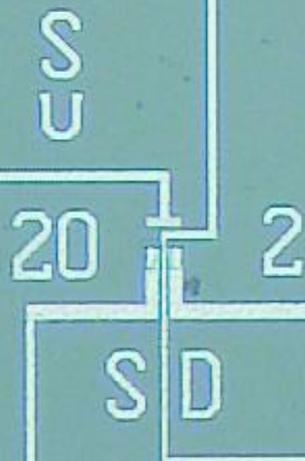


Bachelor Graduation Project Thesis

Portable Parameter Analyser for Organs-on-Chip Power Budget Analysis

Written By:
Eray Albayrak
Maarten Mens

Delft University of Technology



Portable Parameter Analyser for Organs-on-Chip Power Budget Analysis

Bachelor Graduation Project

Maarten Mens - 4590228

Eray Albayrak - 4230418

Supervisors

Dr. Massimo Mastrangeli

Dr. Marco Spirito

Thesis committee

Prof. Dr. Nuria Llombart-Juan

Dr. Massimo Mastrangeli

Dr. Marco Spirito

Carmin De Martino

Hande Aydogmus

Date: June 19, 2020

UNDER EMBARGO UNTIL MAY 1st, 2021

DELFT UNIVERSITY OF TECHNOLOGY

FACULTY OF ELECTRICAL ENGINEERING, MATHEMATICS
AND COMPUTER SCIENCE

BSC ELECTRICAL ENGINEERING PROGRAMME

Abstract

Problems in drug development and personalised disease treatments are the main impulse for the development of OoCs. At the Electronic Components, Technology and Materials (ECTM) group at TU Delft an OoC sensor has been developed and the design is now at a stage where a system can be developed to take the sensor out of the initial research environment, into a complete product that incorporates the sensor. In this thesis a design is proposed for a portable parameter analyser based around the STM32L4 microcontroller platform. It contains an instrumentation amplifier and a communication module to connect to an external device. A power consumption estimation is made for individual system components as well as for the system as a whole. From this it is concluded which components contribute the most to the power consumption and recommendations are made on how to decrease this power consumption.

Preface

Before you lies the thesis "Portable Parameter Analyser for Organs-on-Chip - Power Budget Analysis", which is a part of the three-part project proposed by the ECTM/ELCA group at the Delft University of Technology. This thesis has been written to fulfil the graduation requirements of the BSc Electrical Engineering Programme at the Delft University of Technology.

For the past two months we have been tested on our electro-technical, analytical, and system design skills. It has been a long journey under strange circumstances due to the COVID-19 outbreak. Consequently, all work had to be done from home which was unfortunate as working together with your peers in person is one of the joys of participating in a group project. On top of that the situation challenged our discipline greatly. For the project it meant that no physical demonstrator was allowed, shifting the focus of the project from a balance between theoretical design and practical implementation to a completely theoretical basis.

We would like to express our gratitude towards our supervisors Dr. Marco Spirito and Dr. Massimo Mastrangeli and our daily advisors Carmine De Martino and Hande Aydogmus for their support, supervision, and patience during this project. We wish them good luck with further developments on the OoC system.

Furthermore we would like to thank our colleague project members Maurice van der Maas, Kevin dos Reis Vezo, Jillis Noordhoek, and Yme Wesseling for their collaboration.

—————
Eray Albayrak and Maarten Mens
Delft, June 2020

Contents

Abstract	i
Preface	ii
1 Introduction	1
1.1 Organ-on-Chip (OoC) systems	1
1.2 Problem definition	2
1.3 Currently used set-up	2
1.4 Literature overview	2
1.4.1 ADC comparison	3
1.4.2 Electronics in humid environments	3
1.4.3 Effect of high temperatures on batteries	3
1.5 Project goal	3
1.6 Thesis structure	3
2 Program of Requirements	4
3 Proposed design	5
3.1 Sensing board	6
3.1.1 Instrumentation amplifier	6
3.1.2 Tuning circuit	7
3.2 Interconnection schematic	7
3.3 Development board	7
3.3.1 Micro-controller (STM32L4)	7
3.3.2 Communication Modules	8
3.3.3 Temperature/Humidity sensor (HTS221)	8
3.4 Data flow	9
3.5 Communication protocol	9
4 Measurement routine	10
4.1 Design considerations	10
4.2 Proposed design	11
4.2.1 Measuring routine	11
4.2.2 Initialisation	11
4.2.3 Sampling	11
4.2.4 Data transfer	12
4.2.5 Idle	12
4.2.6 Shutdown	13

5	Power budget analysis	14
5.1	Sensor	14
5.2	Sensing board	14
5.2.1	Instrumentation amplifier	14
5.2.2	Tuning circuit	14
5.3	Discovery board	15
5.3.1	Micro-controller (STM32L4)	15
5.3.2	Bluetooth module (SPBTLE-RF)	16
5.3.3	Measurement state duration	16
5.4	Current consumption for the measurement routine	18
5.4.1	Test case 1: Instant data transfer	18
5.4.2	Test case 2: data transfer after fixed amount of samples or on user request	20
5.4.3	Comparison	21
5.4.4	Test case 3: switch off the in-amp in between sampling	21
5.4.5	Conclusion	21
6	Power management and heat dissipation	23
6.1	Requirements	23
6.2	Power source	23
6.2.1	Battery	23
6.2.2	DC-DC converter	24
6.2.3	Charging chip	24
6.2.4	Battery capacity	24
6.3	Power design	25
6.4	Heat	25
7	Discussion	26
8	Conclusion	27
8.1	Recommendations	27
8.2	Future work	27
	Bibliography	28
	Appendices	30
A	In-amp current consumption simulation	31
B	Power Consumption Calculator	33
C	BlueNRG current consumption estimator	39

Chapter 1

Introduction

1.1 Organ-on-Chip (OoC) systems

Problems in drug development and personalised disease treatments are the main impulse for the development of OoCs [1], which stem from a synthesis of tissue engineering and microfluidic technology. Now that dynamic fluid flow, electro-mechanical stimuli and controlled biochemistry can be purposed to provide a physiologically representative micro-environment to cell cultures, biomedical researchers can get a deeper understanding of the mechanisms and etiology behind the diseases. Various industries opt for human models to minimise toxicological risks where physiological relevancy can increase the certainty required under the increasingly strict regulations. Besides, conventional methods nowadays still include animal testing despite being subject to ethical questioning. In the coming years the remaining technological challenges for OoC systems will be tackled hopefully resulting in the key unmet needs being met.

At the Electronic Components, Technology and Materials (ECTM) group at TU Delft an OoC sensor has been developed and the design is now at a stage where a system can be developed to take the sensor out of the initial research environment, into a complete product that incorporates the sensor. The sensor in question consists of eight Floating Gate Field Effect Transistors (FGFET) designed to measure the changes in electro-chemical charges in a (bio)chemical solution. As can be seen in Figure 1.1 the floating gates of the FGFETs extend into the sensing area, such that charges accumulated on the gates will modulate the gate-source voltage and therefore change the FGFET threshold voltage and drain current [2].

