Design of a system for ambulatory measurement of peripheral temperature

A Degree Thesis

Submitted to the Faculty of the

Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona

Universitat Politècnica de Catalunya

by

Antoni López Giménez

In partial fulfilment

of the requirements for the degree in

Telecommunications Technologies and Services ENGINEERING

Advisor: Juan José Ramos Castro

Barcelona, June 2021

Abstract

The aim of this project is to carry out a study of variations in human peripheral temperature as affected by stress, sleep and/or physical activity. To do this we have divided the project into two parts: the development of the measurement system and the assessment of this end.

In order to be able to carry out the assessment correctly, we need to develop a system that can obtain the peripheral temperature of the subject to be studied and also know if it has been affected by physical activity. For this purpose we have developed a measurement system with a temperature sensor and an accelerometer as well as an RTC to be able to know at what instant of time the data have been collected.

Finally, the assessment consists of the test subject taking the system with him/her for 2-3 days and returning it to download the data for analysis.

Resum

L'objectiu d'aquest projecte és poder realitzar un estudi de les variacions de la temperatura perifèrica de l'ésser humà afectada per l'estrès, somni i/o activitat física. Per a dur-ho a terme hem dividit el projecte en dues parts: el desenvolupament del sistema de mesura i l'estudi d'aquesta finalitat.

Per a poder realitzar l'estudi correctament necessitem desenvolupar un sistema que pugui obtenir la temperatura perifèrica del subjecte a estudiar i a més saber si s'ha vist afectada per l'activitat física. Per això hem desenvolupat un sistema de mesura amb sensor de temperatura i un acceleròmetre a més d'un RTC per a poder saber a quin instant de temps han estat recopilades les dades

Finalment, l'estudi consisteix en el fet que el subjecte a avaluar s'emportarà el sistema durant 2-3 amb ell/ella i serà retornar per descarregar les dades i poder analitzar-les.

Resumen

El objetivo de este proyecto es poder realizar un estudio de las variaciones de la temperatura periférica del ser humano afectada por el estrés, sueño y/o actividad física. Para llevarlo a cabo hemos dividido el proyecto en dos partes: el desarrollo del sistema de medida y el estudio de dicho fin.

Para poder realizar el estudio correctamente necesitamos desarrollar un sistema que pueda obtener la temperatura periférica del sujeto a estudiar y además saber si se ha visto afectada por la actividad física. Para ello hemos desarrollado un sistema de medida con sensor de temperatura y un acelerómetro además de un RTC para poder saber en qué instante de tiempo han sido recopilados los datos.

Finalmente, el estudio consiste en que el sujeto a evaluar se llevará el sistema consigo 2-3 días y será devuelto para descargar los datos y poder ser analizados.

Dedication

I want to dedicate this thesis to my mother because she helped and supported me every day during my bachelor's degree and also for being mother and father at the same time. Without her advice and support, maybe I would not have finished my bachelor's degree.

Thanks mum.

Acknowledgements

I would like to start by thanking Juan Ramos for the opportunity to develop this project with him and helping me to improve my skills and knowledge.

Furthermore, thanks to Alfonso Mendez to support me every day in the lab and share his advice with me during the entire project especially in the hardware part.

Finally, also thanks Marc Mateu, a PhD student, for his support and patience while teaching me and for providing us with a specific programmer essential to develop this project.

Revision history and approval record

Revision	Date	Purpose
0	27/04/2021	Document creation
1	08/06/2021	First Draft of the Contents Organization and introduction
2	09/06/2021	Introduction and budget
3	13/06/2021	Work plan improvement
4	15/06/2021	Project Development
5	16/06/2021	Conclusions and Future development
6	17/06/2021	State of the art
7	18/06/2021	Revision and Results
8	21/06/2021	Last revision and details

DOCUMENT DISTRIBUTION LIST

Name	e-mail
Antoni López Giménez	tonilg95@gmail.com
Juan José Ramos Castro	juan.jose.ramos@upc.edu

Written by:		Reviewed and approved by:	
Date	21/06/2021	Date	21/06/2021
Name	Antoni López Giménez	Name	Juan José Ramos Castro
Position	Project Author	Position	Project Supervisor

Table of contents

List of Figures

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

telecos BCN

List of Tables:

1. Introduction

A great number of scientific surveys discovered that the peripheral temperature of the human body changes around the day depending on the situation. For example, during the night the temperature drops because we don't need energy to do anything but another example that is known is the relationship between temperature and menstrual cycle [1].

The aim of the project is, at least in this case, to develop the measurements system to facilitate the study of the relation between the variations in peripheral temperature and stress, physical activity and sleep.

1.1. Statement of purpose

In order to be able to carry out the assessment described previously I must develop a specific hardware which can obtain the subject's temperature, the time and if he/she was doing sport during the current time and previous.

The hardware should be as small as possible and it can be stored in a box with the form of a smartwatch for user-friendliness.

1.2. Requirements and specifications

1.2.1. Requirements

- The project should be user-friendly
- It can be like a smartwatch
- The battery must have 2-3 days of autonomy
- It must store that 2-3 days of Data

1.2.2. Specifications

As I said previously I should design a specific hardware for the purpose and that hardware should be as small as possible.

In addition, that hardware must have the following components:

- A microcontroller
- A RTC
- A EEPROM memory
- A micro-USB
- A USB charger
- A LDO
- A battery charger
- Two temperature sensors
- An accelerometer
- A battery
- A reset button
- A ten pins header
- Optional: three leds

Furthermore, I also needed a power supply, an oscilloscope, a computer, solder tools and a way to do the PCB.

In terms of software, it has the following specifications:

- Run with Arduino IDE
- It must be able to run in the shortest possible time
- Auto-sleep mode
- Test mode
- Print data in test mode to debug
- The menu must have the following sections
    - STOP → Stop the acquisition mode
    - START → Start the acquisition mode
    - SEND → Send the data read to the EEPROM memory
    - RESET → Reset the EEPROM memory
    - READ → Send the data stored to the terminal

1.3. Methods and procedures

This project was born from a previous project developed under the supervision of Juan José Ramos Castro. The previous project consisted of an Adafruit development board, Adafruit Trinket M0 and an own PCB developed by the student where she implemented the USB charger, LDO, EEPROM memory and two temperature sensors.

That project was not designed to incorporate the measurement of physical activity. For all of this, my project is an improvement of the previous one adding the measurement of physical activity and it implements everything on a single-board.

Due to the fact that some of the components change with respect to the previous one I must develop a new code for my board but it's an interpretation and improvement of the code already made. Some of the code changes are related to these component changes (for example the improvement in the quantity of EEPROM memory) and all of them are based in the GitHub libraries of the components mentioned in particular those concerning the EEPROM memory. [2]

1.4. Work plan

The Work plan was conceived to work in parallel in all project parts. For that, I decided to separate in 4 work packages:

- Work Package 1 (Analysis): Is the part of the project dedicated to do an assessment of the possible solutions in terms of hardware and software.
- Work Package 2 (Hardware): This WP was dedicated to develop and test the specific hardware designed for the project.
- Work Package 3 (Software): The third one, this is the WP dedicated to developing (based on the previous project) a new code for the microcontroller to allowed it the possibility to obtain and store all the data needed.
- Work Package 4 (Documentation): This is the last WP and is dedicated to do and advanced all the documentation, this thesis included, to submit

1.4.1. Work Breakdown Structure

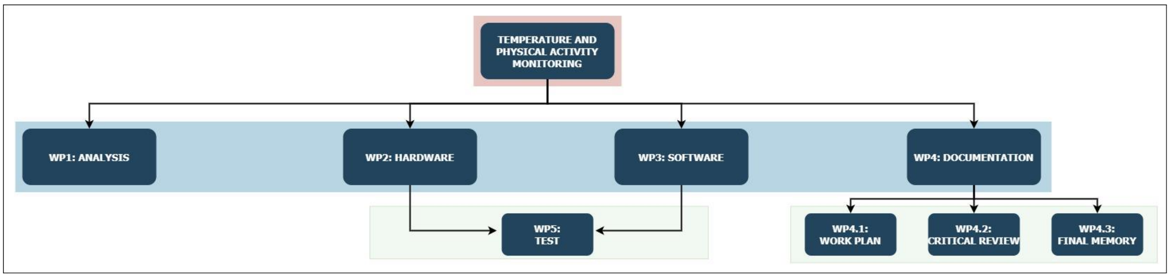

Figure 1: Work Breakdown Structure

Table 1: WP1

Project: Temperature and physical activity monitoring	**WP ref:** WP1	
Major constituent: Analysis	Sheet 1 of 3	
Short description: Work package dedicated to evaluate the prototype and previous project, which language will be used to program the microcontroller and the components which will be used in the project. Do a diagram block to fully understand how it works.	**Start date:** 16/03/2021 **End date:** 22/03/2021	
	Start event: **End event:**	
Internal task T1: Previous Project analysis **Internal task T2:** Find Arduino bootloader **Internal task T3:** Find compatible sensors and components **Internal task T4:** Do a diagram block	**Deliverables:**	**Dates:**

