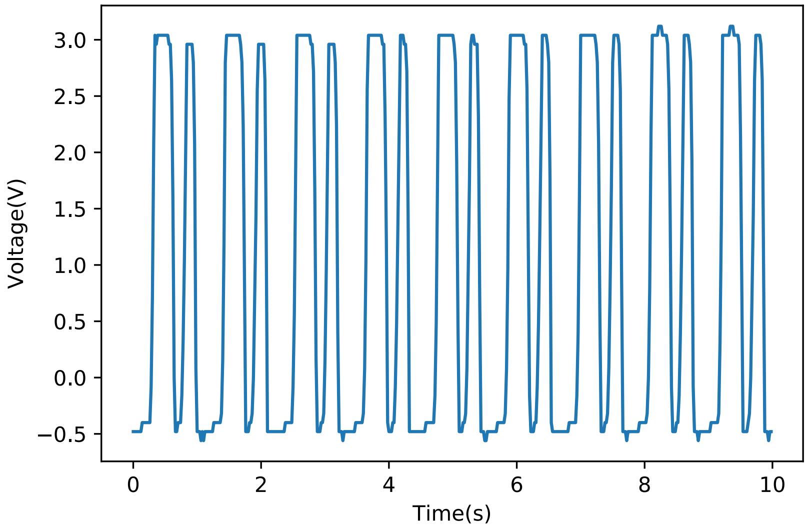

Figure 3.11: Voltage output from the piezoelectric element when the harvester is connected

3.2. MCU programming and testing

To determine the power consumption of the MCU there was looked at the voltage drop over a shunt resistor of 2.1Ω. From this voltage drop the input current of the MCU could be calculated and with these values the power consumption.
In the first test there was looked at the power consumption of the MCU when measuring sensor data. The goal of this test was to check what the influence was of adding extra sensors to the circuit. In this experiment the voltage drop over the shunt resistor will be measured.
In figure 3.12 the power consumption of the development board can be seen.

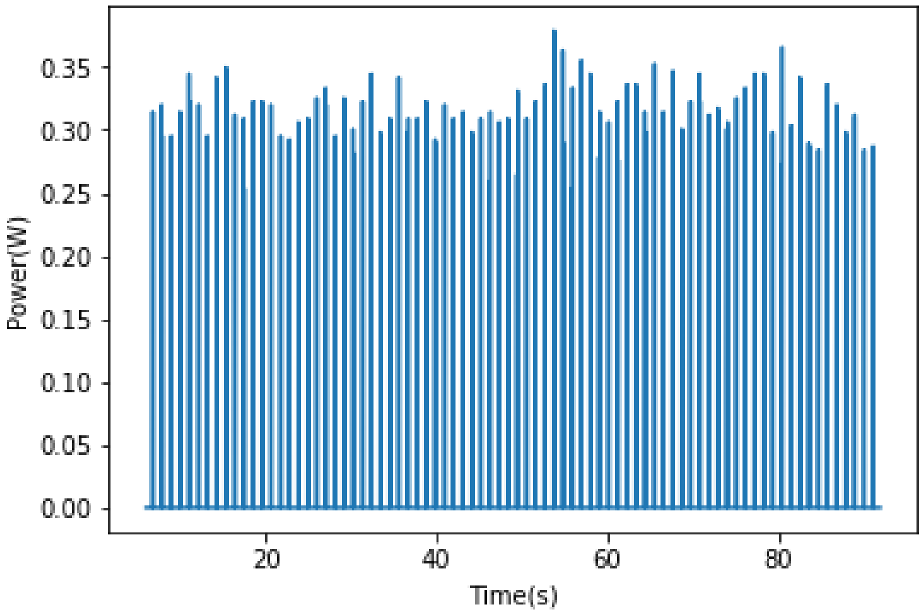

Figure 3.12: Power over time only showing the times a sensor is read out

It can be seen that there is a peak in power every second, this peak is caused by the reading of the sensors. From figure 3.12 there can already by seen that the MCU consumes around 300mW when measuring data from the sensors. As these measurements can be done very quick, less than 0.5

ms the amount of energy per reading can be concluded at less than 150μJ. The power consumption is determined by taking the average of all the sensor readings. In table 3.4 the average power and energy consumption per readout for each amount of sensors can be seen. This energy consumption is calculated with the assumption that a reading would take 0.5 ms.

Table 3.4: Consumed power/energy when different amount of sensors are being used

Amount of sensors	Power needed(mW)	Energy needed(μJ)
2	313	157
4	299	150
6	289	145

The second test was to measure the influence of the amount of data sent on the power consumption. For this the idea was that the measuring phase and the sending phase were clearly distinguishable by letting the MCU wait for 2 seconds after either sending or measuring. The raw data of one of the measurements can be seen in figure 3.13.

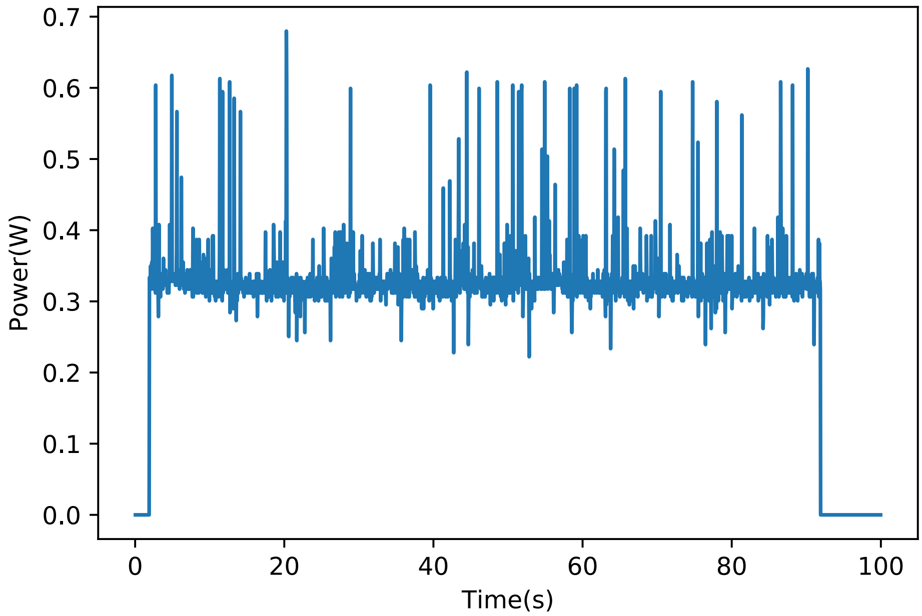

Figure 3.13: Power over time while sending sensor data

As can be seen the idea of the measuring phase and sending phase being distinguishable doesn't apply. No clear peaks can be seen when sending, there can be peaks seen but these are not on the interval at which data is send so are not caused by the BLE connection. To still get an idea what the influence is of sending data, the average power is calculated and compared with the average power when only measuring. The average power for each setting can be seen in table 3.5.

Table 3.5: Average power consumed per second

Amount of datapoints	Power needed(mW)
50	322
250	322
500	331

It is seen that the amount of data points sent doesn't really affect the average power. However sending more data cost more time so more power is consumed in the end. Compared to when only measuring data and not sending the data the development board consumed about 10mW extra.

3.3. Block diagram based on results

From these experiments the block diagram has to be updated. As there is seen that the voltage curve from the PE is not linearly related to the input force anymore when connected to the EH300(see figure 3.11, the PE can't be used as a force sensor without some alterations in the circuit design. A solution would be to add in an extra block in the block diagram which would be able to connect and disconnect the PE and the EH300. When the connection would be made or not is depending on if a force measurement needs to be done or energy needs to be harvested. The functionality to disconnect will be represented by the analog signal conditioning block in the block diagram. This block also represents the attenuation of the voltage of the PE to the ADC inputs of the ESP32S3 as described in section 2.5.1. Also there is seen that the power generation will not be enough to provide the chip with a continuous power supply, see table 3.2. If there is no continuous power supply to the chip it will shut down and the data saved inside the RTC will be lost. To cope with this a rechargeable battery will be added to always have power available and data stays available in the RTC memory. To make sure the battery doesn't run out of power the energy harvested and energy consumption need to be in balance. The updated block diagram can be seen in figure 3.14.

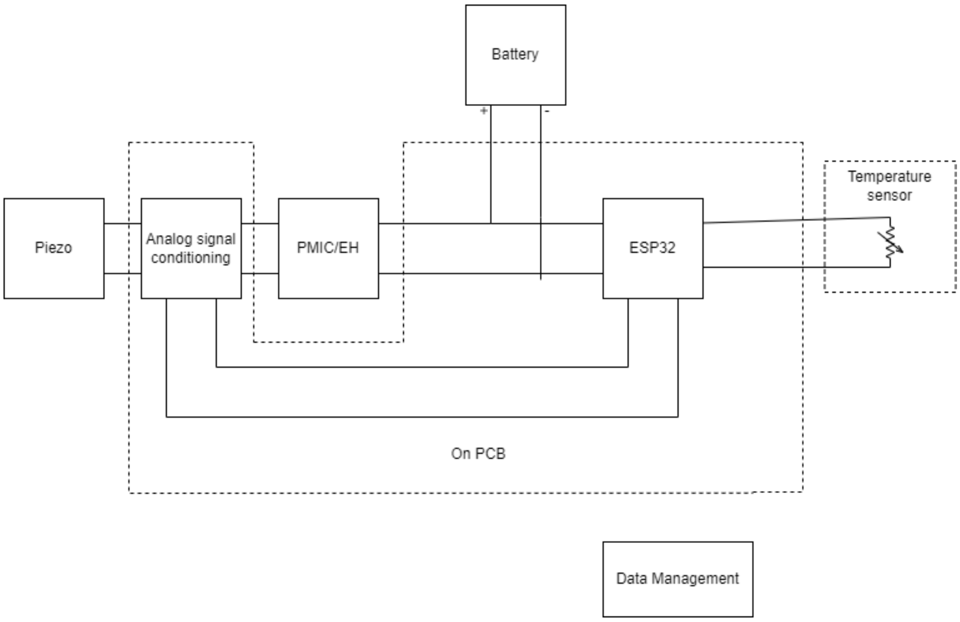

Figure 3.14: Updated block diagram of the circuit.(PMIC:Power managements integrated circuit, EH:Energy Harvester, Piezo: Piezo electric Element)

3.4. PCB design

In this section the design changes to the PCB design are presented. These changes are made based on the results presented before in this chapter after that the final design is shown.

3.4.1. Schematic design

The first step of the PCB design is making the schematic based on the updated block diagram (figure 3.14) from the first experiment.
As stated there is a need for a analog signal conditioning block when switching from harvesting mode to measuring mode. To achieve this there should be a switch decoupling the piezo element from the rest of the circuitry and only to the MCU. A relay is able to switch higher voltages than the supply voltage but these elements are quite large in size and drain a lot of energy, which are both limited. There are analog switches that could be used, these are high voltage analog switches and require not a lot of power. However these switches were out of stock for a longer period so it was decided to design two PCBs, one with a relay as switch and one with a manual switch outside of the hip. In this way the power consumption from the relay can be measured and used to verify whether the circuit is able to switch from harvesting to measuring mode remotely. The other circuit can be used to determine the power

consumption without a high power consuming relay. And thus be used to get an estimate of the power consumption when a low power high voltage analog switch is used.

As stated the maximum voltage is only around 20 volts instead of the earlier mentioned 45 volts. From this information the values or R1 and R2 in figure 2.20 can be set at 9KOhm and 1KOhm respectively. This results in an attenuation of a factor 10. During measuring mode a resistor is placed between the positive and negative electrode of the PE so a similar configuration to the load testing with only the load resistor is achieved. In the results of the load testing(figure 3.2) could be seen that the higher the load resistance the better the linear relation between the force input and voltage output was. For this reason there is chosen for a 10MOhm resistor. Another thing that was seen from figure 3.2 is that the voltage goes from positive to negative, that is why the voltage from the PE has to be measured from both electrodes creating a P_+ and a P_- readout channel.

From the block diagram seen in figure 3.14, there can be clearly seen how many input and output connection points there have to be on the PCB. Because every line that crosses the dotted outline of the PCB should have a connection point. That means 2 from the PE,2 from the battery, 4 for the two sensors, 4 to program the MCU, 2 input from the EH300 and 2 outputs from the EH300. So a total of 16 connection points are needed for the design with the relay. The design without the relay needs two less connection points as the two inputs from the PE and the two inputs from the EH300 will be merged into two inputs after an external switch, so a total of 14 connection points.

The full schematic can be found in figure B.1 in appendix B. Also the corresponding bill of materials can be found there in table B.1.

After placing and routing all the components on the PCB the PCB design with the relay was already at the maximum size that would fit inside the implant. The PCB without the relay had some extra space that could be used so some testpoints, i.e. large vias, were incorporated to be able to measure and debug some channels. There was chosen to place the testpoints at the Power input, GND, P_+ and P_- channel. The result of both the PCB designs can be seen in figure 3.15 and figure 3.16. Also, the relay is highlighted in figure 3.15. As can be seen it takes up a lot of space relative to the other components.

