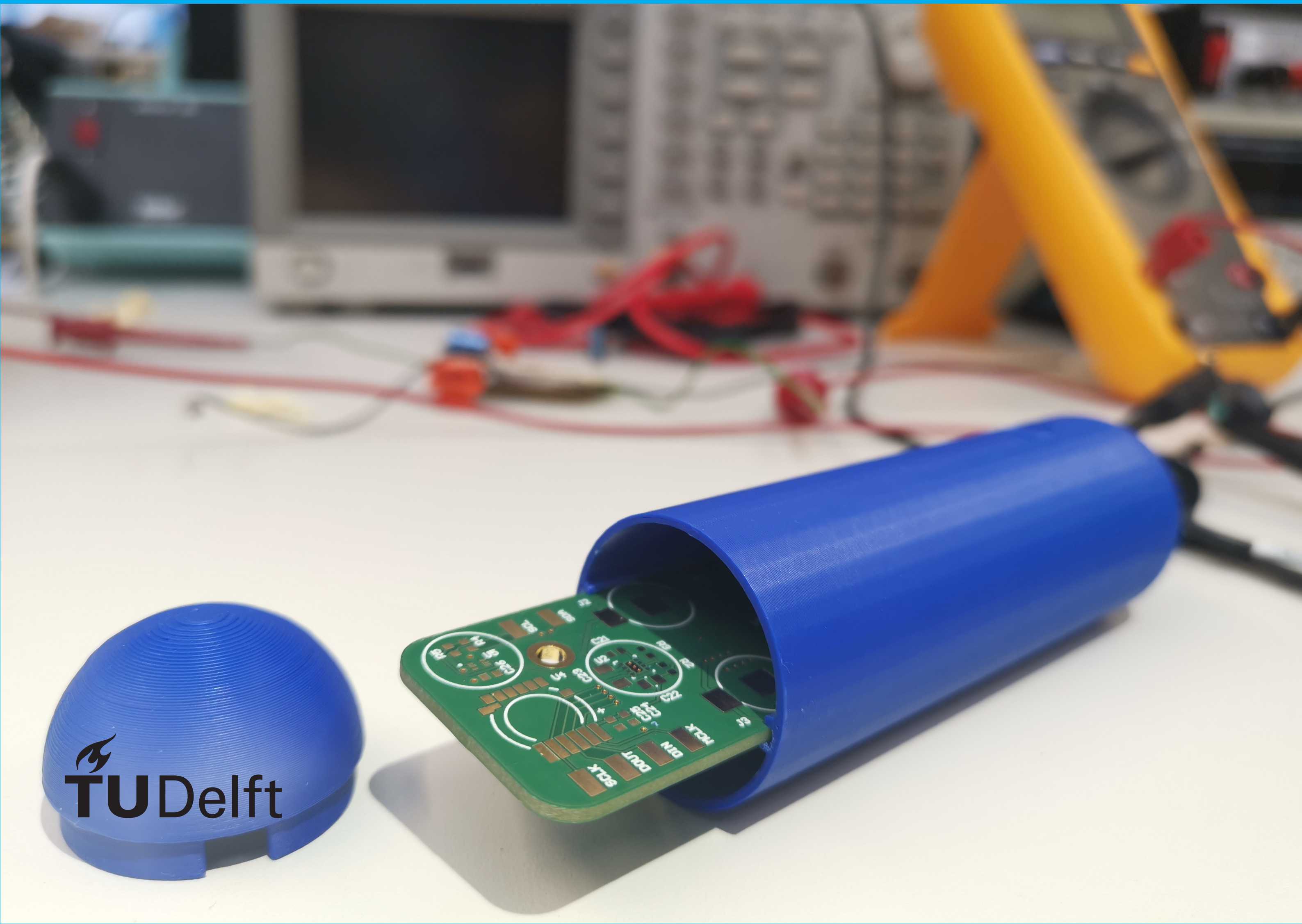
Power management

Endoscopic Pill

A. Kalloe
R. Younis

Endoscopic pill project



Power management

Endoscopic Pill

by

A. Kalloe
R. Younis

to obtain the degree of Bachelor of Science
at the Delft University of Technology,
to be defended on Wednesday July 3, 2019 at 9:30 AM.

Student number: 4473116	4574427	
Project duration:	April 23, 2019 – June 21, 2019	
Thesis committee:	Dr. S. Cotofana,	TU Delft, Chair
	Dr. V. Valente,	TU Delft,Supervisor
	Dr. F. Uysal ,	TU Delft

Preface

This thesis is a part of the Bachelor Graduation Project of Electrical Engineering at Delft University of Technology. This thesis was done under the supervision of Dr. Virgilio Valente. We would like to express our gratitude for his guidance, insight and putting us on the right track. We would also like to thank the Tellegen Hall team for their guidance, patience and advice.

A. Kalloe
R. Younis
Delft, January 2019

Abstract

Endoscopy is a medical procedure in which the inside of the body is looked into. Specifically, the gastrointestinal tract is of interest. This part of the body can be looked into with a swallowable wireless endoscopic pill. The pill contains different components such as different kinds of sensors, a microcontroller, a transmitter and the power supply.

This thesis will discuss the power management of such a pill. The used parts are two CR1025 batteries at a voltage range of 4-6 V and capacity of 30 mAh, a TPS82150 switching converter at an efficiency of 85.8 % with an output voltage of 3.3 V, a TMP112B temperature sensor, a MS5534C pressure sensor and a CC2650 micro controller that consists of a main part (cpu), sensor controller part and a transmitter. The power used by the system was measured to be 2.74 mW per cycle. This resulted in a 57.69 hours runtime which leaves room for other sensors to be implemented and ensures a diagnosis of the whole gastrointestinal tract. All components have been put on a PCB and put into a 3D printed capsule of 123 mm in length and 33 mm in diameter.

Contents

Abstract **v**

1 Introduction **1**
  1.1 Endoscopic pills 1
  1.2 State-of-the-art 1
  1.3 Our project 2
    1.3.1 Overall view 2
    1.3.2 Power management 2

2 Program of Requirements **3**
  2.1 Mandatory requirements 3
  2.2 Trade-off requirements 3

3 Design of the power management unit **5**
  3.1 Batteries 5
  3.2 Converters 6
    3.2.1 Linear converter vs. Switching Converter 7
    3.2.2 Derivation of converter components 8
  3.3 Power management 10
    3.3.1 Current consumption of the system 10
    3.3.2 Temperature threshold 11
    3.3.3 Expected run time 11

4 System Integration **13**
  4.1 Breadboard prototype 13
    4.1.1 Measurement setup 13
    4.1.2 Results 13
  4.2 PCB prototype 16

5 Discussion **19**

6 Conclusion **21**
  6.1 Future recommendation 21
    6.1.1 Power supply 21
    6.1.2 Other sensors 22
    6.1.3 Prototype to manufacturing 22

A Code **23**
  A.1 Transmitter 23
  A.2 Threshold temperature sensor 28

1

Introduction

As the population ages, more advanced health care will be needed and more research will be done for technological advance. This is one of the reasons for the rise of electronic pills, more specific endoscopic pills.

1.1. Endoscopic pills

Endoscopy is a way of looking into the body in the medical world. This is currently done through an endoscope which is a tube with a camera attached to it. This tube is pushed through either the mouth (esophagogastroduodenoscopy) or rectum (colonoscopy) depending on the desired area of diagnosis. This procedure requires a tube to be pushed in the patients which is not desired by the patients and this procedure cannot investigate the small intestine as it cannot be reached [1]. For this reason some diseases cannot be diagnosed such as Crohn's disease, internal bleeding and tumors in the small intestine [2] [3]. Another way of endoscopy, the endoscopic pill, is needed.

An endoscopic pill is a miniaturized electronic pill which can reach areas such as the small intestine. An endoscopic pill can consist of different modules such as a camera, a temperature sensor, a pressure sensor, a microcontroller, a transmitter and power system all in a capsule the size of a large vitamin pill. It has sensors to sense different parameters in the digestive system and deliver real time video images wirelessly to an external console [4]. A few of the challenges in creating the endoscopic pill are making sure the power is well managed so that the pill will reach the desired parts of the GI tract without running out of energy, the wireless communication with a external console and miniaturizing all used parts.

1.2. State-of-the-art

Currently, most endoscopic pills only consist of a single sensor, often a camera, and will work for a limited amount of time as most run on batteries. In Table 3.4 several existing endoscopic pills are shown with the last 2 from research projects.

Table 1.1: Comparison of commercial and research endoscopic pills [4]

	Sensors	**Physical Dimensions**	**Power Source**	**Operation Time**
PillCam	Camera	11 x 26mm	Battery	8h
EndoCapsule	Camera	11 x 26mm	Battery	8h
Sayaka	Camera	9x 23mm	Wireless power	8h
MiroCam	Camera	11 x 24mm	Battery	12h
OMOM capsule	Camera	13 x 28mm	Battery	8h
SmartPill	Ph,Pressure, Temprature	13 x 26mm	Battery	-
Johannessen [5]	Ph, Temprature	12 x 36mm	Battery	42h
Wang [6]	Camera	10 x 190mm	Wireless	-

The pillcam is one of the commercial endoscopic pills. It consists of a camera (CMOS image sensor) and flashing LEDs to ensure clear images, transmitter (RF antenna) and batteries. The patient is required to wear an external recorder to receive the transmitted data. The capsule used by pillcam starts to dissolve 30 hours after ingestion. The pill also has a RFID tag to know its location in the GI tract [7].

1.3. Our project

For designing the system we have split the workload into to three subgroups: the sensor group, the power management group and the data transmission & control group.