Table 2: WP2

Project: Temperature and physical activity monitoring	**WP ref:** WP2	
Major constituent: Hardware	Sheet 2 of 3	
Short description: Work package dedicated to develop the hardware part of the project with Kicad Software. Also it must find the selected components on Farnell shop. Moreover, I will find the correct form to update the bootloader from Arduino in the microcontroller.	**Start date:** 22/03/2021 **End date:** 30/04/2021 **Start event:** **End event:**	
Internal task T1: Evaluation of the components **Internal task T2:** Schematic design **Internal task T3:** PCB design **Internal task T4:** Components soldering	**Deliverables:** Kicad files and PCB files	**Dates:**

Table 3: WP3

<table>
<tr><td>Project: Temperature and physical activity monitoring</td><td colspan="2">WP ref: WP3</td></tr>
<tr><td>Major constituent: Software</td><td colspan="2">Sheet 2 of 3</td></tr>
<tr><td rowspan="2">Short description:
Work package dedicated to develop the software part in the Arduino platform. It must take into account the specifications of the measurements.</td><td colspan="2">Start date: 22/03/2021
End date: 30/04/2021</td></tr>
<tr><td colspan="2">Start event:
End event:</td></tr>
<tr><td>Internal task T1: Evaluation of the functionalities

Internal task T2: Burn the bootloader in the microcontroller

Internal task T3: Develop and program the software</td><td>Deliverables:
Arduino file</td><td>Dates:</td></tr>
</table>

Table 4: WP4

Project: Temperature and physical activity monitoring	**WP ref:** WP4	
Major constituent: Documentation	Sheet 2 of 3	
Short description: Parallel work package, it consists of writing all the necessary documentation and gathers the datasheets of the components to understand, develop and program them correctly.	**Start date:**16/03/2021 **End date:**24/06/2021 **Start event:** **End event:**	
Internal task T1: Datasheets gathered **Internal task T2:** Workplan document **Internal task T3:** Critical Review document **Internal task T4:** Final memory document	**Deliverables:** Workplan, Critical Review, Final Memory.	**Dates:**

Table 5: WP5

Project: Temperature and physical activity monitoring	**WP ref:** WP5	
Major constituent: Test	Sheet 3 of 3	
Short description: The final work package resulting from the WP2 and WP3 that consists of testing the correct performance of the prototype.	**Start date:** 30/04/2021 **End date:** 15/06/2021 Start event: End event:	
Internal task T1: Hardware test **Internal task T2:** Software test **Internal task T3:** Current consumption evaluation **Internal task T4:** Voluntaries research **Internal task T5:** Research conclusions	**Deliverables:**	**Dates:**

1.4.2. Gantt Diagram

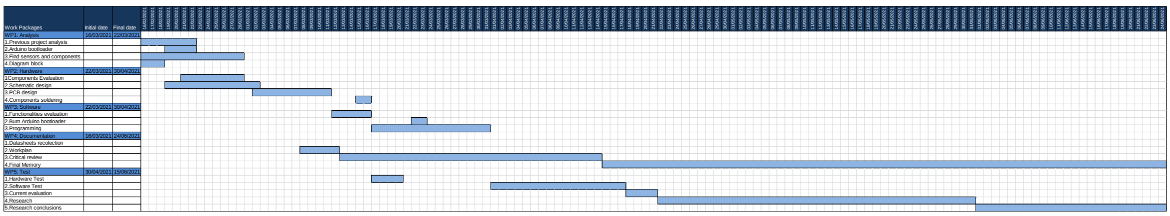

Figure 2: Gantt Diagram

1.5. Incidences

The only change between the final development and the main idea before starting the project was that it was not possible to do the scientific survey. That incident was a consequence of my spinal cord injury and my problems with back pain and the lack of sphincter control with the result that I couldn’t go to the university these days.

Another incident was the problem of finding a correct box to store the final PCB but finally, the previous student allowed us to change her design and I could print a modified version in 3D. Moreover, a great delay for me was finding the correct way to read and write the EEPROM memory because when adapting the code in the GitHub library I made some mistakes.

Finally, the most important incident was the non-possibility to solder of my own, due to lack of experience soldering SMD components, and I needed the help of the lab technician, Alfonso Mendez to solder all the components and the lack of guaranties with the soldier of accelerometer due to its kind of packaging.

2. State of the art

In this section I try to explain in detail why I decided to work with specific software tools and why I decided to choose some specific components and data bus.

2.1. Specific components and I2C bus

First of all, I want to explain the selection of some components and the I2C bus. The I2C bus as designed was a master-slave data bus which allows us to control all the sensors in the project with only 2 wires. The two wires involved in the bus are: SDA and SCL. The SDA (Serial Data) is the line where the data is transferred, understanding as data the direction of the slaves, acknowledgment, data of the slaves, etc. The SCL (Serial clock) is the line which helps all the devices to be synchronized with a clock signal.

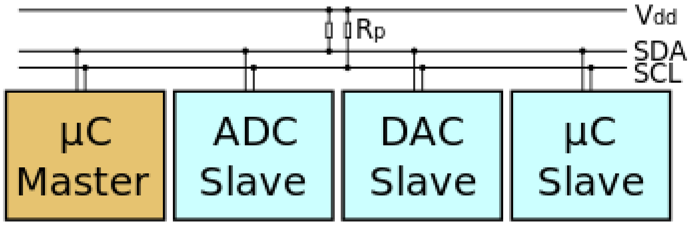

Figure 3: I2C Bus Example

I decide to use this bus because the flexibility it introduces to add new devices and the availability of electronic components (memories, sensors, ADCs, etc..) compatible with this standard. I searched for a micro-controller compatible with it and after doing the research I chose the Atmel SAMD21E18 for its easy compatibility with Arduino Software (the same microcontroller used by the previous student).

2.2. Software tools

2.2.1. KiCad EDA

KiCad EDA is a cross platform and open source electronic design automation suite. With the recommendation of my supervisor Juan Ramos and in addition due to is an open source hardware I worked with it to develop the schematic and PCB for the project.

With KiCAD I could develop in the same software both of them with a great facility and usability with a short lesson about it.

2.2.2. Arduino IDE

The choice of Arduino was more or less for the same reasons as KiCAD but with the difference that I worked previously with it and it was easy to quickly code with it.

Arduino IDE, as KiCAD, is an open-source tool from Arduino that allows you to work easily with the microcontroller and also the sensors involved in the project. With Arduino you have the possibility to work directly with the USB and the terminal and write your full code in Arduino language (It's C++ with more readability) and you can find a great community behind it who develops libraries that you can find in GitHub.

To find some disadvantage to work with Arduino is the absence of a debugger which is useful in some moments of the development of the project. Apart from that, Arduino is a great tool for beginners and experts to work easily with the microcontroller using a IDE multiplatform with big community of developers and multiple libraries.

2.2.3. Microchip Studio

Microchip Studio is the development tool proposed by the manufacturers of the microcontroller. It wasn't a choice it was a necessity. With the fact to buy a microcontroller to implement in a specific board and If you need to work with Arduino, previously you need to install the Arduino bootloader in the flash microcontroller memory to allow you work with Arduino IDE.

In the Adafruit web page, they have a section named "learn" where you can find a great number of tips, projects and advice to develop your projects. In my case, I found a tutorial [3] that helped me to learn how I could install the bootloader with the help of Microchip studio and the .bin file that you can find in their GitHub repository of Arduino bootloader for Adafruit boards. I selected the bootloader for the Adafruit Trinket M0.