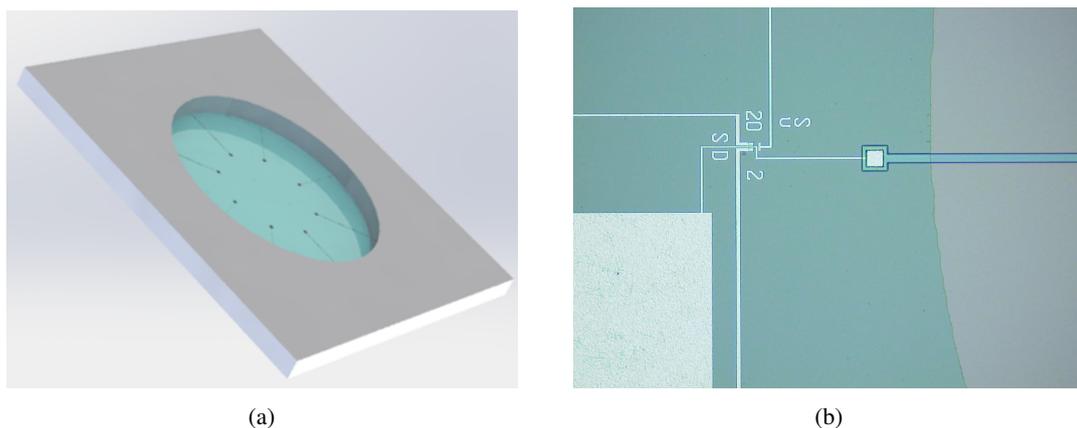


Figure 1.1: a) Model of the sensor showing the 8 floating gates extending into the sensing area. b) Close-up made during the fabrication process, showing one of the transistors with the first polyimide insulation layer for the floating gate extending to the right of the figure [3].

1.2 Problem definition

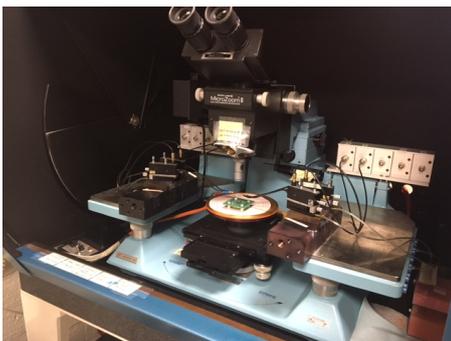
Real-time cell culture condition monitoring is generally done in biological or clinical research environments that in contrast to engineering laboratories, are not equipped with sophisticated electrical measuring equipment. At this stage the end-user has not been given enough thought. Purchasing peripheral equipment with redundant functionality or complexity isn't cost-effective. Moreover source measure units (SMU) aren't necessarily easy to maintain and operate without extensive training. Furthermore, parameter analysers take up too much space to fit inside an incubator. Thus a OoC specific portable parameter analyser is desired. The OoC sensor should measure electro-chemical changes on or in close proximity of the sensing area. The OoC sensor itself is already designed, however it needs to be modelled in order to characterise it and to see where it could be improved in order to increase the sensitivity. For the very small currents and voltages that need to be measured, the Electronic Circuits and Architectures (ELCA) group at the TU Delft already designed an instrumentation amplifier which is given to the group. This design is evaluated together with the rest of the system in order to verify whether it can measure the small changes on the sensor. Three subgroups consisting of two students each were assigned to this project and were each given one of the following tasks:

1. The analysis of the power budget and heat exchange together with an interface design
2. The calibration together with the design of a user-friendly GUI [4]
3. The modeling of the sensor and providing a sensitivity analysis [5]

The analysis of the power budget and heat exchange will be the subject of this thesis.

1.3 Currently used set-up

As mentioned in Section 1.2 the systems currently used are too large to fit inside incubators. The system currently used at the ECTM group at the TU Delft is shown in Figure 1.2. It consists of an experiment set-up containing a microscope and connecting probes for the FGFET sensor (Figure 1.2a) and an SMU (Figure 1.2b). The particular SMU used is the HP4145B [6]. The SMU sets the bias voltage of the control gate, sets the drain-source voltage and measures the drain current of the FGFET sensor.



(a) The microscope and connecting probes for the sensor.



(b) The HP4145B parameter analyser used to set the bias voltages and read the drain current.

Figure 1.2: The measurement setup currently used by the ECTM group in Delft.

1.4 Literature overview

A short overview of the literature study performed will be given in this section.

1.4.1 ADC comparison

When it comes to analog-to-digital converters (ADC) there are many options available concerning architecture. The kind of ADC used will have an impact on power consumption and accuracy of the system. SAR (successive approximation register) ADCs have become very popular due to their excellent energy efficiency for medium resolution and low-to-medium sampling speeds [7]. This makes the SAR ADC very suitable for the application at hand. The method proposed by [8] can be used to calculate power consumption in variously structured SAR ADCs.

1.4.2 Electronics in humid environments

The environment in which electronics are used has an impact on the life time and failure rate of the electronics. It is therefore important to take into account the effects of humidity, temperature cycling, and thermally heavy component placement on local condensation within the enclosure [9] as well as on leakage current [10]. Measures should be taken to protect the electronic components from moisture, dust and gases, like dip coating the PCB [9].

1.4.3 Effect of high temperatures on batteries

As lithium-ion batteries are low cost and widely available they are the most viable option for this system. As the system will be used in higher temperatures than room temperature the effect of high temperatures on lithium-ion batteries should be taken into account as temperature is an important factor affecting the health and safe operation of these batteries. Degradation rates of all components in the battery are increased as a result of high operating temperatures [11].

1.5 Project goal

The goal of this project is to propose a system design for the portable parameter analyser which is compatible with the charge modulated FGFET sensor as described in [2]. A measurement routine, which will describe the interaction between the different sub-components of the system, will be designed. On that design a power consumption analysis will be performed in order to be able to determine the amount of energy storage necessary to operate.

1.6 Thesis structure

In Chapter 2 the system requirements will be described, in Chapter 3 a system design will be proposed, in Chapter 4 the measurement routine design will be given, in Chapter 5 the power consumption analysis for the system is performed, in Chapter 6 a proper power source will be determined. All results will be summarised in Chapter 7 and the thesis will be concluded and recommendations will be made in Chapter 8.

Chapter 2

Program of Requirements

Listed below are the requirements for the total system, from which the ones that are specifically targeted at the power budget subject are highlighted in bold. A distinction is made between mandatory and trade-off requirements.

Mandatory requirements:

- **Must operate within an incubator at 37°C**
- **Must operate within an incubator with a CO₂ concentration of 5%**
- **Must be resistant to the humidity of the incubator (IP59)**
- **Must be able to fit inside most microbiological incubators**
- **Must be battery powered**
- **Must support experiments of at least 6 days**
- **Must sense and amplify the drain current of the FGFET**
- **Must control the drain-source and gate-source voltages of the FGFET**
- **Must sense in the order of μA and control in the order of mV**

- **Measurement data should be available during experiments**

- Must visualise the drain current vs drain voltage (I_D vs V_{DS}) characteristics
- Shall provide a comprehensive GUI for biologists

Trade-off requirements:

- **Minimize power consumption**
- **Minimize heat dissipation**
- **Minimize noise**
- **Maximize current and voltage resolution**
- **Minimize volume**
- Maximize sensitivity by means of FGFET geometry optimization

Chapter 3

Proposed design

To achieve the project goal a new system has to be designed. For this new system a couple of elements are given:

- FGFET Sensor: A sensor developed at the TU Delft.
- Sensing board: A board which is used to convert the measured drain current of the sensor into an analog voltage which can be processed further.
- Discovery board (B-L475E-IOT01A2): This board is used to control the system, perform measurements and communicate the results to an external device.

The proposed design overview is represented in figure 3.1. The production version will be in an enclosure. The system will power the sensor and be able to measure and send the results.