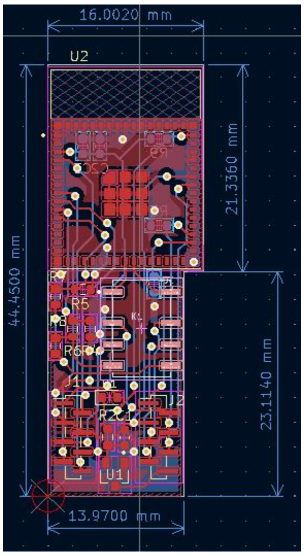

Figure 3.15: PCB with relay in KICAD editor

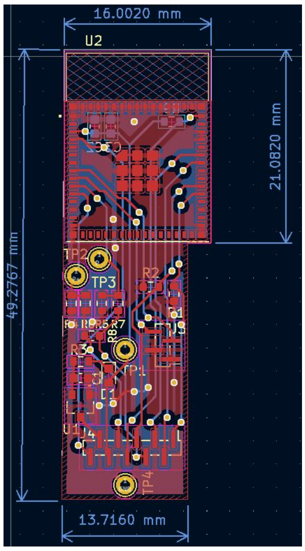

Figure 3.16: PCB without relay in KICAD editor

3.5. Verification of PCB

During the verification of the PCB multiple functionalities will be examined as described in section 2.6. As the experiments were similar to the first load testing the results will also be compared in the discussion.

3.5.1. Power generation

To investigate if the power generation for the designed PCBs was similar to that of the first configuration the EH300 was connected to the PE and PCB. The voltage at the input of the EH300 was measured and the voltage at the power input of the chip, so after the voltage regulator and protection circuit. The results of this experiment can be seen in figure 3.17.

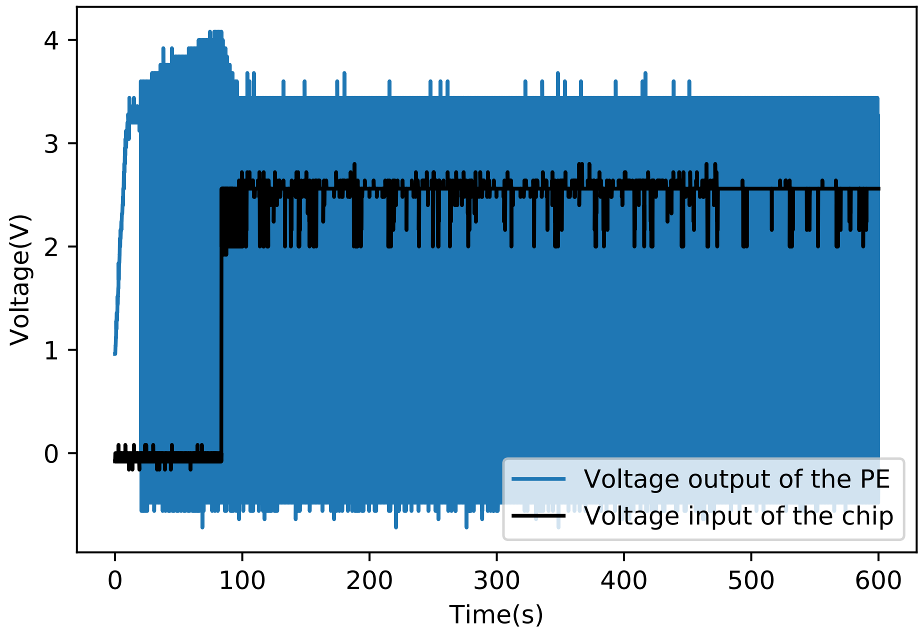

Figure 3.17: Voltage at the input of the EH300 and at the input of the Chip

Here it can be seen that in the first 100 seconds of the measurements the capacitor of the EH300 is charged and when fully charged, at the peak voltage of 4V, it will start to deliver power to the PCB. But as the voltage will be too low to start up the MCU, the energy will not be consumed but will be conserved. The voltage is too low because of the protection circuit after the LDO voltage regulator, this circuit is made of a resistor and a diode. The diode has a voltage drop of around 0.3 Volts and the rest of the voltage will be lost through the resistor. As this issue is caused by the protection circuit after the LDO voltage regulator the test was redone with the resistor and diode removed from the circuit. With this new configuration the voltage after the voltage regulator was indeed 3.3 Volts so enough to either charge the battery or power the chip for a short amount of time.

Also it was seen that the slope at which energy built up in the EH300 seemed a lot higher than achieved during first testing. As the power generation in the EH300 is not influenced by the PCB this would be the result of an error during the first test. To verify that this was the case 2 tests were redone from the first experiments. The test where the PE was attached to the EH300 and the EH300 was delivering it's power to a 10kOhm load and to a 300kOhm load. The results are shown in table 3.6.

Table 3.6: Delivered Energy in 500 seconds of harvesting for different load resistors

Load Resistance($K\Omega$)	Delivered energymJ)
9.83	145
326	140

3.5.2. Power consumption

During the power consumption test the input power was measured from the voltage generator. For this multiple settings were investigated, the setting were only measurements were taken and the setting were the data was sent. This was done for both the PCB configurations, so for the PCB with relay and the PCB without relay. In figure 3.18 and figure 3.19 the power consumption for reading the data can be seen for both PCBs.

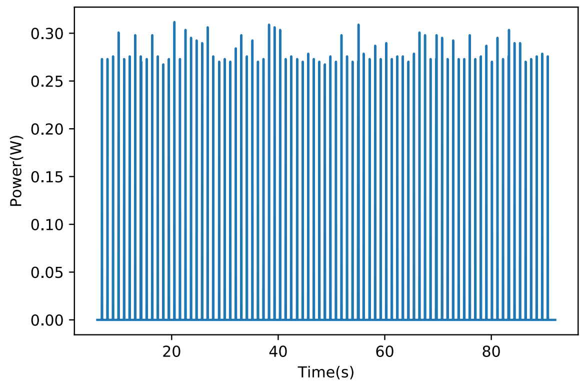

Figure 3.18: Power consumption during reading of sensor data for the PCB without relay

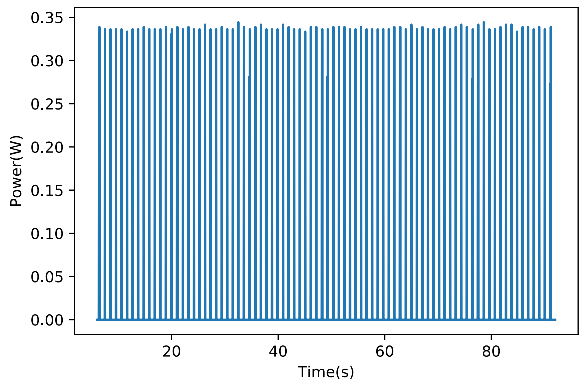

Figure 3.19: Power consumption during reading of sensor data for the PCB with relay

From figure 3.18 and figure 3.19, it can be seen that the PCB with the relay consumes more power than the PCB without. The difference between them is around 50 mW per sensor reading on average. In figure 3.20 and figure 3.21 the power consumption of the circuit can be seen while sending data.

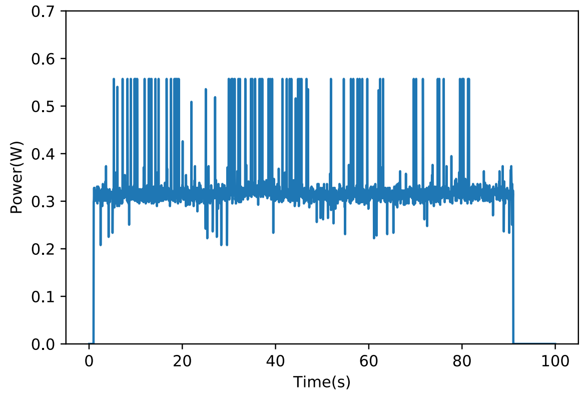

Figure 3.20: Power consumption during sending of sensor data for the PCB without relay

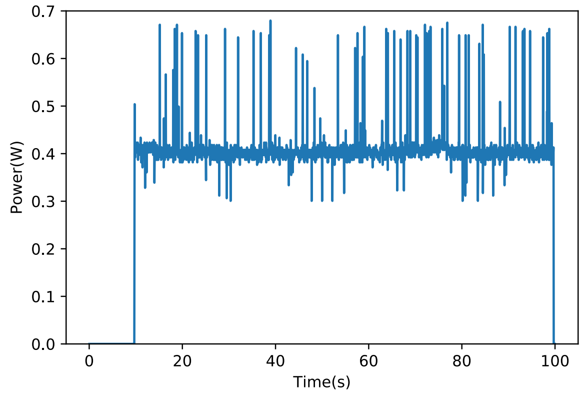

Figure 3.21: Power consumption during sending of sensor data for the PCB with relay

Again it is seen that the power consumption for the PCB with the relay is higher as without the relay. To also see the influence of the relay on the power consumption, there was also a test done were the relay was activated and inactivated. These results can be seen in figure 3.22. Here it can be seen that it is indeed only the influence of the relay and not because of other configuration of the board that the power consumption is higher. In figure 3.23 the power consumption of the PCB when the relay is active is also plotted on top of the power consumption when the relay is turned on and off. Here it can be seen that the power consumption is the same when the relay is active and slightly lower when the relay is inactive.

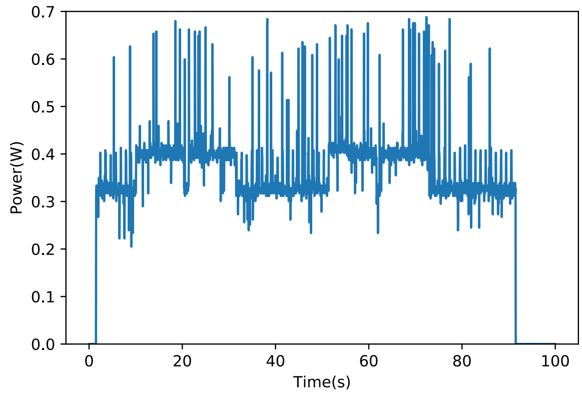

Figure 3.22: The power consumption when the relay is switched through a signal from a phone

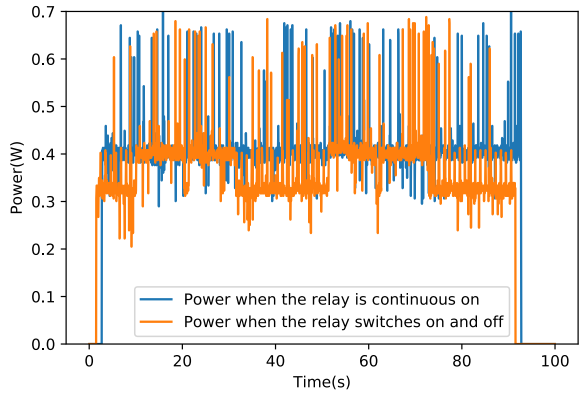

Figure 3.23: Power consumption of the PCB when the relay is active on top of the figure when it is being switched

In table 3.7 the average power consumption during sending and reading can be seen.

Table 3.7: Consumed power during reading and sending of the data with the standard deviation(σ)

PCB	Power consumption reading(mW)	σ	Power consumption sending(mW)	σ
Without relay	282.12	12.329	321.19	28.98
With relay	333.32	16.028	402.83	57.09

3.6. PE as force sensor

To verify how accurate the PE will be as a force sensor multiple measurements had to be taken. The voltage at the input of the PCB, the voltage at the input of the chip and the received digital values on the phone. The first two voltages were measured to compare whether the voltage divider does indeed attenuate the voltage with a factor of ten. And the voltage at the input is compared with digital value read by the phone to verify if the output of the phone correspond with the correct input value.
In figure 3.24 the voltage at the input is multiplied with a factor ten and compared with the input from the piezoelectric element. It can be clearly seen that these are about the same so the voltage division works correctly.

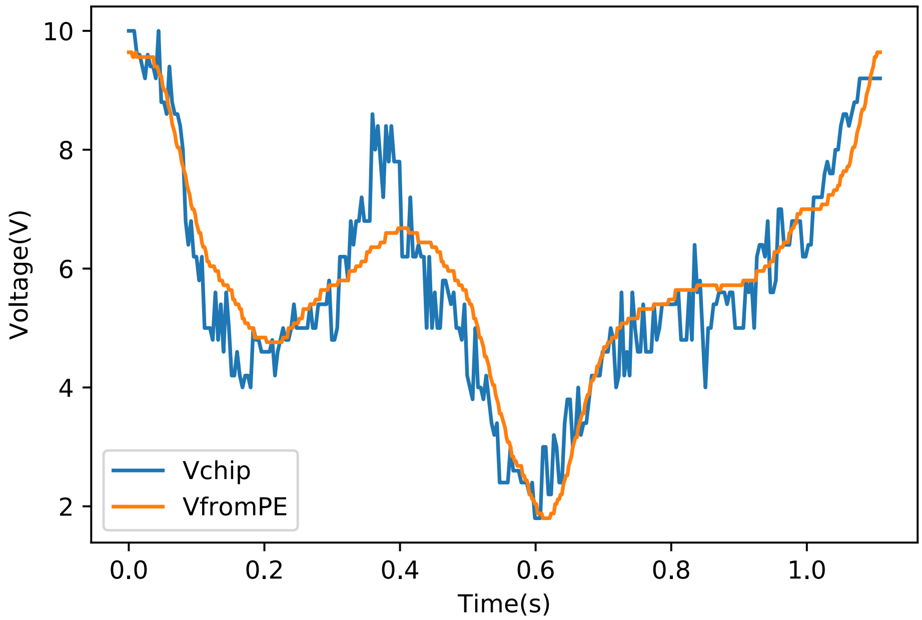

Figure 3.24: Voltage at the input of the chip compared with the voltage from the piezoelectric element

The next step is to compare the digital output of the chip received at the phone with the voltage input at the chip. These results can be seen in figure 3.25.