2.2.4. CoolTermWin

With the necessity to download the data stored during the surveys and the problem of the Arduino terminal, which does not allows you to store what you read in, I needed to find an alternative to write the data in a .txt file. Searching on YouTube, I found [4] a video with alternatives and finally I chose CoolTermWin easy of use.

2.2.5. Freedom Sensor Toolbox

In order to facilitate the development and validate the software for programming and reading the data form the accelerometer we bought a development board for it. The mentioned development board includes specific software where you can manage the register of the accelerometer and know if your setup is correct or not. With it I found my problems of setup and why we couldn't read the specific register which the project needs, thanks to the software have a little explanation about every register and more clear.

2.2.6. Matlab

For a quick download of the data in the memory in the code we write a function that only prints on the terminal data of 1 byte each. With that we need a specific software to translate that number of bytes in a excel file to be able to read correctly and easily the data stored. For that and taking advantage of the Matlab student license of the UPC I wrote an extra code which enabled us to do that operation.

2.2.7. Fusion 360

Finally, with all the project ready to work correctly, I used the Fusion 360 software from AUTODESK to develop the box. The final box, after a great research, is a modification of the box developed by the other student. I used that software because with the student situation you can obtain a free license to work with it.

3. Project Development

This section consists of the explanation of methods and steps followed to achieve the objectives of the project. It explains the research methodology for both hardware and software, and how they were implemented.

The development, as you can read in the previous sections, was divided in two parallel parts: hardware part and software part.

For the first one, we have more differences between the previous project and this one and I based my PCB in different Adafruit development boards and the apple watch dimensions.

For the second one, I decided to base my code in the previous project code due to the similarities but with the improvements due to the changes in the sensors and the addition of the accelerometer.

3.1. Hardware Part

At the beginning of the project I started doing a block diagram to have in mind, roughly speaking, an idea of the connection between all the components of my hardware.

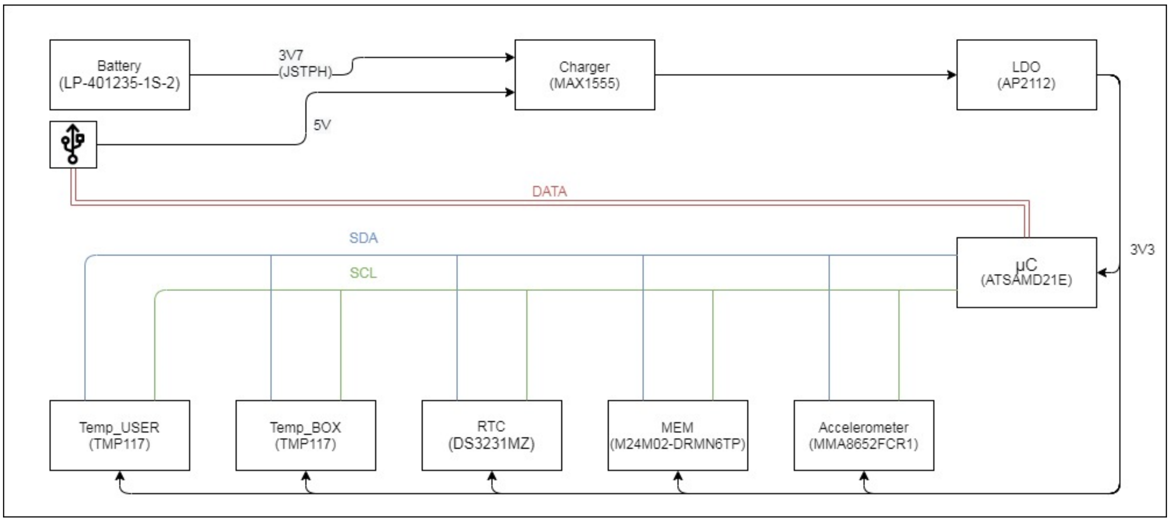

Figure 4: Block Diagram

Then, I was doing some research about the schematics of Adafruit Feather M0 and Adafruit Trinket M0 due to the fact that a great part of the hardware was the same for both. Moreover, I researched which sensors and components were the best for my project focusing on these boards and the previous one.

With all the information and schematics, I started with the schematic of my board:

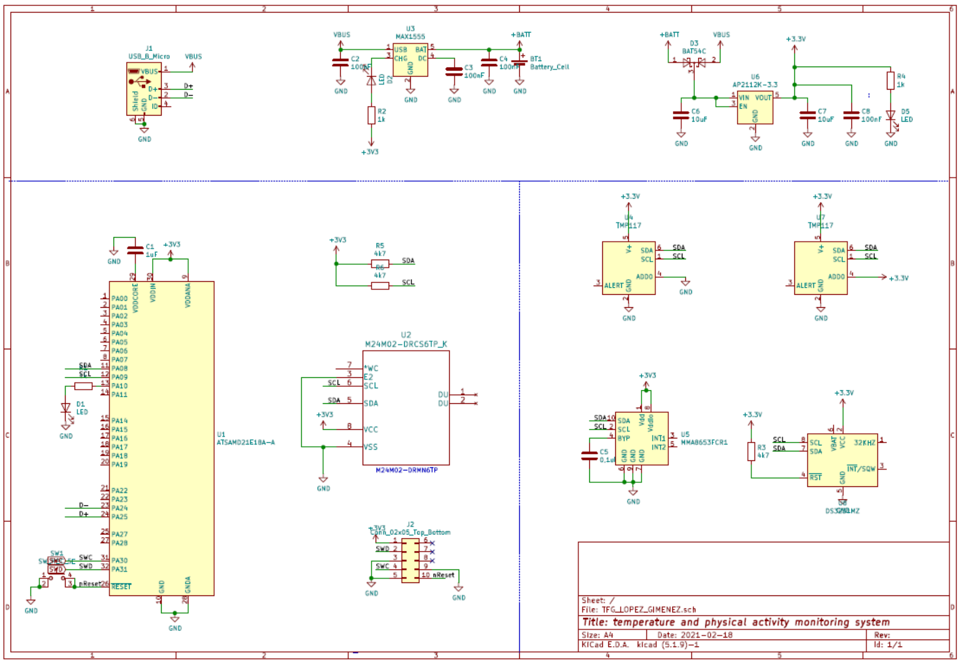

Figure 5: Schematic

I designed a first version of the PCB (prototype) to check the functionality of all the components and to help in the development of the software. Moreover, this board allowed us to identify an correct some wiring problems.

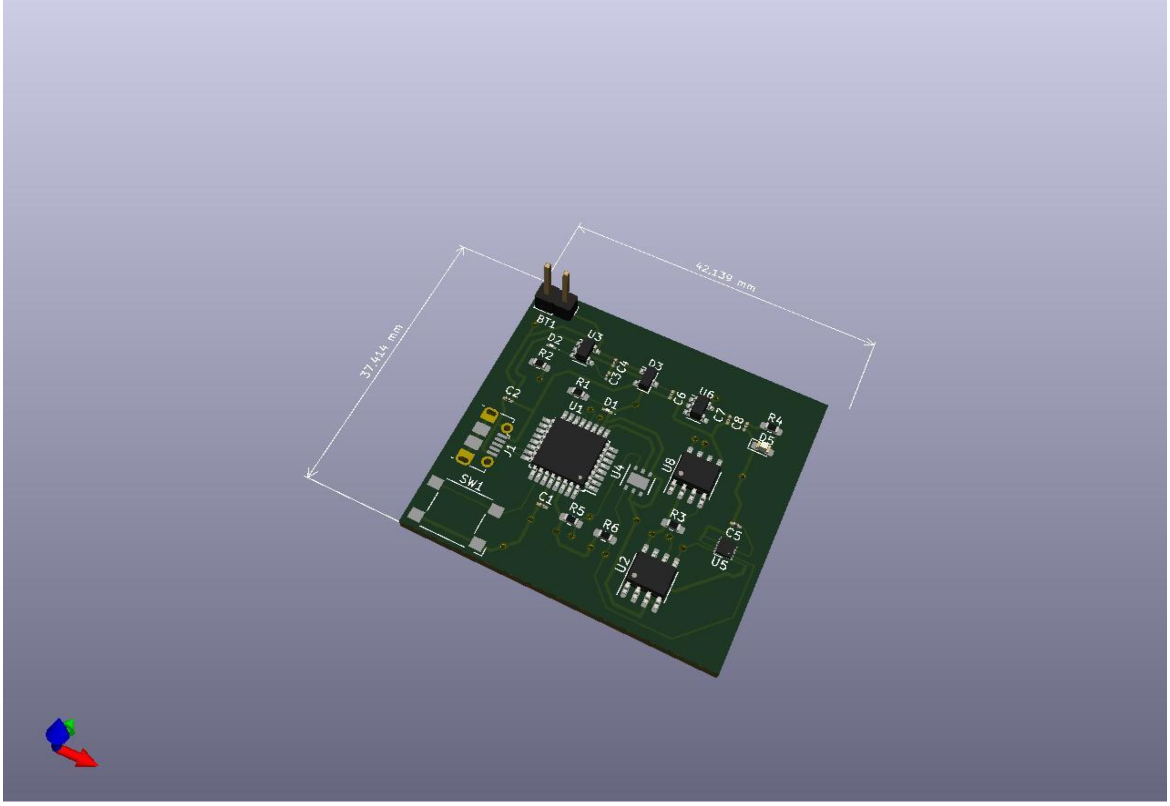

Figure 6: Prototype board (Front)