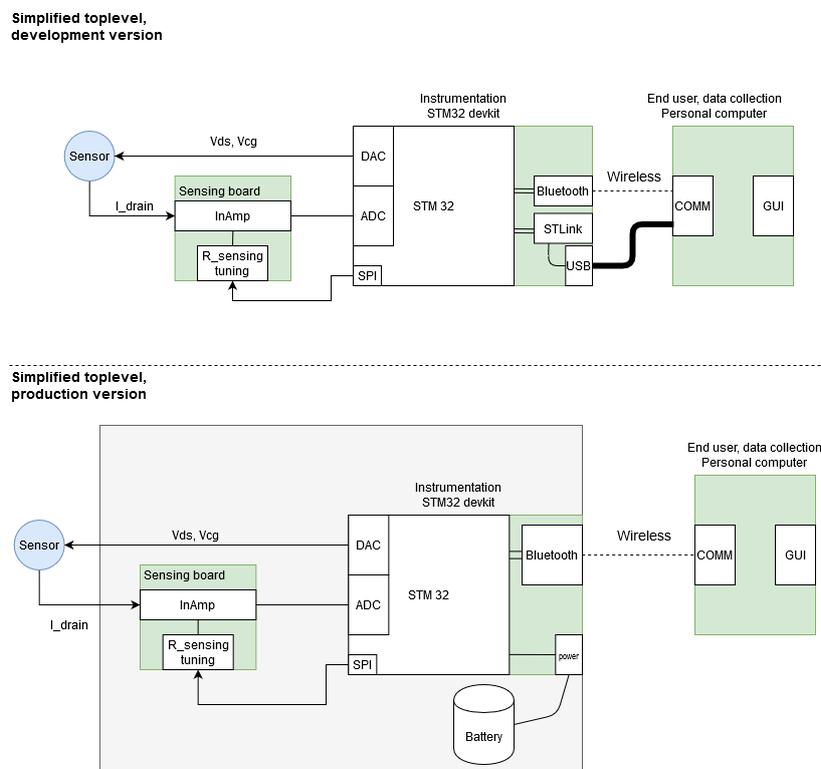


Figure 3.1: Toplevel diagram for the design.

3.1 Sensing board

The sensing board is a board developed by the ECTM/ELCA group to be used to convert the drain current of the FGFET sensor into an analog voltage which can be processed by the analog-to-digital converter in the development board. Figure 3.2 shows the sensing board, it consists of two parts; the instrumentation amplifier (in-amp) and a tuning circuit.

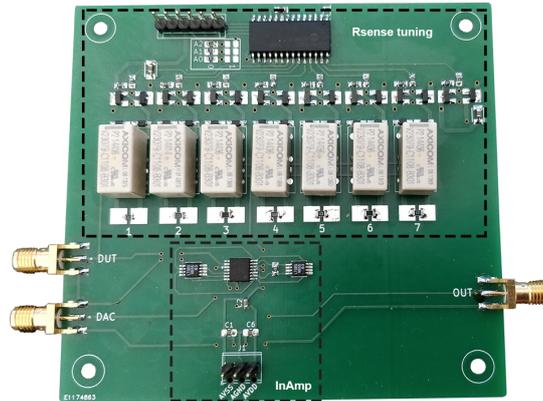


Figure 3.2: The sensing board. The top rectangle indicates the tuning circuit and the bottom rectangle indicates the instrumentation amplifier.

3.1.1 Instrumentation amplifier

The in-amp uses 3 operational amplifiers (op-amps) to convert the current from the sensor to an analog voltage. A schematic for this circuit can be seen in Figure 3.3. The two input op-amps buffer the inputs and have between them a combined gain of 100. The right-most op-amp takes the difference between the outputs of the first two op-amps.

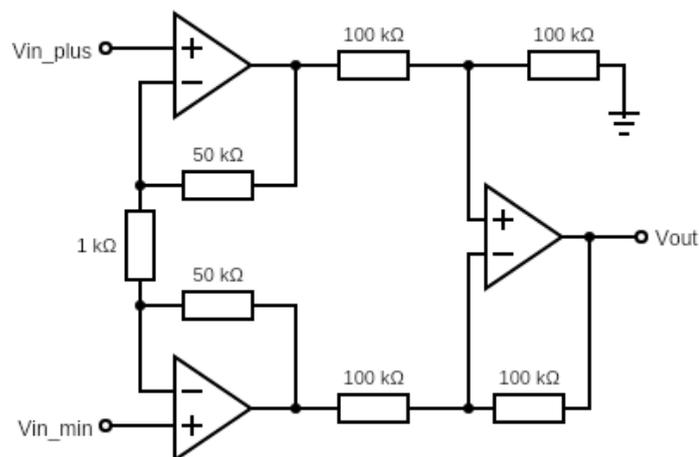


Figure 3.3: Schematic for the instrumentation amplifier.

3.1.2 Tuning circuit

The tuning circuit can be used to change the resistance of R_{sense} . This value determines the sensitivity of the in-amp. The value can be increased or decreased by switching on/off signal relays which introduce an extra resistor in parallel. A total of 7 resistors can be connected in parallel. The signal relays are controlled by a 16 bit I/O Expander (MCP23S17) [12] which is controlled through SPI by the micro-controller.

3.2 Interconnection schematic

Figure 3.4 shows the interconnection between the FGFET sensor equivalent circuit, instrumentation amplifier, analog-to-digital and digital-to-analog converters. A full description of the FGFET sensor can be found in [5]. The sense resistor R_{sense} between the inputs causes a voltage drop as the drain current I_{drain} flows through it which is amplified by the in-amp. The output of the in-amp is connected to an analog-to-digital converter. The op-amp shown in the middle sets a bias voltage/drain-source voltage V_{DS} that is independent of I_{drain} . The control gate voltage V_{CG} and V_{DS} are both controlled by a digital-to-analog converter.

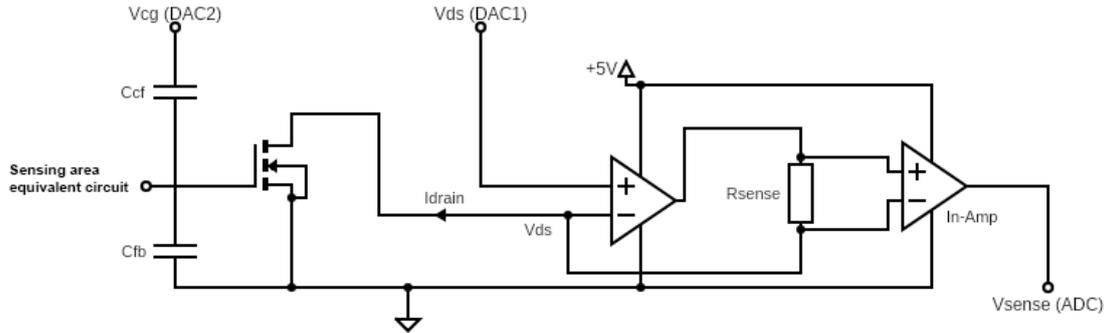


Figure 3.4: Schematic showing the interconnection between the FGFET sensor, instrumentation amplifier, ADC and DACs.

3.3 Development board

The development board available for this project is the Discovery kit B-L475E-IOT01A2 by STMicroelectronics [13]. The kit uses a STM32L4 as micro-controller and features various peripherals that can be used. The elements of the development board that will be used in this project are the STM32L4 MCU, WiFi or Bluetooth module and temperature sensor. For programming this development board the STM32CubeIDE programming environment is used.

3.3.1 Micro-controller (STM32L4)

The STM32L4 is a low powered micro-controller with a maximum system clock frequency of 80 MHz. Some of its features are:

- Operating voltage range: $1.71 \leq V_{DD} \leq 3.6$ V
- 3 Analog-to-Digital Converters (ADC): 12 bits resolution
- 2 Digital-to-Analog converters (DAC): 12 bits resolution
- Several low-power modes.

- Communication: USART, SPI, I²C

The micro-controller performs several functions; it biases the sensor with the Digital-to-Analog converters, it handles the transfer of data and control commands with the GUI through Bluetooth, it handles data storage in flash memory, it can adjust the sensitivity of the in-amp, and it converts the output voltage of the in-amp using an analog-to-digital converter.

Analog-to-Digital converter

The ADCs featured on the micro-controller are SAR ADCs and offer up to 12-bit analog-to-digital conversion. The conversion is performed in several steps. The number of conversion steps is equal to the number of bits in the ADC [14, Table 66].

Its input voltage range is 0 to 3.6 V for a supply voltage of $V_{DD} = 3.6$ V. This results in an input voltage resolution of $\frac{3.6\text{V}}{2^{12}} = 878.9 \mu\text{V}$. The ADC sample rate will be very low, in the order of samples per minute or lower, as will be elaborated upon in Section 4.1.