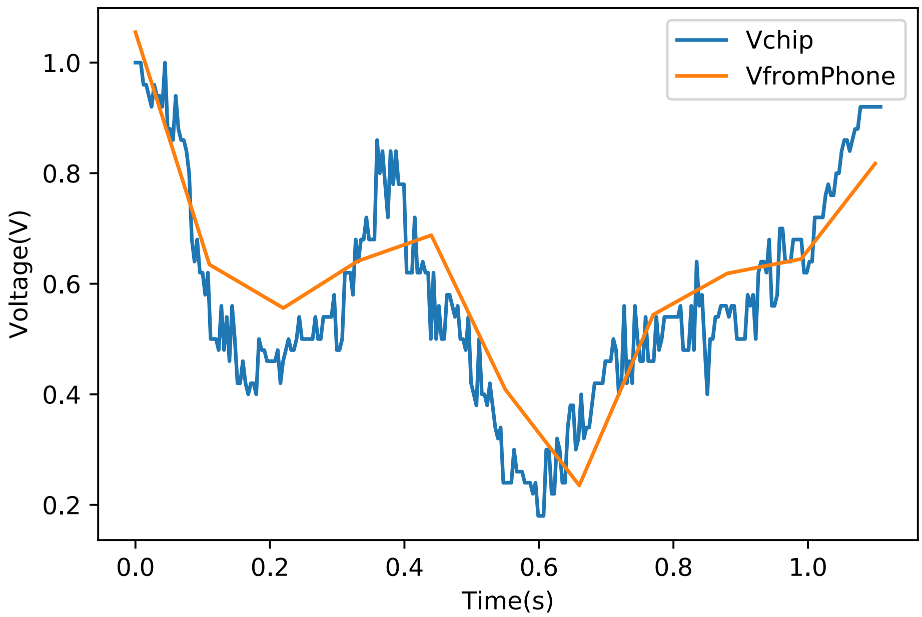

Figure 3.25: Voltage at the input of the chip compared with the signal received from phyphox converted to voltage

In this case it looks almost the same as when connected to a 10kOhm resistor, both in waveform and magnitude. This is most likely due to the fact that the voltage division is build up of a 9kOhm resistor and 1 kOhm resistor. To investigate whether this was the case the measurements for this was done again where the 9kOhm resistor is replaced with a 10 MOhm resistor and the 1kOhm with a 1MOhm resistor. In figure 3.26 the calculated force compared with the machine output force can be seen with the new configuration.

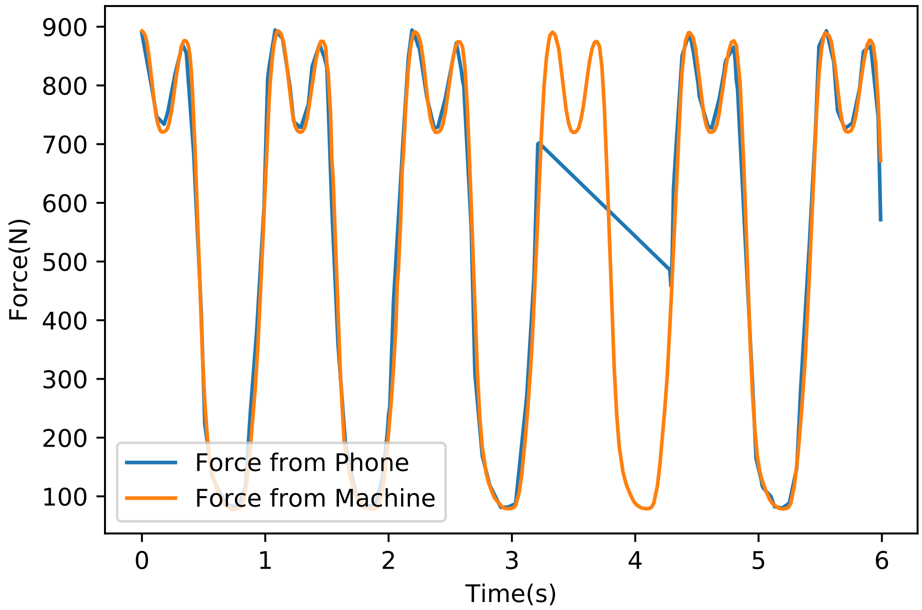

Figure 3.26: Force calculated from the data send to the phone compared with the machine output data

From the figure it can be seen that the calculated force from the data send by the phone matches the machine output data very well. After around 3 seconds in the figure there can be seen a linear line in the data from the phone. This is caused by the way the data is collected. When the PCB receives a 1 from the phone it knows it has to go into measuring mode and switches from harvesting to measuring mode. Then for a period of ten seconds the data from the PE will be send to the phone with a frequency of 20Hz. After those ten seconds it will automatically switch back to harvesting mode and after one second it will check again what the value send from the phone is. As the value was kept at one during measuring it will start to measure the voltage from the PE again. The force based on the data of the phone was calculated by filling in formula 2.10 using Linear regression. The following formulas were used to translate both channels into usable data.

$$Datacurve = P_{+} - P_{-} + max(P_{-}) \tag{3.2}$$

$$Force(N) = Datacurve * 0.45182558 + 65.68531091; \tag{3.3}$$

In formula 3.2 P_{+} is the output from one electrode of the PE and P_{-} is the output from the other electrode of the PE as explained in section 3.4.1. The results from these force measurements can be seen in figure 3.27. It can be seen that both channels need some time to settle to constant maximum values. To make sure that the calculations were not affected by this the NRMSEs were only calculated from the time the output was settled. However all data could be used as the total voltage in the beginning and end is the same as can be seen in figure 3.28 and figure 3.29 respectively. The explanation on why the voltage needs time to settle will be discussed in chapter 4.

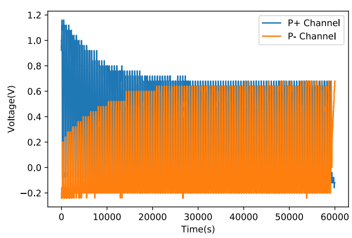

Figure 3.27: Clear difference can be seen between the voltages for the different channels at the beginning

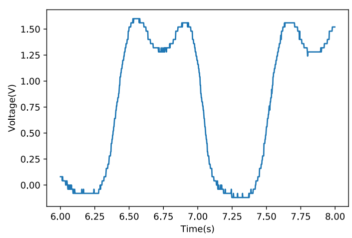

Figure 3.28: The produced output of the data at the beginning

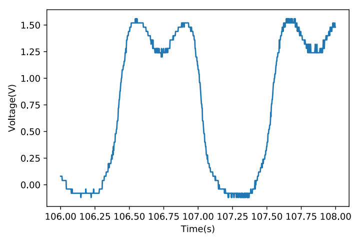

Figure 3.29: The produced output of the data at the end

From the test with 0.9Hz and the normal gait cycle the NRMSE was 0.0403 with a σ of 0.00671. To verify whether this coefficient is also accurate for other cases, tests were done with different frequencies and forces as stated in section 2.6.3. The results can be seen in table 3.8.

Table 3.8: Normalized Root Mean Square Error(NRMSE) for measured forces at different loading conditions

Condition	NRMSE	σ
0.45 Hz	0.0314	0.0131
0.9 Hz	0.0403	0.00671
0.8 Hz	0.0369	0.00555
1.0 Hz	0.0447	0.00655
1.8 Hz	0.0798	0.0108
0.9 Hz Low force	0.114	0.0102

3.7. End result

The last part of this project was to make sure everything could fit in the hip implant(figure 3.30). To make sure it fitted the capacitors of the EH300 were replaced with a smaller version and placed at the bottom of the implant. The batteries are placed inside the bottom stem which is also hollow for a small part. After that the PCB is placed on top of the EH300. The last step was to connect the wires. In figures B.3 and B.4 in appendix B two different views can be seen of the implant with the circuit. One where only the EH300 is inside, and one where the PCB is placed on top but without the head of the implant.

Figure 3.30: The final result with the wires connected

4
Discussion

In this section first a table is presented which gives an overview of what is achieved in this research together with comparable projects, see table 4.1. After that the results will be discussed in more detail. After the results are discussed there will be a section that focuses on how this research can move forward.

Table 4.1: Overview of characteristics of different smart implants

Research	Implant type	Power source	Power Harvested	sensors used	Communication	Power consumption
This research	Hip	PE and Battery	877.31μW	Temperature and Force	BLE	300-320 mW
Bergmann et al. [17]	Hip	Inductive	-	Temperature and Force	Inductive	5mW
lange et al. [14]	Hip	PE	729.9 μW	-	-	-
safaei et al.[12]	Knee	4 PEs	12μW	Force	-	-
Almouahed et al.[25]	Knee	PE	4.28 mW	Force	-	-
Soares et al. [26]	Hip	PE and electromagnetic	567.4μW	-	-	-
Gaalen et al. [15]	Hip	Battery	-	Accelerometer	Bluetooth	-

4.1. Results discussion

4.1.1. Load testing

This PE gives an average output power of 877.31μW during one gait cycle. Compared to some energy harvesting systems in orthopedic implants this is a lot more, 12 μW [27] for a knee implant, 0.6 μW for a hip implant [26]. But there are also studies that score higher or similar. A study that scores higher is by Almouahed et al. where they started with a prototype that could harvest 1.81mW and later 4.28mW [25] . In this configuration they were able to place four piezoelectric elements in the knee meaning that on average each element could produce 1.07mW. A piezoelectric element that was custom made for an instrumented hip implant by Lange et al. scores in similar range [28]. They were able to harvest 729.9μW. From this research could also be seen that by adapting the piezoelectric element to a specific design the power output could be improved a lot. As in their first design they were only able to harvest 29.8μW.

In figure 3.5 the average power to load resistance was plotted. The results seen are comparable with what is expected, a quick rise in average power to go down smoothly. In other harvesting circuits similar curves are seen [12, 14]. To compare the value of the load at which maximum power is reached with other research is not valid. This can only be done if those researches had used the same element. For a more accurate maximum power point(MPP), there could have been more resistor values tested with values around 163KOhms. There are also other techniques to achieve the MPP, these are very well documented in a paper by Brenes et al. [29].

During the design of this research there was a PE used that was not able to sustain forces higher than 900N. It was not possible to see how much power could be harvested when loading the PE with the actual force acting on the hip during a normal gait cycle. This is because that would destroy the element. Another PE is available by the same supplier which can sustain forces up to 1960N, this is the PK4GYP1. This force would be high enough when using the standard load set by Orthoload [22] for an average human with a body weight of 75Kg. But for higher loading sets it is also not enough. The d_{33} is also higher namely $2.1 * 10^5 pC/N$ compared with $6 * 10^4 pC/N$ of the current PE. The current element is hermetically sealed where the PK4GYP1 is not meaning that it wouldn't be suited to implant this element with the current design. In the end it was opted to not also investigate how much power could be harvested with the other element with in the back of the head that a custom PE could improve the results more, as seen in the research by Lange et al. [28]. A recommendation is to look into a custom piezoelectric element which would be best suited for this application.

In the methodology the theoretical power generation was calculated to be $2.13mJ$ per gait cycle. If all this energy can be delivered during one gait cycle this would mean an average power output of $1.92mW$ per gait cycle. This is more than double as the highest measured average power of 877.31μW. Multiple reasons can be the cause of this. The first is that the load impedance is not exactly the same as the internal impedance of the piezoelectric element which would cause a less than optimal power transfer. Other options are that some of the force is transferred into the stainless-steel tube around the piezoelectric element or there is charge leakage in the piezoelectric element or the oscilloscope.

In figure 3.8 and figure 3.9 it was seen that by changing the loading frequency the amount of power harvested during a gait cycle also changed. This was expected as the impedance of the PE is frequency dependant as seen by formula 2.8. This is something that has to be taken into account when choosing for the optimal technique for achieving the MPP, this technique should be able to work with alternating input impedance as the walking pattern of most people is very irregular.

There was seen in table 3.2 that the loading resistance at the output of the EH300 was not of influence to the energy harvested. This is as expected as the load resistance is not connected to the circuit until the EH300 closes the switch. When it is connected the influence will be minor as the resistor will then be in parallel to the capacitor which has a impedance of around 27 Ohms. As the loading resistors have a higher impedance, all around factor 10 or higher, there is almost no change in equivalent impedance of the total circuit.

As mentioned in section 3.5.1 during the validation of the PCB the slope at which energy was build up in the EH300 seemed a lot higher than during first testing. As the power generation in the EH300 is not influenced by the PCB this would be the result of a error during the first test. To verify that this was the case 2 tests were redone from the first experiments. The test where the PE was attached to the EH300 and the EH300 was delivering it's power to a 10kOhm load and to a 300kOhm load. These values were chosen so both the scenarios were represented. Not all resistor values were redone because it was seen that the load resistance didn't influence the amount of energy harvested. From these tests it was seen that the power generated was more than doubled. The cause for this can be explained to the placing of the oscilloscope probes. During the first testing the probe was placed between the poles of the input of the EH300, this causes no problem when no other probes are used. But when multiple probes are used the oscilloscope makes a common ground, so it pulls all grounds of the probes the same. This would probably result in only harvesting half of the energy, only when one side is more positive and the other not as this side will be pulled to ground by the probe.

On average there was seen that in 500 seconds there was 140 mJ harvested from the PE. This means that per gait cycle the average power would be 311μW. This is lower than the maximum that was seen in the results before. The reasons for this can be that the EH300 is not load matched to the piezoelectric element or that there is energy lost in the EH300 itself. An improvement would be to design a matching circuit to harvest the maximum power from the piezoelectric element. Advised is to first design a new PE and after that design a matching harvesting circuit accordingly. In this way not only the harvesting efficiency can be improved but also there can be changes in space as the designed PCB and EH300 doesn't have to be stacked anymore.

The piezoelectric element is used as a force sensor. It is seen that the NRMSE of the force sensor is 0.0519 when the voltage over the 975KOhm resistor was used. From the data send to the phone the NRMSE was 0.0403 for the loading cycle with 0.9Hz and 900N maximum force. The improvement can be explained because now the piezoelectric element is placed over a load of 10MOhm instead of the

975KOhm. The voltage over the higher resistor follows the gait cycle much better than for lower values. In figure 4.1 the voltage over the 10MOhm resistor is compared with the voltage over the 975KOhm resistor. It can be seen that especially in the second peak there is an improvement when the PE is connected to the 10MOhm load.