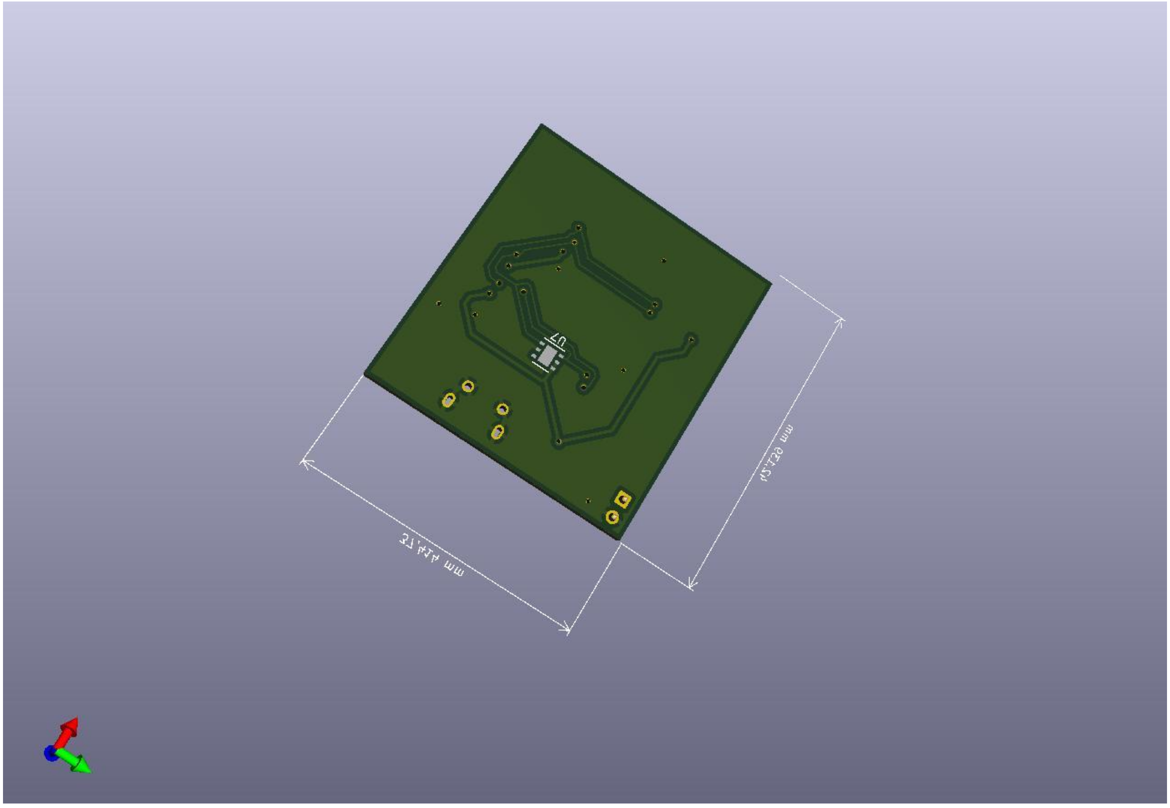

Figure 7: Prototype board (Bottom)

Finally, with all errors corrected and with the correct dimensions defined, I designed the final PCB that was manufactured in the lab or the Instrumentació Electrònica i Biomèdica research group.

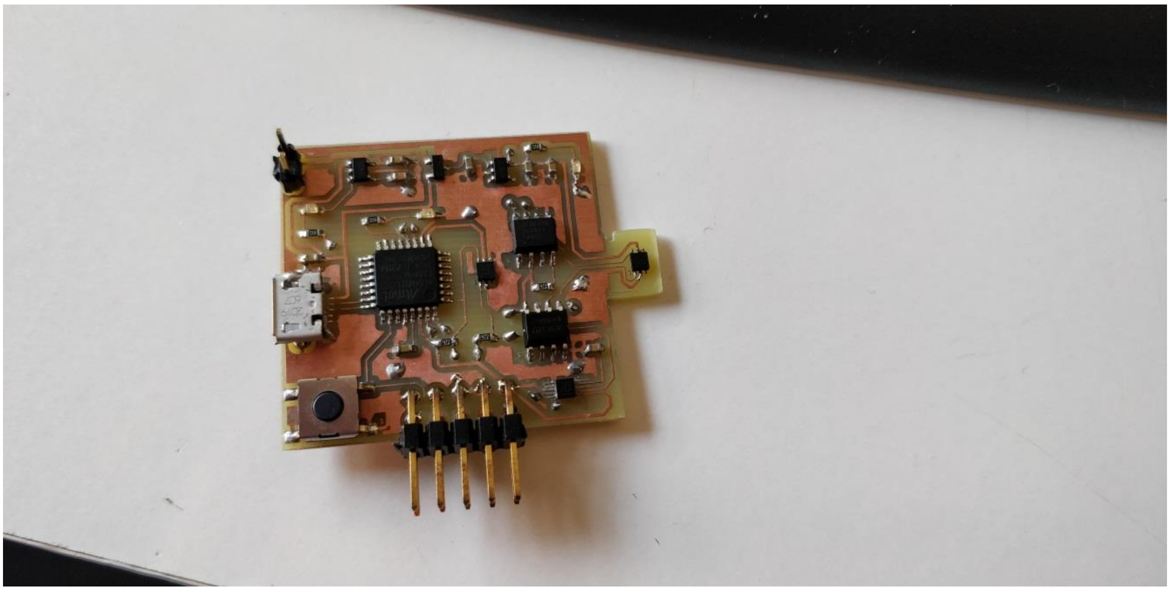

Figure 8: Final Board (Front)

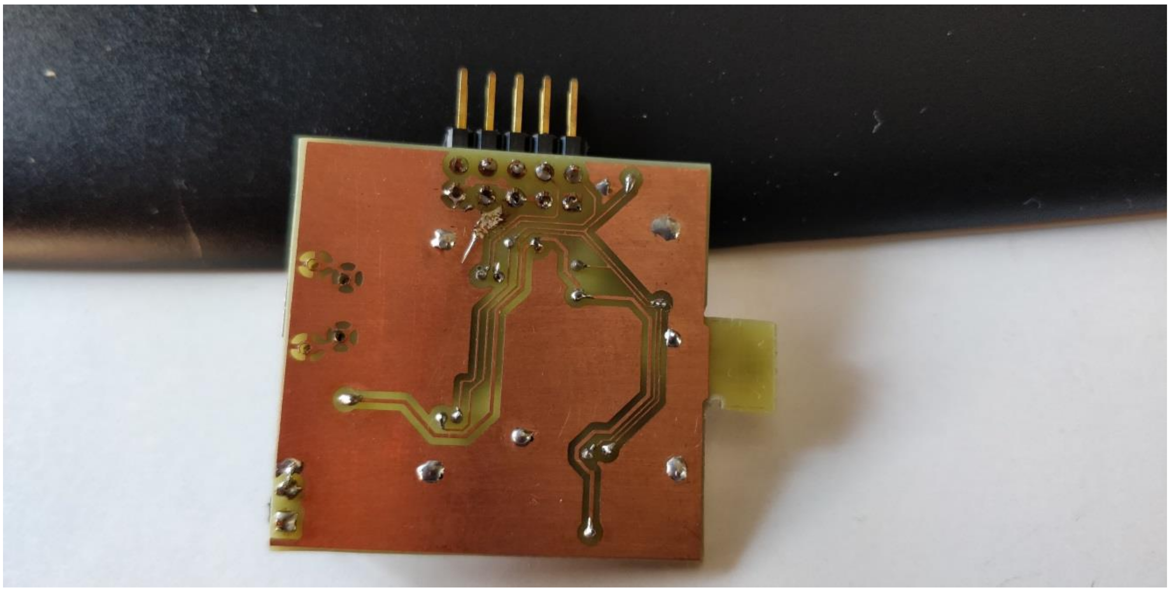

Figure 9: Final Board (Bottom)

3.2. Software Part

For this part, it wasn't necessary to do research about how to code because this one has been based on the previous one. As a consequence of that, I read carefully and in detail and I took the more important parts and the structure to develop my own code with appropriate changes.

