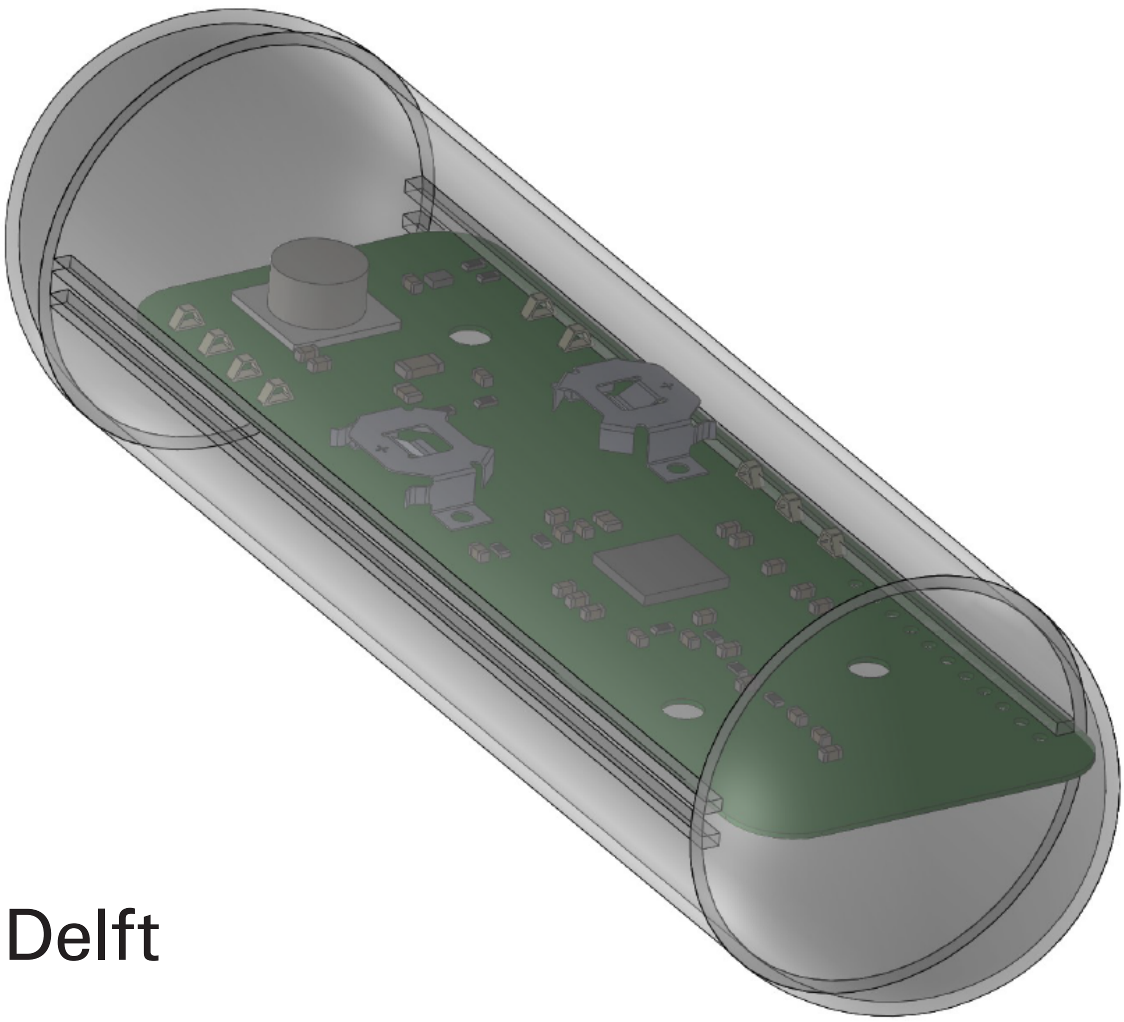
Sensors and design of an endoscopic pill

BSc Thesis

Zakaria Abdellaoui
Esad Beydilli



TU Delft

Sensors and design of an endoscopic pill

BSc Thesis

by

Zakaria Abdellaoui
Esad Beydilli

to obtain the degree of Bachelor of Science
at the Delft University of Technology,
to be defended on Wednesday July 3, 2019 at 13:30.

Project duration:	April 21, 2019 – July 5, 2019	
Supervisor:	Dr. V. Valente	
Thesis committee:	Dr. ir. J. J. A. Baselmans,	TU Delft, Chair
	MSc L. Pakula,	TU Delft

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft

Abstract

Endoscopic devices are used in the medical world to inspect the gastrointestinal tract. Because there are some complications with wired endoscopy, such as reaching the small intestine, there is a need for new methods to do endoscopy. Therefore an endoscopic pill is invented which can be used to do measurements in the gastrointestinal tract. This endoscopic pill is a small device with small sized electronic sensing elements.

This thesis goes through the implementation process of sensors inside an endoscopic pill in order to be able to perform measurements inside the human body. The sensors used in the endoscopic pill are a temperature and a pressure sensor. Besides that, this thesis also goes through the design process of the 3D design of the capsule, which is later 3D printed. The goal of the project is to make a prototype which contains functioning sensors and which can be made swallowable during future work.

Preface

Before you lies the thesis ”Sensors and design of an endoscopic pill”. This thesis is about implementing sensors inside an endoscopic pill and the 3D design of the endoscopic capsule. This thesis is written as a part of the Bachelor Graduation Project at Delft University of Technology, alongside the development of a product. The project is commissioned under supervision of dr. Virgilio Valente. We would like to thank dr. Valente for his guidance during the project and the employees at the Tellegen Lab for providing us with all required instrumentation.

We also want to thank our colleagues and group members, Ahmed Aouichi, Aschraf Gardouh, Avinash Kalloe and Rami Younis. We would like to thank you for your cooperation, it was a pleasure to work with you.

We hope you enjoy reading this thesis.

Zakaria Abdellaoui
Esad Beydilli
Delft, June 2019

Contents

1 Introduction

Endoscopic devices are used in the medical world to inspect the gastrointestinal tract. They are commonly used for diagnosing diseases such as gastrointestinal bleeding, Crohn's disease and tumors inside the bowel [13]. Endoscopy is traditionally performed using a flexible tube with a camera attached at its end. This method does have challenges, as ingesting the tube into a patient may be uncomfortable. The major problem with traditional endoscopy has been the inability to reach certain areas, especially the lower part of the small intestine [16]. Because of this, evaluation of obscure gastrointestinal bleeding has frequently been unsatisfactory [13].

In 2001, the first commercially available endoscopic pill, designed by Given Imaging, received clearance by the Food and Drug Administration [14]. Since then, capsule endoscopy has been widely used in the medical field as a viable, non-invasive alternative to traditional endoscopy. This endoscopy method involves a device which is the size of a pill and which can send relevant information from the gastrointestinal tract to a computer. This thesis covers the implementation of different sensors into a swallowable pill-sized device in order to perform measurements inside the digestive tract.

The goal of this project is not to deliver an actual swallowable pill, but rather a prototype with functioning sensors inside a capsule which later can be made smaller, at the size of a pill, while having the same functionality. The prototype may be too large to be swallowable and may not be waterproof and biocompatible, but serves as a step towards making a swallowable one.

1.1. Systems in the pill

The endoscopic pill consists of sensors, batteries, control unit, transmitter and a power management unit(see figure 1.1).

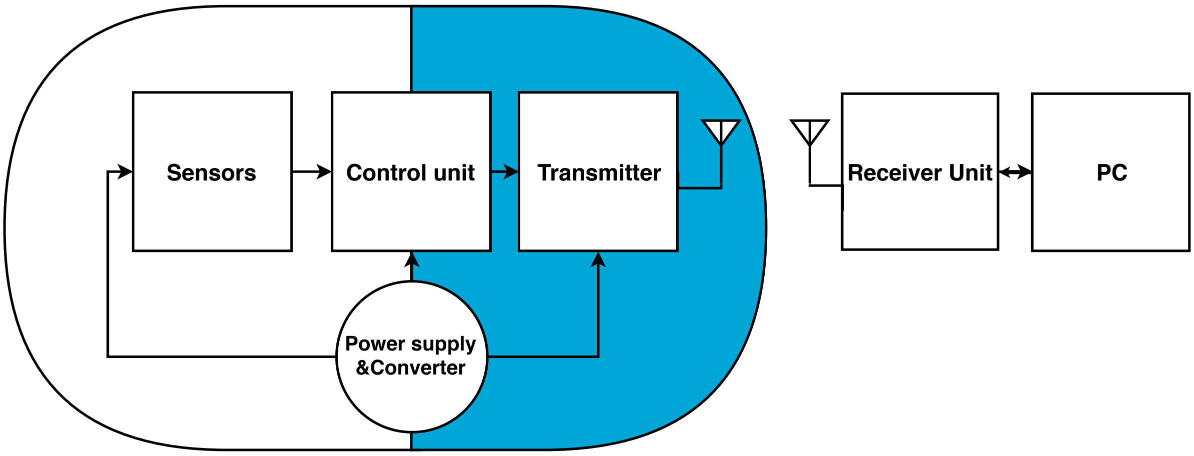


Figure 1.1: Endoscopic capsule on system level

Externally there is a receiver which is connected to a PC, this PC will show the result of the measurements done by the sensor inside the pill. The power management unit makes sure that every unit in the pill receives enough power and current. The sensors in the pill send their data to the micro controller, which makes the antenna send the data to an external receiver. For designing this endoscopic pill, the workload has been split into three different subgroups. A subgroup is working on the power management, a subgroup working on the sensors and another subgroup working on data-transmission. The subgroup writing this thesis is responsible for the sensors, another responsibility of this group is making a 3D-design of the capsule. In the 3D-design the exposure of the sensors should be taken in account, if necessary.

1.2. State-of-the-art analysis

Currently available swallowable telemetry devices are often equipped with an image sensor [12] [15]. Most of these devices work with a battery which can last between 8 and 12 hours [24]. For monitoring the stomach and the small intestine, this is sufficient. However, for performing measurements through the entire traject including the large intestine, the battery would have to last between 24 and 48 hours [28]. While wireless power transfer with inductive coils is a possible solution [3], this may limit the mobility of the patient, and thus may be uncomfortable.

Telemetry capsules without image sensors are also available. Instead of taking images, they can, for example, measure temperature, pH and pressure and have much longer battery lives than capsules with cameras, often longer than 5 days [6].

1.3. Possible sensors

The possible sensors which cab be used in the endoscopic were observed. For classifying a sensor as potential, the relevance of the measurements it offers is taken into accountant. Does it provide relevant measurements in the gastrointestinal tract, which can be used by doctors to diagnose some disease. For determining the sensors which will be proceeded to work on, the feasibility is analyzed.

1.3.1. Bleeding sensor

In the gastrointestinal tract bleeding might occur. This might be a symptom for different diseases. For example it might indicate there is cancer in the gastrointestinal tract [23]. Therefore a sensor which can measure blood is a good sensor to have in the endoscopic pill.

A sensor which is useful for sensing blood in digestive system is a bacteria on chip sensor [19]. For this sensor a bacteria is used which has the ability to detect heme, a component of blood [19]. When there is blood a cell on the sensor will glow. To detect the glowing of the sensor, a phototransistor is used. But this sensor is not commercially available and therefore could not be used in this project.