Digital-to-Analog converters

The micro-controller features two 12-bit DACs which will be used to set the bias voltages of the FGFET sensor. The output voltage range is 0 to 3.6 V with the output buffer disabled and when supplied with a voltage of $V_{DD} = 3.6$ V [14, Table 72].

The DAC needs to control the control gate voltage V_{CG} as well as the drain-source voltage V_{DS} of the FGFET sensor. For the control gate voltage a value of $V_{CG} = 3.0$ V has been selected. V_{CG} is ideally as high as possible and $V_{CG} = 3.0$ V is the highest value tested within the DAC's output voltage range [5]. For this V_{CG} the saturation region of the FGFET starts at $V_{DS} = 1.45$ V. As the sensor needs to operate in its saturation region the drain-source voltage needs to be higher than 1.45 V [5]. A drain-source voltage of $V_{DS} = 1.65$ V has been selected as this will prevent the in-amp output voltage to go over 3.3 V making sure the ADC input will not be overloaded. Why this happens will be explained in more detail in Section 5.2.1.

The output buffers available on the DACs will not be necessary as the DAC does not need to drive a high load (meaning low impedance). The control gate is a capacitor with a DC voltage applied to it, so its impedance is high and no current runs through it. The DAC controlling V_{DS} is connected to the op-amp driving I_{drain} acting as a buffer, meaning the DAC does not have to provide high current.

3.3.2 Communication Modules

There are two modules available for wireless communication between the system and an external device. The WiFi module makes it possible for the system to work as an IoT device, however, within the scope of this project this is not interesting. Bluetooth has to be connected to an external device in close proximity which gives less flexibility. A benefit of using Bluetooth, however, is its simplicity in use and lower power consumption. The choice is made to use Bluetooth as it will suit the application at hand very well.

Bluetooth module (SPBTLE-RF)

A version 4.1 Bluetooth Low Energy module [15] is available on the development board. This module is low powered, has an operating voltage between 1.7 and 3.6 V, and can be interfaced with through SPI.

3.3.3 Temperature/Humidity sensor (HTS221)

The development board features a temperature sensor, the HTS211 [16]. This sensor is a capacitive digital sensor which is able to measure relative humidity and temperature. The sensor can be used to monitor the humidity and temperature within the enclosure to prevent harm to the circuit. Furthermore, the temperature could be used to optimise measurements that may be affected by heat. This sensor has a 16-bit ADC, can be supplied with a voltage between 1.7-3.6 V, and can be interfaced with through SPI and I²C.

3.4 Data flow

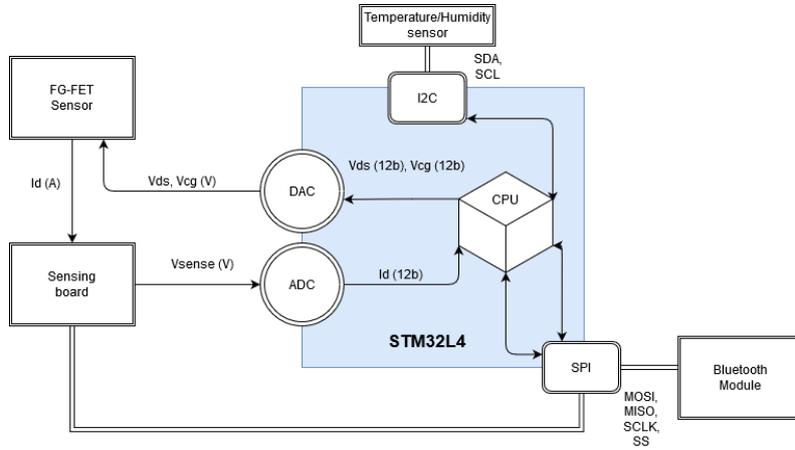


Figure 3.5: Flowchart of data flow and control signals

Figure 3.5 shows a data flow diagram for the system. The drain current from the FGFET sensor is converted into a voltage which is amplified by the instrumentation amplifier on the sensing board. V_{sense} is then converted to a digital value using the ADC. The drain-source voltage and control gate voltage of the FGFET sensor are controlled by two DACs. The temperature and humidity sensor is connected to the micro-controller through I²C and both the Bluetooth module and sensing board tuning circuit are connected through SPI.

3.5 Communication protocol

[4] describes the communication protocol proposed to communicate between the graphical user interface (GUI) and the micro-controller. A set of commands has been defined for the user to request certain data and control certain parameters. In order to estimate the power consumption contribution of the communication module the length of these commands must be known.

Table 3.1 shows all commands and their lengths in bytes. All REQUEST commands have a length of 1 byte as well as the STATE_MOVE, STATE_FORCE, and SHUTDOWN commands. The SET_RESISTOR command has a length of 1 byte as there are 7 resistors on the sensing board which can be switched on and off. Each resistor is controlled by a separate bit. The SET_BIAS command consists of 2 bytes: 1 selector bit to select the bias voltage and 12 bits for the value. These lengths could be reduced in the future as [4] mentions, as there is no need to use whole bytes for most commands. For now these lengths will be used, to give a worst-case scenario.

Table 3.1: Communication protocol commands and their lengths in bytes.

Command	Length (in bytes)
STATE_MOVE	1
STATE_FORCE	1
REQUEST_MEAS_ALL	1
REQUEST_INFO	1
SHUTDOWN	1
REQUEST_MEAS	1
SET_RESISTOR	1
SET_BIAS	2

Chapter 4

Measurement routine

The experiments that will be performed using this system involve biochemical processes. These processes are inherently slow and the experiments can run for multiple days. There are several requirements; it is important for the system to be able to support these long experiments; the system needs to be flexible enough to be able to support shorter experiments as well; the user must be able to configure the experiment length, sample rate, and bias voltages for the sensor; the user must be able to view the measured data during the experiment; there needs to be some form of protection against data loss in case of a loss of power. On top of these requirements the requirement of low power consumption is a leading factor in designing the routine.

These requirements dictate how measurements are performed by the system. The method with which measurements are performed can be described in a measurement routine. The system's main tasks can be identified as sampling the to be measured parameters, storing the sampled values, and transferring the stored data to an external device. This can be translated into multiple stages or states, an initialisation state, a sampling state, a data transfer state, and an idle state. These states together form the measurement routine and will be further elaborated upon in Section 4.2.

4.1 Design considerations

As mentioned above the changes occurring in biochemical processes are slow, in the order of minutes. This means that the sampling rate can be relatively low, also in the order of samples per minute. A sample rate of $\frac{1}{60}$ Hz will be used as a base for further calculations.

The frequency with which the data transfer state is used might have an impact on the power consumption of the system as well as on the data redundancy and measurement data accessibility. Because of this three different cases are considered. These cases have both advantages and disadvantages.

1. **Case 1:** Transfer data after each sample taken. The advantage of this option is that it gives the most real time representation of data to the user. This option also gives the most redundancy as each sample is being transmitted immediately as well as being stored in internal flash memory. A disadvantage is that the Bluetooth connection needs to be established every time, which is costly in terms of power consumption as will be shown in Section 5.3.2.
2. **Case 2:** Transfer data after a set of samples has been acquired. This option offers a less real time representation of data to the user. It also offers less redundancy than the first option. The power consumption is decreased depending on the amount of samples acquired before transmitting them.
3. **Case 3:** Transfer data on user request. The user decides when data is transmitted to the external device. This will decrease power consumption for multi-day experiments as data is not transmitted unnecessarily during the night. This option offers the least amount of redundancy out of the three options, depending on the frequency with which the user requests data.

4.2 Proposed design

As low power consumption is crucial for this system, it is important to understand what effect the measurement routine has on the power consumption of the system. The first option described above will be used as reference when analysing the power consumption in Chapter 5. The third option however will be the preferred design as it offers greater flexibility for the user, and is expected to have a lower power consumption. In this section, the measurement routine and its states will be described.