1.3.1. Overall view

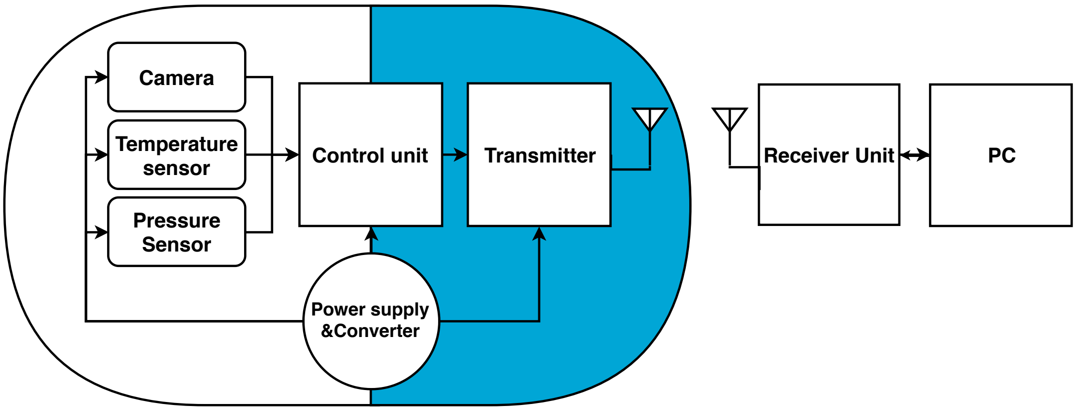


Figure 1.1: Schematic representation of the endoscopic pill

In Figure 1.1 the different parts and how they are connected to each other can be seen. The batteries and converter are in the pill as energy storage and to supply all different components of the right amount of power and voltage. The different sensors will measure the data and this will be sent through to the microcontroller (control unit). In the microcontroller this data is processed and then transmitted through the transmitter. The micro controller (control unit) consists of a sensor controller, CPU and transmitters The internal sensor controller is first used to read the sensor values and stores those in RAM while the other parts of the microcontroller are turned off. Next, the sensor controller turns off and the main microcontroller (CPU) turns on, reads the sensor data from RAM and sends it to the transmitter after any processing if needed. The transmitter then sends the data to an external unit that visualizes the results on a screen. Also part of the control unit is the power management unit which will ensure the needed run time.

1.3.2. Power management

As for any electrical system a power source of some kind is needed. For the endoscopic pill this is a battery or some type of wireless powering. Most commercial electronic pills use batteries [4], but due to their relatively large size wireless powering is being looked into [8]. Wireless power is implemented through inductive powering, far field directive power beaming and non-directive low-power far-field harvesting [9]. For inductive powering of the pill, the patient swallows the pill (receiving module) and lays down with the transmitting unit (coil) around the patient. The efficiency of this system is dependent on where the pill is in the body (distance between the two units) [6]. As for far-field powering of the pill this is done with a transmitter device that sends out a electromagnetic field which is then received on the pill (antenna) and stored [9]. For this project batteries have been chosen as wireless powering would be out of reach for this project. A converter will be used to ensure a constant voltage with a low enough voltage ripple that is does not effect the other components (sensors and micro controller). To ensure a longer battery lifetime and getting the needed run time managing of the different components is needed by implementing thresholds and deciding when components would be on or not. This is what the power management group is about.

2

Program of Requirements

The endoscopic pill will consists of the different components as mentioned in Chapter 1 these will have to fit inside a pill. As for all projects there are several requirements or goals for the system. This section covers the requirements concerning the pill and power management, 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 are not fulfilled, the product will be incomplete. The mandatory requirements concerning the prototype of the endoscopic pill are as follows:

1. A prototype with all parts integrated on a single PCB in a 3D printed capsule.
2. The total volume of all implemented sensors must be smaller than the volume of the 3D printed capsule.
3. A supply that can deliver enough peak and average power so the sensors and micro controller don't fail during operation.
4. The output voltage of the converter is compatible with multiple sensors.
5. All possible sensors must measure with such frequencies that the total run time is at least 8 hours (in order to at least capture the small intestine).
6. The product must contain a temperature sensor.
7. Microcontroller (control unit) that can command when to read sensors or send data.
8. The fabrication and construction of the prototype shall not cost more than 250 euro as set by the Tellegen hall crew.
9. The project has to be finished in 9 weeks.

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:

1. The endoscopic pill should have the shape of a capsule with an outside diameter of 11 millimeters and length of 23 millimeters, to traverse through the GI tract.
2. >24h run time through power management so it can measure during the whole time while it is in the GI tract [10].

3. The material of the capsule should be biocompatible and not harmful to living tissue.
4. The product should have an image sensor to visualize the intestines.
5. The pill should have multiple sensors such as a camera, pH sensor, blood sensor or pressure sensor.

3

Design of the power management unit

The endoscopic pill needs a power supply and as mentioned in Chapter 1 this can be done by both wireless powering and batteries. But as wireless powering is out of reach for this project, batteries are chosen. But as the batteries are discharged, the voltage that is supplied will go down. This is why a converter will be used to supply a constant voltage for the different components. In this chapter the design choices for these parts of the power system as well as a threshold implementation for data transmission will be discussed. An overview of the different parts of the system can be seen in Figure 3.1

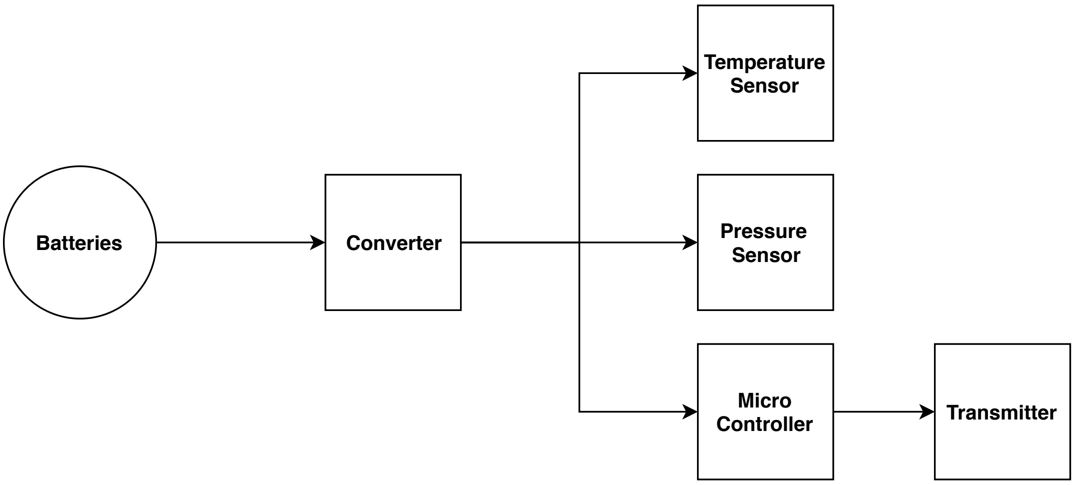


Figure 3.1: Power management unit block diagram

3.1. Batteries

Most of the commercial endoscopic pills are battery powered [4]. The battery powered endoscopic pills use coin cell batteries. This is for their size and capacity. These coin cell batteries are primary batteries which means they are for one use only. Secondary batteries are rechargeable batteries and can be used multiple times. Coin cell batteries are typically used in watches and medical devices such as pacemakers. Lithium and silver-oxide coin cell batteries are the most used chemistry in battery powered endoscopic pills [4]. This is due to their high energy density and light weight. Lithium has have some hazards such as it being highly flammable and potentially explosive [11]. As the Lithium manganese dioxide battery has a higher voltage, greater energy density and are of the safe, leakproof and non-corrosive type [12], it was chosen over silver-oxide batteries, see table 3.1.

As mentioned in chapter 2 the pill will have a diameter of 11 mm therefore the CR1025 coin batteries have been chosen because of the 10 mm diameter of the battery. These lithium coin batteries have a nominal voltage of 3 V and are discharged at 2 V as seen in Figure 3.2.

Table 3.1: Lithium vs. silver oxide coin battery

	Lithium(CR1025)	**Silver-oxide(SR1130SW)**
Voltage[V]	3.0	1.55
Dimensions[mm]	10x2.5	11.6x3.1
Capacity[mAh]	30	60
Cutoff voltage [12] [V]	2.0	1.2
Energy density [12] [mWh/cm^3]	610	500

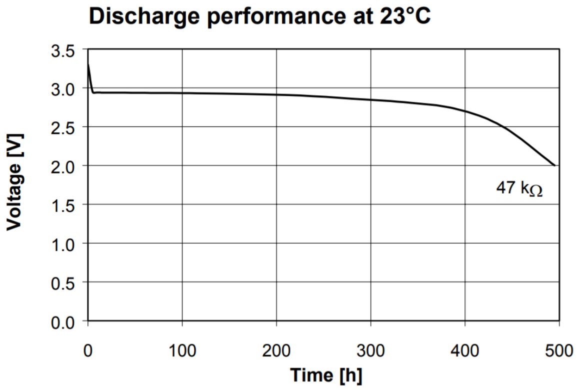


Figure 3.2: Discharge rate CR1025 battery [13]

Given that several sensors, a microcontroller and an antenna should be used, it was decided to put two of these batteries in series to ensure a 24 hour run time.

Table 3.2: Resulting specifications of power supply

	2 batteries in series
Voltage [V]	4-6
Capacity [mAh]	30
Dimensions [mm]	10 x 5

With a voltage of 6 V and capacity of 30 mAh, 180 mWh of energy and for a 24 hour run time the average power will be 7.5 mW. The needed energy and power will be discussed in Section 3.3.

3.2. Converters

For the battery, the output voltage and battery capacity have a specific range as mentioned in Section 3.1. As the battery discharges the voltage drops. This asks for converters as the several components in the endoscopic pill (see Chapter 1) all need a certain voltage to work. This is possible to achieve by either linear converters or switching converters [14] [15].