1.3.2. pH sensor

The second sensor considered was the pH sensor. The pH level is an important physic-chemical parameter that can provide useful information in clinical management of diseases within the gut [7]. Therefore it might be useful to measure the pH level in the digestive system. The most commonly used types of pH sensors are ISFET sensors, this ISFET sensor can measure local pH of the stomach acid and the intestine [8]. An ISFET sensor is a Ion Sensitive Field-Effect Transistor. ISFET sensors are expensive and therefore not feasible for this project. There are alternative pH sensors, but those are too big to fit inside the pill.

1.3.3. Temperature sensor

The degree of body temperature is an important indication of health as well as illness and often constitutes the basis for deciding whether or not to initiate treatment [22]. There are a lot of different temperature sensors available, therefore it is decided to make a temperature sensor for inside the capsule.

1.3.4. Pressure sensor

Another measurement which might help diagnosing patients on a disease is pressure. It helps by diagnosing motility disorders, motility is the contraction of the muscles that propel contents in the GI tract. Pressure in the gastrointestinal tract can also offer information of some disease [10]. Therefore a pressure sensor is a good optional sensor. There are already some pressure sensors in the market which can be used. There are also some concept for making a pressure sensor, which can be used to make an own pressure sensor. Therefore the choice is made to add this sensor in the endoscopic pill.

1.3.5. camera sensor

An image sensor is useful for taking pictures of the gastrointestinal tract. For the image sensor, the two most common used types are CMOS and CCD. While CCD image sensors can produce higher quality images and have better signal-to-noise ratio, CMOS image sensors cost less per performance, consume less power and thus are more suitable for the endoscopic pill [18].

Most CMOS image sensors have raw image data (used to determine the color of the current pixel), and horizontal and vertical synchronous signals (used to locate the current pixel) at their output. In order to save power, the raw data should be compressed, as it is found that image compression can reduce approximately 48% of the total power consumed by the capsule [25]. This can be done by storing one frame from the image sensor in a memory, performing near-lossless compression on the frame [27], and storing it back in the memory, after which the frame is ready to be transmitted.

It is also useful to serialize the compressed frame, which is in parallel format, before sending it to the microcontroller for transmission [26]. An advantage of using a serial output is that it requires fewer pins on the microcontroller, and thus makes it easier to route the signals on a PCB. The Serial Peripheral Interface, for example, is an interface which consists of only four wires: SCLK (Serial Clock), MOSI (Master Output Slave Input), MISO (Master Input Slave Output) and SS (Slave Select) [29]. This is much fewer than the typical output of an image sensor, which could consist of 8 data wires, 2 synchronous wires (horizontal an vertical) and a pixel clock wire.

In order to picture inside the human body with an endoscopic pill with a camera, a light source is required. For the purpose of illuminating inside the body, LEDs are the preferred light source because of their long life, stability, reliability and their ease of integration [2].

Software has to be written for the microcontroller in order to configure the image sensor, and for the computer in order to translate received serial data into actual frames. During this project, writing software to translate bitstreams and designing the serial interface would not leave enough time for working on the other sensors. Therefore, the image sensor is not added to the product.

Table 1.1: An overview of all considered sensors

Sensor type	Medical use	Feasible for project
Temperature	Detecting illness	Yes
pH	Detecting diseases in the gut	No
Pressure	Detecting motility disorders	Yes
Blood	Detecting symptoms of cancer	No
Camera	Imaging diseases, e.g. inflammatory bowel diseases	No

1.4. Thesis structure

The next Chapter, which is Chapter 2, lists all requirements whith which the end product must or should preferably comply. Chapter 3 goes through the design process of the sensors and the 3D design of the capsule. Chapter 3.4 explains the implementation steps for the prototype. Chapter 5 covers the testing process and the results of the implementation of sensors and 3D design.

2 Programme of requirements

As stated in the introduction, chapter 1, this subgroup is responsible of the sensors inside the endoscopic pill. This section covers the requirements concerning the sensors and 3D-design, divided into mandatory requirements and trade-off requirements.

2.1. Mandatory requirements

Mandatory requirements are constraints with which the product always needs to comply. Making the product meet these requirements is essential during the design process. If any of the mandatory requirements is not fulfilled, the product will be incomplete. The mandatory requirements concerning the sensors and 3D-design of the endoscopic pill are as follows:

MAN1 The total volume of all implemented sensors must be smaller than the volume of a capsule which is swallowable, which has a diameter of 11 mm, length of 23 mm and a volume of 1837.3 mm^3.

MAN2 At every request that is sent to any of the sensors, information must be returned by the sensor within 1 second.

MAN3 All sensors must measure with such frequencies that the total battery life is at least 8 hours (in order to at least capture the small intestine).

MAN4 All sensors must have an operating voltage of 3.3 V.

MAN5 The product must contain at least one sensor which can respond to incoming commands.

2.2. Trade-off requirements

Trade-off requirements are preferred for the product to comply with. These requirements offer additional features which may be beneficial for the end user and are as follows:

TO1 The product should have an image sensor to visualize the intestines.

TO2 The product should have a temperature sensor.

TO3 The product should have a pressure sensor.

TO4 The product should have a pH sensor.

TO5 The product should have a bleeding sensor which is able to measure blood.

TO6 All sensors should measure with such frequencies that the total battery life is at least 24 hours (in order to also capture the large intestine).

TO7 The endoscopic pill should have the shape of a capsule with a diameter of 11 millimeters and length of 23 millimeters, resulting in a total volume of 1837.3 mm^3.

TO8 The material of the capsule should be biocompatible and non-harmful to living tissue.

2.3. Sensor-specific requirements

The sensors which may be implemented as trade-off requirements also have their own specific requirements, which are listed per sensor in the following sections.

2.3.1. Image sensor

IMG1 The image sensor should have a resolution of at least 100x100 pixels.

IMG2 The image sensor should be able to take at least one frame per two seconds.

IMG3 The image sensor should be able to take RGB images by using a Bayer color filter array.

IMG4 The image sensor should be a CMOS sensor.

IMG5 The image sensor should have a serial output.

IMG6 The image sensor should be exposed in the design of the pill.

2.3.2. Temperature sensor

TMP1 The temperature sensor should have a resolution of 0.1 °C or less.

TMP2 The temperature sensor should operate at the temperature range between 30 °C and 45 °C

TMP3 The inaccuracy of the temperature sensor should not be more than ±1 °C.

2.3.3. Pressure sensor

PRS1 Inside the pressure range between 0.9891 bar and 1.0116 bar [10].

PRS2 The pressure sensor should be able to measure with an inaccuracy less than 1 millibar.

PRS3 The pressure sensor should have a resolution of 0.1 millibar or less.

PRS4 The pressure sensor should be exposed in the design of the pill.

2.3.4. pH sensor

PH1 The pH sensor should have an operating range from 1 to 7.9 pH [4].

PH2 The pH sensor should have a resolution of 0.1 pH.

PH3 The pH sensor should have an accuracy of 0.1 pH.

2.3.5. Blood sensor

BLD1 Should be able to detect heme molecules.

3 Design process

In this project a temperature sensor and a pressure sensor is used for the endoscopic pill. These were chosen because it measures valuable parameters inside the gastrointestinal tract. These parameters can be used to help doctor to diagnose patients. For the sensors first how to design a sensor was looked into. After this was done the sensors were chosen. When choosing the sensors the requirement in Chapter 2.3 were kept in mind. Another important specification which was kept in mind, is the size of the sensor.

3.1. Temperature sensor

For the temperature sensor, a thermocouple and a NTC thermistor was looked into. These are two different types of temperatures sensors. A thermocouple is a thermoelectric sensor which consists of two junctions of two metals. One of the junctions is kept at a known temperature, the other junction is the one measuring the temperature [17]. Due to temperature difference between those two junctions, there will be a voltage across those junctions, this voltage is used to measure the temperature. For this project a thermocouple would not be a useful sensor, since for one junction constant temperature is necessary. This can not be done in the endoscopic pill.

The NTC thermistor, is a resistance with a negative temperature coefficient. This means when the temperature increases, resistance decreases. This can be used in a temperature, by connecting a resistor with know resistance in parallel. By supplying a known voltage to this circuit, the voltage across the thermistor can be used to determine the temperature. The circuit of such a sensor is given in figure 3.1.

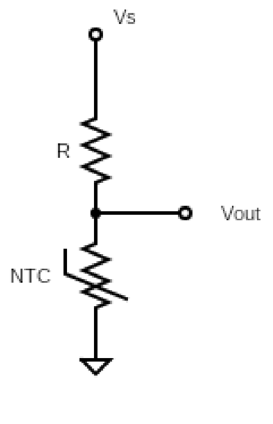


Figure 3.1: Circuit for temperature sensor with a thermistor

.

From the measured output voltage, and the supplied voltage, the resistance of the thermistor can be calculated. From the resistance the temperature can be retrieved. The resistance of the thermistor can be calculated by using Kirchhoff's Current Law and Ohm's Law 3.1.

$$R_{NTC} = \frac{R \cdot V_{out}}{V_s - V_{out}} \tag{3.1}$$

Before starting to work with a thermistor, its suitability for this project should be observed. The temperature range and accuracy should be known. For this project the B57861S thermistor was available. To determine the accuracy of this thermistor, the temperature vs resistance plot was made. This was made by using the R/T characteristics given in [1]. This table shows the resistor value at a certain temperature. For describing the relation between the temperature and resistor value, the Steinhart and Hart Equation can be used [21], which is given in equation 3.2

$$T = \frac{1}{A + B\log(R) + C(\log(R)^3} \tag{3.2}$$

The variables A, B, C are the so called Steinhart-Hart coëfficients, by determining these a temperature vs resistance plot. These coefficients can be calculated by filling in three different values for temperature and resistor, this is done by a MATLAB script. The temperature vs resistance plot can be plotted using those coefficients. This resulted in the graph shown in figure 3.2

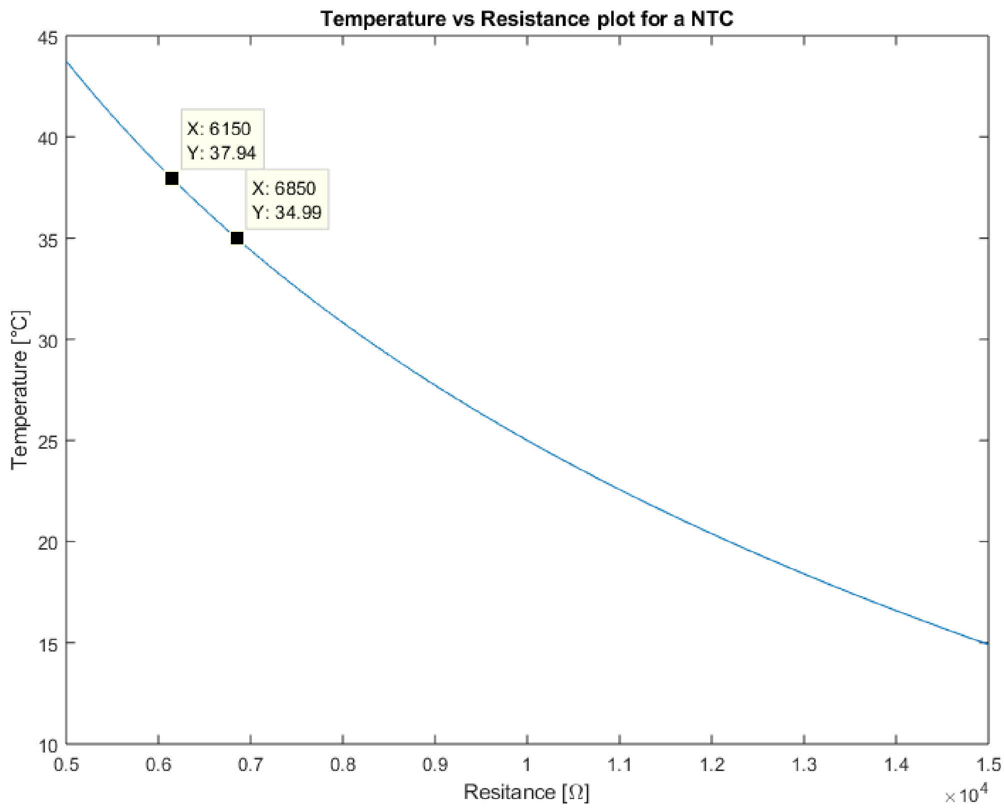


Figure 3.2: Temperature vs resistance plot for B57861S thermistor

.