4.2.1 Measuring routine

Figure 4.1 shows a state diagram of the routine. The routine starts by initialising the system after which it starts iterating through the sampling and idle states until a certain criterion is met, e.g. a number of samples is reached or a time limit is exceeded, which are both user definable. On user request the system will move to the data transfer state to transfer all data collected up until that point.

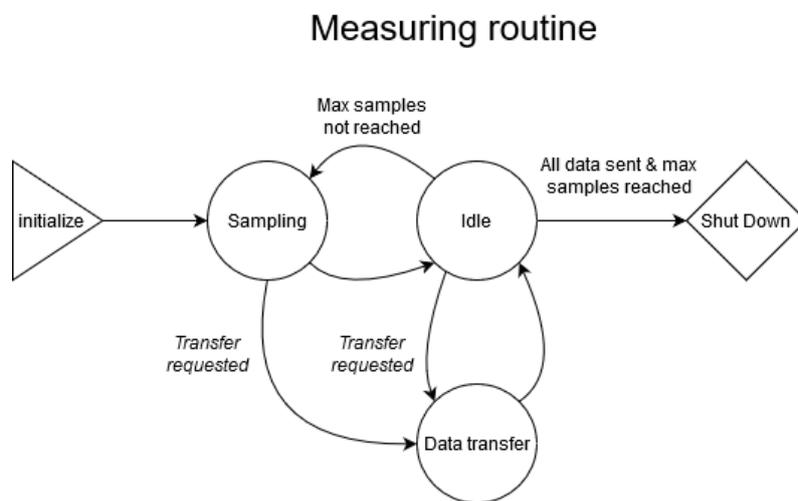


Figure 4.1: Measuring routine state diagram.

4.2.2 Initialisation

During the initialisation state the Flash memory is erased to ensure enough storage space for the new experiment. Experiment parameters are reset. A Bluetooth connection is set up to allow the user to send new experiment parameters and start the experiment. The system makes a transition to the sampling state.

4.2.3 Sampling

During the sampling state data of 4 parameters is acquired; the voltage output of the in-amp, the temperature, humidity and the Real Time Clock value. These values are then stored in flash memory. When all values are stored the routine enters the idle state. If during the sampling state a data transfer request is received, the next state will be the data transfer state in stead of the idle state.

ADC sampling

The Analog-to-Digital converter will convert its input voltage to a 12-bit digital value. In order to eliminate input signal noise multiple samples can be taken and the average of these samples can be calculated. This only works well if the input signal is very constant [17] which is the case for the experiments that this system will be used for.

Temperature and humidity sensor sampling

The data on the HTS211 can be accessed through I²C and contains the temperature (16 bits) and the humidity (16 bits) at that time. These values are read and stored to be used for calibrating the system and to give information on the environment the system is in.

RTC sampling for timestamp

To have the data make sense to the user the measurement points have to be put in chronological order. To this end the Real Time Clock is sampled and the value is stored. This value contains date and time information which can be useful to the user during experiments. The date needs to be stored as the system must support multi-day experiments. The stored RTC value is 32 bits long.

Storing data in flash memory

Combining the sampled data from the ADC, temperature and humidity sensor, and real time clock the total size of each measurement point is $12 + 32 + 32 = 76$ bits. The micro-controller includes 1 MByte of internal flash memory, divided into two 512 KByte banks. From one of the banks the program code is executed. The other bank can be used to store data. Each bank consists of 256 2 KByte pages and each page can store 256 64-bit words. The flash memory endurance is only 10 kilocycles per page [14]. These cycles could be burned through very fast if no proper software management is implemented.

One method for managing data storage on flash memory is EEPROM emulation. EEPROM emulation can be implemented in different ways and there is a method available for the STM32L4xx micro-controllers [18]. Some advantages of EEPROM emulation are; robustness against power failures and asynchronous resets; it implements a wear levelling algorithm to increase emulated EEPROM cycling capability; it does data granularity management, meaning it will combine small sized data elements into 32-bit data elements before storing these elements, optimising the amount of storage; it has good data recovery properties as it implements CRC.

However, this all comes at a cost. EEPROM emulation needs a significant amount of overhead. As this method requires two sets of pages of equal size to store one set worth of data this reduces the available storage to half its capacity. Per set of pages an even number of guard pages can be used to further improve endurance, by default this is 1 guard page. Per page there are 4 64-bit words used as page header, so 252 data elements can be stored per page. This means that for 256 2 KByte pages only $(\frac{256}{2} - 1) \cdot 252 = 32004$ data elements can be stored. Each 64-bit data element consists of a 16-bit virtual address, 16-bit CRC, and 32-bit data value. So in order to store one measurement point three 64-bit words are required, leaving room for $32004/3 = 10668$ measurement points on the internal flash memory. This is a worst-case scenario as in reality multiple 12-bit ADC values can be stored in the 32-bit data element. When assuming a sample rate of 1 sample per minute a maximum experiment duration of roughly 7.4 days can be achieved. Decreasing the sampling rate will increase this duration. This meets the requirement that the parameter analyser should support experiments of 6 days.

4.2.4 Data transfer

When a transfer request is sent by the user while in idle state, the system is interrupted and enters the data transfer state. In this state a Bluetooth connection between the system and the external device is established. All stored values until that point, which have not yet been transferred will be transferred. If the requests parameter changes, this will be executed. If not it returns back into the idle state. This data request could occur while the system is in the sampling state. When this happens the system finishes sampling and storing and instead of entering the idle state it will immediately enter the data transfer state.

4.2.5 Idle

The idle state is the state the routine will spend most time in. When entering the idle state a timer will start counting. When it reaches the sampling time set by the user it will trigger an interrupt. On this interrupt the system will check whether the set amount of samples is reached. If so, the system will transfer any

data that might not have been transferred yet and will then move to the shutdown state. If the set amount of samples has not been reached the system will move to the sampling state to continue iterating.

4.2.6 Shutdown

The system will enter the shutdown mode when the experiment is done. In this state all unneeded peripherals are shut down and parameters are reset. The system will stay in this state until it loses power or it is reset after which it returns to the initialisation state.

Chapter 5

Power budget analysis

In order to find a proper power source for the system an estimation has to be made on the power consumption of the system. Moreover, the power consumption will have an impact on the heat dissipation of the system which will impact the environment in which the system is used. To determine the system's expected power consumption an analysis will be performed on the individual system components.

5.1 Sensor

The sensor is a floating-gate FET-based ion sensor. The power consumption depends on the bias voltages and the drain current flowing through the sensor. Simulations done by Group C [5] show that at $V_{DS} = 1.65$ V and $V_{CG} = 3.0$ V, the drain current is $I_D = 360 \mu\text{A}$. Which results in a power consumption of $594 \mu\text{W}$.

5.2 Sensing board

The sensing board consists of two parts, an instrumentation amplifier and a tuning circuit.

5.2.1 Instrumentation amplifier

In order to simulate the current consumption for the instrumentation amplifier the software Advanced Design System (ADS) is used. The amplifier is being fed with a 5 V supply voltage and the supply current is measured using a current probe as can be seen in Figure A.1. A common mode voltage V_{common} is applied to the negative input. This common mode voltage equals the drain-source voltage set by the DAC which is $V_{DS} = 1.65$ V (see Section 3.3.1). The differential voltage between the positive and negative inputs of the amplifier is swepted from 0 to 50 mV which covers the whole output voltage range when the supply voltage is 5 V as the amplifier gain is 100 [4]. The resulting supply current is shown in Figure 5.1. The plot shows the operating region from 0 V to 3.3 V and a saturation region for a differential voltage larger than 3.3 V. In fact, when looking at the in-amp output voltage in Figure A.2 a similar behaviour is observed. Due to the common mode voltage of 1.65 V the output voltage is limited to twice that value. When plotting the output voltage for multiple values of V_{common} in Figure A.3 its effect can be observed. The average current consumption within the operating region is 2.7 mA.