What went wrong was my bad idea to start writing the full code without the prototype board because I made a lot of mistakes that could have been less if I had waited a little bit to have it. But mistakes aside, my process to write my full code was:

1. Read accurately the previous code
2. Find the correct libraries to be able to work with the specific components.
3. Write the core of the code: consists of writing the parts without thinking about the specific functions or parts of code.
4. Write the specific code of each sensor
5. Test

As I said previously the code was written without the prototype board which is a mistake because with the board you can write every specific part and test with more exactitude if all works correctly.

For a better understanding of the software I did a flow diagram that you can see in the next picture:

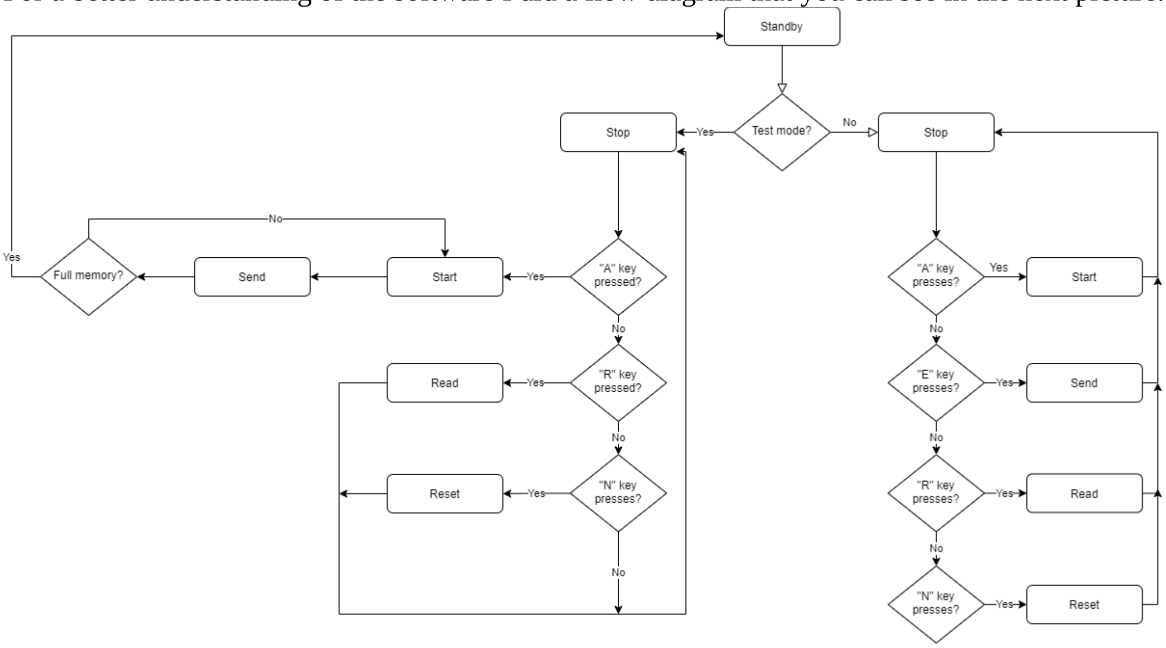

Figure 10: flow diagram

As you can see in the previous picture in test mode we can verify the all functions separately but if we are doing the study and the test mode is not activated the Read and Reset mode will done the same if we are in the test mode but if we Start the acquisition mode starts a cyclical code which it will only stop when the memory will be fulled.

4. Results

In this section I will explain some results obtained during the test parts of hardware and software to confirm the correct functionality of all and the fulfilment of the requirements and specifications.

Firstly, If we talk in terms of current consumption I did two different assessments with the final board. The first one, is to evaluate the average current consumption in a normal situation and the second one was to evaluate the average consumption without a specific LED which took the main part of the consumption. The results were:

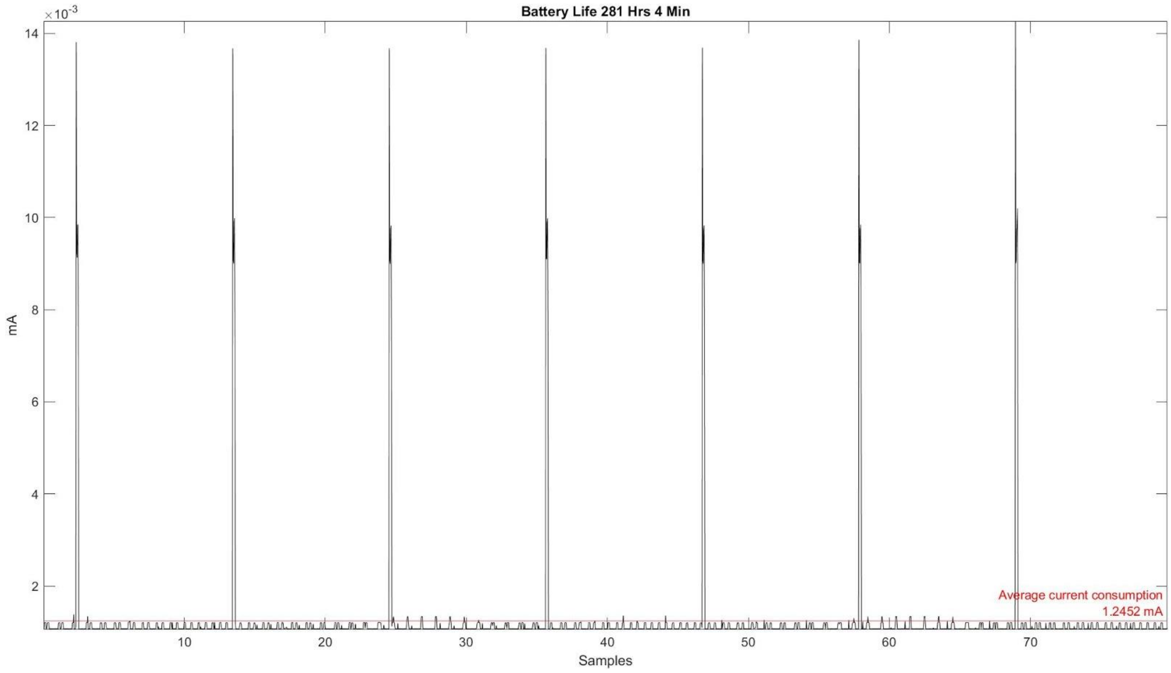

Figure 11: consumption with LED

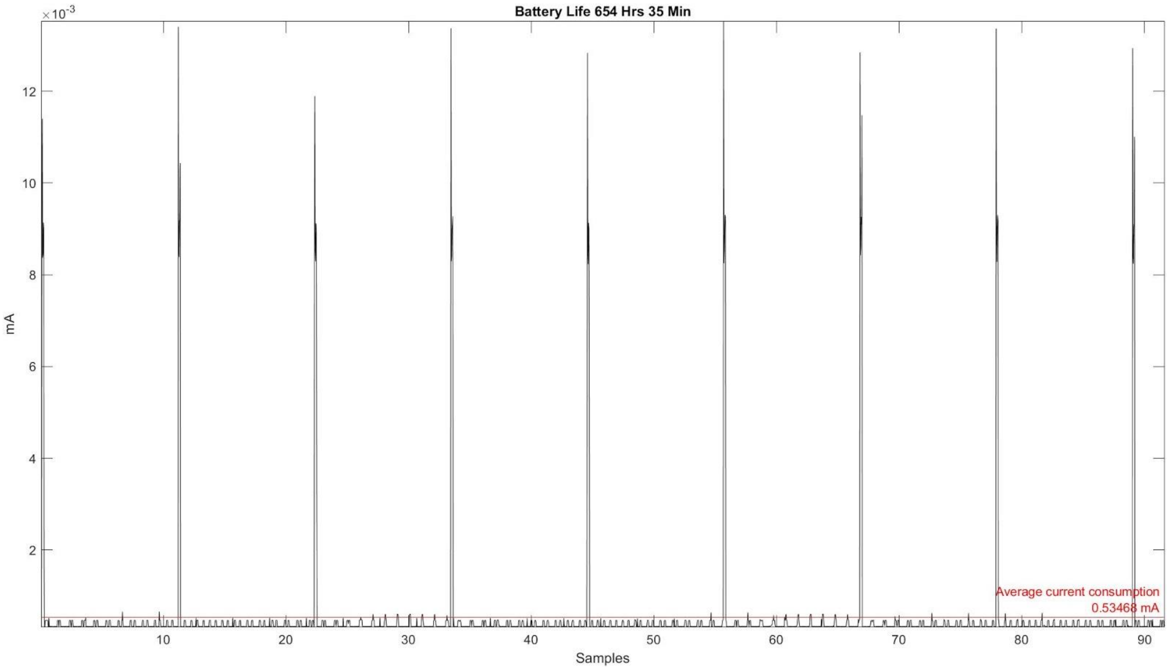

Figure 12: consumption without LED

As you can see in the figures above with the LED we have a consumption of 1.24 mA which represents a battery life of 281h (approximately 12 days). Besides that, if we don't have the specific LED the consumption would be 0.53 mA which represents 654h (approximately 28 days). If we did that modification we can improve the battery life by a 2.32 factor.

Now, we will talk about the performance of the LED talked previously. In the next figures you can see two different scenarios of why these LEDs were added. In the next case you can see how the green led (responsible for the higher consumption of the board) shows us the correct behaviour of the board when it is connected to a battery or USB power supply.

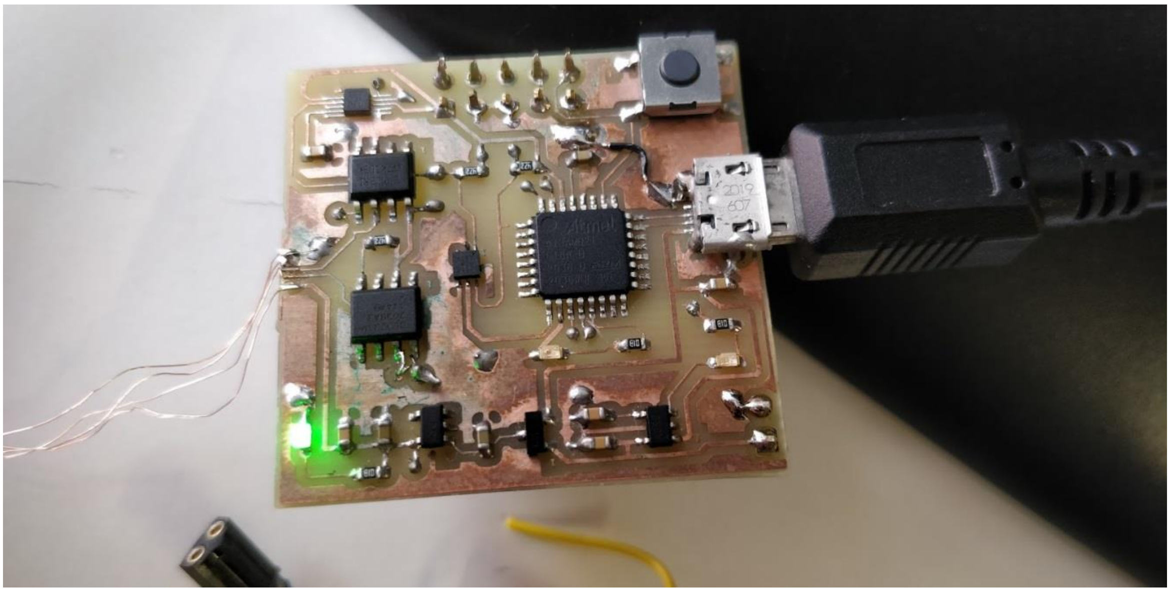

Figure 13: green LED with USB power supply

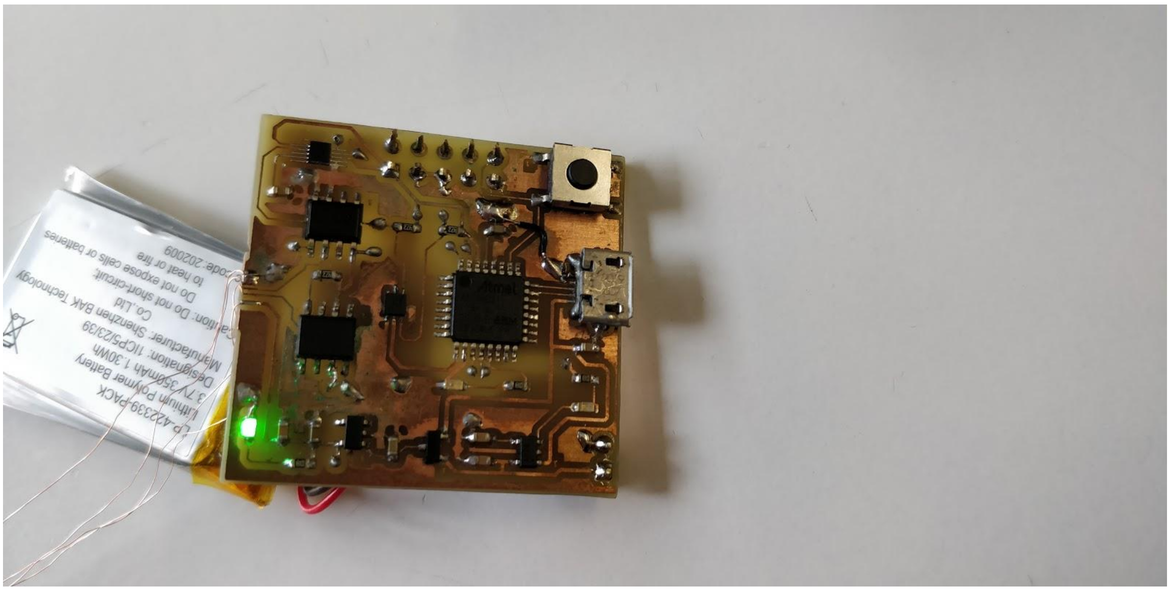

Figure 14: green LED with Battery power supply

Apart from the green led I added a blue LED which indicates the board is charging and it is shown in the next figure:

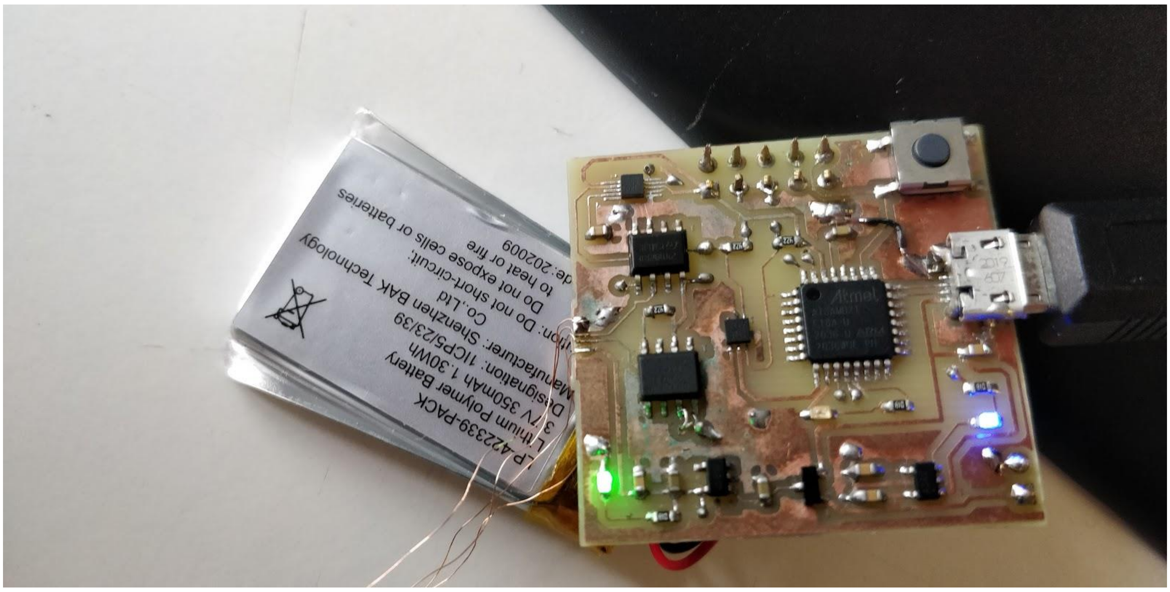

Figure 15: LEDs performance with charger mode

At the beginning the data stored in the EEPROM memory was downloaded in the terminal with a specific format and that results in a long time to download the full memory, for that I made some changes to the code to reduce that time to 9 min. The change was to download all the data in 8 bits format without any format and then I wrote a different code with Matlab to translate the data in a good format to read but it must be said that the new code takes an extra time around 60 sec to translate in the specific format. In the following picture you can see an example of that format.

	1 Date	2 Time	3 Pheripherical_temperature	4 Internal_temperature	5 Physical_Activity
1	"30/4/2021"	"17:43:52"	"0"	"28.79"	"NO"
2	"30/4/2021"	"17:43:55"	"0"	"28.76"	"NO"
3	"30/4/2021"	"17:43:58"	"0"	"28.67"	"NO"
4	"30/4/2021"	"17:44:01"	"0"	"28.58"	"NO"
5	"30/4/2021"	"17:44:04"	"0"	"28.5"	"NO"
6	"30/4/2021"	"17:44:07"	"0"	"28.42"	"NO"
7	"30/4/2021"	"17:44:10"	"0"	"28.36"	"NO"
8	"30/4/2021"	"17:44:13"	"0"	"28.3"	"NO"
9	"30/4/2021"	"17:44:16"	"0"	"28.26"	"NO"
10	"30/4/2021"	"17:44:19"	"0"	"28.21"	"NO"
11	"30/4/2021"	"17:44:22"	"0"	"28.18"	"NO"
12	"30/4/2021"	"17:44:25"	"0"	"28.14"	"NO"
13	"30/4/2021"	"17:44:28"	"0"	"28.11"	"NO"
14	"30/4/2021"	"17:44:31"	"0"	"28.09"	"NO"
15	"30/4/2021"	"17:44:34"	"0"	"28.06"	"NO"
16	"30/4/2021"	"17:44:37"	"0"	"28.04"	"NO"

Figure 16: example 1 of final data format

	1 Date	2 Time	3 Pheripherical_temperature	4 Internal_temperature	5 Physical_Activity
1	"26/5/2021"	"12:29:22"	"0"	"27.69"	"YES"
2	"26/5/2021"	"12:29:25"	"0"	"27.67"	"NO"
3	"26/5/2021"	"12:29:28"	"0"	"27.68"	"NO"
4	"26/5/2021"	"12:29:31"	"0"	"30.43"	"NO"
5	"26/5/2021"	"12:29:34"	"0"	"31.5"	"NO"
6	"26/5/2021"	"12:29:37"	"0"	"31.52"	"NO"
7	"26/5/2021"	"12:29:41"	"0"	"31.52"	"NO"
8	"26/5/2021"	"12:29:44"	"0"	"31.52"	"NO"
9	"26/5/2021"	"12:29:47"	"0"	"31.1"	"YES"
10	"26/5/2021"	"12:29:50"	"0"	"28.87"	"YES"
11	"26/5/2021"	"12:29:53"	"0"	"28.32"	"YES"
12	"26/5/2021"	"12:29:56"	"0"	"28.01"	"YES"
13	"26/5/2021"	"12:29:59"	"0"	"27.64"	"YES"
14	"26/5/2021"	"12:30:02"	"0"	"27.5"	"YES"
15	"26/5/2021"	"12:30:05"	"0"	"27.46"	"YES"
16	"26/5/2021"	"12:30:08"	"0"	"27.36"	"YES"

Figure 17: example 2 of final data format

Finally, below you can find a test during approximately 20 min walking around the laboratory. As you can see the sensitivity must be adjusted to eliminate the spurious. In this case, the Pheripherical temperature corresponds to room temperature and internal temperature the temperature inside the box.

Table 6: Final project test

Date	Time	Pheripherical_temperature	Internal_temperature	Physical_Activity
21/6/2021	12:54:14	32.21	33.4	NO
21/6/2021	12:54:25	32.41	33.25	NO
21/6/2021	12:54:36	32.5	33.05	NO
21/6/2021	12:54:47	32.55	32.82	NO
21/6/2021	12:54:58	32.62	32.6	NO
21/6/2021	12:55:09	32.67	32.39	NO
21/6/2021	12:55:20	32.71	32.19	NO
21/6/2021	12:55:31	32.75	32.02	NO
21/6/2021	12:55:42	32.81	31.86	NO
21/6/2021	12:55:53	32.84	31.7	NO
21/6/2021	12:56:04	32.86	31.56	NO
21/6/2021	12:56:15	32.89	31.42	NO
21/6/2021	12:56:26	32.94	31.3	NO
21/6/2021	12:56:37	32.96	31.18	NO
21/6/2021	12:56:48	32.96	31.09	YES
21/6/2021	12:56:59	32.93	30.99	NO
21/6/2021	12:57:10	32.9	30.89	YES
21/6/2021	12:57:21	32.8	30.8	YES
21/6/2021	12:57:32	32.38	30.77	YES
21/6/2021	12:57:43	32.1	30.79	NO
21/6/2021	12:57:54	32.01	30.85	NO
21/6/2021	12:58:05	31.85	30.89	NO
21/6/2021	12:58:16	31.72	30.99	NO
21/6/2021	12:58:27	31.88	31.03	NO
21/6/2021	12:58:38	31.83	31.05	NO
21/6/2021	12:58:50	32.03	31.17	NO
21/6/2021	12:59:01	32.66	31.31	YES
21/6/2021	12:59:12	32.75	31.42	YES
21/6/2021	12:59:23	32.56	31.47	NO
21/6/2021	12:59:34	32.37	31.51	YES
21/6/2021	12:59:45	32.21	31.54	NO
21/6/2021	12:59:56	32.07	31.57	YES
21/6/2021	13:00:07	31.86	31.6	YES
21/6/2021	13:00:18	32.1	31.62	NO
21/6/2021	13:00:29	31.93	31.64	NO
21/6/2021	13:00:40	31.93	31.71	NO
21/6/2021	13:00:51	31.63	31.79	NO
21/6/2021	13:01:02	31.39	31.82	NO

21/6/2021	13:01:13	31.21	31.83	NO
21/6/2021	13:01:24	31.06	31.82	NO
21/6/2021	13:01:35	30.93	31.8	NO
21/6/2021	13:01:46	30.85	31.77	NO
21/6/2021	13:01:57	30.86	31.75	NO
21/6/2021	13:02:08	31.28	31.74	NO
21/6/2021	13:02:19	31.16	31.75	NO
21/6/2021	13:02:30	31.15	31.75	NO
21/6/2021	13:02:41	31.21	31.75	NO
21/6/2021	13:02:52	31.35	31.75	NO
21/6/2021	13:03:03	31.32	31.78	NO
21/6/2021	13:03:14	31.29	31.76	NO
21/6/2021	13:03:25	31.29	31.74	NO
21/6/2021	13:03:36	31.28	31.72	NO
21/6/2021	13:03:47	31.28	31.72	NO
21/6/2021	13:03:59	31.29	31.73	NO
21/6/2021	13:04:10	31.31	31.73	NO
21/6/2021	13:04:21	31.32	31.73	NO
21/6/2021	13:04:32	31.45	31.74	NO
21/6/2021	13:04:43	32.01	31.74	NO
21/6/2021	13:04:54	31.83	31.78	NO
21/6/2021	13:05:05	31.65	31.8	NO
21/6/2021	13:05:16	31.85	31.82	NO
21/6/2021	13:05:27	31.8	31.94	NO
21/6/2021	13:05:38	31.79	32.1	NO
21/6/2021	13:05:49	31.65	32.23	NO
21/6/2021	13:06:00	31.96	32.27	NO
21/6/2021	13:06:11	31.94	32.25	NO
21/6/2021	13:06:22	31.9	32.23	NO
21/6/2021	13:06:33	31.92	32.21	NO
21/6/2021	13:06:44	32.01	32.18	NO
21/6/2021	13:06:55	31.95	32.15	NO
21/6/2021	13:07:06	31.88	32.1	NO
21/6/2021	13:07:17	31.8	32.03	NO
21/6/2021	13:07:28	31.73	31.95	NO
21/6/2021	13:07:39	31.79	31.85	NO
21/6/2021	13:07:50	31.87	31.77	NO
21/6/2021	13:08:01	32.12	31.71	NO
21/6/2021	13:08:12	32.16	31.71	NO
21/6/2021	13:08:23	32.15	31.72	NO
21/6/2021	13:08:34	31.91	31.75	NO
21/6/2021	13:08:45	31.82	31.77	NO
21/6/2021	13:08:56	31.98	31.85	NO
21/6/2021	13:09:07	31.97	32	NO