As can be seen in figure 3.2 two points are marked, this is to show the worst-case scenario accuracy of the thermistor. The tolerance of the thermistor is ±1%, if for example the temperature is ±35 °C, the resistor value is 6850Ω. Given the tolerance is ±1%, in the worst case the resistance of the resistor is 6150Ω, so the error is the ±3 °C. Therefore this sensor might not be a good sensor for this, since it does not satisfy requirement TMP3.

Since there were no thermistors available with higher accuracy, the next types of temperature sensors looked into is a prefabricated temperature sensor. Most of these sensors are digital temperature sensors. A selection of the prefabricated sensors was made, this selection is shown in table 3.1, with some of their specs. For the accuracy given in table 3.1, the accuracy between 25 to 45°C has been observed. Some of the resolutions is given since it depends on on the Analogue to Digital Converter.

Table 3.1: Selection of sensors with their specs

	Operating voltage	Temperature range	Accuracy	Resolution
LMT01-SP	2V to 5.5V	–50°C to 150°C	1.5°C	0.0625°C
TMP112B	1.4V to 3.6V	–40°C to 125°C	±0.1°C	0.0625°C
TMP235	2.3V to 5.5V	–50°C to 150°C	±0.1°C	-

Because of requirement TMP3 is not satisfied by LMT01-SP, this sensor is left out consideration. So the choice between TMP112B and TMP235 has to be made. The choice was made to use the TMP112B, since it has a I^2C interface which is a universal communication protocol and good to use. Besides for the TMP235 an good ADC converter should be found.

TMP112B is a digital temperature sensor, which measures temperature by using a diode temperature sensor. The current trough the diode changes due to temperature, because of this current change the voltage across the temperature diode changes. This voltage gives information about the temperature. The voltage across the diode will be converted to a digital value of 12 bit by an ADC converter in the sensor. This value will be saved in a register as 2 bytes in which the 4 least significant bits are ignored for reading temperature, since the temperature is indicated by 12 bits. The LSB of those 12 bits is equal to 0.0625°C. So to translate the value stored in the temperature register value it has to be multiplied by 0.0625. A schematic overview of the described data stream inside the temperature sensor is given in figure 3.3.

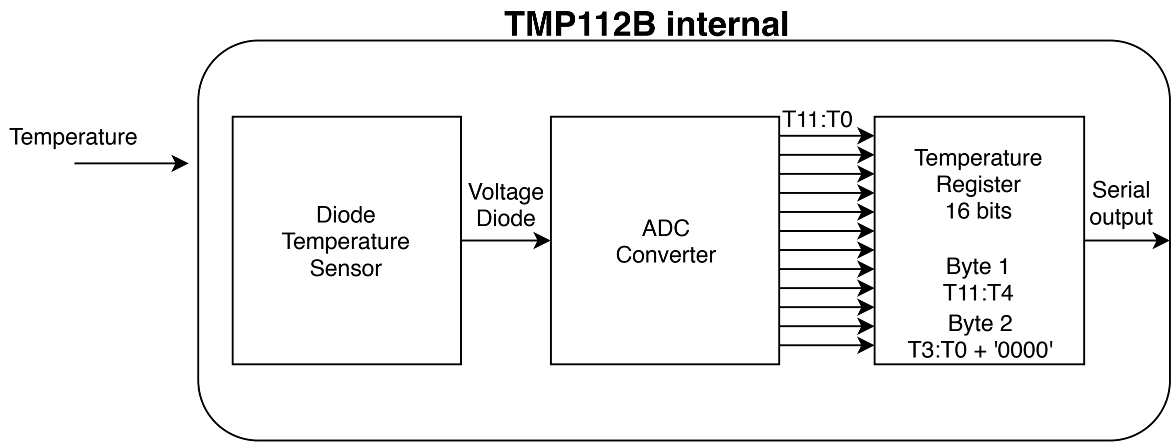


Figure 3.3: Schematic internal data stream overview of TMP112B

.

3.2. Pressure sensor

Among many different available types of pressure sensors, a piezoresistive pressure sensor is chosen for this project, as it is the easiest available type of pressure sensor, is more sensitive to change in pressure, can respond in milliseconds and only requires one pressure port, as it does not need to compare the measured pressure like the absolute or differential pressure sensors [20].

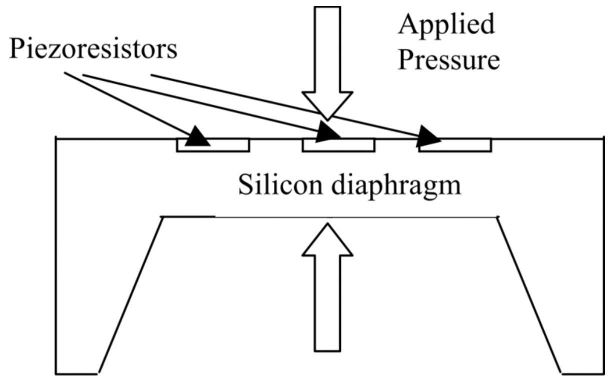


Figure 3.4: Schematic overview of a piezoresistive pressure sensor [20].

Piezoresistive pressure sensors measure the pressure with a Wheatstone bridge of piezoresistors, as shown in figure 3.4. A change in pressure results in the bending of the silicon diaphragm, which results in a change in resistance in the piezoresistors, resulting in a change in current running through the bridge. This change in current is then measured and the pressure is obtained.

For the pressure sensor in this project, the MS5534C barometer module [11] is used, shown in figure 3.5.

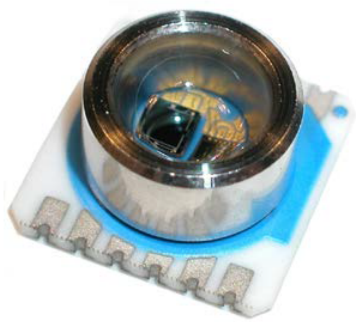


Figure 3.5: The MS5534C barometer module [11].

This module contains a piezoresistive pressure sensor, an ADC-interface and a memory component. These components are shown in the block diagram of the module in figure 3.6. The ADC-interface is used for converting the analog output voltage of the pressure sensor to a digital value. The memory contains predefined calibration coefficients, which is defined by the manufacturer and is used for minimizing errors during pressure measurement. The multiplexer switches between retreiving data from the sensor and delivering power to the digital interface. The pressure sensor on the MS5534C module is surrounded by silicone gel in order to avoid damage by humidity. The resolution of this sensor is 0.1 millibar.

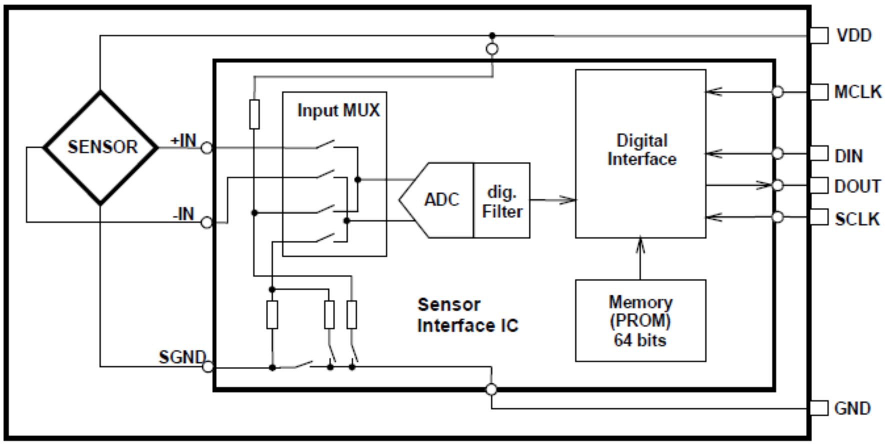


Figure 3.6: Block diagram of the MS5534C barometer module [11].

As stated in requirement PRS2, the pressure sensor should be able to measure accurately between 0.9891 bar and 1.0116 bar. Figure 3.7 from the datasheet of the sensor [11] shows that in that pressure range, the pressure error is close to 0 mbar for 25 °C and 60 °C. Because the temperature range inside the human body is somewhere between these two temperatures, we can assume that the pressure error is close to 0 mbar for the temperature range inside the human body. Figure 3.8, also from the datasheet [11], in which the pressure error is plotted over temperature, also confirms this assumption. The figure shows that inside the temperature range of the human body, the pressure error is almost zero, except for the case in which the pressure is 300 mbar. However, this amount of pressure is unlikely to occur inside the human body.

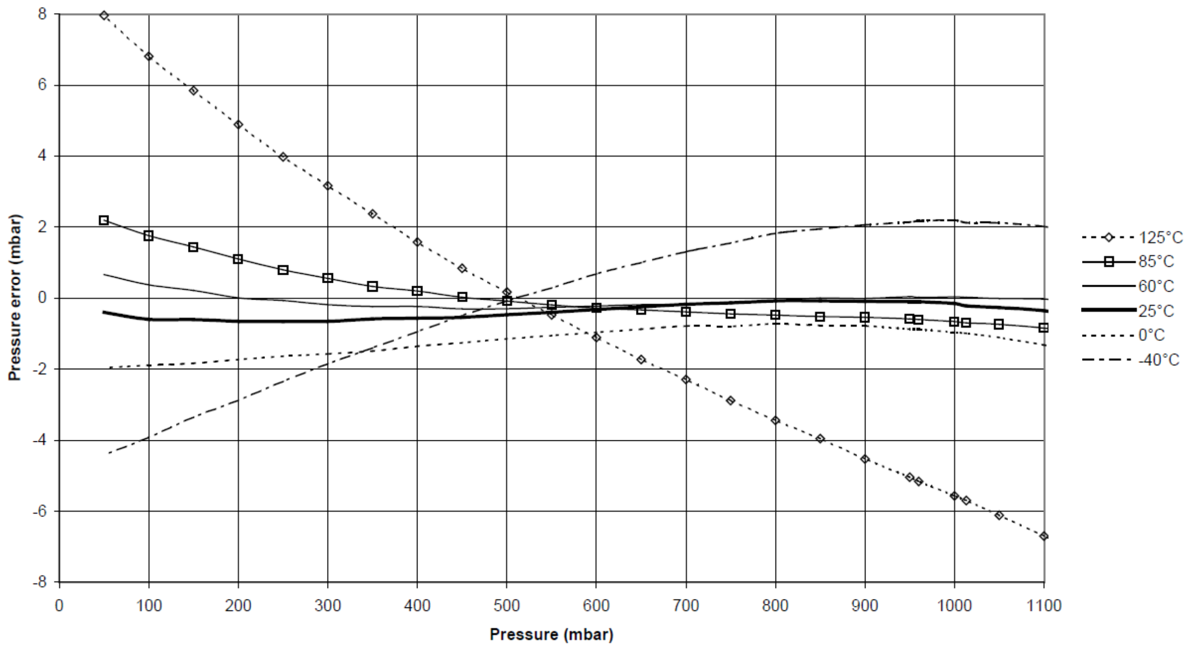


Figure 3.7: Absolute pressure accuracy after calibration using predefined calibration coefficients, for the MS5534C barometer module [11].