Table 3.3: Supply ranges of the different components

	Supply voltage [V]
Temperature sensor(TMP112B)	1.4-3.6
Pressure sensor(MS5534C)	2.2-3.6
Microcontroller(CC2650)	1.8-3.8

In Table 3.3 the voltage supply ranges of the different components can be seen. From these values, it was decided that a converter would be used which has an output voltage of 3.3 V. The battery has a nominal voltage of 3 V and will drop to 2 V when discharged. In Table 3.3 you can see the chosen sensors could work directly from the battery, but since we have the requirement to be able to use different sensors a working voltage of 3.3 V is chosen since this is a typically used voltage level and the chosen sensors work with this as well. This does mean a converter will be needed. Two batteries in series will be used, this way only a buck converter (high to lower voltage) is needed. This can be done through either a switching or linear converter all with their own pros and cons.

3.2.1. Linear converter vs. Switching Converter

In order to choose between whether to use the linear converter or the buck converter these two were compared to each other in size, efficiency, components off chip and voltage ripple.

A linear converter as implied in the name consists of a linear component the resistor which will regulate the output voltage. As can be seen from Equation 3.1 The difference between the input voltage and output voltage is dissipated in the resistor (in heat) which means the higher the difference between these two, the lower the efficiency.

$$P_{Dissipated} = (V_{In} - V_{Out}) \cdot I_{Load} \tag{3.1}$$

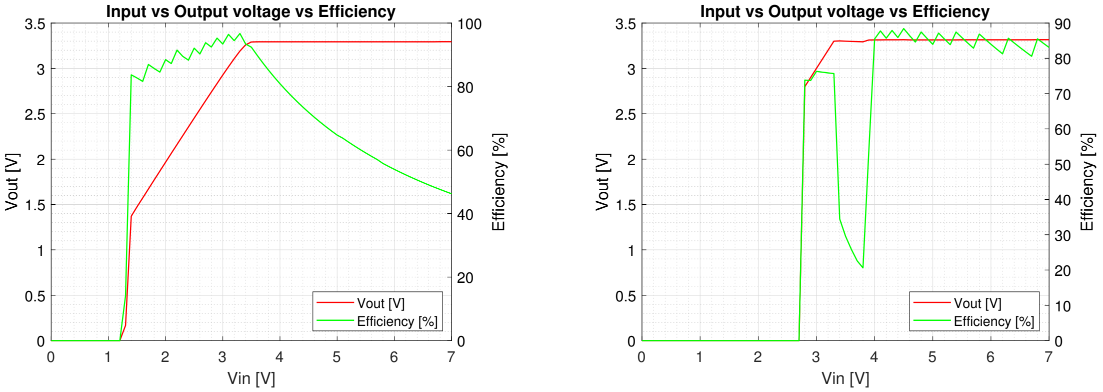
A low power linear converter requires less external components like capacitors or inductors. For our application, size is very important, linear converters being small in overall size will make it possible to fit the converter in the endoscopic pill.

As for the switching converter it consists of switching elements, inductors and capacitors. The voltage has to go from 6 to 3.3V (Buck converter), this is done with a switch that is opened and closed at a specific duty cycle and frequency to get the desired voltage. During the time that the switch is closed the current in the inductor increases and energy is stored in the inductor. When the switch is open the energy that was stored in the inductor is moved to the capacitor resulting in the right voltage at the output. The switching converter has a higher efficiency than linear converters.

The linear converter LT1762EMS8-3.3 and the switching converter TPS82150SILT have been tested and compared, see table 3.4. Linear converters have a lower efficiency when compared to a switching converter but they use less components, so typically less area is used [14].

Table 3.4: Linear vs switching converters

	Linear	**Switching**
Voltage ripple (at 2.25mA)	43mV	148mV
Components off-chip	2	5
Chip area	14.7mm^2	8.4mm^2
Average efficiency (4-6V input)	65.8%	85.8%



(a) Vout vs Efficiency (LT1762EMS8-3.3)

(b) Vout vs Efficiency (TPS82150SILT)

Figure 3.3: Linear vs switching converter at a load of 3.3mA

In figures 3.3 the output voltage and efficiency of both the linear and switching converter can be seen. These were measured with the same load and power supply. The load was chosen given that the average power is 7.5mW (Section 3.1) during a 24 hour runtime. As the battery has a nominal voltage of 3V and are discharged at 2V the voltage range of 4-6V is taken into consideration. In this voltage range the efficiency of these two converters are 65.8% for the linear converter and 85.8% for the switching converter. Given the efficiency in figure 3.3b and data in table 3.4 such as the voltage ripple the switching converter was chosen. Even though the voltage ripple is higher, it would not effect the different components as the voltage would stay in the working range of the components.

3.2.2. Derivation of converter components

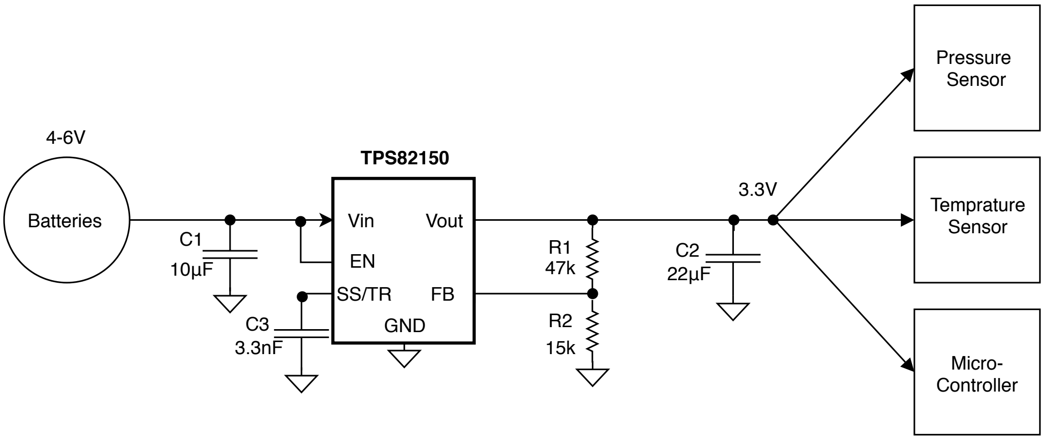


Figure 3.4: Voltage conversion

In Figure 3.4 the connection between the batteries and converter (and its schematics) to the other components can be seen. In the converter schematic R1 and R2 have a voltage division which will decide the output voltage [16] :

$$V_{Out} = V_{FB} \cdot (1 + \frac{R1}{R2}) \tag{3.2}$$

Given from the datasheet that $V_{FB} = 0.8V$, R1 and R2 need to have a ratio of 3.125 to ensure an output voltage of 3.3V. R1 has been chosen to have the value of 47kΩ and R2 15kΩ. Also R2 should not be higher than 100kΩ to achieve high efficiency at light load as stated in [16] These values resulted in a output voltage of 3.3 V as seen in Figure 3.5 and voltage ripple of 148 mV as seen in Table 3.4. The input capacitor C1 minimizes the input voltage ripple and was recommended to be 10 μF. C2 was 22 μF to minimize the voltage ripple.

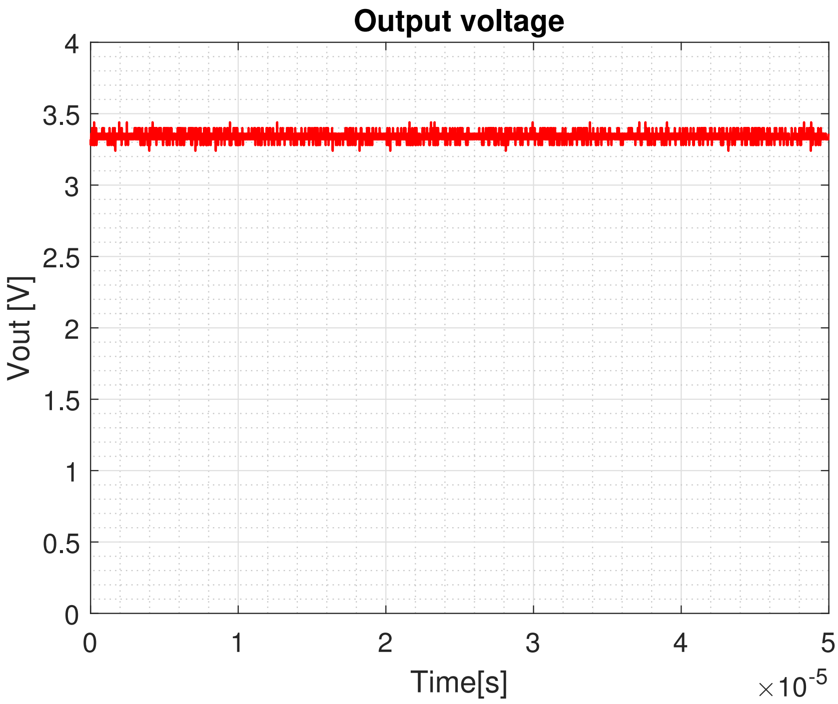


Figure 3.5: Output Voltage buck converter(TPS82150)