5.2.2 Tuning circuit

The tuning circuit adjusts the sensitivity of the in-Amp. To do this it uses an I/O expander (MCP23s17), signal relays (P2 Relay V23079), transistors and resistors. Since the tuning circuit only consumes power when it is set in the initialisation stage, the power consumption is negligible in comparison to the instrumentation amplifier and will not be included in the estimations.

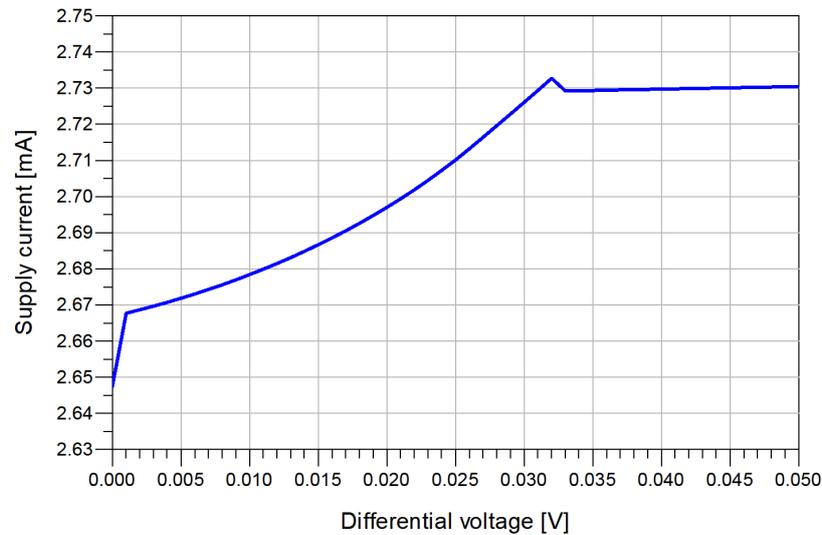


Figure 5.1: The consumed current plotted against the differential input voltage for the instrumentation amplifier.

5.3 Discovery board

In order to make an estimation of the power consumption for the Discovery board the consumption of the measurement routine states (see chapter 4) and different Bluetooth module states must be determined. STMicroelectronics provides software to estimate the power consumption of the micro-controller (STM32CubeIDE: Power Consumption Calculator) and Bluetooth module (BlueNRG Current Consumption Estimator). Using this software the power consumption of the different states and operations can be estimated.

5.3.1 Micro-controller (STM32L4)

Power modes

The micro-controller can operate at different clock frequencies and in different power modes. The power modes available are:

1. **(Low-power) run mode**
In run mode all peripherals are active. This is the most power consuming mode. To lower the power consumption the low-power run (LPRun) mode can be used in which the system clock frequency is limited to 2 MHz and a lower-power regulator is used. Peripherals use a separate clock.
2. **(Low-power) Sleep mode**
In sleep mode only the system clock is stopped. It can be woken up with interrupts/events. All peripherals continue to operate. Like the LPRun mode, the low-power sleep (LPSleep) mode supports a lower system clock frequency and uses the low-power regulator.
3. **Stop modes**
Different stop modes are available which have the lowest power consumption while still retaining content in memory and in registers. Depending on the stop mode (1, 2 or 3) a certain amount of peripherals are active.
4. **Standby mode**
In standby mode the CPU and all peripherals are powered off. The SRAM memory is still on. The 'brown out reset (BOR)' and RTC keep active to make sure operation can continue on wake up.

5. Shutdown mode

In Shutdown mode the micro-controller fully stops. SRAM is powered off. Only the RTC stays active.

What power mode is used is dependent on what peripherals need to be active and what clock frequency will be used. As this system needs to be low power consuming the clock frequency is preferably as low as possible depending on the peripherals' required clock frequencies. The LPRun and LPSleep modes allow for a clock frequency of 2 MHz down to 100 KHz. The minimum clock frequency for the ADC is 1.4 MHz. Since there is only choice between 1 and 2 MHz around that frequency for the system clock, a system clock frequency of 2 MHz is chosen. For this reason the LPRun mode will be used for all active states (Initialisation, Sampling, Data transfer). Which power mode will be used for the passive (Idle) state will depend mainly on the required peripherals during this state.

Table 5.1 shows which peripherals need to be active in which states. Since the DAC needs to be active in the idle state only two modes are suitable for this state; the LPSleep and Stop1 modes [19, Table 4]. As SPI needs to be active in order to receive transfer requests the Stop1 mode falls short as this mode does not support SPI. LPSleep will thus be used during the idle state. To further reduce the power consumption during LPSleep the system clock speed is reduced to 100 kHz.

Measurement state power consumption

The current consumption I_{state} for each state in the measurement routine can be calculated by summing the current consumption for the CPU and all active peripherals for a given power mode and clock frequency. The peripherals which need to be active in each state are listed in Table 5.1.

As stated earlier the current consumption can be calculated accurately by using the Power Consumption Calculator provided by STMicroelectronics. A full description on this method can be found in Appendix B. The resulting I_{state} values are shown in Table 5.1. The average current consumption I_{avg} can be calculated as

$$I_{avg} = \frac{\sum(I_{state} \cdot T_{state})}{T_{total}} \quad (5.1)$$

where T_{state} is a state duration and T_{total} is the sum of all T_{state} . The average current consumption will be useful when comparing the different measurement routines described in Chapter 4 and can be used to determine the total amount of current consumed over the course of an experiment, which will be a leading factor in determining the necessary battery capacity.

5.3.2 Bluetooth module (SPBTLE-RF)

In the datasheet for this module [15, Table 3] some results can be found on the current consumption during different operations. The maximum current consumed is 7.72 mA, which will only occur during the set-up of a connection. Using the 'BlueNRG Current Consumption Estimator' an estimation of the duration time and power consumption can be made with the specific parameters of the system (see Appendix C). In the initialisation and data transfer state information is communicated between the system and the computer. In table 5.2 the results of the estimation can be found.

From this table it can be seen that when a connection is made 3.79 mA of current is consumed and depending on the amount of data 0.61 mA of current is consumed per package. Setting up a connection is relatively expensive which means it is more efficient to send multiple packages in one connection instead of sending packages one by one.

5.3.3 Measurement state duration

The duration for each state T_{state} must be determined to allow for accurate estimations on power consumption. The results from this section are summarised in Table 5.1.

Table 5.1: MCU's power consumption for each measuring routine state, assuming a 3.6 V supply voltage.

State	Active peripherals	Power mode	T_{state}	I_{state}
Initialisation	Flash memory SPI	LPRun	3.109s	310.8 μ A
Sampling	ADC DAC x2 Flash memory I2C RTC Timer	LPRun	12.16 ms	1.020 mA
Idle	DAC x2 SPI RTC Timer	LPSleep	60 s	651.3 μ A
Data transfer	DAC x2 SPI Flash memory RTC Timer	LPRun	≥ 2.461 ms*	991.4 μ A
Shutdown		LPRun, shutdown	≥ 1 ms	27.6 μ A

* The duration of data transfer state for x measurement points is:

$$T_{state} = 1.35 + x \cdot 0.61 + x \cdot 0.501 \text{ ms (Section 5.3.3)}$$

Table 5.2: Bluetooth module's power consumption for each Bluetooth state, assuming a 3.6 V supply voltage.

Operation	Power mode	Measurement state	Duration	Current consumption
Waiting	Sleep	Sampling Idle		2.4 μ A
Connecting	Active	Initialisation Data Transfer	1.35 ms	3.79 mA
Data transfer (1 package, 12 bytes)	Active	Initialisation Data Transfer	0.61 ms	0.41 mA

Initialisation state

The time it takes to initialise and erase the Flash memory will be maximal when all pages need to be erased [18, Table 6]. With a maximum page erase time of $t_{ERASE,max} = 24.47$ ms [14, Table 53] and 127 flash EEPROM pages (128 pages – 1 guard page) the maximum erase time will be $127 \cdot 24.47 = 3.108$ s.