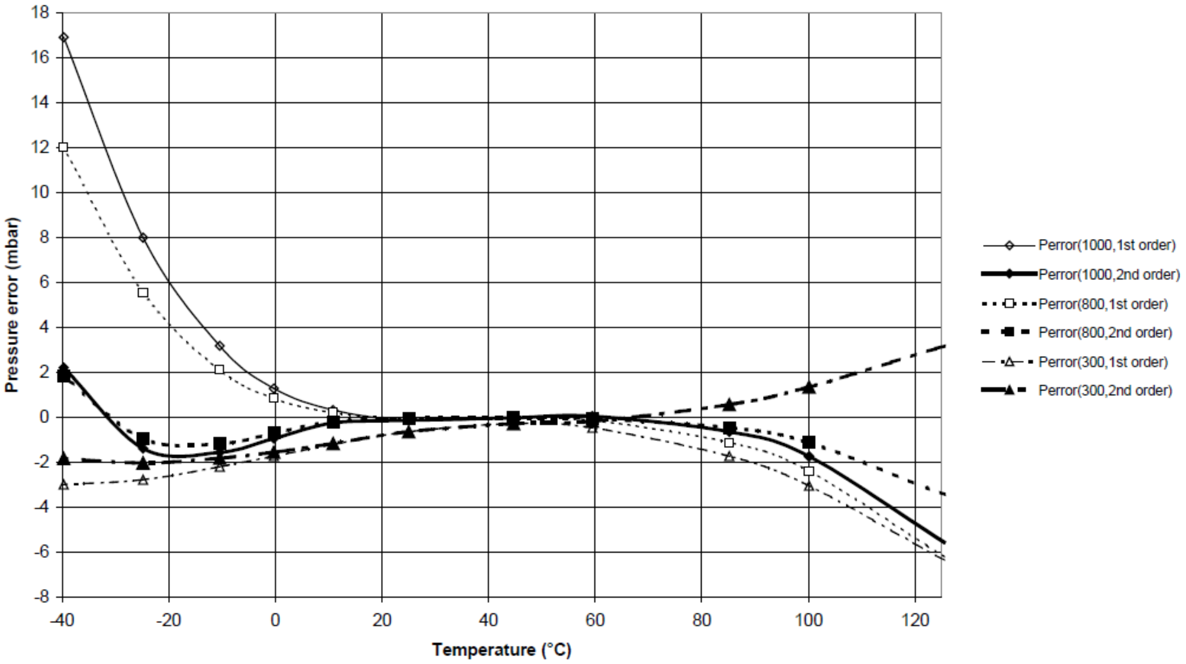


Figure 3.8: Pressure error accuracy vs temperature, for the MS5534C barometer module. 2nd order includes additional calibration [11].

3.3. Combination temperature and pressure sensor

For now the design of the sensors separately has been looked at, but there has also a point to combine these two designs. For combining those two sensor it was chosen to sequentially retrieve the data from the sensors. These values will then be combined in one packet by the data transmission subgroup and send to the computer.

3.4. 3D design

The design of the shell around the product has the shape of a capsule and is 3D printed. The dimensional constraints given in requirement TO7 should be followed in order to make the capsule swallowable. However, the prototype in this project is much larger than that, as this design is focussed on having functional sensors instead of swallowability. The design is able to hold a PCB containing all sensors.

The program Solidworks is be used to make the capsule design. It is constructed from a hollow cylinder and two hollow hemispheres, to be 3D printed. Inside the cylinder, rails are added in order to hold the PCB in place. This cylinder is shown in figure 3.9.

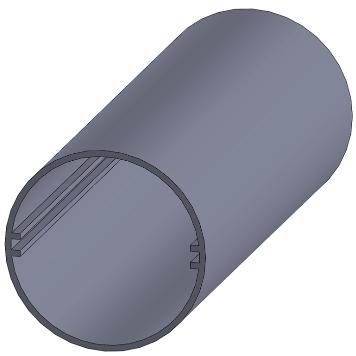


Figure 3.9: The cylinder of the 3D-design, with rails to keep the PCB in place.

The hollow hemispheres have a round part of which the radius is 0.2 mm smaller than the radius of the cylinder. They serve as caps for the cylinder and can fit tightly at each end. Figure 3.10 shows a cap.

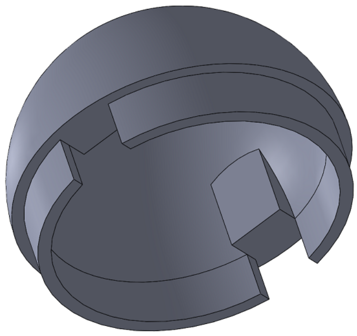


Figure 3.10: A cap of the 3D design, designed to fit tightly at one of the ends of the cylinder.

Besides the 3D design of the prototype, an effort has been made in making a concept model for the smaller, pill-sized design. Figure 3.11 shows this model. The hole in the design is meant for the pressure sensor. As stated in requirement PRS3, the pressure sensor should be exposed in order to be able to measure. If another sensor than the pressure sensor is used that requires to be exposed, like the image sensor, this hole should be adjusted.

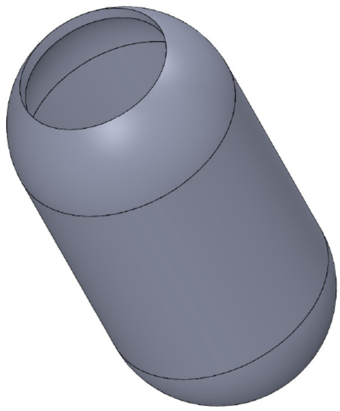


Figure 3.11: The 3D design of a pill-sized capsule with a hole for a pressure sensor.

4

Prototype implementation

For the implementation the two sensors were at first implemented separately. After those two sensors were implemented, they were combined. In this chapter it will be explained how those sensors were implemented and how they were combined.

4.1. Temperature sensor

For implementing the temperature sensor, the Sensor Controller of the microcontroller is used. This sensor controller is ideal for interfacing external sensors and for collecting analog and digital data autonomously while the rest of the system is in sleep mode[5]. To work with this unit of the microcontroller, Sensor Controller Studio should be used, this program generates a Sensor Controller interface which can be used in Sensor Controller Studio. Code Composer Studio is an integrated development environment that supports the microcontroller. In Sensor Controller Studio the Serial Clock was set on 400 kHz, for this frequency the average current in the sensor is lower. In Sensor Controller Studio there are already some predefined functions available for the I^2C protocol. For example a start and stop function to start or stop sending data to the sensor.

The code in Sensor Controller Studio consists of four parts, Initialization Code, Execution Code, and Event Handler A Code. In the initialization the temperature sensor is configured, the values in the configuration register is kept on its default value (see Appendix A.1.1). This means there will be 4 conversions per second, this is chosen to be sure to keep the temperature up to date. The sensor begins in the Initialization Code and then goes to the Execution Code. In this part of the code the time of reading values from temperature sensor is scheduled (see Appendix A.1.2). The sensor is implemented to measure 1 time per second, this is considered to be enough data to get sufficient information about the gastrointestinal tract. This is done by using a Real-Time Clock and a Timer. The timer is used to trigger the Event Handler Code(see Appendix A.1.3), in this code the value from the temperature register is read. When this is done an interrupt is generated by the Sensor Controller, this alerts the main CPU that a new value has arrived. This value can then be further processed in the main controlled for data transmission. When the Event Handler Code is finished it returns to the Execution code. In the Execution code the Real-Time Clock is used to schedule the next execution, as said before it is chosen to execute one execution per second. So the next execution is scheduled for the next second, this means that Execution Code will be run again. A schematic overview of the code in Sensor Controller Studio is given in figure 4.1. After the program is finished writing, a set of files can be generated which can be used in Code Composer.

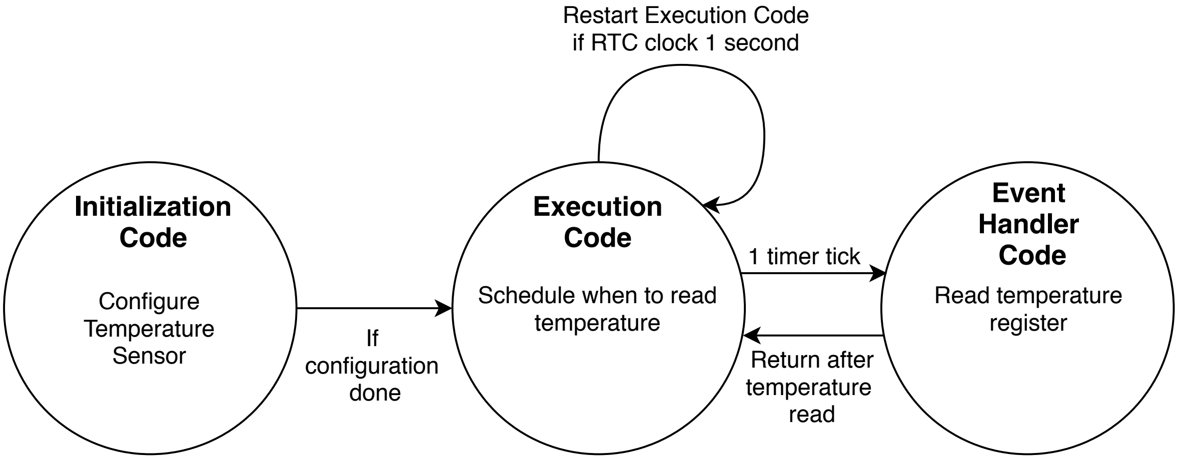


Figure 4.1: Schematic overview of working principle of the code in Sensor Controller Studio

.

Before the microcontroller and temperature sensor can be connected, the sensor has first to be soldered on a breakout board, to make it possible to connect microcontroller pins to the temperature sensor's pins. The temperature sensor requires also a pullup resistors on the SCL and SDA pin. There is also a bypass capacitor on the supply needed as shown in figure 4.2.