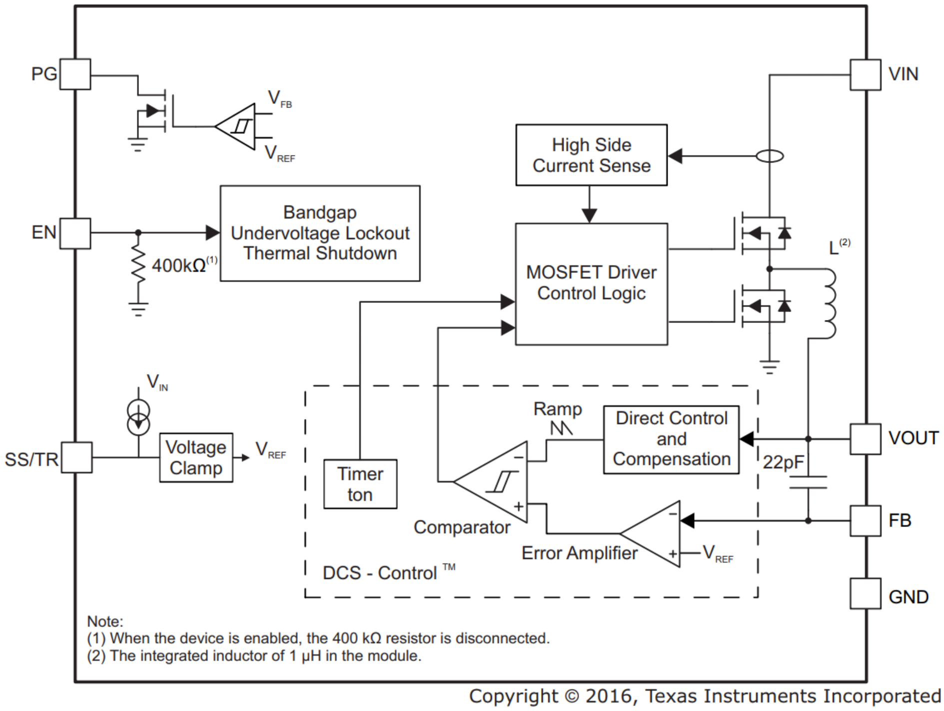


Figure 3.6: TPS82150 block diagram [16]

The TPS82150 is a buck converter and uses a control mode called DCS-Control [17], see Figure 3.6 for a block diagram. It is based on a hysteretic topology. The feedback pin is used to make the output of an error amplifier high if the output voltage is too low. That output is fed into a comparator, the other input of the comparator is a ramp which slope and frequency are decided based on the output voltage. The output will be a PWM signal that will control the MOSFETs.

3.3. Power management

As mentioned in Chapter 2 the pill should measure and send data for at least 24 hours to ensure it works throughout the whole the GI tract. For this to work another important part of the system, power management, has to be implemented.

To process and transmit the sensor data while still using as little energy and area as possible a microcontroller with integrated transmitter and sensor controller is used. To ensure longer lifetime of the pill the different components can be turned off or put in standby mode when they are not necessarily needed. There are several situations where this would be very useful as the temperature sensor does not need to measure the temperature continuously, since it does not rapidly change in the body, a threshold can be implemented to decide when the temperature data should be transmitted [18], the same can be said about the pressure sensor. When turning off components to preserve energy and increase battery life controllable switches are needed, but this introduces latency as the components need to be reinitialized [19]. In our system everything can be controlled in software directly from the microcontroller so that issue is not present.

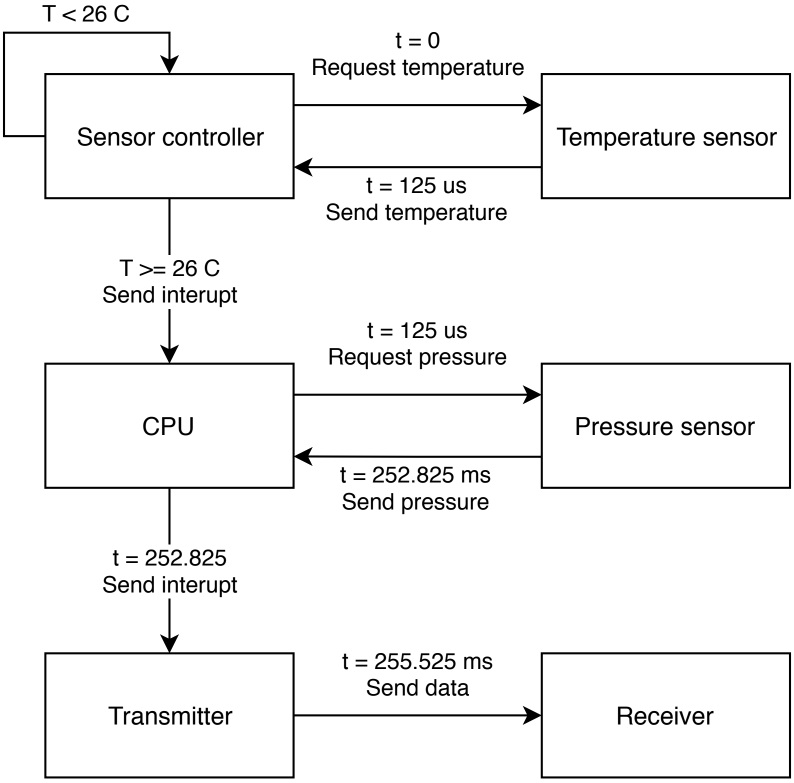


Figure 3.7: Flow chart single cycle

In figure 3.7 a flow chart of one measurement cycle is shown. It shows that the total time that the microcontroller and other parts are busy is 255.525 ms, for the rest of the cycle which takes 1 s all parts will be in a low power mode. In 3.8 the timing of this is shown in combination with the expected currents. The code running on the CPU is available in Appendix A.1.

3.3.1. Current consumption of the system

In this Subsection the current consumption of the different components of the endoscopic pill are discussed.

It was decided that the temperature sensor will measure once every second and the pressure sensor 4 times every second, this is to ensure an high accuracy value for the pressure value. The temperature component consists of a temperature sensor and the sensor controller part that is integrated in the microcontroller. The pressure component uses the pressure sensor and main cpu in the microcontroller to retrieve and convert the data.

Table 3.5: Retrieve and conversion time of the different components during one cycle

	Retrieve and conversion time [ms]
Temperature component	0.125
Pressure component	252.7
Transmitting	2.7

In table 3.5 the retrieve and conversion time for the different components are shown, for the pressure sensor this is value is due to the 4 measurements in its conversion besides that the pressure sensor that was chosen, the MS5534C, uses both a temperature and pressure measurement for an accurate value of the pressure. The transmitter has a set pre-transmission state and afterwards transmits the data bits (4 bytes + header). All the values in table 3.5 were measured by the sensor and data-transmission group.

Table 3.6: Expected current consumption of the different components

	Average current [A]	**Peak current [A]**
Pressure sensor(MS5534C)	$3.64 \cdot 10^{-6}$	$0.91 \cdot 10^{-3}$
Temperature sensor(TMP112B)	$1.25 \cdot 10^{-6}$	$10 \cdot 10^{-6}$
Sensorcontroller(CC2650)	$19.68 \cdot 10^{-6}$	$393.6 \cdot 10^{-6}$
Microcontroller(CC2650)	$0.7325 \cdot 10^{-3}$	$2.93 \cdot 10^{-3}$
Transmitter(CC2650)	$16.26 \cdot 10^{-6}$	$9.1 \cdot 10^{-3}$
Total	$\mathbf{0.77 \cdot 10^{-3}}$	$\mathbf{13.34 \cdot 10^{-3}}$

In table 3.6 the expected current consumption of the different components can be found. The averages were calculated using the conversion times from table 3.5 and peak currents taken from the datasheet from the components [20][21][22]. Using the data in tables 3.5 and 3.6 figure 3.8 has been made as an estimation of the current consumption in one cycle. In the figure only the first 300 ms are shown since the current stays low for the other 700 ms until the cycle starts again.

3.3.2. Temperature threshold

The endoscopic pill will be powered by batteries and will continuously measure the sensor outputs this data has to be transmitted but as a 24 hour run time ensures the GI tract is fully inspected [10]. A threshold margin should be implemented that if the sensed temperature has not changed in the given margin. Such that only if the temperature would have a 0.1 change that it would be transmitted. But due to time constraints it was not implemented yet. As an alternative a threshold is implemented so the temperature data is only transmitted when a certain threshold is passed. The body has a temperature around 37°C +/- 0.25 [23] but patients who have hypothermia can have body temperatures of 35°C or lower. The threshold should then be put at 30°C to ensure the pill does not send data when it is outside of the body assuming the patient takes the pill in a room temperature environment. In this endoscopic pill it is implemented that the sensors retrieve and convert their data after each other, this is done by sending a signal from sensor controller to main cpu in the micro controller. As the temperature threshold was implemented, the pressure sensor is not allowed to do its measurement and conversion if the threshold is not reached, resulting in less power used. This threshold was tested and the receiver does not receive any transmitted data as the temperature was under the threshold of 26°C which was put for the test. The code running on the sensor controller is available in Appendix A.2.

3.3.3. Expected run time

Given that the batteries are 3V and put in series, at 30 mAh capacity the amount of energy from these batteries is 180 mWh and at an efficiency of 85 % of the switching converter this results in 153 mWh of energy. Now taking into account the total average current of the different components from table 3.6 and that the system would keep measuring and transmitting this data at the one measurement per cycle. Provided that all the components work on 3.3V the predicted amount of power used is 2.54 mW per cycle. This would result in a 60.24h run time. This gives the possibility to add more or other sensors that would use more power.