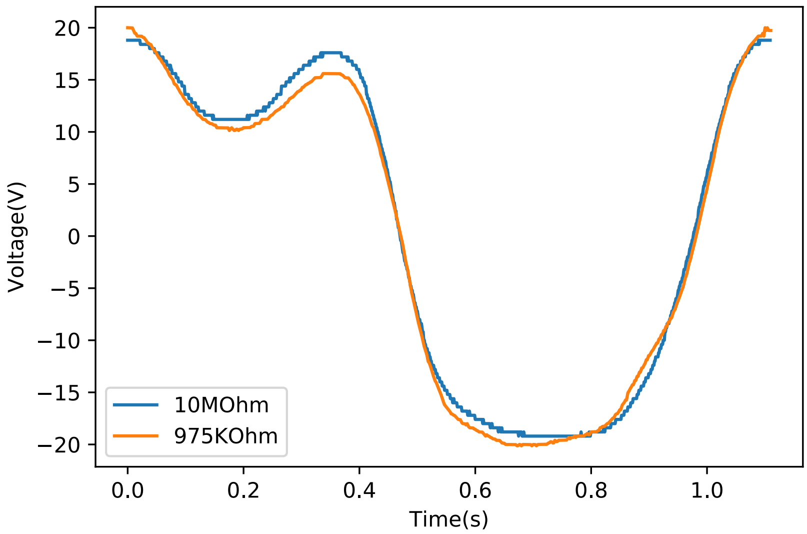

Figure 4.1: Voltage over 10Mohm load resistor compared with a 975KOhm load resistor

A shortcoming in this research is that the force on the PE is placed perpendicular. This was done because the element is very susceptible to shear forces. However, this also means that the accuracy of the PE as a force sensor is not representative when using it during normal walking. It is yet to be seen how the element behaves when loaded under an angle. Especially it is important to see if a small load can give the same output as a high load under an angle. If this will be the case a solution to distinguish these forces, i.e. high load under angle and low load perpendicular, could be to incorporate a gyroscope in the design. In this way the angle of the hip can be determined and from that information the actual force on the hip calculated based on an hip angle to force translation factor.

During the force measurements, see figure 3.27, there was seen that the PE needed some time to settle the voltage output to a steady output at both sides of the element. This is caused by the time constant of the circuit. The time constant of the circuit is R*C, with R being the load resistance between the electrodes and C the capacitance of the PE. Without going into much detail, based on the formulas for charging and discharging a capacitor it can be said that it charges to around 100% in 5 times the time constant.

$$V_C = V_S * (1 - e^{-\frac{t}{RC}}) \tag{4.1}$$

$$V_C = V_S * e^{-\frac{t}{RC}} \tag{4.2}$$

Here is V_C the voltage of the capacitor, V_S the source voltage and t the time.
The time constant of the circuit which is used while using the PE as a force sensor is 12s, meaning that it would take 60 seconds to completely charge or discharge the capacitor of the PE. This is also the time it takes for the voltage output to settle and discharge at the end in figure 3.27, so it can be concluded that the time to settle comes from choosing a very high resistance. This time constant is also the explanation why the charge build up over the 218Ohm and 10KOhm resistor was so small and had extra zero voltage crossings, as seen in figure 3.3 and figure 3.4. For the 218Ohm resistor it would only take 1ms to completely discharge and for the 10KOhm resistor it would take 60ms. Whereas from the 163KOhm resistor this would take 1s meaning that it wouldn't discharge a lot before a force was applied on it again.

4.1.2. MCU consumption testing

From the results could be seen that the power consumption of the development board was around 300mW when only measuring and 320mW when also sending. When doing the power consumption measurements with the PCB without the relay it improved slightly, 282.12 mW when measuring and during sending the result was the same namely 321.19mW. The results for the PCB with the relay were worse than both the development board and the PCB without the relay. It was concluded that the power consumption of the PCB with the relay was 50mW higher when only reading and 80mW higher when sending. This power is lost in the coils of the relay. However this is a significant increase in power consumption this is not as high as the rated value of 140mW according to the datasheet [30]. Although this increase is less than expected it still confirms that a circuit with a relay as switch is not preferred. So should the high voltage analogue switches be available again these get the preference over the relay.

When comparing the power consumption with circuits for similar application this circuit consumes on the high side. In a paper by Soares dos Santos et al. [31] they mention the average power consumption of a few designs in other researches. Most of these range from 5mW till 30 mW, but there is also a research which consumes 500 mW. So in terms of power consumption there are steps to be made.

One of the main power consuming modules inside the PCB is the MCU. As said in section 2.2.4 the decision for the ESP32s3 was made because it was recommended to start with and because of the widely available support for it. The downside is that this is not the best low power MCU there is on the market. But on the other hand it is a very easy and approachable MCU with a lot of modules available ready to be used. In this way it was easier to see what is possible within the hip and how to control certain aspects. However this means that in terms of power consumption there are steps to be made. More low power MCU's would be MSP430 [15], RE01B [32] or ATM33e [33]. From the datasheets there can be assumed that the power consumption will range between 10mW and 33 mW when active for these 3 MCUs [34, 35, 36].

During the power consumption test of the development board there was seen that the addition of extra passive sensors seemed to result in less power consumption. This is a strange phenomenon because to use the thermistors as a sensor the MCU sends power from a general output pin trough a Thermistor and a resistor. As the thermistor has a resistance of about 10KOhm and the reference resistor also has a resistance of 10KOhm the total resistance would be 20KOhm. As the GPIO pins have a voltage of 3.3 Volt this would mean that the extra power consumed by the sensor circuit would be $P = (3.3)^2/20000 = 545\mu W$. As the total amount of energy consumed is around 300mW the addition of one sensor would not have a big impact. So the decrease seen in power consumption can be more explained to the noise than it is an effect of adding sensors. Because adding extra passive sensors that require energy to perform should always result in extra energy consumption. However when another MCU is going to be used the power consumption of the sensor will become of influence.

During the measurement of the power consumption when sending data a clear peak was expected when sending data over Bluetooth, but this was not the case. One of the reasons is that the general power consumption of the chip is so high that adding a function will not results in a significant increase in energy consumption. Another reason could be that the code for the BLE connection is not well written. The connection should only be made while sending data and after that the connection should be closed. When the connection is not closed the power consumption will also not decrease. Some alternatives for the code have been used but each version resulted in about the same power consumption without clear peaks.

Another issue that can affect the power consumption can be the distance at which the phone is from the chip. For these experiments the phone was placed at a distance of 10 cm from the chip. This is similar to a phone placed inside your pocket next to you hip. There was not looked at other distances but to send data further there is need for more energy so more energy consumption. The datasheet reports an increase of 70mA in peak current consumption when increasing the signal from 17dBm to 20.5dBm [24].

4.1.3. PCB design

A major complication for the PCB design was the availability of products. Specifically, in this design there is made use of a relay instead of an high voltage analog switch. If a high voltage analog switch were to be used this could significantly improve the performance both in size and power consumption. However during this project each high voltage analog switch that was suitable to use was out of stock at multiple suppliers. As the switches are either normally open or normally closed the high voltage analog would need power during harvesting and measuring mode. In this way it influences the total power consumption of the circuit unambiguous of harvesting of measuring mode. So to really measure whether it would make a big difference on the performance this should still be tested.

The current PCB design has a two layer PCB configuration, this means that the front and back can be used to route signals from component to component on the chip. As for this design the backside of the PCB is empty with components but the routing is quite full already an improvement in the future could be to use more layers in the design. This would create more space for routing so more elements could be included. Another option would be to stack another PCB on top, this could only work if the EH300 would be integrated into the PCB design as currently the height limit is reached.

While placing the PCB with the EH300 inside of the implant some complications were encountered. The first problem faced was that for the connection points the BC070 connector sockets from GCT were used. These are normally used for board to board connections and not to attach wires to it. It is possible to connect wires to it but it is very prone to also connect to the neighboring connector pin. Also the stability at which the BC020 connects to the BC070 is less as expected. Alternative for the connection points is the 1-2367197-0 by TE connectivity. Another point of interest taken during the placement inside the hip was that if it is possible to be able to integrate a module/component on the PCB to do it. This is because when a component is not on the PCB it has to be connected via a wire, as all these wires together can take up a lot of space there is need for minimization.

In the current design there was chosen to made use of BLE for the communication between the implant and the outside. BLE is a wireless communication technique that operates at a frequency of 2,4GHz. But as frequencies above 1 GHz experience a lot more scattering than frequencies below this when sending signals through human tissue this can cause some problems [37]. To test how it would behave the circuit could be placed inside a hip implant inside a phantom of human tissue. From this can be determined if the signal will be strong and reliable enough to use BLE. If this is deemed not good enough there are also different MCU's available which can send at sub GHz frequencies, an example is the CR1312R by texas instruments [38]. But as a research from Christoe et al. deemed Bluetooth a promising way of sending information from within the body [39] this is probably not necessary.

4.1.4. Putting numbers into perspective

In the results there was seen that every 500 seconds there is 140mJ of energy harvested. With a gait cycle of 1.11 seconds per gait this means that in about 450 gait cycles 140mJ is harvested. In one gait cycle there are two steps, this means that is takes 900 steps to generate 140mJ. In the United States the average steps per day is about 5100 steps whereas another study showed that the average amount of steps in Switzerland is about 7000 steps per day[40][41]. As there are large differences between countries, age, gender and BMI the choice is made to work with the lower amount of 5000 steps per day. With 5000 steps per day the total amount of energy available per day is $\frac{5000}{900} * 140 = 778mJ$. It was seen that reading data costs around 150μJ and sending the data costs around $90mJ$. If the data is send every eight hours and every minute the sensors are read the total energy consumption per day will be $90 * 3 + 480 * 3 * 0.15 = 486mJ$. As seen there is room to use some more energy but as one reading per minute will be more than enough for diagnostics, this extra energy can be used either for adding additional functionalities to the design or to use when the circuit is switched from harvesting to measuring mode as one measurement of 10 seconds consumes $10 * 300 = 3000mJ$.

4.2. Future research

In the first part of the discussion there was already looked at some parts that need to be investigated in the future. In this section those options will be further elaborated and some more points of interest added.

The most important future research would be to have a piezoelectric element that is suitable to sustain enough force and is optimally designed for the hip implant. There was seen that by adapting the PE to a specific application this could significantly improve the power harvested. During this design process it is wise to take the shear forces into account for the placement of the PE. This has to be done because PEs are fragile when loaded with shear forces. There are also PEs that are specifically build to use shear forces but this will limit the element in the transverse direction. Although the piezoelectric shear coefficient is higher than the coefficients for axial and transverse mode [42] in the current configuration it would probably still be best to look into an element in transverse mode. To get a better picture on how the forces will act on the hip implant first computational simulations must be done. Based on those results the PE can be designed accordingly.

The next step would be to design a matching harvester circuit which will be able to match the piezoelectric elements impedance for maximum power transfer. In a recent article by Li et al. [43] multiple circuit designs are analysed. It was seen that synchronized switch harvesting on inductor (SSHI) circuits, synchronous electric charge extraction (SECE) circuits, inductor-less circuits and synchronized multiple-bias-flip (SMBF) circuits could be used to enhance the performance of the Piezo electric harvester. With the SSHI being able to harvest more energy than the SECE, Shen et al. even showed in a simulation a maximum MPP tracking efficiency up to 99.1% [44]. But the SECE doesn't require an additional circuit for impedance matching this results in less volume of the system. The paper by Brenes et al. helps designers with a choice for each available technique by presenting the advantages and disadvantages of each design. The advantage of designing this yourself and not using it as a module is that it can be included on the custom PCB requiring less connection points and more options in the use of the space that is available.

As the current design of the altered hip implant is not yet mechanical tested it is important to investigate whether the design is strong enough to be used inside the body. As the upper part of the hip is made hollow without any mechanical testing or modelling it could be that some alterations have to be made to strengthen the implant. It can be that these reinforcements take away some space which can't be used anymore for the placement or wiring of the circuit. In the best case some additional material can be taken away and more space will be available.

In the circuit design there is now made use of the MCU in module form with attached antenna. By designing a custom antenna there is no need to use the module anymore and the width at the top is only bound to the size of the chip, which for most MCU chips is 7mmx7mm.

In this research the goal was to look if it would be feasible to have an electronic circuit inside a hip implant based on on the market modules. There was not looked at restrictions from a medical regulations perspective. There are a lot of rules to comply to before you may implant something in a body. For this a CE mark has to be given to a certain product. In the future designs it is recommended to consider MDR related requirements as well.

5 Conclusion

During this research the goal was to design an electronic system which fitted inside a hip implant. The end result of this project was a circuit that could fit inside the implant and was able to accurately send information to a phone. The current configuration works with a PE harvester from Thorlabs inside the neck of the implant. The energy harvested will be collected by the EH300 module from Advanced linear devices which is connected to a custom made PCB. On the PCB the ESP32-S3-MINI module is present to be able to measure and send information from two thermistors. Also the PE can be used as a force sensor when the MCU receives a signal from the phone to switch from harvesting to measuring mode. It was seen that frequency at which the PE was loaded influenced the amount of energy harvested, where a higher frequency resulted in more energy harvested per gait cycle. Although the current configuration is able to measure and send information from within the hip to a phone, there are improvements to be made.

References

[1] Bhakti Shah. *Joint Replacement Surgery*. 2023. URL: https://rheumatology.org/patients/joint-replacement-surgery.