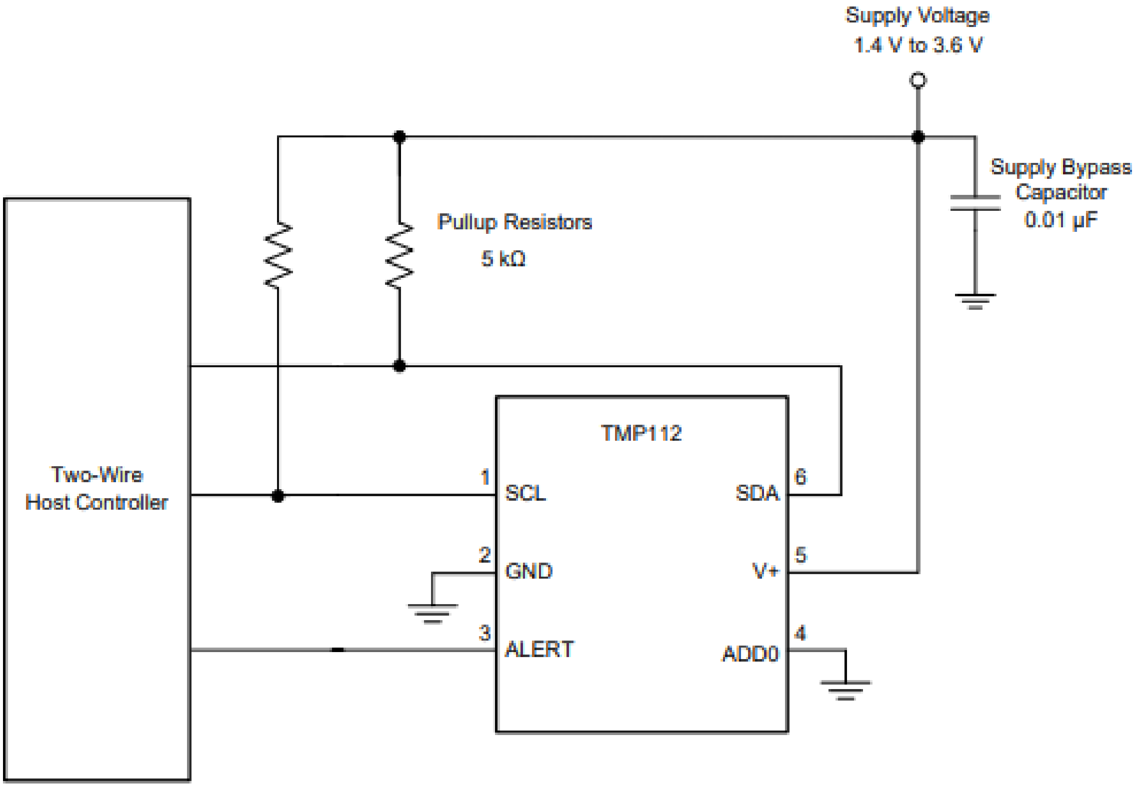


Figure 4.2: Schematic overview of the connection between sensor and microcontroller

.

4.2. Pressure sensor

Unlike the temperature sensor, the pressure sensor implemented in this project does not have a universal interface like I2C. Instead, it uses a 3-wire serial interface to communicate with a controller, which consists of the serial clock wire (SCLK), data input wire (DIN) and the data output wire (DOUT). Besides, the pressure sensor requires an external crystal clock with a frequency of 32.768 kHz. When testing the pressure sensor, a signal generator was used to mimic this crystal clock, as there was no external crystal clock available. Later during this project, it was managed to forward the signal of a crystal clock available in the microcontroller to an output pin. After this, the signal generator was not needed anymore.

Before being able to obtain pressure, the microcontroller needs to request the calibration coefficients, which are predefined by the manufacturer and stored in the memory of the MS5534C module. A total of 6 coefficients is stored in 4 words of 16 bits in the memory. Figure 4.3 shows an example, in which sequences for requesting the first and third word and the resulting data format are shown. The microcontroller can request the coefficients word per word. Appendix A.2.5 shows the microcontroller code for requesting all calibration coefficients, which uses the code in Appendix A.2.4 for requesting each word. After that,

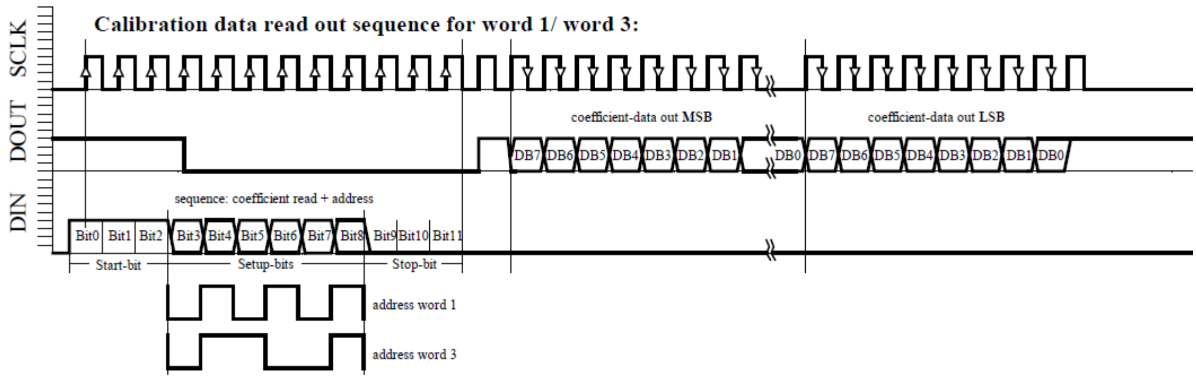


Figure 4.3: The process of requesting words 1 and 3 from the pressure sensor by a controller [11].

the microcontroller can ask the sensor for measuring and sending the temperature (also used for correcting the measured pressure) and pressure, both in words of 16 bits. This is shown in figure 4.4. The figure shows that it takes 33 ms for the pressure sensor to send the pressure data after the request sequence is sent by the controller. The code for this is shown in Appendix A.2.9, in which the average gets taken of n samples of pressure measurements in order to increase accuracy.

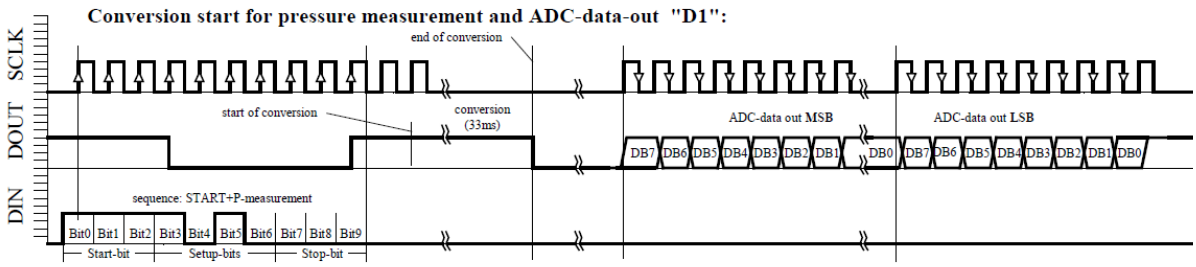


Figure 4.4: The process of requesting pressure from the sensor by a controller [11].

4.3. Sensors combined

Now the two sensors have been implemented separately, they should be combined. First the microcontroller needs to request the calibration coefficients as stated in chapter 4.2. Then the functioning of the temperature sensor will be started, this is done by running the Initialization Code. Every time when an alert is generated by the Sensor Controller, the program in Code Composer will go to the callback function of the temperature sensor. In this code the value read by Sensor Controller can be retrieved. Since we want to measure pressure after the temperature, the pressure will be determined when the program arrives in callback function of the temperature sensor. So then both parameters are obtained sequentially.

5 Testing & Results

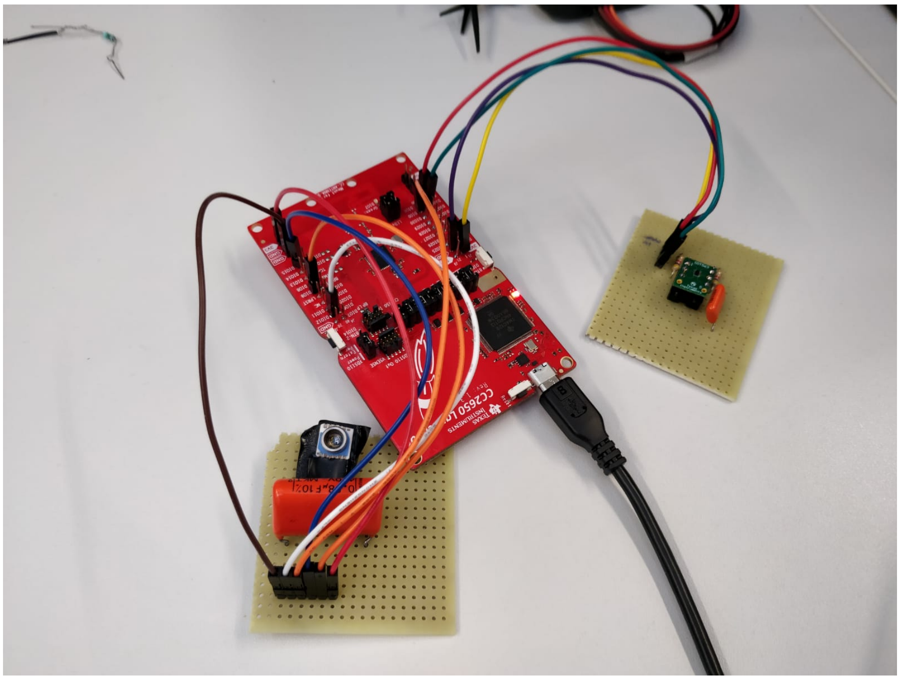


Figure 5.1: Test setup with a TI Launchpad CC2650 microcontroller connected to the temperature sensor and the pressure sensor.

In order to confirm that the two implemented sensors works correctly, both seperately and combined, various tests have been done on the system. The following two sections describe the test methods and results for tests with the two sensors seperately. Section 5.2 covers the methods and results for testing the two sensors combined. Figure 5.1 shows the test setup in which the two sensors are connected to a microcontroller. The operating voltage of the sensors in this setup are 3.3V.

5.1. Temperature sensor

For the implementation of the temperature sensor, it was first tested in the Sensor Controller Studio program. This program offers a possibility to plot the value of the temperature register. The result after implementing the sensor circuit and code in Sensor Controller Studio is show in figure 5.2. It can be seen from the figure that the temperature in the beginning is around 25 °C. But then it changes, this is caused by a finger which is placed on the temperature sensor. As it can be seen it responds to this external temperature change. This is considered as good functioning of the temperature sensor

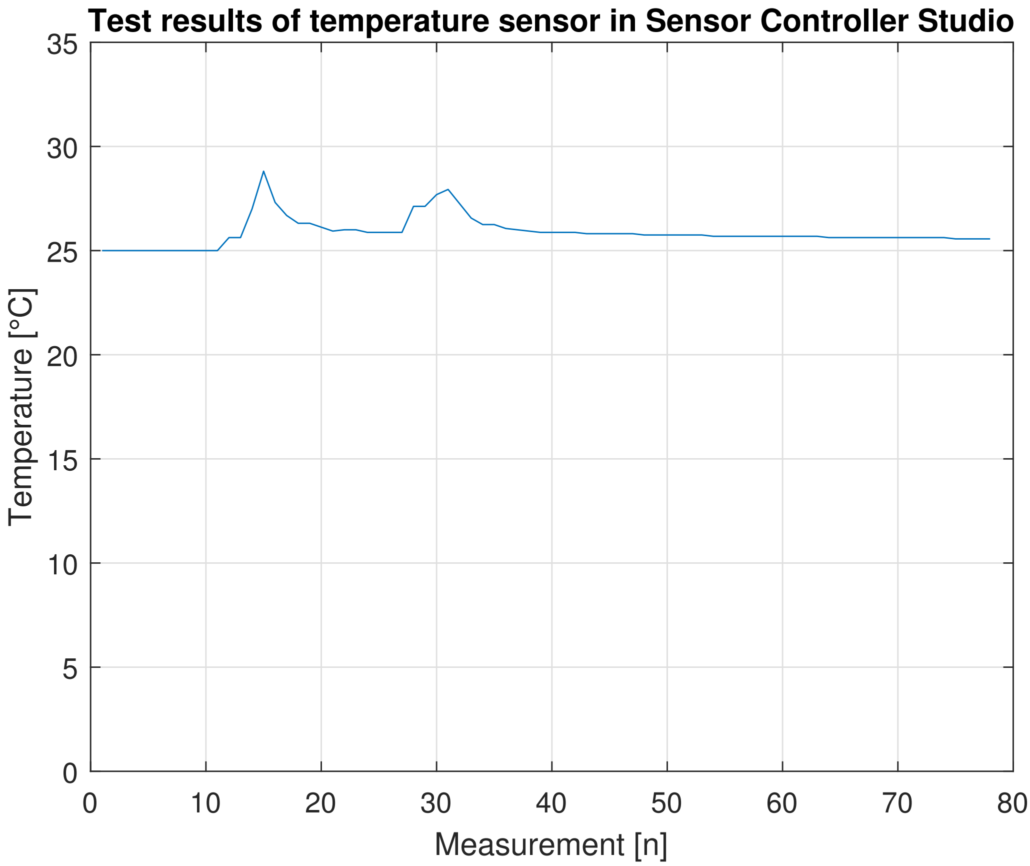


Figure 5.2: Result of measurement done by temperature sensor in Sensor Controller Studio