21/6/2021	13:09:18	31.75	32.07	NO
21/6/2021	13:09:29	31.65	32.08	NO
21/6/2021	13:09:40	31.69	32.06	NO
21/6/2021	13:09:52	31.71	32.03	NO
21/6/2021	13:10:03	31.71	32.01	NO
21/6/2021	13:10:14	31.8	32	NO
21/6/2021	13:10:25	31.83	31.99	NO
21/6/2021	13:10:36	31.86	31.98	NO
21/6/2021	13:10:47	31.98	31.97	NO
21/6/2021	13:10:58	31.98	31.99	NO
21/6/2021	13:11:09	32.43	32	NO
21/6/2021	13:11:20	32.58	32.03	NO
21/6/2021	13:11:31	32.43	32.07	NO
21/6/2021	13:11:42	32.36	32.1	NO
21/6/2021	13:11:53	32.35	32.21	NO
21/6/2021	13:12:04	32.35	32.38	NO
21/6/2021	13:12:15	32.38	32.56	NO
21/6/2021	13:12:26	32.37	32.73	NO
21/6/2021	13:12:37	32.33	32.82	NO
21/6/2021	13:12:48	32.34	32.87	NO
21/6/2021	13:12:59	32.34	32.97	NO
21/6/2021	13:13:10	32.3	33.03	NO
21/6/2021	13:13:21	32.23	33	NO
21/6/2021	13:13:32	32.17	32.93	NO
21/6/2021	13:13:43	32.11	32.83	NO
21/6/2021	13:13:54	32.14	32.71	NO
21/6/2021	13:14:05	32.14	32.63	NO
21/6/2021	13:14:16	32.17	32.54	NO
21/6/2021	13:14:27	32.27	32.48	NO
21/6/2021	13:14:38	32.23	32.42	NO
21/6/2021	13:14:49	32.22	32.37	NO
21/6/2021	13:15:00	32.22	32.32	NO
21/6/2021	13:15:11	32.19	32.28	NO
21/6/2021	13:15:22	32.16	32.26	NO
21/6/2021	13:15:33	32.28	32.25	NO
21/6/2021	13:15:44	32.5	32.26	NO
21/6/2021	13:15:55	32.72	32.28	NO
21/6/2021	13:16:06	32.75	32.34	NO
21/6/2021	13:16:17	32.88	32.41	NO
21/6/2021	13:16:28	32.86	32.47	NO
21/6/2021	13:16:39	32.89	32.53	NO
21/6/2021	13:16:50	32.94	32.57	NO
21/6/2021	13:17:01	32.84	32.6	NO
21/6/2021	13:17:12	32.71	32.62	NO

5. Budget

This project consists in developing a specific hardware to allow us the opportunity to do an assessment of the target. For that, we required a prototype and a final PCB.

To develop the PCB, the Software for the microcontroller and the box I needed a medium to large range in terms of performance. All the software and tools used would be open-source with the exception of Fusion 360 software from Autodesk and Matlab. Both aren't open-source software but I can access it for free due to my student status.

Furthermore, to develop correctly the project needs a multimeter and power supply to check the short circuits in the prototype PCB both also provided by the university..

Table 7: Components Summary

Component	Commercial Price (€)	Quantity	Subtotal(€)
ATSAMD21E18A-AU	3.3	2	6.6
AP2112K-3.3TRG1	0.655	2	1.31
DS3231MZ+	6.24	2	12.48
TMP116AIDRVR	4.1	3	12.3
M24M02-DRMN6TP	2.93	2	5.86
MMA8652FCR1	1.96	2	3.92
MAX1555EZK+T	1.61	2	3.22
B3FS-1000	0.18	2	0.36
MOLEX105017-0001	0.724	2	1.448
BAT54C,215	0.144	2	0.288
YOBLP422339PACK	19.96	2	39.92
		TOTAL	87.706

In terms of personnel, the project consists of a main worker and extra help of laboratory technician. The main worker is listed as a junior engineer and the average salary for one is 9€/h approximately for working a total of 430h which consists of 20h per week during 17 weeks. A laboratory technician is considered the same as a Senior Engineer and approximately a salary of 35€/h and 20 hours of work.

Table 8: Personal Salaries Summary

Workers	Salary/Hour (€)	Total Hours	Cost (without taxes) (€)
Junior Engineer	9	430	3870
Laboratory Technician	35	20	700
		TOTAL	4570

With that the total budget of the project is 4657,71€.

6. Conclusions

In this section I want to separate the project conclusions and my personal conclusions about me during the project.

On the one hand, I talk about the project conclusions. The project, finally, it's finished but later than the original deadline. This has happened because we had some problems during the development due to my mistakes in the design of the prototype board and also my lack of experience in SMD components soldering. Moreover, the start objectives that I thought would be difficult, were not. For example, finding the correct Arduino bootloader and uploading it on the board, it was so easy with the help of Adafruit web page and the programmer that Marc let me.

The prototype developed can record two temperatures and the activity of the patient/subject for up to 28 days, more than the initial designing. It will allow the research group to make long term studies of the peripherical temperature and activity.

On the other hand, I would also like to explain my own conclusions. At the beginning I needed a great number of hours of research to learn about things that I believed impossible but with patience and the help of Juan it was really quick and easy. It was the first time that I designed a full PCB and I'm really happy with the results despite the errors. Maybe the most difficult part for me was the way to design the correct box to store the final hardware which I thought it would be easier.

Finally, it was a project that gave me a great deal of motivation and I learned a great number of things, tips and engineering solutions which will help me in my future career.

7. Future Development

Some improvements that could be interesting is to replace some components for another or the same component but with another kind of package. For example, I work with the Atmel SAMD21 E18 in a TQFP package and it can be replaced by the QFN package which can give us an improvement in terms of dimensions due to the second one having half an area.

That situation can be extended to all components but with the problems that these components can't be soldered at the lab facilities and will have to be outsourced.

Also an improvement could be to do an specific algorithm embedded in the microcontroller to detect the kind of exercise which the subject was doing to reduce the amount of memory needed but taking into account the problem of autonomy due to the probably increase of power consumption.

Bibliography:

[1] "Temperatura basal y días fértiles". Sanitas. [Online]. Available: https://www.sanitas.es/sanitas/seguros/es/particulares/biblioteca-de-salud/embarazo-maternidad/quedarme-embarazada/temperatura-basal/index.html

[2] "M24M02_EEPROM". GitHub. [Online]. Available: https://github.com/kriswiner/M24M02_EEPROM

[3] Brent Rubell. "How to Program SAMD Bootloaders". Adafruit. [Online]. Available:https://learn.adafruit.com/how-to-program-samd bootloaders/programming-the-bootloader-with-atmel-studio

[4] Sayanee Basu. "6 ways to access the serial monitor for Arduino". YouTube. [Online]. Available: https://www.youtube.com/watch?v=UwOLAL9z2Rk

Glossary

RTC: Real time clock

LDO: Low Dropout Regulators

PCB: Printed circuit board

WP: Work Package

TQFP: Thin Quad Flat Package

QFN: Quad Flat No-Lead Package

UPC: Universitat politecnica de Catalunya

SMD: Surface-mount technology