[2] Jasvinder A. Singh et al. “Rates of Total Joint Replacement in the United States: Future Projections to 2020–2040 Using the National Inpatient Sample”. In: *The Journal of Rheumatology* 46.9 (2019), pp. 1134–1140. ISSN: 0315-162X. DOI: 10.3899/jrheum.170990. eprint: https://www.jrheum.org/content/46/9/1134.full.pdf. URL: https://www.jrheum.org/content/46/9/1134.

[3] Bozic KJ et al. “he epidemiology of revision total hip arthroplasty in the United States”. In: *Bone Joint Surg Am* (Jan. 2009). DOI: 10.2106/JBJS.H.00155.

[4] A Postler et al. “Analysis of Total Knee Arthroplasty revision causes.” In: *BMC Musculoskelet Disord.* (Feb. 2018). DOI: 10.1186/s12891-018-1977-y.

[5] Grayson Kelmer et al. “Reasons for Revision: Primary Total Hip Arthroplasty Mechanisms of Failure.” In: *Journal of the American Academy of Orthopaedic Surgeons* 29.2 (Jan. 2021), pp. 78–87. DOI: 10.5435/JAAOS-D-19-00860.

[6] D. Capón-García, A. López-Pardo, and M.T. Alves-Pérez. “Causes for revision surgery in total hip replacement. A retrospective epidemiological analysis”. In: *Revista Española de Cirugía Ortopédica y Traumatología (English Edition)* 60.3 (2016), pp. 160–166. ISSN: 1988-8856. DOI: https://doi.org/10.1016/j.recote.2016.03.004. URL: https://www.sciencedirect.com/science/article/pii/S1988885616300049.

[7] van den Kieboom J et al. “Periprosthetic joint infection is the main reason for failure in patients following periprosthetic fracture treated with revision arthroplasty.” In: *rch Orthop Trauma Surg.* 142.12 (Dec. 2022), pp. 3565–3574. DOI: 10.1007/s00402-021-03948-3.

[8] Utkarsh Anil, Vivek Singh, and Ran Schwarzkopf. “Diagnosis and Detection of Subtle Aseptic Loosening in Total Hip Arthroplasty”. In: *The Journal of Arthroplasty* 37.8 (2022), pp. 1494–1500. ISSN: 0883-5403. DOI: https://doi.org/10.1016/j.arth.2022.02.060. URL: https://www.sciencedirect.com/science/article/pii/S088354032200211X.

[9] Karthikeyan. P. Iyengar et al. “Significant capabilities of SMART sensor technology and their applications for Industry 4.0 in trauma and orthopaedics”. In: *Sensors International* 3 (2022), p. 100163. ISSN: 2666-3511. DOI: https://doi.org/10.1016/j.sintl.2022.100163. URL: https://www.sciencedirect.com/science/article/pii/S2666351122000080.

[10] C. O’Connor and A. Kiourti. “Wireless Sensors for Smart Orhopedic Implant”. In: *Journal of Bio-and Tribo-Corrosion* 3 (2017).

[11] Salil Sidharthan Karipott et al. “An Embedded Wireless Temperature Sensor for Orthopedic Implants”. In: *IEEE Sensors Journal* 18.3 (2018), pp. 1265–1272. DOI: 10.1109/JSEN.2017.2780226.

[12] Mohsen Safaei, R Michael Meneghini, and Steven R Anton. “Force detection, center of pressure tracking, and energy harvesting from a piezoelectric knee implant”. In: *Smart Materials and Structures* 27.11 (Sept. 2018), p. 114007. DOI: 10.1088/1361-665x/aad755. URL: https://doi.org/10.1088/1361-665x/aad755.

[13] James Davis et al. “Design and performance simulation of a triboelectric energy harvester for total hip replacement implants”. In: *Health Monitoring of Structural and Biological Systems XV*. Ed. by Paul Fromme and Zhongqing Su. Vol. 11593. International Society for Optics and Photonics. SPIE, 2021, pp. 232–244. DOI: 10.1117/12.2583343. URL: https://doi.org/10.1117/12.2583343.

[14] Hans-E. Lange et al. “Performance of a Piezoelectric Energy Harvesting System for an Energy-Autonomous Instrumented Total Hip Replacement: Experimental and Numerical Evaluation”. In: *Materials* 14.18 (2021). ISSN: 1996-1944. DOI: 10.3390/ma14185151. URL: https://www.mdpi.com/1996-1944/14/18/5151.

[15] Jolien B. van Gaalen et al. “Versatile smart hip implant technology using 3D metal printing”. In: *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*. 2016, pp. 2731–2734. DOI: 10.1109/ISCAS.2016.7539157.

[16] Thomas A. G. Hall, Frederic Cegla, and Richard J. van Arkel. “Simple Smart Implants: Simultaneous Monitoring of Loosening and Temperature in Orthopaedics With an Embedded Ultrasound Transducer”. In: *IEEE Transactions on Biomedical Circuits and Systems* 15.1 (2021), pp. 102–110. DOI: 10.1109/TBCAS.2021.3052970.

[17] Georg Bergmann et al. “High-Tech Hip Implant for Wireless Temperature Measurements In Vivo”. In: *PLOS ONE* 7.8 (Aug. 2012), pp. 1–7. DOI: 10.1371/journal.pone.0043489. URL: https://doi.org/10.1371/journal.pone.0043489.

[18] Bojing Shi, Zhou Li, and Yubo Fan. “Implantable Energy-Harvesting Devices”. In: *Advanced Materials* 30.44 (2018), p. 1801511. DOI: https://doi.org/10.1002/adma.201801511. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/adma.201801511. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201801511.

[19] Evgeny Katz and Paolo Bollella. “Fuel Cells and Biofuel Cells: From Past to Perspectives”. In: *Israel Journal of Chemistry* 61.1-2 (), pp. 68–84. DOI: https://doi.org/10.1002/ijch.202000039. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/ijch.202000039. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/ijch.202000039.

[20] Susie Maestre. *What is Piezoelectric Effect?* 2022. URL: https://www.circuitbread.com/ee-faq/what-is-piezoelectric-effect.

[21] *Hermetically Sealed Piezoelectric Actuator, -25 – 210 °C Operating Temp. Range, 150 V, 12.0 µm, PH24SRZW*. Thorlabs. 2021.

[22] Georg Bergmann et al. “Standardized Loads Acting in Hip Implants”. In: *PLOS ONE* 11.5 (May 2016), pp. 1–23. DOI: 10.1371/journal.pone.0155612. URL: https://doi.org/10.1371/journal.pone.0155612.

[23] *EH300/301 EPAD ENERGY HARVESTING MODULES*. Advanced Linear Devices, Inc. 2015.

[24] *ESP32-S3-MINI-1/ESP32-S3-MINI-1U/Datasheet*. Espressif. 2023.

[25] S. Almouahed, C. Hamitouche, and E. Stindel. “Optimized Prototype of Instrumented Knee Implant: Experimental Validation”. In: *IRBM* 38.5 (2017), pp. 250–255. ISSN: 1959-0318. DOI: https://doi.org/10.1016/j.irbm.2017.06.005. URL: https://www.sciencedirect.com/science/article/pii/S1959031817300660.

[26] Marco P. Soares dos Santos et al. “Multi-source Harvesting Systems for Electric Energy Generation on Smart Hip Prostheses”. In: *Biomedical Engineering Systems and Technologies*. Ed. by Joaquim Gabriel et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 80–96. ISBN: 978-3-642-38256-7.

[27] Mohsen Safaei, R. Michael Meneghini, and Steven R. Anton. “Energy Harvesting and Sensing With Embedded Piezoelectric Ceramics in Knee Implants”. In: *IEEE/ASME Transactions on Mechatronics* 23.2 (2018), pp. 864–874. DOI: 10.1109/TMECH.2018.2794182.

[28] Hans-E. Lange, Rainer Bader, and Daniel Kluess. “Design Study on Customised Piezoelectric Elements for Energy Harvesting in Total Hip Replacements”. In: *Energies* 14.12 (2021). ISSN: 1996-1073. DOI: 10.3390/en14123480. URL: https://www.mdpi.com/1996-1073/14/12/3480.

[29] A Brenes et al. “Maximum power point of piezoelectric energy harvesters: a review of optimality condition for electrical tuning”. In: *Smart Materials and Structures* (2020).

[30] *AXICOM IM RELAY*. TE Connectivity. 2023.

[31] Marco P. Soares dos Santos et al. “Instrumented hip implants: Electric supply systems”. In: *Journal of Biomechanics* 46.15 (2013), pp. 2561–2571. ISSN: 0021-9290. DOI: https://doi.org/10.1016/j.jbiomech.2013.08.002. URL: https://www.sciencedirect.com/science/article/pii/S0021929013003825.

[32] Renesas Electronics Corporation. *Renesas Adds Bluetooth 5.0 to Ultra-Low Power RE Family for Battery Maintenance-Free IoT Devices*. 2021. URL: https://www.renesas.com/us/en/about/press-room/renesas-adds-bluetooth-50-ultra-low-power-re-family-battery-maintenance-free-iot-devices.

[33] Atmosic Technologies. *ATM33 Series*. 2023. URL: https://atmosic.com/products_atm3/.

[34] *MSP430FR604x, MSP430FR504x 16-MHz MCU up to 64KB FRAM, 12-Bit High-Speed 8-MSPS Sigma-Delta ADC, and Integrated Sensor AFE*. Texas Instruments. 2021.

[35] *RE01B Group Product with 1.5-Mbyte Flash Memory*. Renesas. 2020.

[36] *ATM3201/ATM3221/ATM3231, Exteme Low Power Bluetooth 5.0 SoC with energy Harvesting*. Atmosic. 2022.

[37] Bradley D. Nelson et al. "Wireless Technologies for Implantable Devices". In: *Sensors* 20.16 (2020). ISSN: 1424-8220. DOI: 10.3390/s20164604. URL: https://www.mdpi.com/1424-8220/20/16/4604.

[38] *CC1312R SimpleLink™ High-Performance Sub-1 GHz Wireless MCU*. Texas Instruments. 2020.

[39] Michael J. Christoe et al. "Bluetooth Signal Attenuation Analysis in Human Body Tissue Analogues". In: *IEEE Access* 9 (2021), pp. 85144–85150. DOI: 10.1109/ACCESS.2021.3087780.

[40] David R.jr Bassett et al. "Pedometer-Measured Physical Activity and Health Behaviors in U.S. Adults." In: *Medicine science in Sports Exercis* 42.10 (Oct. 2010), pp. 1819–1825. DOI: 10.1249/MSS.0b013e3181dc2e54.

[41] Maria M. Sequeira et al. "Physical Activity Assessment Using a Pedometer and Its Comparison with a Questionnaire in a Large Population Survey". In: *American Journal of Epidemiology* 142.9 (Nov. 1995), pp. 989–999. ISSN: 0002-9262. DOI: 10.1093/oxfordjournals.aje.a117748. eprint: https://academic.oup.com/aje/article-pdf/142/9/989/200175/142-9-989.pdf. URL: https://doi.org/10.1093/oxfordjournals.aje.a117748.

[42] Mohammad Malakooti and Henry Sodano. "Piezoelectric energy harvesting through shear mode operation". In: *Smart Materials and Structures* 24 (May 2015). DOI: 10.1088/0964-1726/24/5/055005.

[43] Di Li et al. "Recent progress and development of interface integrated circuits for piezoelectric energy harvesting". In: *Nano Energy* 94 (2022), p. 106938. ISSN: 2211-2855. DOI: https://doi.org/10.1016/j.nanoen.2022.106938. URL: https://www.sciencedirect.com/science/article/pii/S2211285522000234.

[44] Jiahui Shen, Yinshui Xia, and Huakang Xia. "ReL-SSHI rectifier based piezoelectric energy harvesting circuit with MPPT control technique". In: *Microelectronics Journal* (2022).