After the temperature sensor is considered working in Sensor Controller Studio, it has also been tested in Code Composer Studio. This is done by printing the digital value of the temperature sensor. The result obtained is shown in figure 5.3. The temperature sensor will print every second a new value. The C in the figure is to separate the value of every separate measurement. The binary value read from the figure is '0000 0001 1001 0101', the decimal value of this binary vector is 405. To convert this to a temperature it should be multiplied by 0.0625°C. This results in a temperature of 25.3125°C, this temperature was checked by using u mercurial thermometer. The temperature can be read accurately by one tenth degree Celsius, and estimated by one hundredth degree Celsius. The temperature was read to be 25.3°C, so because of this it can be considered the temperature sensor is accurate enough, since the mercurial thermometer and the sensor give the same value, when the temperature is taken to one tenth of degree Celsius.

```
0000000110010101C0000000110010101C0000000110010101C0000000110010101C0
0101C0000000110010101C0000000110010101C0000000110010101C0000000110010
110010101C0000000110010101C0000000110010101C0000000110010100C00000001
00000110010101C0000000110010100C
```

Figure 5.3: Result of printing the digital temperature value separated by c every second.

5.2. Pressure sensor

Figures 4.3 and 4.4 from the previous chapter showed the way data should be sent to and received from the pressure sensor. To confirm whether the sensor reacts correctly to command sequences, the data transmission between the sensor and the controller has been examined with a logic analyzer.

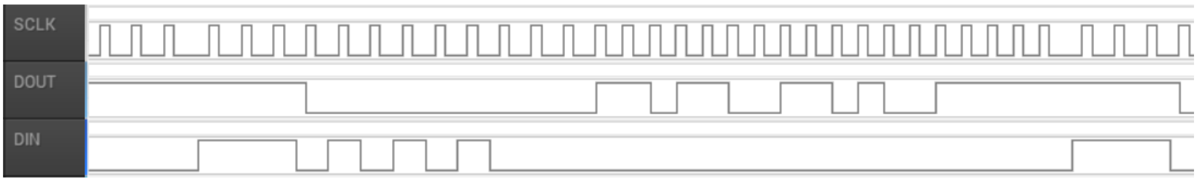


Figure 5.4: Logic analyzer data showing the result of requesting word 1 from the pressure sensor. Pracictal result of figure 4.3. The response at DOUT is the binary number 1011001101001111, which corresponds to the decimal number 45903.

Figure 5.4 shows the result given by the logic analyzer when the first word in the memory, containing calibration data, is requested. The response of the sensor is found in DOUT. The sequence in figure 5.4 corresponds to 45903 in decimal.

Figure 5.5 shows the sequence for starting a pressure measurement being sent at the DIN port of the pressure sensor. It can be seen that each signal, including the serial clock SCLK, becomes zero after the sequence is sent. This is the phase in which the conversion for pressure measurement occurs and is expected to take 33 ms according to figure 4.4.

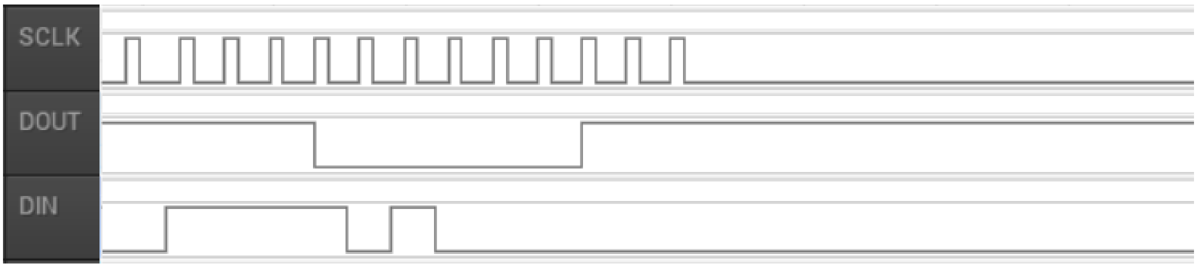


Figure 5.5: Logic analyzer data showing that the sequence for pressure measurement is being sent to the DIN pin of the pressure sensor.

33 ms after the sequence is sent, the pressure data starts returning, as shown in figure 5.6. The returned pressure data is 18263 in decimal. After calculation using the calibration coefficients, the calculated pressure is found to be 1.0185 bar.

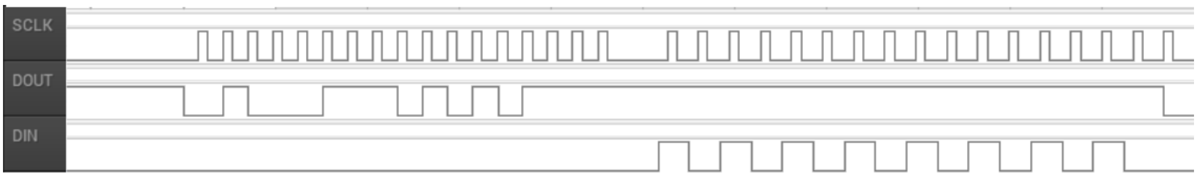


Figure 5.6: Logic analyzer data showing the result of the sequence sent in figure5.5, 33 milliseconds after the pressure data is requested. The response is the binary number 0100011101010111, which is 18263 in decimal.

Testing the pressure sensor has been more difficult than the temperature sensor, because it is more difficult to accurately adjust the pressure of an area than changing the temperature of that area. This makes the callibration of the pressure sensor difficult. Figure 5.7 shows a few samples of pressure data in one-tenth millibars, while the pressure is manually increased and decreased. This shows that the measured pressure changes relative to the changes in the actual pressure. However, unfortunately, the accuracy of the measurement of pressure cannot easily be tested.

```
pres: 10318
pres: 10595
pres: 11189
pres: 12293
pres: 13393
pres: 14483
pres: 15123
pres: 10724
pres: 10180
pres: 10181
pres: 10181
```

Figure 5.7: Pressures measured by the pressure sensor, in one-tenth millibars.

5.3. Sensors combined

After the two sensors are tested separately, they now have to be combined and tested together. The setup shown in Figure 5.1 is connected to a power source and the software for the functionality of the two sensors and transmission of their data is uploaded to the microcontroller. Another setup consisting of only a microcontroller is connected to a computer via USB, which is programmed by the data transmission subgroup to receive data. Using these two setups, the simultaneous working of the temperature sensor and the pressure sensor is tested. This way, the transmission of measurement values also gets tested.

After combining both sensors, the measurements of the sensors are plotted in a custom GUI, shown in figure 5.8. Some external temperature and pressure changes were applied to the sensors, and as it can be seen the sensors react to it.

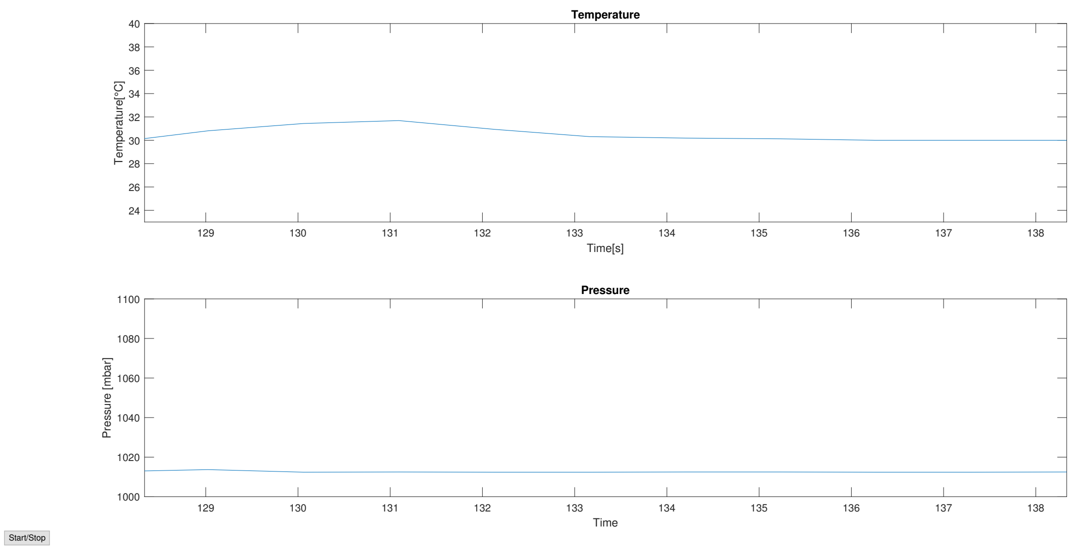


Figure 5.8: A custom GUI showing the measurements of the temperature sensor and pressure sensor over time.

To test whether the amount of times the sensor value read is once per seconde, a logic analyzer is use. The logic analyzer used is the Saleae Logic 8. The pins analysed in the logic analyzer are the data out and in pins of the pressure sensor. These one chosen because it says something about the functioning of the pressure sensor. For the temperature sensor the SDA pin is looked at. In figure 5.9 the results of those pin can be seen, it shows when a pin is active. As it can be seen after every time the temperature sensor is active, the pressure sensor is active. It can also be seen that the pressure sensor is active for multiple times, this is because of the average it does want to measure. It takes the average of four measurements. Another thing which can be seen is that after one second the temperature sensor has been active it will be active again. This is because there will be one measurement per second.

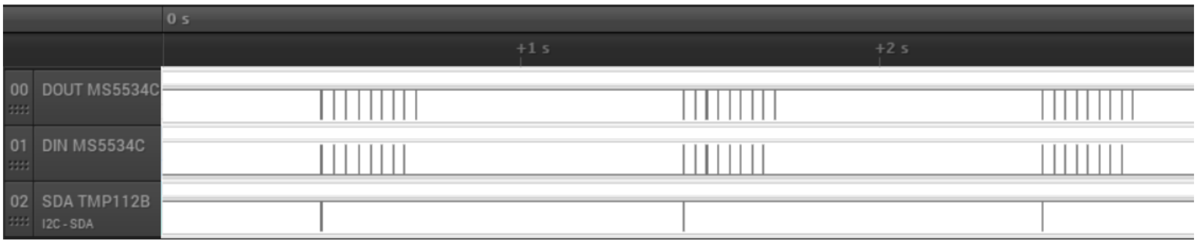


Figure 5.9: Test with logic analyzer of the DIN, DOUT(of pressure sensor) and SDA pin(of temperature sensor).