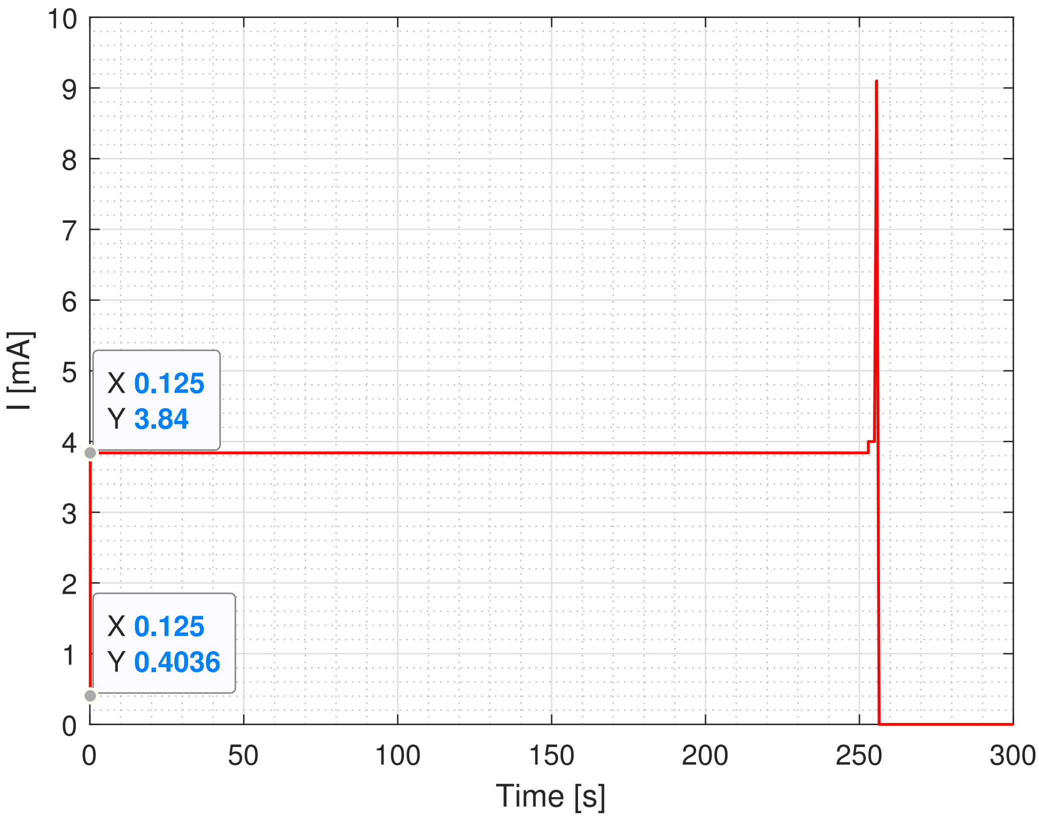


Figure 3.8: Current prediction one cycle

4

System Integration

With all parts separately completed it was time to integrate them into one system. First all parts where integrated on a breadboard prototype after that a PCB was designed so it can be put into a pill shaped enclosure for testing.

4.1. Breadboard prototype

The test setup is build around a development board with the used microcontroller, the CC2650-Launchpad. The used temperature and pressure sensor are small SMD components that were soldered on a breakout board with some supporting components. Multiple test were done, first a test if all parts and software were functional by supplying it with USB via the development board, once everything was working measurements on the power could be done.

4.1.1. Measurement setup

A benchtop powersupply, the Tektronix PWS4205, was used as power source. It was set to 3.3V to simulate the output of the converter. Since the converter has a voltage ripple of 148mV and switching components, which brought noise into the measurement this made the data not useable to see high current peaks. This measurement can be seen in figure 4.2

Multiple fluke 177 multimeters were used to measure the minimum, average and maximum current of the full system, the temperature sensor and the pressure sensor. A resistor of 10Ω 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 was used to get the waveforms in figure 4.3a and figure 4.3b. A picture of the complete test setup can be seen in figure 4.1.

4.1.2. Results

From the measured data in figure 4.3a it can be seen that in one full cycle of 1 second there is 250ms with the current at 3.0-3.5 mA this is where the pressure sensor is read and a 2 ms peak of 9.6 mA at the end that is the transmission, the temperature sensor read out could not be seen as the current value is lost in the noise. Figure 4.3b is a zoomed in measurement at the transmission, here it is visible that at the end of reading the sensors the data is transmitted during 2 ms and directly after that goes back to a low power mode for the rest of the cycle.

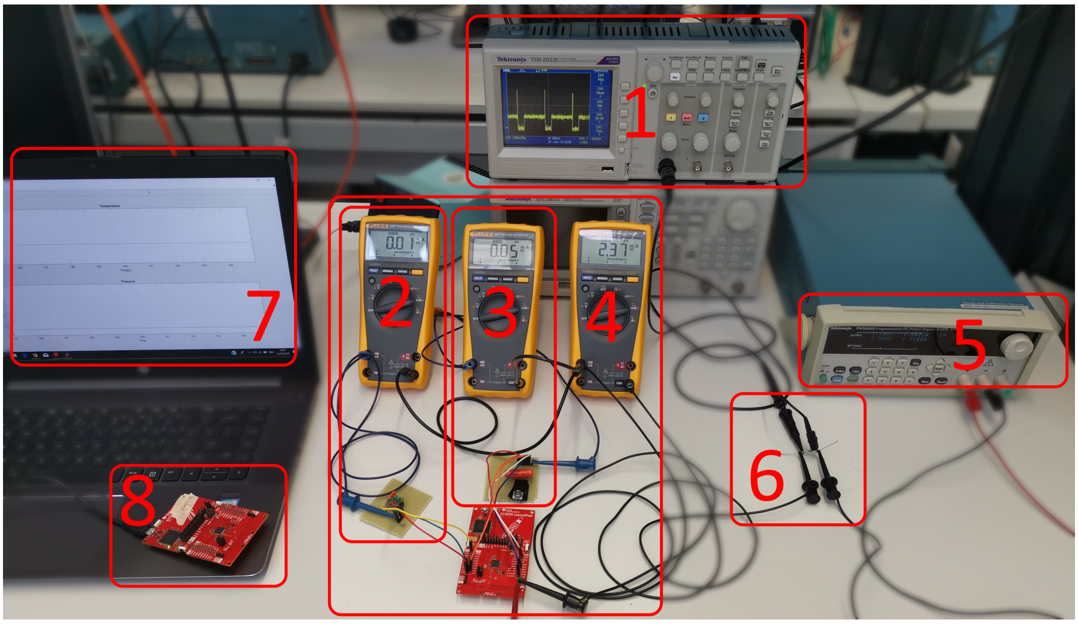


Figure 4.1: 1 Tektronix TDS2022C oscilloscope measuring total current with 6 shunt resistor, 2 Fluke 177 measuring temperature sensor current, 3 Fluke 177 measuring pressure sensor current, 4 Fluke 177 measuring total current, 5 Tektronix PWS4205 power supply, 7 laptop showing measured data which uses 8 receiver via usb

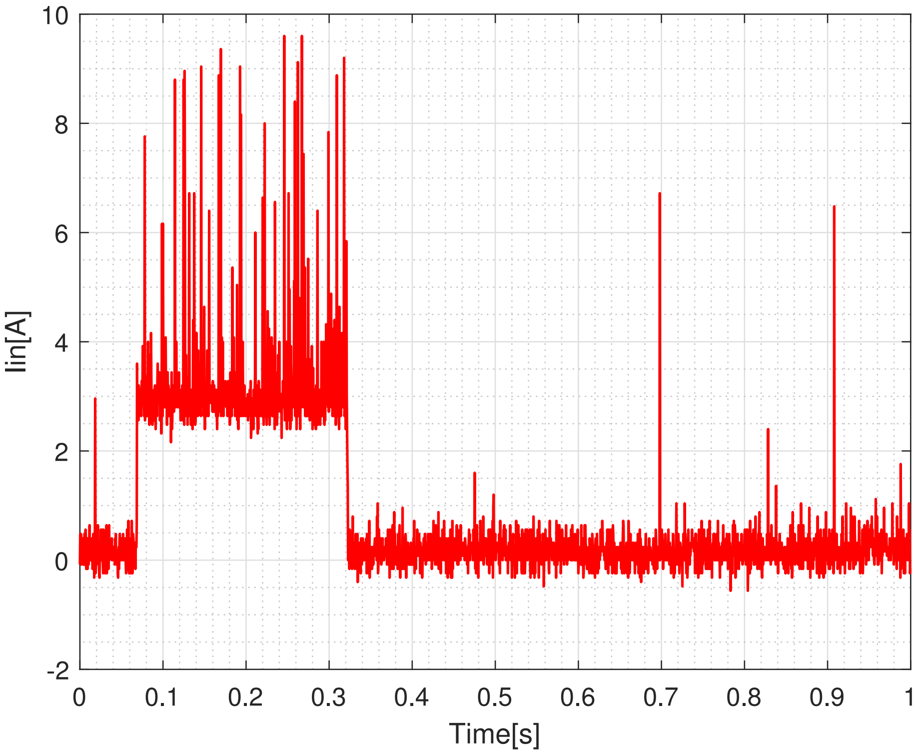
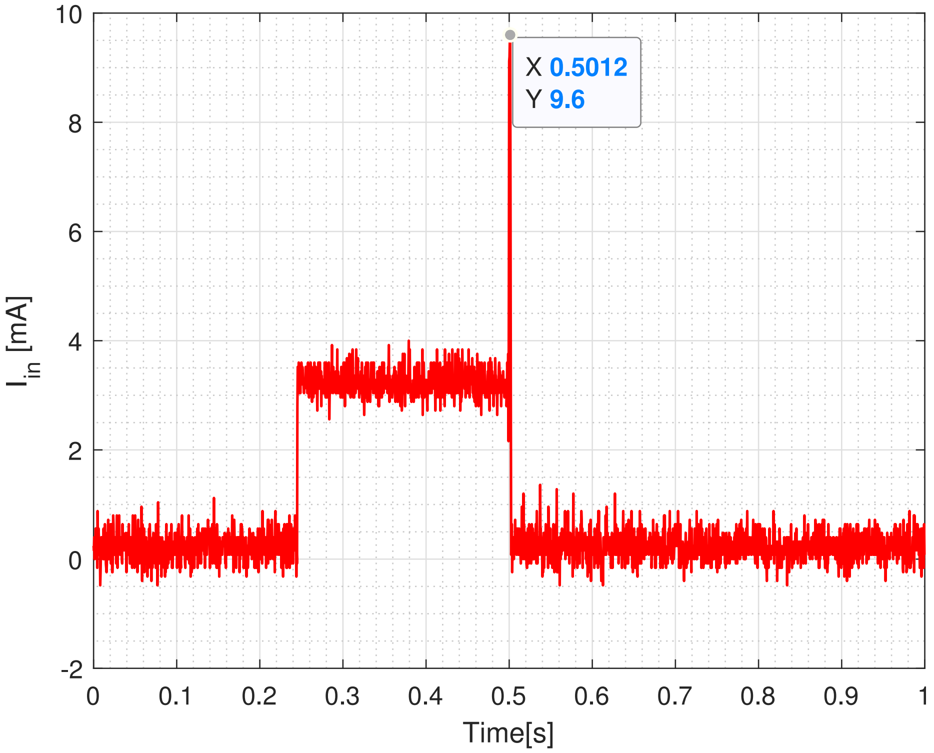
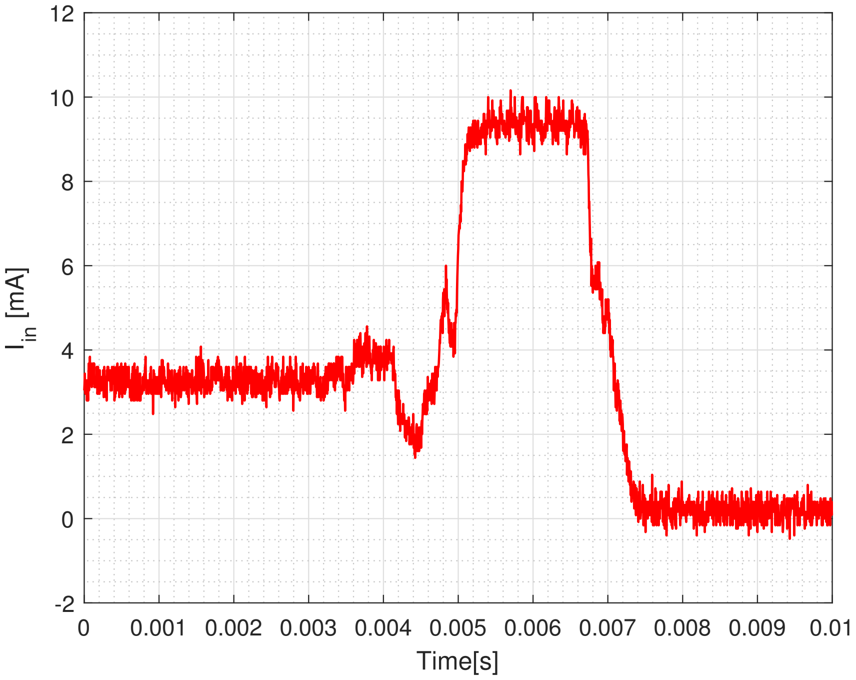