During initialisation the GUI will send at least the SET_RESISTOR command and twice the SET_BIAS command, once for each bias voltage. Using the lengths defined in Section 3.5 the package length will then be 5 bytes. The time it takes to receive this package is 1.45 ms as the simulation in Figure C.3 shows.

The total duration of the initialisation state thus is $T_{state} = 3.109$ s.

Sampling state

The amount of time it takes to perform a single conversion using a SAR ADC is $t_{CONV} = t_s + 12.5$ where t_s is the sampling time and the 12.5 comes from the fact that it is a 12-bit ADC [14, Table 66]. t_s can be set according to the time it takes for the capacitor present in the sample-and-hold circuit to fully charge. t_s needs to be large enough for the capacitor to fully charge, otherwise a lower voltage will be sampled than the actual voltage. However, since the input voltage approaches a DC voltage this capacitor will always be charged and the voltage across this capacitor will follow the input voltage. Thus, any value for t_s can

be chosen. $t_s = 2.5$ clock cycles is chosen, which is the minimum value for t_s [14, Table 66], in order to minimise the duration of sampling. As was mentioned in 4.2.3 input signal noise can be eliminated by taking multiple samples and averaging their values. A value of 50 samples is chosen heuristically. The total time to sample the ADC is then $50 \cdot (2.5 + 12.5) = 750$ clock cycles, which is $750/(2 \cdot 10^6) = 375 \mu\text{s}$.

Following the I²C protocol it takes 76 bits to read the temperature and humidity data (32 bits of data and 44 bits of addressing and acknowledgements), using I²C Fast Mode+ available on the micro-controller [14] a clock frequency of 1 MHz can be reached, meaning that the sampling takes $76/(1 \cdot 10^6) = 76 \mu\text{s}$.

The Flash memory write speed for a single 64-bit word is $97 \mu\text{s}$ at a 80 MHz system clock [18, Table 6]. For a system clock of 2 MHz the single word write speed is $97 \cdot \frac{80}{2} = 3.880$ ms. In reality this time will be shorter as the Flash memory latency will be lower for lower system clock frequencies [19, Figure 4]. Every sample consists of three 64-bit words as explained in Section 4.2.3 so the total write time will be $3 \cdot 3.880 = 11.64$ ms.

The total duration for the sampling state then becomes $T_{state} = 0.375 + 0.078 + 11.64 = 12.16$ ms.

Data transfer state

During the data transfer state the data has to be read from Flash memory. The read time for the last stored value is $8.9 \mu\text{s}$ and the read time for the first stored value is $331 \mu\text{s}$ [18, Table 6]. An average read time of $(8.9 + 331)/2 = 167 \mu\text{s}$ for a single data element will be assumed, meaning that a single measurement point is read in $3 \cdot 167 = 501 \mu\text{s}$.

Table 5.2 shows that setting up the Bluetooth connection takes 1.35 ms and transferring a single 12 byte data package takes 0.61 ms. Transferring a single measurement point will then take $1.35 + 0.61 = 1.96$ ms.

The total duration of the data transfer state for a single measurement point becomes $T_{state} = 2.461$ ms.

5.4 Current consumption for the measurement routine

As mentioned in chapter 4 there are multiple options for measuring the required data. The 3 options introduced were; case 1: transfer data after each sample; case 2: transfer data after a set of samples have been acquired; case 3: transfer data on user request. In this section an estimation will be made on the consumption of case 1 and on case 2. The following testing conditions will be applied:

- 720 samples are taken with a sample rate of 1 sample per minute, which takes approximately 12 hours
- The sensor and instrumentation amplifier are always on.
- The duration of the initialisation and shutdown state is set to 1 minute.

5.4.1 Test case 1: Instant data transfer

This options starts with initialising the system after which it starts sampling and transferring data. Since in this routine data is transferred immediately after sampling, it moves from the sampling state to the data transfer state. When this is done it goes into the idle state to wait 60 seconds before it repeats these steps. It keeps doing this for 720 iterations, after which the system moves to the shutdown state. Figure 5.2 shows a state diagram for this routine.

Using the current consumption per state as listed in Table 5.1 the total current consumption can be determined which can be used to calculate the required battery capacity for the system to work for at least the required 6 days. Using the information about the measurement routine states, the Bluetooth modes, the determined current consumption of the instrumentation amplifier and the sensor an overview of all total state durations $T_{state,total}$ and I_{state} is made in Table 5.3. Using Equation 5.2 the current consumption (in Ah) per state can be calculated.

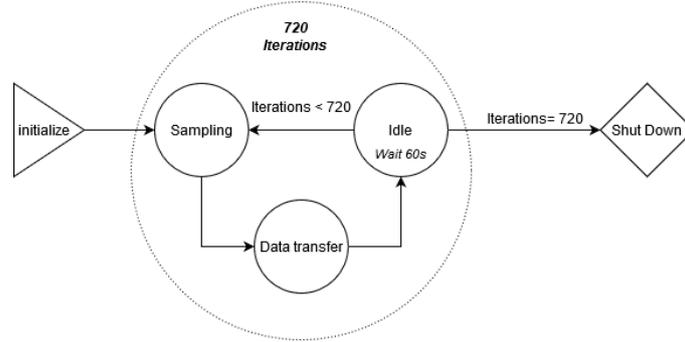


Figure 5.2: Test case with instant data transfer

$$\text{Current consumption} = \frac{T_{state,total} \cdot I_{state}}{3600} \quad (5.2)$$

MCU state	# Iterations	T_{state}	$T_{state,total}$	I_{state}	Current consumption
Initialisation	1	60 s	60 s	0.311 mA	5.18 μ Ah
Sampling	720	12.16 ms	8.76 s	1.02 mA	2.48 μ Ah
Idle	720	60 s	43,200 s	0.651 mA	7.82 mAh
Data transfer	720	2.461 ms	1.77 s	0.99 mA	0.49 μ Ah
Shutdown	1	60 s	60 s	27.6 μ A	0.46 μ Ah
Bluetooth module	# Iterations	T_{state}	$T_{state,total}$	I_{state}	Current consumption
Waiting	720	60.012 s	43,200 s	2.40 μ A	28.81 μ Ah
Connecting	721	1.35 ms	0.97 s	3.79 mA	1.02 μ Ah
Data transfer	721	0.61 ms	439.81 ms	0.41 mA	50.09 nAh
Other components			$T_{state,total}$	I	Current consumption
Sensor			12 hours	0.36 mA	4.32 mAh
Instrumentation amplifier			12 hours	2.70 mA	32.40 mAh
Temp. and Humidity sensor			12 hours	25 μ A	0.3 mAh
Average current consumption $I_{avg} = 3.73$ mA					
Total test case current consumption: 44.94 mAh					

Table 5.3: Current consumption for test case instant data transfer

Using Equation 5.1 the average consumption is determined to be $I_{avg} = 3.73$ mA, so for the system to work for the required 6 days as stated in the Chapter 2 the minimum required battery capacity is $6 \cdot 24 \cdot 3.73$ mA = 537.12 mAh.

What is interesting to observe is that although the Idle state consumes less current than some other states, it is the largest contributor to the current consumption of all the states due to the MCU spending the most time in it. The instrumentation amplifier is the largest current consumer overall, due to it always being active.

Regarding Bluetooth; the current needed to set up a connection is independent on the amount of packages send. This means that it is more efficient to send a set of data instead of sending packages individually, but since the contribution to the consumption of Bluetooth is quite low in comparison to the MCU, this effect is negligible.