A

Methodology

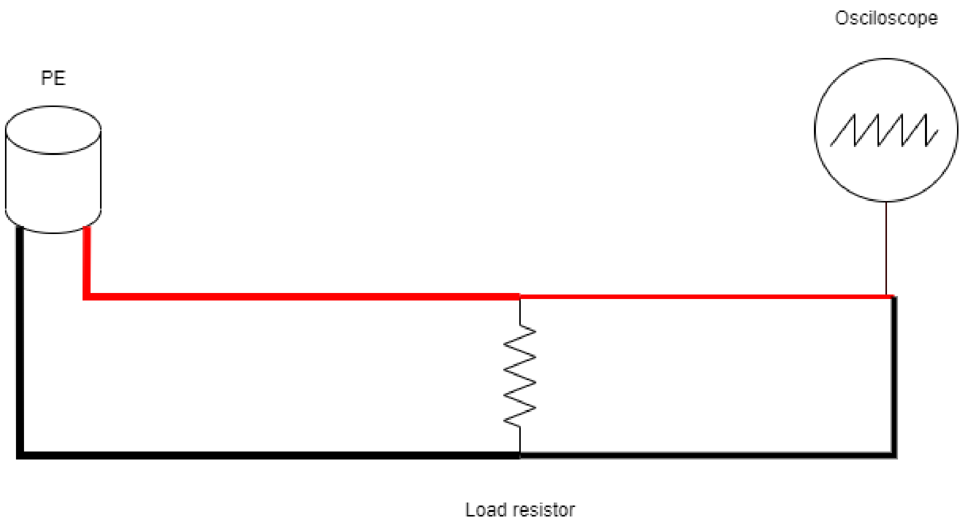

Figure A.1: Connection overview of the Loading test with the PE only connected to a load resistor

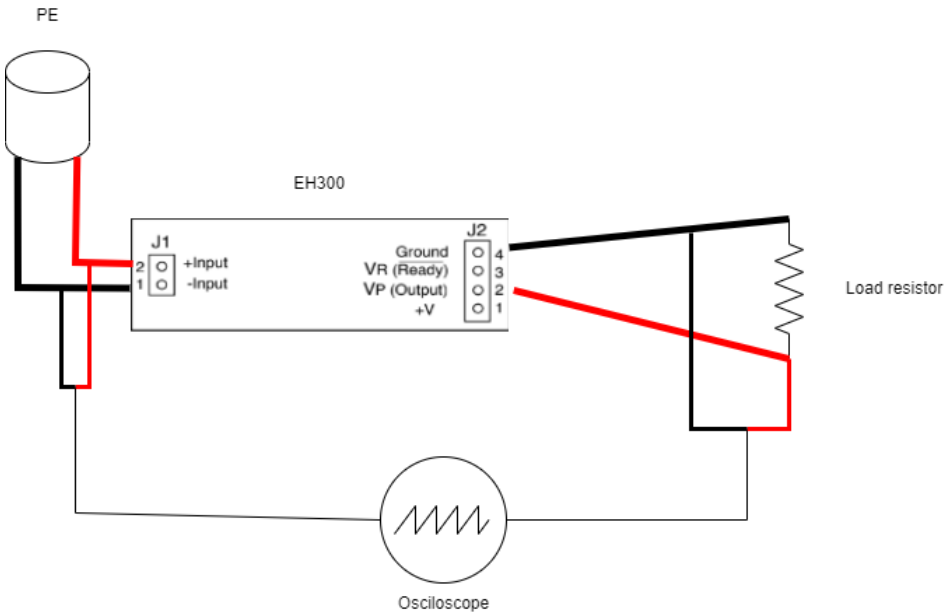

Figure A.2: Connection overview of the Loading test with the EH300 connected to the PE

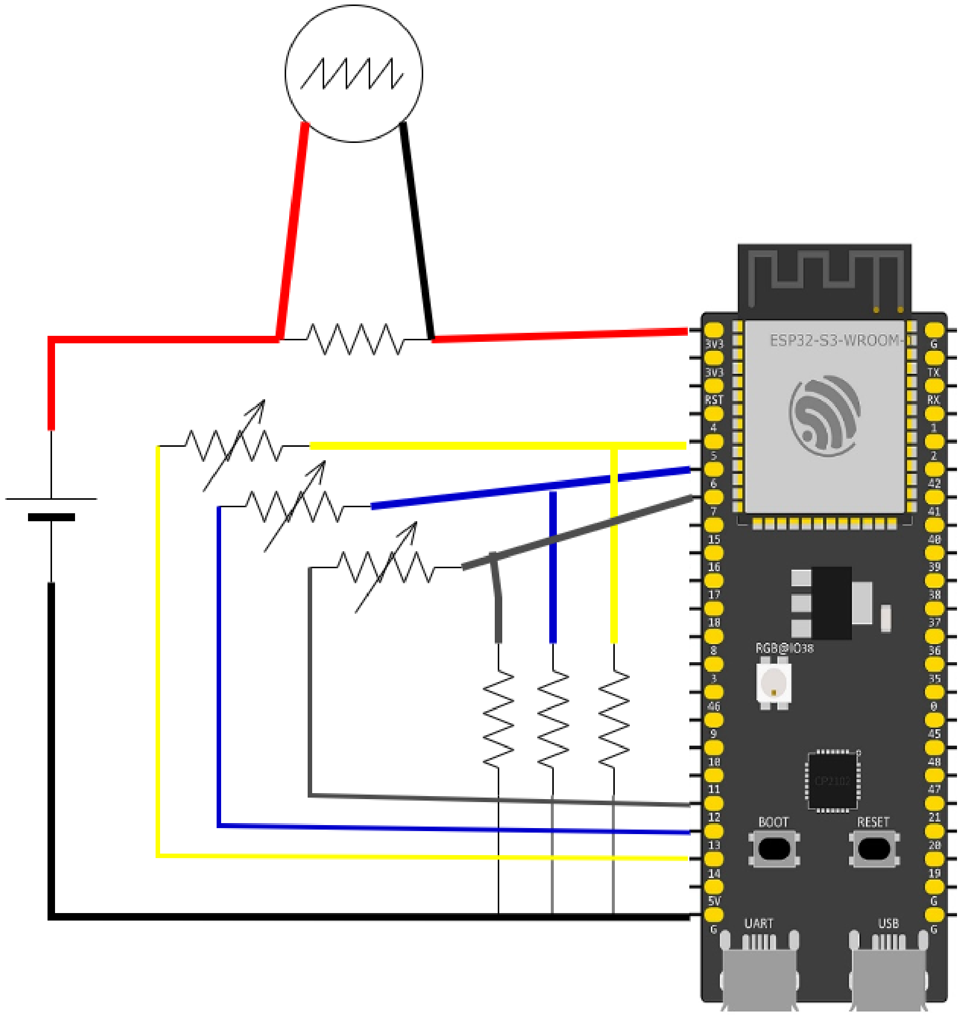

Figure A.3: Connection overview of the power consumption measurement test

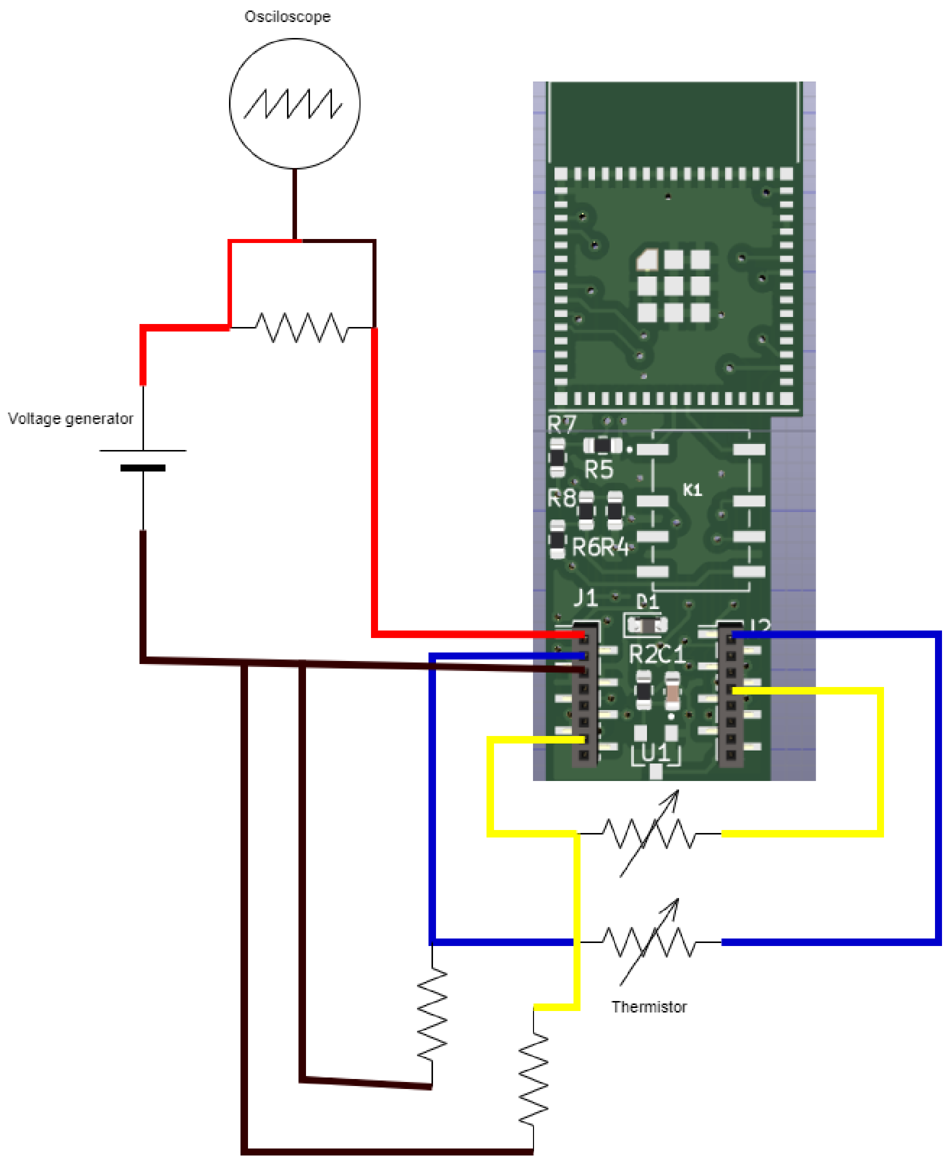

Figure A.4: Connection overview of the power consumption of the PCB

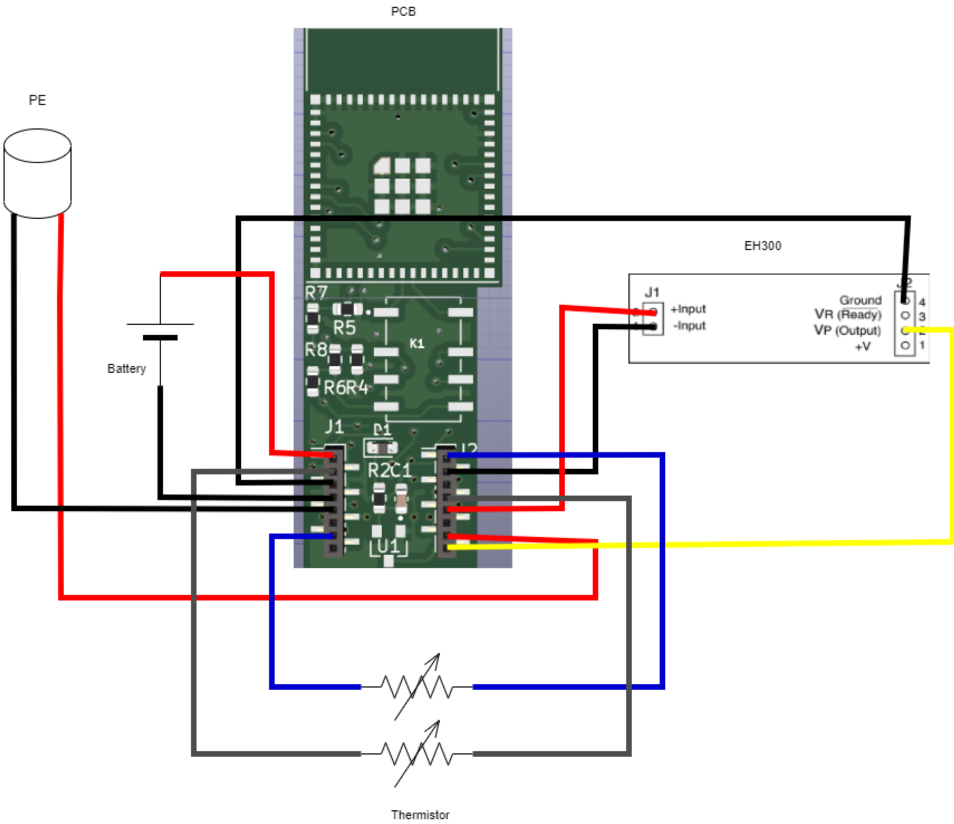

Figure A.5: Connection overview of the Loading test with the PCB

Time(s)	Force(N)
0	285,76
0,022	355,54
0,044	463,45
0,067	593,57
0,089	704,04
0,111	795,81
0,133	856,70
0,156	892,85
0,178	898,64
0,200	884,34
0,222	852,96
0,244	800,67
0,267	760,89
0,289	731,75
0,311	716,61
0,333	710,40
0,356	714,82
0,378	728,69
0,400	747,74
0,422	775,16
0,444	801,55
0,467	846,77
0,489	867,41
0,511	891,66
0,533	899,89
0,556	884,78
0,578	837,39
0,600	756,34
0,622	656,20
0,644	529,99
0,667	376,51
0,689	271,34
0,711	207,26
0,733	176,72
0,756	151,91
0,778	133,53
0,800	115,74
0,822	100,78
0,844	92,07
0,867	87,58
0,889	82,25
0,911	83,64
0,933	80,09
0,956	78,20
0,978	81,33
1,000	94,22
1,022	120,29
1,044	162,65
1,067	219,59
1,089	248,70
1,111	285,76