Figure 4.2: Current consumption during one cycle with converter

In table 4.1 the currents from the sensors and the total system are shown. The multimeter has a resolution of 0.01 mA [24] so with average current expected to be lower than that, the measurements will only indicate a proper peak current for the sensors. The peak currents do show that the peak current of the pressure sensor is lower than expected 3.6. This could be because the peak is very short and



(a) Full cycle



(b) Peak current during transmission

Figure 4.3: Current consumption doing one measurement per cycle

the sensor uses less current for the rest of the time. The measurements do show that the average current of full system is only 0.06 mA higher than the expected value from table 3.6, which could be because of non-ideality of the components such as the capacitors, inductors and resistors

Table 4.1: measured current consumption from multimeter

	min	**max**	**avg**
Full system	0.03 mA	3.07 mA	0.83 mA
Temperature	*0.00 mA*	0.02 mA	*0.01 mA*
Pressure	*0.00 mA*	0.08 mA	*0.02 mA*

These results show that the predicted consumption in figure 3.8 is of the same shape and order of magnitude. With the average current that was measured with the multimeter 0.83 mA (taken from Section 4.1) at 3.3 V and energy calculated from the two batteries after the converter 158 mWh (as stated in section 3.3.3) the run time of the endoscopic pill will be 57.69 hours. This is slightly lower than the run time found in 3.3.3 of 60.21 hours.

These measurements were also done for if the sampling frequency of the sensors and transmitting of the data 3 times per cycle. The average current of the full system became 2.38 mA which is three times as high as seen in Table 4.1 as expected as now the data is measured and transmitted three times in one cycle (one second). Figure 4.4 shows the measurement of this cycle. But it was decided by the sensor group that one measurement per cycle would give enough information about the GI tract.

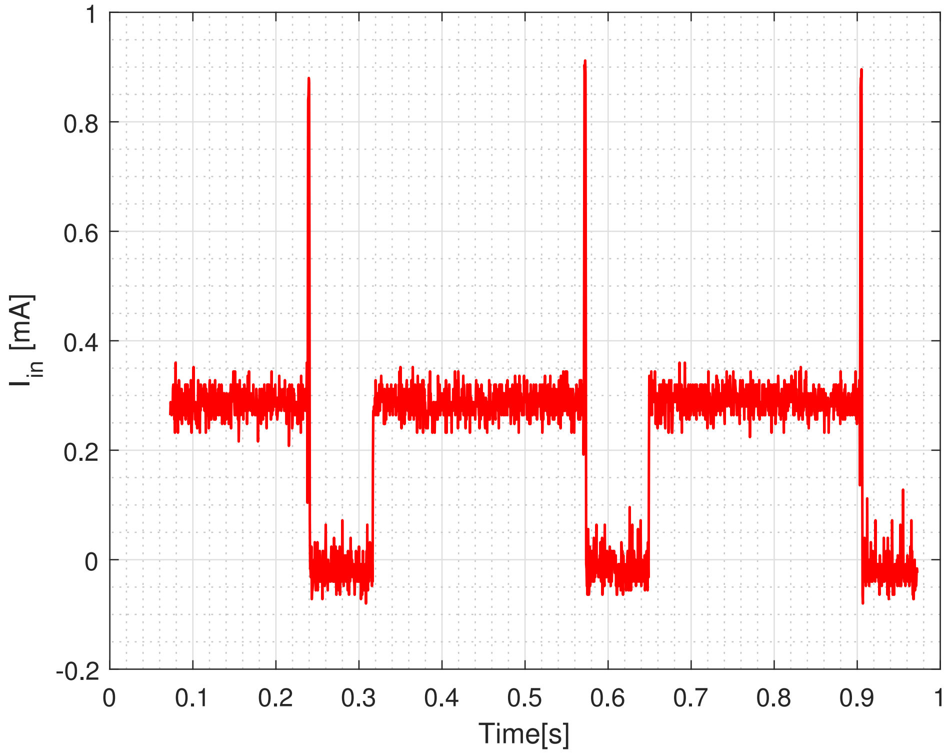


Figure 4.4: Current consumption during three measurements per cycle

4.2. PCB prototype

Since the breadboard prototype is fully working a PCB could be designed. If figure 4.5 the design is shown. All components are put on one side for ease of assembly. The separate parts are all put into white circles, these are used to indicate how the PCB can be further miniaturized. This will be discussed in Section 6.1.

There are no measurements done yet at the time of writing since it has not been assembled.

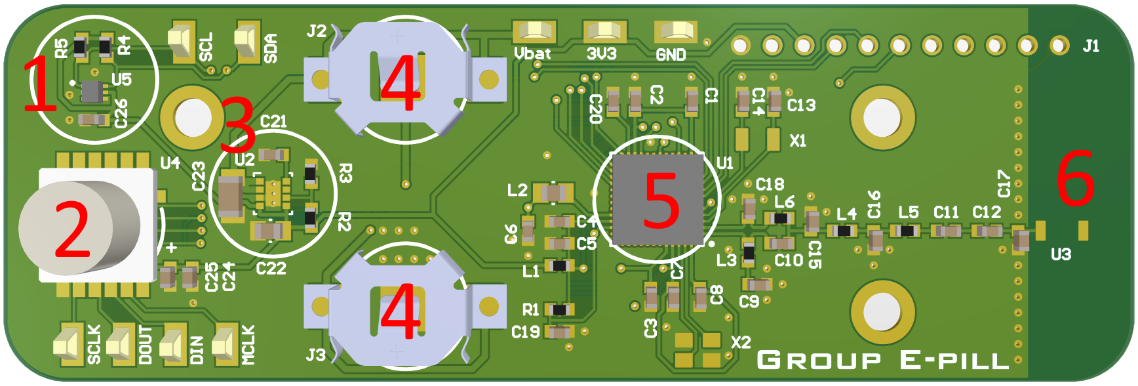


Figure 4.5: PCB design, 1 temperature sensor, 2 pressure sensor, 3 converter, 4 batteries, 5 microcontroller, 6 antenna

5

Discussion

In the end an endoscopic capsule was designed and tested on two different frequencies and it was decided that enough information is retrieved from having one measurement per cycle instead of three per cycle. This resulted in using three times less energy than for three measurements and leaving energy for other sensors to be implemented.

The converter that was chosen after some comparison between linear and switching converters as it was decided to use batteries and a step-down voltage was needed to supply the different components. The buck converter (TPS82150) has an efficiency of 85.8% at a load of 7.5mW, as seen in Section 3.2.1, this load was an over estimation of the load of the system. With the system constantly retrieving and transmitting the data from the sensor 1 measurement per cycle the system would work for 57.69 hours showing that one battery would have been enough to reach the 24 hour run time but the choice of 2 batteries was made as also other sensors would be able to be implemented.