At last the current trough the microcontroller when connecting those two sensors, was measured. For this measurement the Tektronix PWS4205 powersupply was used to supply 3.3V to the microcontroller. This is also the voltage the microcontroller will get from the power subgroup. A resistor of 10 Ω was put in series with the ground of the whole system to act as a shunt resistor so an oscilloscope, the Tektronix TDS 2022C, could measure the voltage over it which corresponds to the current, this resulted in the graph shown in figure 5.10

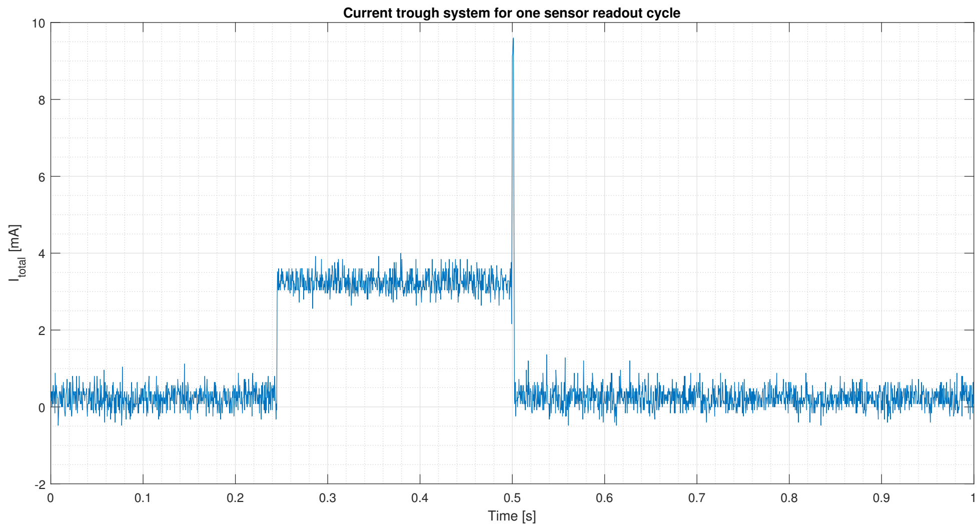


Figure 5.10: Total current trough system when temperature and pressure connected and transmission is active.

In figure 5.10 it can be seen that for a time of 260 ms there is a current of around 3.5 mA. This current is caused by the conversion done in the pressure sensor. It does 8 conversion which takes 8 times 35 ms. The peak in the figure is caused by the transmission of data. So the average current in this system is around 0.91 mA. Provided that all the components work on 3.3V the amount of power used is 3.00 mW per cycle. Given the battery supplies 153mWh, the endoscopic pill will last 51 hours.

5.4. 3D design

The parts shown in figure 3.9 and 3.10 have been 3D printed and the result of that is shown in figure 5.11. The PCB containing all the sensors fits in the capsule and the caps at each end can easily be removed and attached.

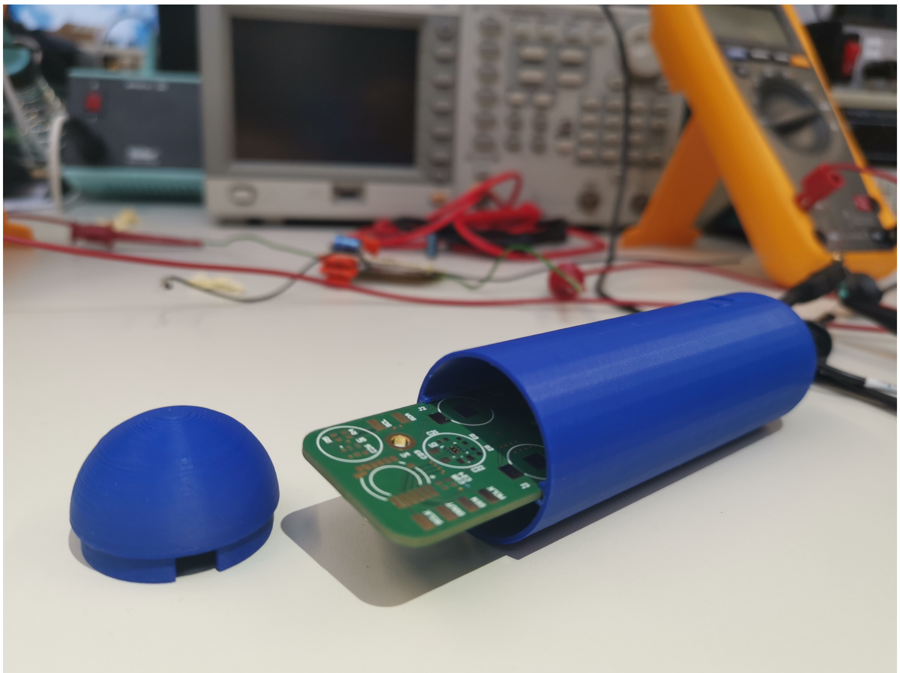


Figure 5.11: The 3D printed result of the capsule design, in which the PCB containing the sensors can fit.

6 Discussion

For the sensors the choice was made to choose a sensor which is prefabricated. These sensors indeed where chosen with the requirements of Chapter 2 kept in mind. The advantage of this is that all specs are already given. Another advantage of choosing a prefabricated sensor, is the implementation of the sensors are already explained in the manual.

A disadvantage of this method might be that the components can not be modified. This might be needed to exclude some parts of the component, because it is not necessary in the project. The size can also not be made smaller, it might be useful to make some components smaller, since the goal is to make the endoscopic pill as small as possible.

The tests done in Chapter 5 were considered to be satisfied the requirements. One downside of the tests is that it was not really tested at the range of body temperature. This is because these tests need a self-designed setup which can simulate these circumstances, but the resources in this project were limited. Therefore it could not be tested at the range of body temperature and pressure. The sensors used were not calibrated before starting working with it. This could be done to ensure a higher accuracy, but the accuracy without calibration was already good enough. This accuracy is given by the data sheet of the components.

Something which came to the attention using the temperature sensor and which can be seen in Figure 5.2, is the temperature decreases slowly after applying an external increase of temperature. This is probably because cooling down of the sensors take some time. However such abrupt change in temperature will not occur, so this is not a problem in the endoscopic pill.

Collaborations with other subgroups was needed in order to test the interactions between components. The correct sending and arrival of temperature values and pressure values required collaboration with the data transmission subgroup. The power efficiency of temperature and pressure measurements depended on collaboration with the power management subgroup.

7 Conclusion

The goal to deliver a prototype with functioning sensors inside a capsule which later can be made smaller, at the size of a pill, is achieved. It is chosen to use a temperature and a pressure sensor. Those two sensors are already small enough to be included in a endoscopic pill. But the pressure sensor might be made smaller, to make more space for other sensors. There is also a 3D design made, which has a diameter of 11 mm and a length of 23 mm. This capsule is shown in figure 3.11, as can be seen in the figure the capsule has a hole in it. This hole is made to expose the pressure sensor. At the moment it is achieved to make the circuit on a PCB of 30 mm by 90 mm, for this PCB there is also a capsule designed this is shown in figure 5.11. As it can be seen from figure 5.9, asking for data in pressure and getting it back takes around 250ms, for the pressure sensor it gets immediately the data back. So the sensors satisfy requirement MAN2. Since those sensor also can operate on 3.3V, it also satisfies requirement MAN4.

The temperature sensor chosen is TMP112B, this temperature sensor is very accurate at the range of body temperature. At bodytemperature range it has an accuracy of around 0.1°C. The temperature range it works on is from -50°C to 150°C. The resolution of the temperature sensor is 0.0625°C. Whith those specs it satisfies all the requirement, defined in Chapter 2.3.2. The protocol for interfacing the temperature sensor with the microcontroller is I^2C.

For the pressure sensor MS5534C is chosen, the pressure sensor implemented in this project does nothave a universal interface like I^2C. The resolution of this sensor is 0.1 millibar. The pressure sensor has an accuracy of almost 0 mbar at body temperature and it also operates on the desired pressure range which is defined in requirement PRS1.

Combining these two sensor it was chose to have a frequency of 1 Hz for determining the temperature and pressure. The readout of these sensor is done sequentially. From figure 5.10, the current during one readout cycle can be seen. The power used in this cycle is around 3.00 mW per cycle. Given the battery supplies 153mWh, the endoscopic pill will last 51 hours.

8 Future recommendations

8.1. Other possible sensors

The sensors mentioned in chapter 1.3 that are not implemented during this project are all feasible sensors which can added during future works. It is easy to add an additional sensor in our project when it already has a predefined interface like I^2C.

Another thing which could be done is stripping down the pressure sensor, since this is one is much larger than the temperature sensor. The working principle of the pressure sensor is know, so it might be an idea to build an own pressure sensor.

For using a image sensor the amount of measurement done should be considered again, since the image sensor might need much power. This can cause the lifetime of the endoscopic pill to be too low.

8.2. Capsule design

The prototype designed during this project is not at a swallowable size, waterproof, or made of a biocompatible material. It also does not have holes for exposing certain sensors, like pressure sensors and image sensors. These are essential requirements in order to perform measurements inside the human body.

Instead of having removable caps, in order to keep the capsule waterproof, the caps should be glued to the cylinder using biocompatible adhesive [9]. For an image sensor to work, the capsule would need to have a transparent part, in order for the sensor to take images of the outside world without making contact with humidity. Pressure sensors would either need to be sticking out of the capsule, or there should be a hole with a waterproof tube inside the capsule leading towards the sensor.

A
Code

A.1. Temperature sensor

A.1.1. Initialization Code

```
//configure tmp112b
i2cStart();
i2cTx(I2C_OP_WRITE | ALS_I2C_ADDR);
i2cTx(ALS_REG_CFG);
i2cTx(ALS_CFG_ONE_SHOT >> 8);
i2cTx(ALS_CFG_ONE_SHOT >> 0);
i2cStop();
// Schedule the first execution
fwScheduleTask(1);
```

A.1.2. Execution Code

```
// Read the result after ~1 milliseconds + a 20% margin
evhSetupTimer1Trigger(0, 1, 2);

//Schedule the next execution
//1 second
fwScheduleTask(100);
```

A.1.3. Event Handler Code

```
// If a measurement was successfully started during the last execution ...
if (state.i2cStatus == 0x0000) {

    // Select the result register
    i2cStart();
    i2cTx(I2C_OP_WRITE | ALS_I2C_ADDR);
    i2cTx(ALS_REG_RESULT);

    // If successful ...
    if (state.i2cStatus == 0x0000) {
        U16 resultRegH;
        U16 resultRegL;

        // Read the result
        i2cRepeatedStart();
```

```
        i2cTx(I2C_OP_READ | ALS_I2C_ADDR);
        i2cRxAck(resultRegH);
        i2cRxAck(resultRegL);
        i2cStop();

        // Convert the result into 12-bit
        U16 value = (resultRegH << 4) | (resultRegL >> 4);
        output.value = value;

        // Notify the application with the result is above the high threshold
        // if (value > 416) { //26 graden
        //     fwGenAlertInterrupt();
        //}


        //Generate alert for main CPU
        fwGenAlertInterrupt();

    } else {
        i2cStop();
    }
}
```

A.2. Pressure sensor

A.2.1. SendCommand

```
// send command MS bit first
void SendCommand(unsigned long cmd, size_t nbits)
{
    while(nbits--)
    {
        if(cmd & (unsigned long)(1 << nbits))
            PIN_setOutputValue(ledPinHandle, IOID_15, 1);
        else
            PIN_setOutputValue(ledPinHandle, IOID_15, 0);

        PIN_setOutputValue(ledPinHandle, IOID_12, 1);
        PIN_setOutputValue(ledPinHandle, IOID_12, 0);
    }
}
```

A.2.2. ResetSensor

```
/* Reset the sensor */
void ResetSensor()
{
    SendCommand(0x155540, 21); // 1010101010101010 + 00000
}
```