Table A.1: Overview of force input for the Electropuls machine

B
Results

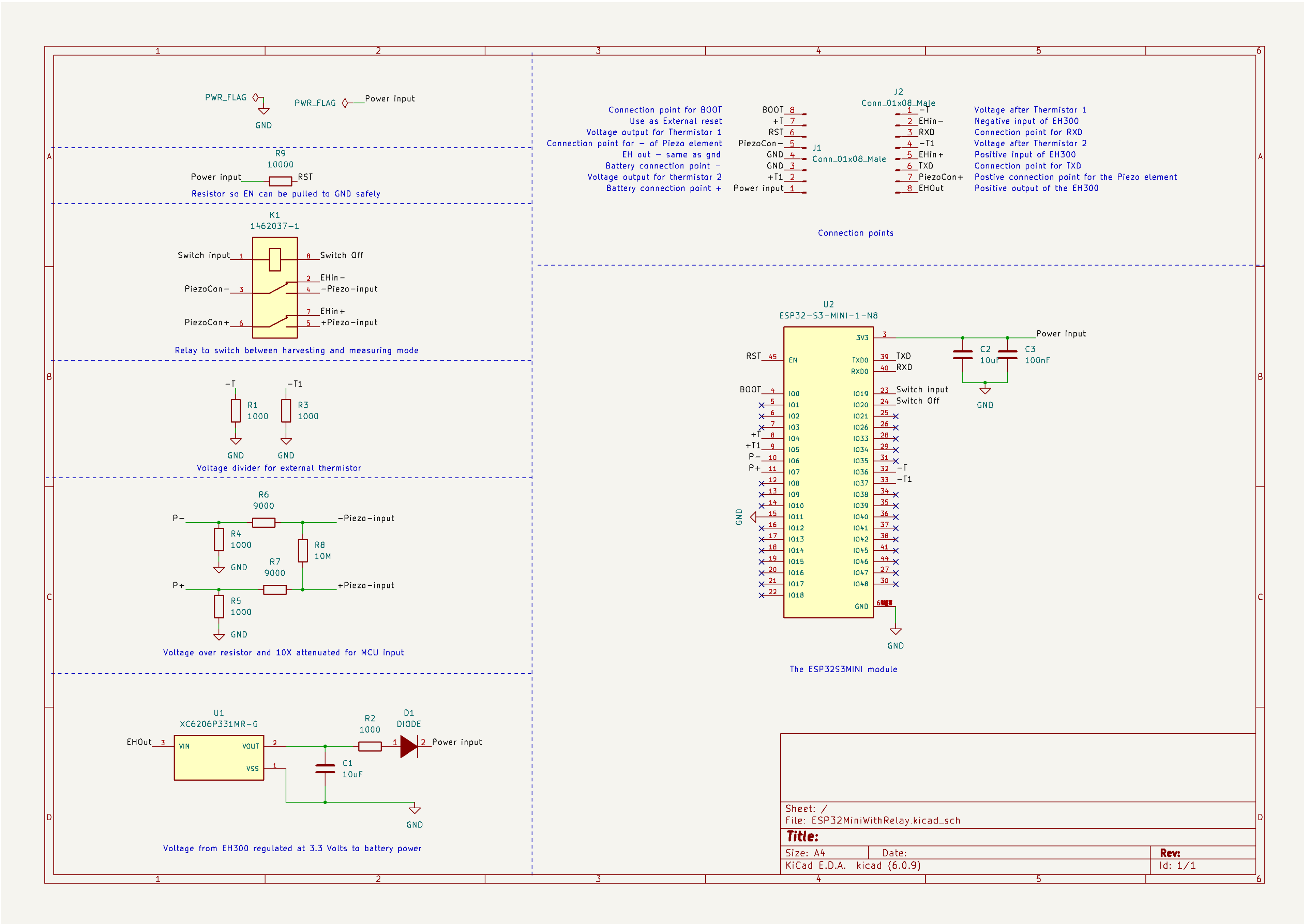

Figure B.1: Schematic of the PCB design with relay

Table B.1: Bill of materials PCB design

Designator	Manufacturer	Part number
U2	Espressif	ESP32-S3-MINI-1-N8
U1	Torex semiconductor	XC6206P331MR-G
J1,J2	GCT	BC070-08-A-1-L-C
R1,R2,R3,R4,R5	Multicomp Pro	MCMR06X1001FTL
R6,R7	Multicomp Pro	MCWR06X9101FTL
R8	Yageo	RC0603JR-0720ML
C1,C2	KEMET	C0603C106M8PAC7411
C3	KEMET	C0603C104K5RACAUTO
D1	KYOCERA AVX	SD0603S040S0R2
K1	axicom te connectivity	IM01GR
R9	yageo	RC0603JR-1010KL

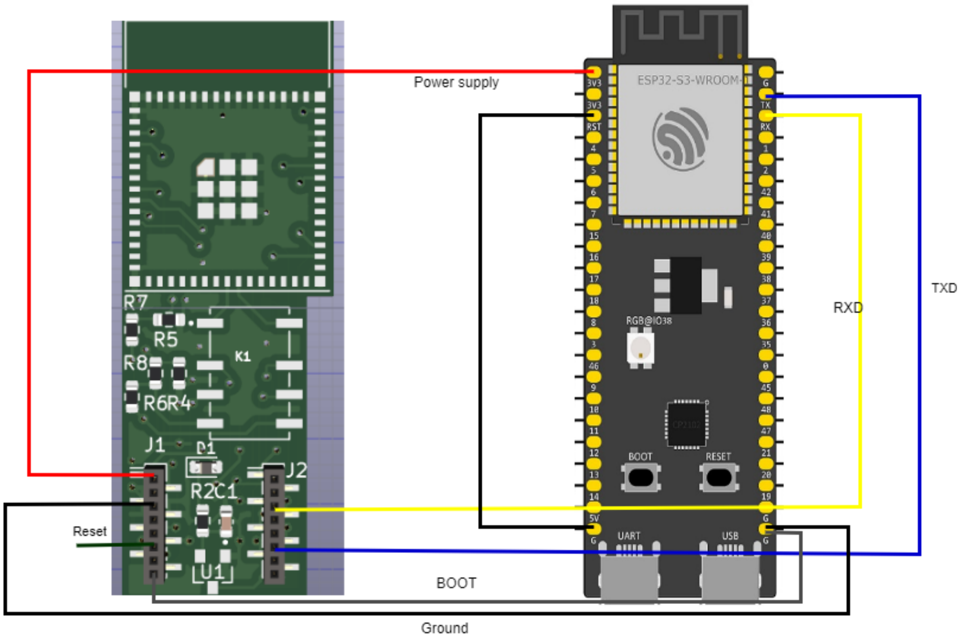

Figure B.2: Connections to be made to programm the MCU

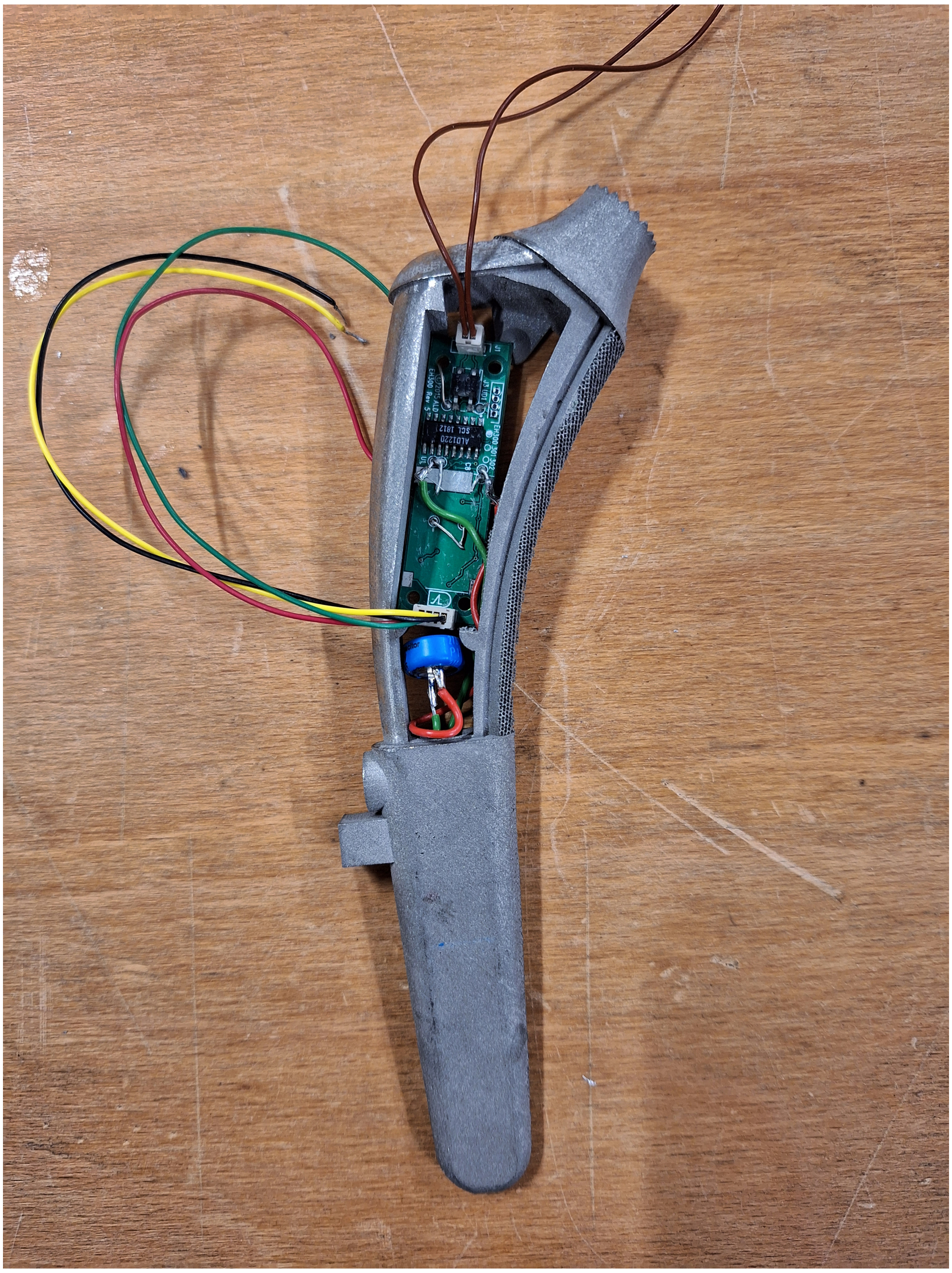

Figure B.3: The EH300 placed inside the hip implant with the capacitor placed in the space below

Figure B.4: The final results of the instrumented implant without the head

C

Arduino codes used

```
#include <BLEDevice.h>
#include <BLEServer.h>
#include <BLEUtils.h>
#include <BLE2902.h>
#include <driver/adc.h>
#include <phyphoxBle.h>

#define maxData 100 //define how much data values you want to measure each time before going to sleep
#define sleepTime  2 //define how many seconds you want the MCU to go to sleep after measuring/sending
RTC_DATA_ATTR int Teller = 0;
RTC_DATA_ATTR float Sensout1[maxData];
RTC_DATA_ATTR float Sensout2[maxData];


void setup() {
  // put your setup code here, to run once:
 Serial.begin(115200);
 adc1_config_channel_atten( ADC1_CHANNEL_3, ADC_ATTEN_11db );
 adc1_config_channel_atten( ADC1_CHANNEL_4, ADC_ATTEN_11db );
 adc1_config_channel_atten( ADC1_CHANNEL_5, ADC_ATTEN_0db );
 adc1_config_channel_atten( ADC1_CHANNEL_6, ADC_ATTEN_0db );
 pinMode(36, OUTPUT);
 digitalWrite(36,HIGH);

 pinMode(37,OUTPUT);
 digitalWrite(37,HIGH);
 pinMode(19, OUTPUT);
 digitalWrite(19,LOW);
 pinMode(20,OUTPUT);
 digitalWrite(20,HIGH);
 delay(10);
 digitalWrite(20,LOW);

  PhyphoxBLE::start("ESP32S3");
 // PhyphoxBLE::configHandler = &receivedData;    // used to receive data from PhyPhox.

```

```
  PhyphoxBleExperiment Voltmeter;

  Voltmeter.setTitle("BLEdata");
  Voltmeter.setCategory("Arduino Experiments");
  Voltmeter.setDescription("This experiment will plot the measured voltage
       over time.");

  //View
  PhyphoxBleExperiment::View firstView;
  firstView.setLabel("Rawdata"); //Create a "view"

  //Graph
  PhyphoxBleExperiment::Graph firstGraph;      //Create graph which will
      plot random numbers over time
  firstGraph.setLabel("Voltmeter");
  firstGraph.setUnitX("s");
  firstGraph.setUnitY("V");
  firstGraph.setLabelX("time");
  firstGraph.setLabelY("Voltage");
//Graph2
  PhyphoxBleExperiment::Graph secondGraph;      //Create graph which will
      plot random numbers over time
  secondGraph.setLabel("Voltmeter");
  secondGraph.setUnitX("s");
  secondGraph.setUnitY("V");
  secondGraph.setLabelX("time");
  secondGraph.setLabelY("Voltage");
  PhyphoxBleExperiment::Graph derdeGraph;      //Create graph which will
      plot random numbers over time
  derdeGraph.setLabel("Voltmeter");
  derdeGraph.setUnitX("s");
  derdeGraph.setUnitY("V");
  derdeGraph.setLabelX("time");
  derdeGraph.setLabelY("Voltage");
  PhyphoxBleExperiment::Graph vierdeGraph;      //Create graph which will
      plot random numbers over time
  vierdeGraph.setLabel("Voltmeter");
  vierdeGraph.setUnitX("s");
  vierdeGraph.setUnitY("V");
  vierdeGraph.setLabelX("time");
  vierdeGraph.setLabelY("Voltage");
  //  PhyphoxBleExperiment::Graph vijfdeGraph;      //Create graph which
      will plot random numbers over time
  //vijfdeGraph.setLabel("Voltmeter");
// vijfdeGraph.setUnitX("s");
// vijfdeGraph.setUnitY("V");
// vijfdeGraph.setLabelX("time");
// vijfdeGraph.setLabelY("Voltage");
  /* Assign Channels, so which data is plotted on x or y axis
     first parameter represents x-axis, second y-axis
     Channel 0 means a timestamp is created after the BLE package arrives
         in phyphox
     Channel 1 to N corresponding to the N-parameter which is written in
         server.write()
  */