The power management of the components should have been controlled by a threshold margin when to transmit data from the sensors (data needed to change by 0.25% to be transmitted). But due to time constraints a simple threshold barrier was implemented. This threshold barrier is implemented so the system does not transmit data and use energy when it is not in the right environment (the GI tract).

Also if the desired sensors work in the range of the battery voltage 4-6 V voltage conversion will not be needed and power will be saved.

6

Conclusion

To conclude the endoscopic pill consists of a temperature sensor and a pressure sensor which use the different parts of the microcontroller to retrieve and convert the data and transmit this data all in one cycle which last 1 second. For the power supply the batteries were put in series and the voltage had to be converted from 6 V to 3.3 V. The converter type that was chosen was the buck converter which is a switching converter. The voltage conversion was done at an efficiency of 85.8 %, see section 3.2.1.

As stated in Table 3.5 the different components are on for a total of 255.52 ms and are in low power mode for 744.5 ms. With the average current and supplied voltage the power used by the components was 2.541 mWh as stated in Section 3.3.3. The resulting available energy from the batteries is 153 mWh, as calculated in 3.3.3. With the available energy and the power used of the components in one cycle, the pill would continuously sense and transmit the data for 57.69 hours taking into account the non-ideality of the components. This would leave room for other sensors to be implemented as the 24 hour run time that was mentioned in the requirements, Chapter 2, is reached.

Table 6.1: Characteristics of components used

	$\mathbf{V_{in}}$**[V]**	$\mathbf{I_{peak}}$ **[A]**	$\mathbf{I_{avg}}$**[A]**	$\mathbf{P_{avg}}$**[W]**	$\mathbf{t_{on}}$ **1 Cycle [ms]**
Temperature sensor	3.3	$10 \cdot 10^{-6}$	$1.25 \cdot 10^{-6}$	$4.13 \cdot 10^{-6}$	0.125
Sensor controller	3.3	$393.6 \cdot 10^{-6}$	$19.68 \cdot 10^{-6}$	$64.94 \cdot 10^{-6}$	
Pressure sensor	3.3	$0.91 \cdot 10^{-3}$	$3.64 \cdot 10^{-6}$	$12.01 \cdot 10^{-6}$	252.7
Micro controller	3.3	$2.93 \cdot 10^{-3}$	$0.7325 \cdot 10^{-3}$	$2.42 \cdot 10^{-3}$	
Transmitter	3.3	$9.1 \cdot 10^{-3}$	$16.26 \cdot 10^{-6}$	$53.66 \cdot 10^{-6}$	2.7
Total	**3.3**	$\mathbf{13.34 \cdot 10^{-3}}$	$\mathbf{0.77 \cdot 10^{-3}}$	$\mathbf{2.541 \cdot 10^{-3}}$	**255.525**

A threshold was also implemented to ensure the system only transmits when it is inside of the body (body temperature range) with the test showing success in only transmitting when the temperature was over 26°C which was the threshold for that measurement 3.3.2. Also measurements with the PCB still need to be done as it was not yet assembled.

6.1. Future recommendation

Given that the project was done in 9 weeks and a specific budget ,there are several recommendations which would improve the product in the near-future. These recommendations range from having smaller components to ensure the product would fit in a swallowable pill(overall size),its power supply and components used.

6.1.1. Power supply

As for the power supply several options are there to reduce the size of the product Carbon nanotubes (Buckytubes) is a technology with great potential to supply these pills of power due to their high surface area[8].

In later stages of the project wireless power could be looked into, this has the potential to make the pill work as long as required but with less batteries(or non at all) so there would be more space for interesting sensors.

6.1.2. Other sensors

An interesting sensor that could be added is a camera, which would give a real time view of the GI tract. Since a camera uses more power than the sensors that where tested it should be possible to manage it just like the other sensors. This way the camera can also be put in standby mode when going trough less interesting parts of the GI tract so it uses less power but it can also work on less frames per second [8]. Also with the camera are the LEDs which will only flash at the moment of image capture for low-power consumption [25]. Besides a camera other sensors that can detect pH, blood or other parameters could be used as well.

To accommodate a larger amount of sensors, which means an increase in power usage, the power management could need an improvement. Power could be managed by dynamically changing which sensors are measured on the basis of camera images or by externally requesting data from a specific sensor by sending a message to the transceiver.

6.1.3. Prototype to manufacturing

There are multiple parts that have to change to go from the prototype to making a ”real” pill. With the most important part being making it smaller.

To shrink the PCB it should be designed in 10mm diameter disks with components on both sides, this way they can be stacked on top of each other and the batteries. The connection between the PCBs could be done with flex PCB.

Only the die of the ICs should be used and repackaged into a smaller IC package with all pull up resistors and other external components that are required included in it as well as much as possible.

To make the antenna the matching circuit has to be made smaller, this can be done both by redesigning it for a smaller antenna and by using smaller components.

A

Code

A.1. Transmitter

```

/*
 * Titel : Transmitter
 * Module: Digital IC of the pill
 * Author: Group E
 */

/* XDCtools Header files */
#include <xdc/std.h>
#include <xdc/runtime/System.h>

/* BIOS Header files */
#include <ti/sysbios/BIOS.h>
#include <ti/sysbios/knl/Clock.h>
#include <ti/sysbios/knl/Task.h>

/* TI-RTOS Header files */
// #include <ti/drivers/I2C.h>
#include <ti/drivers/PIN.h>
// #include <ti/drivers/SPI.h>
#include <ti/drivers/UART.h>
// #include <ti/drivers/Watchdog.h>

/* Board Header files */
#include ”Board.h”
#include <scif.h>
#include <driverlib/rf_prop_mailbox.h>

/* SmartRF settings */
#include ”smartrf_settings/smartrf_settings.h”
#include <driverlib/aon_ioc.h>


#define TASKSTACKSIZE        1024
#define TX_TASK_PRIORITY     3
#define TMP_TASK_PRIORITY    2
#define PR_TASK_PRIORITY     1
#define TMP_ARGUMENT         100000 / Clock_tickPeriod
#define PR_ARGUMENT          100000 / Clock_tickPeriod
#define TX_ARGUMENT          100000 / Clock_tickPeriod
#define PAYLOAD_LENGTH       30
#define N 100
#define wait_time 10000000
#define PRESSURE_SAMPLES     4

/* Task variables */
Char txTaskStack[TASKSTACKSIZE];
Task_Struct txTask;
Task_Params taskParams;

/* Semaphore variables */
Semaphore_Struct semStruct1;
Semaphore_Handle semTX;
Semaphore_Params semParams;

/* RF variables */
RF_Params rfParams;
```

```c
RF_Object rfObject;
RF_Handle rfHandle;

/* UART variables */
UART_Handle uart;
UART_Params uartParams;

/* Temporary packet variables */
uint32_t time;
uint8_t packet[PAYLOAD_LENGTH];

/* Pin driver handle */
PIN_Handle ledPinHandle;
PIN_State ledPinState;
PIN_Config ledPinTable[] = {
        Board_LED0 | PIN_GPIO_OUTPUT_EN | PIN_GPIO_LOW | PIN_PUSHPULL | PIN_DRVSTR_MAX, // Red led
        Board_LED1 | PIN_GPIO_OUTPUT_EN | PIN_GPIO_LOW | PIN_PUSHPULL | PIN_DRVSTR_MAX, // Green led
        IOID_0 | PIN_INPUT_EN | PIN_GPIO_LOW | PIN_PUSHPULL | PIN_DRVSTR_MAX,         // DOUT
        IOID_12 | PIN_GPIO_OUTPUT_EN | PIN_GPIO_LOW | PIN_PUSHPULL | PIN_DRVSTR_MAX,     // SCLK
        IOID_15 | PIN_GPIO_OUTPUT_EN | PIN_GPIO_LOW | PIN_PUSHPULL | PIN_DRVSTR_MAX,     // DIN
        IOID_21 | PIN_GPIO_OUTPUT_EN | PIN_GPIO_LOW | PIN_PUSHPULL | PIN_DRVSTR_MAX,     // MCLK
        PIN_TERMINATE
};

int temperature;
int pressure;


/* reading pressure sensor: */

unsigned int coefficients_[6];

// 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);
    }
}

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

/* 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;
}

/* Read the coefficient from the sensor */
size_t ReadCoefficient(unsigned char addr)
{
    // 111 + 6bit coeff addr + 000 + 1clk(send0)
    unsigned long cmd = (unsigned long)0x1C00 | (((unsigned long)addr) << 4);
    SendCommand(cmd,13);
    return ReadWord();
}

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

```
    unsigned int wa = ReadCoefficient(0x15);

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

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

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

#ifdef DEBUG
    //    for(size_t i=0; i<6; ++i)
    //        {
    //            Serial.print“(Coefficient “);
    //            Serial.print(i + 1, DEC);
    //            Serial.print”( : “);
    //            Serial.println(coefficients_[i], DEC);
    //    }
#endif
}

/* Calculate the pressure value */
long ConvertPressureTemperature()
{
    const long UT1  = (coefficients_[4] << 3) + 20224;
    const long dT   = (long)temperature  - UT1;
    const long TEMP = 200 + ((dT * (coefficients_[5] + 50)) >> 10);
    const long OFF  = (coefficients_[1] <<2) + (((coefficients_[3] -512) * dT) >> 12);
    const long SENS = coefficients_[0]  + ((coefficients_[2] * dT) >> 10)  + 24576;
    const 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;
}

/* Read one sample from the sensor */
void TriggerTemperatureSample(void)
{
    // 111 + 1001 + 000 + 2clks(send 0)
    ResetSensor();
    SendCommand(0xF20, 12);
}

void TriggerPressureSample(void)
{
    // 111 + 1010 + 000 + 2clks(send 0)
    ResetSensor();
    SendCommand(0xF40, 12);
}