A.2.3. ReadWord

```
/* Read one word from the sensor */
unsigned int ReadWord(void)
{
```

```
    unsigned int w;
    unsigned int clk = 16;
    w = 0;
    while(clk--)
    {
        PIN_setOutputValue(ledPinHandle, IOID_12, 1);
        PIN_setOutputValue(ledPinHandle, IOID_12, 0);
        w |= (PIN_getInputValue(IOID_0) << clk);
    }
    PIN_setOutputValue(ledPinHandle, IOID_12, 1);
    PIN_setOutputValue(ledPinHandle, IOID_12, 0);

    return w;
}
```

A.2.4. ReadCoefficient

```
/* Read the coefficient from the sensor */
size_t ReadCoefficient(char addr)
{
    long cmd = (long)0x1C00 | (((long)addr) << 4);
    SendCommand(cmd,13);
    return ReadWord();
}
```

A.2.5. ReadCoefficients

```
/* Read the coefficients from the sensor */
void ReadCoefficients(void)
{
    int wb = ReadCoefficient(0x16);
    int wa = ReadCoefficient(0x15);

    coefficients[0] = (int)((wa >> 1) & (int)0x7FFF);
    coefficients[4] = (int)(((wa & 0x1) << 10) | ((wb >> 6) & (int)0x3FF));
    coefficients[5] = (int)(wb & 0x3F);

    wb = ReadCoefficient(0x1A);
    wa = ReadCoefficient(0x19);

    coefficients[3] = (int)((wa >> 6) & 0x3FF);
    coefficients[1] = (int)(((wa & 0x3F) << 6) | (wb & 0x3F));
    coefficients[2] = (int)((wb >> 6) & 0x3FF);
}
```

A.2.6. ConvertPressureTemperature

```
/* Calibrate pressure and temperature values */
long ConvertPressureTemperature()
{
    long UT1  = (coefficients[4] << 3) + 20224;
    long dT   = (long)temperature - UT1;
    long TEMP = 200 + ((dT * (coefficients[5] + 50)) >> 10);
    long OFF  = (coefficients[1] <<2) + (((coefficients[3]-512)*dT) >> 12);
    long SENS = coefficients[0] + ((coefficients[2] * dT) >> 10) + 24576;
```

```
    long X     = ((SENS* ((long)pressure  - 7168)) >> 14) - OFF;
    pressure     = ((X * 10) >> 5) + 2500;
    temperature = TEMP;

    long T2 = 0, P2 = 0;
    if(TEMP < 200)
    {
        T2 = (11 * (coefficients[5]+24) * (200-TEMP) * (200-TEMP)) >> 20;
        P2 = (3 * T2 * (pressure-3500)) >> 14;
        pressure     = pressure - P2;
        temperature = temperature - T2;
    }

    return pressure;
}
```

A.2.7. TriggerTemperatureSample

```
/* Perform one temperature measurement */
void TriggerTemperatureSample(void)
{
    // 111 + 1001 + 000 + 2clks(send 0)
    ResetSensor();
    SendCommand(0xF20, 12);
}
```

A.2.8. TriggerPressureSample

```
/* Perform one pressure measurement */
void TriggerPressureSample(void)
{
    // 111 + 1010 + 000 + 2clks(send 0)
    ResetSensor();
    SendCommand(0xF40, 12);
}
```

A.2.9. AcquireAveragedSample

```
/* Read the average value by reading n samples */
void AcquireAveragedSample(const size_t nSamples)
{
    long pressAccum = 0;
    int n;
    for(n = nSamples; n; n--)
    {
        TriggerTemperatureSample();
        while(PIN_getInputValue(IOID_0))
            ;
        temperature = ReadWord();
        TriggerPressureSample();
        while(PIN_getInputValue(IOID_0))
            ;
        pressure = ReadWord(); // read pressure
        pressAccum += ConvertPressureTemperature();
    }
```

```
    long pressAvg = pressAccum / nSamples;
    pressure = pressAvg;
}
```

A.2.10. calc_pressure

```
/* acquire the pressure value from the sensor */
void calc_pressure(){
    AcquireAveragedSample(PRESSURE_SAMPLES);
}
```

A.3. Temperature & pressure sensor combined

A.3.1. scTaskAlertCallback

```
// SCIF driver callback
// Temperature and pressure sensor are combined
void scTaskAlertCallback(void) {

    scifClearAlertIntSource();
        //reading pressure sensor
    calc_pressure();
        //reading temperature sensor (from Sensor Controller)
    temperature = scifTaskData.tmp112.output.value;
    Semaphore_post(semTX); //run the transmitter task
    scifAckAlertEvents();
}
```

Bibliography

[1] EPCOS AG. Ntc thermistors for temperature measurement, 2013. URL https://www.tdk-electronics.tdk.com/inf/50/db/ntc_13/NTC_Mini_sensors_S861.pdf.

[2] K Aswani, T Maggiano, and J Clements. Led technology advances endoscopy. *Biophotonics International*, 25:20–23, 02 2018.

[3] R. Carta and R. Puers. Wireless power and data transmission for robotic capsule endoscopes. In *2011 18th IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*, pages 1–6, Nov 2011. doi: 10.1109/SCVT.2011.6101305.

[4] DF Evans, G Pye, R Bramley, AG Clark, TJ Dyson, and JD Hardcastle. Measurement of gastrointestinal ph profiles in normal ambulant human subjects. *Gut*, 29(8):1035–1041, 1988.

[5] Texas Instruments. Cc2650, 2016. URL http://www.ti.com/lit/ds/symlink/cc2650.pdf.

[6] Richard J Saad and William L Hasler. A technical review and clinical assessment of the wireless motility capsule. *Gastroenterology & hepatology*, 7:795–804, 12 2011.

[7] E. A. Johannessen, L. Wang, C. Wyse, D. R. S. Cumming, and J. M. Cooper. Biocompatibility of a lab-on-a-pill sensor in artificial gastrointestinal environments. *IEEE Transactions on Biomedical Engineering*, 53(11):2333–2340, Nov 2006. ISSN 0018-9294. doi: 10.1109/TBME.2006.883698.

[8] Erik Johannessen, Tong Boon Tang, Lifang Wang, Lili Cui, Mansour Ahmadian, Nizamettin Aydin, Alexander Astaras, Alan F. Murray, B.W. Flynn, Tughrul Arslan, Steven Beaumont, and David R. S. Cumming. An ingestible electronic pill for real time analytical measurements of the gastro-intestinal tract. 01 2002. doi: 10.1007/978-94-010-0295-0_60.

[9] Ji Hye Kim and Ihn Han. Biocompatible hybrid allyl 2-cyanoacrylate and hydroxyapatite mixed for bio-glue. *Journal of adhesion science and Technology*, 31(12):1328–1337, 2017.

[10] Youngin Kim, Geunho Lee, Sukho Park, Byungkyu Kim, Jong-Oh Park, and Jin-ho Cho. Pressure monitoring system in gastro-intestinal tract. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 1321–1326. IEEE, 2005.

[11] *Barometic Module*. Measurement Specialities, 6 2011.

[12] Medtronic. Pillcam. URL https://www.medtronic.com/.

[13] Miguel Muñoz-Navas. Capsule endoscopy. *World Journal of Gastroenterology: WJG*, 15 (13):1584, 2009.

[14] Samuel N. Adler. The history of time for capsule endoscopy. *Annals of Translational Medicine*, 5:194–194, 05 2017. doi: 10.21037/atm.2017.03.90.

[15] Olympus Optical. Endocapsule. URL https://www.olympus-europa.com/.

[16] D. Panescu. An imaging pill for gastrointestinal endoscopy. *IEEE Engineering in Medicine and Biology Magazine*, 24(4):12–14, July 2005. ISSN 0739-5175. doi: 10.1109/MEMB.2005.1463383.

[17] Paulus Petrus Leonardus Regtien. *Electronic instrumentation*, volume 2. VSSD, 2005.

[18] Robert Repas. Cmos image sensors. *Machine Design*, 79(23):61, Dec 13 2007. URL https://search-proquest-com.tudelft.idm.oclc.org/docview/217194756?accountid=27026. Copyright - Copyright Penton Media, Inc. Dec 13, 2007; Document feature - Diagrams; Last updated - 2017-10-31; CODEN - MADEAP; SubjectsTermNotLitGenreText - United States–US.

[19] M. Scudellari. Synthetic Bacteria Drive New Ingestible Gut Sensor, 2018. URL https://www.webmd.com/digestive-disorders/bleeding-digestive-tract#1.

[20] R. Singh, Low Lee Ngo, Ho Soon Seng, and F. N. C. Mok. A silicon piezoresistive pressure sensor. In *Proceedings First IEEE International Workshop on Electronic Design, Test and Applications '2002*, pages 181–184, Jan 2002. doi: 10.1109/DELTA.2002.994611.

[21] John S Steinhart and Stanley R Hart. Calibration curves for thermistors. In *Deep Sea Research and Oceanographic Abstracts*, volume 15, pages 497–503. Elsevier, 1968.

[22] Märtha Sund-Levander, Christina Forsberg, and Lis Karin Wahren. Normal oral, rectal, tympanic and axillary body temperature in adult men and women: a systematic literature review. *Scandinavian Journal of Caring Sciences*, 16(2):122–128. doi: 10.1046/j.1471-6712.2002.00069.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1046/j.1471-6712.2002.00069.x.

[23] The National Institute of Diabetes and Digestive and Kidney Diseases Health Information Center. Symptoms & Causes of GI Bleeding. URL https://www.niddk.nih.gov/health-information/digestive-diseases/gastrointestinal-bleeding/symptoms-causes. (2016, July).

[24] Amy Wang, Subhas Banerjee, Bradley A Barth, Yasser M Bhat, Shailendra Chauhan, Klaus Gottlieb, Vani Konda, John T Maple, Faris Murad, Patrick R Pfau, D Pleskow, Uzma D Siddiqui, Jeffrey Tokar, and Sarah A Rodriguez. Wireless capsule endoscopy. *Gastrointestinal endoscopy*, 78, 10 2013. doi: 10.1016/j.gie.2013.06.026.

[25] Zhihua Wang, Xiang Xie, Xinkai Chen, and Xiaowen Li. *Design Considerations of Low-Power Digital Integrated Systems for Implantable Medical Applications*, pages 119 – 162. 05 2011. ISBN 9781118016497. doi: 10.1002/9781118016497.ch5.

[26] Martin Wäny. Miniature size camera modules for medical endoscopy. image sensors london conference 2016. 03 2016.

[27] Xiang Xie, Guolin Li, XiaoWen Li, Chun Zhang, ZhiHuaWang , XinKaiChen , and HsiaoWeiSi . A new high quality image compression method for digital image sensors with bayer color filter arrays. pages S3/3 – 13, 01 2005. ISBN 0-7803-8665-5. doi: 10.1109/BIOCAS.2004.1454125.

[28] Marcia Yu. M2a capsule endoscopy. a breakthrough diagnostic tool for small intestine imaging. *Gastroenterology nursing : the official journal of the Society of Gastroenterology Nurses and Associates*, 25:24–7, 01 2002.

[29] You Zhi, Chen Shu-zhen, and Huang Ji-wu. Serial peripheral interface communication between mcu em78p447a and rf reader ic mf rc530. *Wuhan University Journal of Natural Sciences*, 10(3):550–554, May 2005. ISSN 1993-4998. doi: 10.1007/BF02831143. URL https://doi.org/10.1007/BF02831143.