```

```
  firstGraph.setChannel(0,1);
  secondGraph.setChannel(0,2);
  derdeGraph.setChannel(0,3);
  vierdeGraph.setChannel(0,4);
 // vijfdeGraph.setChannel(0,5);
  //Edit
  PhyphoxBleExperiment::Edit myEdit;
  myEdit.setLabel("Editfield");
  myEdit.setUnit("u");
  myEdit.setSigned(false);
  myEdit.setDecimal(false);
  myEdit.setChannel(1);
  myEdit.setXMLAttribute("max=\"10\"");
   //Export
  //PhyphoxBleExperiment::ExportSet mySet;       //Provides exporting the data to excel etc.
  //mySet.setLabel("mySet");

  //PhyphoxBleExperiment::ExportData myData1;
  //myData1.setLabel("myData1");
  //myData1.setDatachannel(1);

  //PhyphoxBleExperiment::ExportData myData2;
  //myData2.setLabel("myData2");
  //myData2.setDatachannel(2);

    //PhyphoxBleExperiment::ExportData myData3;
  //myData3.setLabel("myData3");
  //myData3.setDatachannel(3);

  //PhyphoxBleExperiment::ExportData myData4;
  //myData4.setLabel("myData4");
  //myData4.setDatachannel(4);

  firstView.addElement(firstGraph);            //attach graph to view
  firstView.addElement(secondGraph);
  firstView.addElement(derdeGraph);
  firstView.addElement(vierdeGraph);
  firstView.addElement(myEdit);
  //mySet.addElement(myData1);                     //attach data to exportSet
  //mySet.addElement(myData2);
  //mySet.addElement(myData3);                     //attach data to exportSet
  //mySet.addElement(myData4);
  //firstView.addElement(vijfdeGraph);
  Voltmeter.addView(firstView);
  //Voltmeter.addExportSet(mySet);//Attach view to experiment
  PhyphoxBLE::addExperiment(Voltmeter);       //Attach experiment to server

}
void loop() {

    float Sensor1;
    int raw_dataN;
```

```
    int check1;
    float check12;
    raw_dataN=adc1_get_raw(ADC1_CHANNEL_3);
    //check1=analogRead(4);
    //check12=check1*1.00;
    Sensor1=raw_dataN*1.00;
    float Sensor2;
    int raw_datasens2;
    raw_datasens2=adc1_get_raw(ADC1_CHANNEL_4);
    //raw_datasens2=analogRead(5);
    Sensor2=raw_datasens2*1.00;
    Sensout1[Teller]=Sensor1;
    Sensout2[Teller]=Sensor2;


    float raw_dataP1;
    float raw_dataP2;
    raw_dataP1=0;
    //raw_dataP1=analogRead(6);
    int check2;
    float check22;
    //check2=analogRead(7);
    //check22=check2*1.00;
    raw_dataP2=0;
    //raw_dataP2=analogRead(7);
    PhyphoxBLE::write(Sensor1,Sensor2,raw_dataP1,raw_dataP2);
    Serial.println("P1");
    Serial.println(raw_dataP1);
    Serial.println("P2");
    Serial.println(raw_dataP2);
    float readInput;
    int editValue;
    PhyphoxBLE::read(readInput);
    editValue = readInput;
    if(editValue==1){
     pinMode(19, OUTPUT);
     digitalWrite(19,HIGH);
     pinMode(20,OUTPUT);
     digitalWrite(20,LOW);
     int i=0;
     for(i;i<=100;i++){
     raw_dataP1=adc1_get_raw(ADC1_CHANNEL_5)*1.00;
    raw_dataP2=adc1_get_raw(ADC1_CHANNEL_6)*1.00;
    PhyphoxBLE::write(Sensor1,Sensor2,raw_dataP1,raw_dataP2);
    delay(100);
    Serial.println(i);}}
    pinMode(19, OUTPUT);
     digitalWrite(19,LOW);
     pinMode(20,OUTPUT);
     digitalWrite(20,HIGH);
     delay(100);
     digitalWrite(20,LOW);
    delay(1000);
    Teller++;
}
```

```
#include <BLEDevice.h>
#include <BLEServer.h>
#include <BLEUtils.h>
#include <BLE2902.h>
#include <driver/adc.h>
#include <phyphoxBle.h>



#define maxData 10 //define how much data values you want to measure each
    time before going to sleep
#define sleepTime  1 //define how many seconds you want the MCU to go to
    sleep after measuring/sending
RTC_DATA_ATTR int Teller = 0;
RTC_DATA_ATTR float Sensout1[maxData];
RTC_DATA_ATTR float Sensout2[maxData];



void setup() {
Serial.begin(115200);
  // put your setup code here, to run once:
long int tsetup=micros();
adc1_config_channel_atten( ADC1_CHANNEL_3, ADC_ATTEN_11db );
adc1_config_channel_atten( ADC1_CHANNEL_4, ADC_ATTEN_11db );

pinMode(39,OUTPUT);
digitalWrite(39,HIGH);

pinMode(40,OUTPUT);
digitalWrite(40,HIGH);

pinMode(19, OUTPUT);
digitalWrite(19,LOW);
pinMode(20,OUTPUT);
digitalWrite(20,HIGH);
long int tsetupeind=micros();\
long int setuptijd=tsetupeind-tsetup;
Serial.println("Time to set up:");
Serial.println(setuptijd);
}

void loop() {
    long int runt=micros();
    float Sensor1;
    int raw_dataN;
    raw_dataN=adc1_get_raw(ADC1_CHANNEL_3);
    Sensor1=raw_dataN*3.30/4095.00;

    float Sensor2;
    int raw_datasens2;
    raw_datasens2=adc1_get_raw(ADC1_CHANNEL_5);
    Sensor2=raw_datasens2*3.30/4095.00;
    Sensout1[Teller]=Sensor1;
    Sensout2[Teller]=Sensor2;
    Serial.println("Value1");
```

```
    Serial.println(Sensout1[Teller]);
    Serial.println("Value2");
    Serial.println(Sensout2[Teller]);
    long int sends=micros();
    if(Teller==maxData){
     PhyphoxBLE::start("ESP32S3");
     for (int i=0;i<=maxData;i++){
     PhyphoxBLE::write(Sensout1[i],Sensout2[i]);
     Serial.println("In Loop");
     delay(10);

     };
     Teller=0;
     } else{    Teller++;
   Serial.println(Teller);};
   long int runteind=micros();
   long int runsend=runteind-sends;
   long int runtot=runteind-runt;
   Serial.println("Time to measure");
   Serial.println(runtot);
   Serial.println("Time to send");
   Serial.println(runsend);
   Serial.println("Going to sleep");
   delay(1);
   esp_sleep_enable_timer_wakeup(sleepTime * 1000000);
   esp_deep_sleep_start();


  }
```

```
   #include <BLEDevice.h>
#include <BLEServer.h>
#include <BLEUtils.h>
#include <BLE2902.h>
#include <driver/adc.h>
#include <phyphoxBle.h>







#define maxData 100 //define how much data values you want to measure each time before going to sleep
#define sleepTime  2 //define how many seconds you want the MCU to go to sleep after measuring/sending
RTC_DATA_ATTR int Teller = 0;
RTC_DATA_ATTR float Sensout1[maxData];
RTC_DATA_ATTR float Sensout2[maxData];


void setup() {
  // put your setup code here, to run once:
Serial.begin(115200);
adc1_config_channel_atten( ADC1_CHANNEL_3, ADC_ATTEN_11db );
```

```
adc1_config_channel_atten( ADC1_CHANNEL_4, ADC_ATTEN_11db );
adc1_config_channel_atten( ADC1_CHANNEL_5, ADC_ATTEN_0db );
adc1_config_channel_atten( ADC1_CHANNEL_6, ADC_ATTEN_0db );
pinMode(36, OUTPUT);
digitalWrite(36,HIGH);

pinMode(37,OUTPUT);
digitalWrite(37,HIGH);
pinMode(19, OUTPUT);
digitalWrite(19,LOW);
pinMode(20,OUTPUT);
digitalWrite(20,HIGH);
delay(10);
digitalWrite(20,LOW);

  PhyphoxBLE::start("ESP32S3");
 // PhyphoxBLE::configHandler = &receivedData;   // used to receive data
    from PhyPhox.

  PhyphoxBleExperiment Voltmeter;

  Voltmeter.setTitle("BLEdata");
  Voltmeter.setCategory("Arduino Experiments");
  Voltmeter.setDescription("This experiment will plot the measured voltage
      over time.");

  //View
  PhyphoxBleExperiment::View firstView;
  firstView.setLabel("Rawdata"); //Create a "view"

  //Graph
  PhyphoxBleExperiment::Graph firstGraph;      //Create graph which will
     plot random numbers over time
  firstGraph.setLabel("Voltmeter");
  firstGraph.setUnitX("s");
  firstGraph.setUnitY("V");
  firstGraph.setLabelX("time");
  firstGraph.setLabelY("Voltage");
 //Graph2
  PhyphoxBleExperiment::Graph secondGraph;      //Create graph which will
     plot random numbers over time
  secondGraph.setLabel("Voltmeter");
  secondGraph.setUnitX("s");
  secondGraph.setUnitY("V");
  secondGraph.setLabelX("time");
  secondGraph.setLabelY("Voltage");
  PhyphoxBleExperiment::Graph derdeGraph;      //Create graph which will
     plot random numbers over time
  derdeGraph.setLabel("Voltmeter");
  derdeGraph.setUnitX("s");
  derdeGraph.setUnitY("V");
  derdeGraph.setLabelX("time");
  derdeGraph.setLabelY("Voltage");
  PhyphoxBleExperiment::Graph vierdeGraph;      //Create graph which will
     plot random numbers over time
  vierdeGraph.setLabel("Voltmeter");
```

```
  vierdeGraph.setUnitX("s");
  vierdeGraph.setUnitY("V");
  vierdeGraph.setLabelX("time");
  vierdeGraph.setLabelY("Voltage");
  //  PhyphoxBleExperiment::Graph vijfdeGraph;      //Create graph which will plot random numbers over time
  //vijfdeGraph.setLabel("Voltmeter");
// vijfdeGraph.setUnitX("s");
// vijfdeGraph.setUnitY("V");
// vijfdeGraph.setLabelX("time");
// vijfdeGraph.setLabelY("Voltage");
  /* Assign Channels, so which data is plotted on x or y axis
     first parameter represents x-axis, second y-axis
     Channel 0 means a timestamp is created after the BLE package arrives in phyphox
     Channel 1 to N corresponding to the N-parameter which is written in server.write()
  */

  firstGraph.setChannel(0,1);
  secondGraph.setChannel(0,2);
  derdeGraph.setChannel(0,3);
  vierdeGraph.setChannel(0,4);
// vijfdeGraph.setChannel(0,5);
  //Edit
  PhyphoxBleExperiment::Edit myEdit;
  myEdit.setLabel("Editfield");
  myEdit.setUnit("u");
  myEdit.setSigned(false);
  myEdit.setDecimal(false);
  myEdit.setChannel(1);
  myEdit.setXMLAttribute("max=\"10\"");
   //Export
  //PhyphoxBleExperiment::ExportSet mySet;       //Provides exporting the data to excel etc.
  //mySet.setLabel("mySet");

  //PhyphoxBleExperiment::ExportData myData1;
  //myData1.setLabel("myData1");
  //myData1.setDatachannel(1);

  //PhyphoxBleExperiment::ExportData myData2;
  //myData2.setLabel("myData2");
  //myData2.setDatachannel(2);

    //PhyphoxBleExperiment::ExportData myData3;
  //myData3.setLabel("myData3");
  //myData3.setDatachannel(3);

  //PhyphoxBleExperiment::ExportData myData4;
  //myData4.setLabel("myData4");
  //myData4.setDatachannel(4);

  firstView.addElement(firstGraph);           //attach graph to view
  firstView.addElement(secondGraph);
  firstView.addElement(derdeGraph);
```

```
  firstView.addElement(vierdeGraph);
  firstView.addElement(myEdit);
  //mySet.addElement(myData1);                    //attach data to exportSet
  //mySet.addElement(myData2);
  //mySet.addElement(myData3);                    //attach data to exportSet
  //mySet.addElement(myData4);
  //firstView.addElement(vijfdeGraph);
  Voltmeter.addView(firstView);
  //Voltmeter.addExportSet(mySet);//Attach view to experiment
  PhyphoxBLE::addExperiment(Voltmeter);       //Attach experiment to server

}
void loop() {

    float Sensor1;
    int raw_dataN;
    int check1;
    float check12;
    raw_dataN=adc1_get_raw(ADC1_CHANNEL_3);
    //check1=analogRead(4);
    //check12=check1*1.00;
    Sensor1=raw_dataN*1.00;
    float Sensor2;
    int raw_datasens2;
    raw_datasens2=adc1_get_raw(ADC1_CHANNEL_4);
    //raw_datasens2=analogRead(5);
    Sensor2=raw_datasens2*1.00;
    Sensout1[Teller]=Sensor1;
    Sensout2[Teller]=Sensor2;


    float raw_dataP1;
    float raw_dataP2;
    raw_dataP1=0;
    //raw_dataP1=analogRead(6);
    int check2;
    float check22;
    //check2=analogRead(7);
    //check22=check2*1.00;
    raw_dataP2=0;
    //raw_dataP2=analogRead(7);
    PhyphoxBLE::write(Sensor1,Sensor2,raw_dataP1,raw_dataP2);
    Serial.println("P1");
    Serial.println(raw_dataP1);
    Serial.println("P2");
    Serial.println(raw_dataP2);
    float readInput;
    int editValue;
    PhyphoxBLE::read(readInput);
    editValue = readInput;
    if(editValue==1){
     pinMode(19, OUTPUT);
     digitalWrite(19,HIGH);
```

```
      pinMode(20,OUTPUT);
      digitalWrite(20,LOW);
      int  i=0;
      for(i;i<=100;i++){
      raw_dataP1=adc1_get_raw(ADC1_CHANNEL_5)*1.00;
    raw_dataP2=adc1_get_raw(ADC1_CHANNEL_6)*1.00;
    PhyphoxBLE::write(Sensor1,Sensor2,raw_dataP1,raw_dataP2);
    delay(100);
    Serial.println(i);}}
    pinMode(19, OUTPUT);
      digitalWrite(19,LOW);
      pinMode(20,OUTPUT);
      digitalWrite(20,HIGH);
      delay(100);
      digitalWrite(20,LOW);
    delay(1000);
    Teller++;
  }
```