/* Read the average value by reading multiple samples */
void AcquireAveragedSampleCm(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;
}
```

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

/* reading temperature  sensor: */

// SCIF driver callback: Sensor Controller task code has generated an alert interrupt
void scTaskAlertCallback(void) {

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

/* Task: */

/* Transmitter task */
void TaskFunction(UArg arg0, UArg arg1){

    //initialisation of the rf parameters
    RF_Params_init(&rfParams);
    RF_cmdPropTx.pktLen = PAYLOAD_LENGTH;
    RF_cmdPropTx.pPkt = packet;
    RF_cmdPropTx.startTrigger.triggerType = TRIG_ABSTIME;
    RF_cmdPropTx.startTrigger.pastTrig = 1;
    RF_cmdPropTx.startTime = 0;
    rfHandle = RF_open(&rfObject, &RF_prop, (RF_RadioSetup*)&RF_cmdPropRadioDivSetup, &rfParams);
    RF_postCmd(rfHandle, (RF_Op*)&RF_cmdFs, RF_PriorityNormal, NULL, 0);

    while(1){
        Semaphore_pend(semTX, BIOS_WAIT_FOREVER); //blocking the task

        packet[0] = (uint8_t)(temperature >> 8);
        packet[1] = (uint8_t)(temperature);
        packet[2] = (uint8_t)(pressure >> 8);
        packet[3] = (uint8_t)(pressure);

        /* sending data to the antenna */
        RF_CmdHandle cmdHandle = RF_postCmd(rfHandle, (RF_Op*)&RF_cmdPropTx, RF_PriorityNormal, NULL, 0);
        if (cmdHandle<0)
        {
            while(1);
        }
        RF_EventMask result2 = RF_pendCmd (rfHandle, cmdHandle, 0);
        RF_yield(rfHandle);
    }
}

/* Initialisations: */

/* Tasks initialisation */
void tasksInit(){

    Task_Params_init(&taskParams);
    taskParams.stackSize = TASKSTACKSIZE;
    taskParams.priority = TX_TASK_PRIORITY;
    taskParams.stack = &txTaskStack;
    taskParams.arg0 = TX_ARGUMENT;
    Task_construct(&txTask, TaskFunction, &taskParams, NULL);
}

/* Semaphore initialisation */
void semaphoresInit(){

    Semaphore_Params_init(&semParams);
    Semaphore_construct(&semStruct1, 1, &semParams);
    semTX = Semaphore_handle(&semStruct1);
}

/*initialisation of the pressure sensor*/
void pressure_init(){

    ResetSensor();
    ReadCoefficients();
    IOCPortConfigureSet(IOID_9, IOC_PORT_AON_CLK32K, IOC_STD_OUTPUT);
    AONIOC32kHzOutputEnable();
}

/*initialisation of the Sensor Controller and its interrupt*/
void temperature_init(){

```

```
    // Initialize the SCIF operating system abstraction layer
    scifOsalInit();
//    scifOsalRegisterCtrlReadyCallback(scCtrlReadyCallback);
    scifOsalRegisterTaskAlertCallback(scTaskAlertCallback);

    // Initialize the SCIF driver
    scifInit(&scifDriverSetup);

    // Enable RTC ticks, with N Hz tick interval
    scifStartRtcTicksNow(0x00010000 / N);

    // Start the "TMP112" Sensor Controller task
    scifStartTasksNbl(1 << SCIF_TMP112_TASK_ID);
}

/*initialisation of the pins*/
void pins_init(){
    ledPinHandle = PIN_open(&ledPinState, ledPinTable);
}

/*
 * ======== main ========
 */
int main(void){

    Board_initGeneral();
    tasksInit();
    semaphoresInit();
    pins_init();
    pressure_init();
    temperature_init();
    BIOS_start(); //start RTOS kernel

    return (0);
}
```

A.2. Threshold temperature sensor

```
// Configure and start the next measurement
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();

// Read the result after ~100 milliseconds + a 20% margin
evhSetupTimer1Trigger(0, 120, 2);

// Schedule the next execution
fwScheduleTask(100);

// 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 (4-bit exponent + 12-bit mantissa) into 16-bit fixed-point
        U16 value = (resultRegH << 4) | (resultRegL >> 4);
        output.value = value;

           // Notify the application if the result is above the high threshold
           if (value > cfg.highThreshold) {
                   fwGenAlertInterrupt();
           }

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

Bibliography

[1] ASGE, *Understanding capsule endoscopy*. [Online]. Available: `https://www.asge.org/home/for-patients/patient-information/understanding-capsule-endoscopy`.

[2] P. Swain, “Wireless capsule endoscopy and crohn’s disease”, *Gut*, vol. 54, no. 3, pp. 323–326, 2005, ISSN: 0017-5749. DOI: `10.1136/gut.2004.047282`. eprint: `https://gut.bmj.com/content/54/3/323.full.pdf`. [Online]. Available: `https://gut.bmj.com/content/54/3/323`.

[3] L. E. Moore, “The advantages and disadvantages of endoscopy”, *Elsevier*, vol. 18, pp. 250–253, 2003. DOI: `10.1016/S1096-2867(03)00071-9`.

[4] T. D. Mehmet R. Yuce, “Easy-to-swallow wireless telemetry”, *IEEE Microwave Magazine*, vol. 13, pp. 90–101, 2012. DOI: `10.1109/MMM.2012.2205833`.

[5] 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*, vol. 53, no. 11, pp. 2333–2340, Nov. 2006, ISSN: 0018-9294. DOI: `10.1109/TBME.2006.883698`.

[6] P. J. Kundong Wang Guozheng Yan and D. Ye, “A wireless robotic endoscope for gastrointestine”, *IEEE Transactions on Robotics*, vol. 24, pp. 206–210, 2008. DOI: `10.1109/TRO.2008.915418`.

[7] D. Fitzpatrick, “Implantable electronic medical devices”, in. Elsevier, 2015, ch. 15.

[8] P. Swain, “The future of wireless capsule endoscopy”, *World Journal of Gastroenterology*, vol. 14, pp. 4142–4145, 2008. DOI: `10.3748/wjg.14.4142`.

[9] D. C. Zoya Popovic Erez Falkenstein and R. Zane, “Low-power far-field wireless powering for wireless sensors”, *Proceedings of the IEEE*, vol. 101, pp. 1397–1409, 2013. DOI: `10.1109/JPROC.2013.2244053`.

[10] M. 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*, vol. 25, pp. 24–7, Jan. 2002.

[11] T. S. Diego Lisbona, “A review of hazards associated with primary lithium and lithium-ion batteries”, *Process Safety and Environmental Protection*, vol. 89, pp. 434–442, 2011. DOI: `10.1016/j.psep.2011.06.022`.

[12] T. P. Crompton, *Battery Reference Book*. Elsevier, 2000, ISBN: 9780080499956.

[13] R. batteries, *Cr1025*, CR1025 Lithium 3 V, 2006. [Online]. Available: `https://www.celltech.fi/fileadmin/user_upload/Celltech/Prod.sheets/CR1025_v06.pdf`.

[14] K. Iniewski, “Cmos biomicrosystems: Where electronics meet biology”, in. Wiley, 2011, ch. 5.

[15] L. Benini, G. Castelli, A. Macii, and R. Scarsi, “Battery-driven dynamic power management”, *Design & Test of Computers, IEEE*, vol. 18, pp. 53–60, Apr. 2001. DOI: `10.1109/54.914621`.

[16] T. Instruments, *Tps82150silt*, Buck converter TI, 2017. [Online]. Available: `http://www.ti.com/lit/ds/symlink/tps82150.pdf`.

[17] C. Glaser, *High-efficiency, low-ripple dcs-control™ offers seamless pwm/power-save transitions*, 2013. [Online]. Available: `http://www.ti.com/lit/an/slyt531/slyt531.pdf`.

[18] P. Valdastri, A. Menciassi, A. Arena, C. Caccamo, and P. Dario, “An implantable telemetry platform system for in vivo monitoring of physiological parameters”, *IEEE Transactions on Information Technology in Biomedicine*, vol. 8, no. 3, pp. 271–278, Sep. 2004, ISSN: 1089-7771. DOI: `10.1109/TITB.2004.834389`.

[19] G. D. M. Luca Benini Alessandro Bogliolo, “A survey of design techniques for system-level dynamic power management”, *IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS*, vol. 8, pp. 299–316, 2000. DOI: `10.1109/92.845896`.

[20] T. Instruments, *Ms5534c*, Micro controller TI, 2016. [Online]. Available: `https://www.te.com/commerce/DocumentDelivery/DDEController?Action=showdoc&DocId=Data+Sheet%5C%7FMS5534C%5C%7FB%5C%7Fpdf%5C%7FEnglish%5C%7FENG_DS_MS5534C_B.pdf%5C%7FCAT-BLPS0032`.

[21] ——, *Tmp112b*, Temperature sensor TI, 2018. [Online]. Available: `http://www.ti.com/lit/ds/symlink/tmp112.pdf`.

[22] ——, *Cc2650*, Micro controller TI, 2016. [Online]. Available: `http://www.ti.com/lit/ds/symlink/cc2650.pdf`.

[23] V. E. D. Bene, “Clinical methods: The history, physical, and laboratory examinations. 3rd edition”, in. Butterworth Publishers, 1990, ch. 218.

[24] Fluke, *Fluke multimeter*, Micro controller TI, 2016. [Online]. Available: `http://assets.fluke.com/manuals/175_____umeng0000.pdf`.

[25] X. Xie, G. Li, X. Chen, X. Li, and Z. Wang, “A low-power digital ic design inside the wireless endoscopic capsule”, *IEEE Journal of Solid-State Circuits*, vol. 41, no. 11, pp. 2390–2400, Nov. 2006, ISSN: 0018-9200. DOI: `10.1109/JSSC.2006.882884`.