5.4.2 Test case 2: data transfer after fixed amount of samples or on user request

In this test case it is assumed that during the progress of the experiment it is not necessary to see the results immediately. This means that less time has to be spent in the data transfer state and less power is dissipated through the use of Bluetooth. With regards to the frequency with which data is transferred this test case is considered similar to the case in which data is sent on request.

In this test case the same calculation will be made for the same amount of data points (still a sampling rate of once every minute), but instead of sending data immediately the samples are sent after 60 minutes which means the data will be sent 12 times with each time having 60 packages.

This routine starts in the initialising state, after which sampling starts until 60 samples have been taken. The system then moves to the data transfer state to send the 60 samples. It continues until 720 samples have been taken and sent after which the system goes to the shutdown state. Figure 5.3 shows a state diagram for this routine.

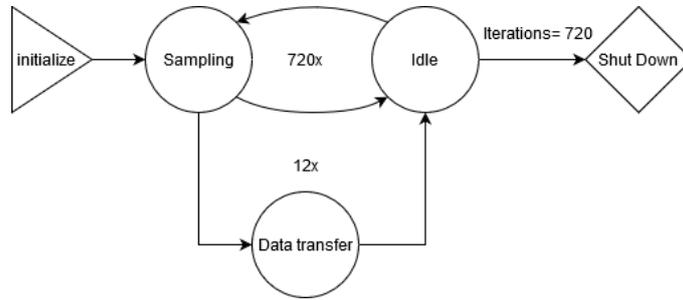


Figure 5.3: Test case with data transfer after 60 minutes

Again using the information about the measurement routine states, the Bluetooth modes, the determined current consumption for the instrumentation amplifier and sensor $T_{state,total}$, I_{state} and the total current consumption are given in Table 5.4.

Table 5.4: Current consumption for test case instant data transfer

MCU state	# Iterations	T_{state}	$T_{state,total}$	I_{state}	Current consumption
Initialisation	1	60 s	60 s	0.311 mA	5.18 μ Ah
Sampling	720	12.16 ms	8.76 s	1.02 mA	2.48 μ Ah
Idle	720	60 s	43,200 s	0.651 mA	7.82 mAh
Data transfer	12	2.461 ms	29.50 ms	0.99 mA	8.13 nAh
Shutdown	1	60 s	60 s	27.6 μ A	0.46 μ Ah
Bluetooth module	# Iterations	T_{state}	$T_{state,total}$	I_{state}	Current consumption
Waiting	720	60.012 s	43,200 s	2.40 μ A	28.81 μ Ah
Connecting	13	1.35 ms	17.60 ms	3.79 mA	50.10 nAh
Data transfer	721	0.61 ms	439.81 ms	0.41 mA	50.09 nAh
Other components			$T_{state,total}$	I	Current consumption
Sensor			12 hours	0.36 mA	4.32 mAh
Instrumentation amplifier			12 hours	2.70 mA	32.40 mAh
Temp. and Humidity sensor			12 hours	25 μ A	0.3 mAh
Average current consumption $I_{avg} = 3.73$ mA					
Total test case current consumption: 44.94 mAh					

Using Equation 5.1 the average consumption is determined to be $I_{avg} = 3.73$ mA, which is equal to the result from test case 1, which means that the frequency with which data is transferred has no significant impact on the current consumption of the system.

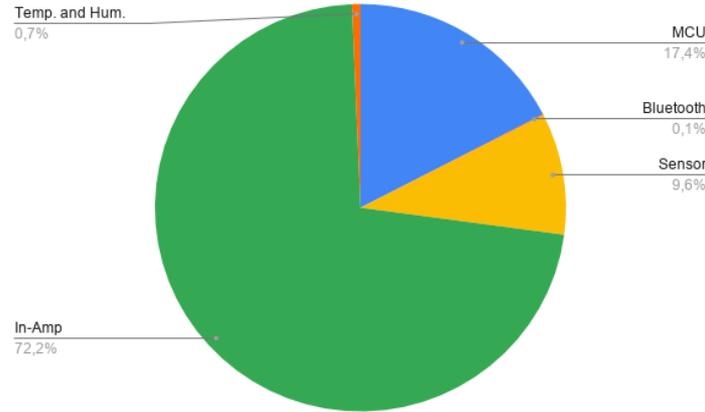


Figure 5.4: Visualisation of the contribution of individual system components to total test case current consumption for test case 1.

5.4.3 Comparison

From the test results the effect of changing the amount of data transfer states seems insignificant. This can be explained by the fact that by far the largest contributor to the current consumption is the instrumentation amplifier. This is visualised in Figure 5.4 which shows the individual component contribution to the total test case current consumption in percentages. As the in-amp is such a large consumer in the system it would be interesting to see what the effect would be of switching it off in between sampling. This will be done in a third test case described in Section 5.4.4.

5.4.4 Test case 3: switch off the in-amp in between sampling

For this test case the in-amp will be switched off in between sampling. An on-time of $T_{on} = 12.16$ ms is chosen which is equal to the MCU's sampling state time (Table 5.2). The same test conditions as for test case 1 are used, so the in-amp will be on for 720 iteration. Table 5.5 shows the average and total current consumption for each component which are calculated from Table 5.3 except for the in-amp. When looking at the visualisation of the contribution to current consumption in Figure 5.5 and from Table 5.5 it can be seen that the in-amp is the lowest contributor to current consumption and the MCU the largest. The average current consumption for the total system is now $I_{avg} = 1.04$ mA, which is a significant decrease from the $I_{avg} = 3.73$ mA from test case 1 and 2.

Table 5.5: The average and total current consumption per component.

Component	$I_{avg,component}$	Total current consumption
MCU	0.65 mA	7.82 mAh
Bluetooth module	2.48 μ A	29.9 μ Ah
Sensor	0.36 mA	4.32 mAh
Instrumentation amplifier	0.90 μ A	10.8 μ Ah
Temp. and humidity sensor	25.0 μ A	0.30 mAh
Average current consumption $I_{avg} = 1.04$ mA		
Total test case current consumption: 12.50 mAh		

5.4.5 Conclusion

From the test cases it can be concluded that the frequency with which data is transferred to the external device does not result in a significantly different current consumption. Changing the time the in-amp is

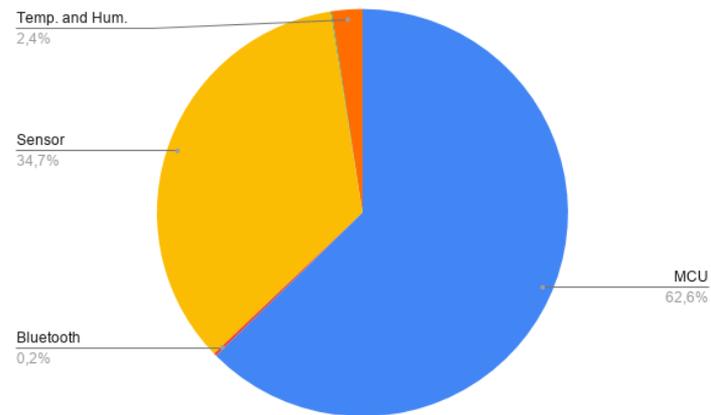


Figure 5.5: Visualisation of the contribution of individual system components to total test case current consumption for test case 3.

switched on, however, does have a significant effect on current consumption.

Chapter 6

Power management and heat dissipation

6.1 Requirements

In the Program of requirements (Chapter 2) a set of requirements is given for the system which influences the power management design. The following requirements have to be taken into consideration:

- **The system is to be used inside microbiological incubators at a temperature of 37°C.** This means that the battery and components should be able to work at higher ambient temperatures.
- **The system must be able to fit inside all microbiological incubators.** This impacts the size the system and the battery can have.
- **Must be battery powered for 6 days.** A battery must be used and it should have enough capacity to power the system for 6 days. No external power cords should be needed.

Considering these requirements a power design is made and a power source is chosen.

6.2 Power source

A battery will be used as power source for the entire system. This will eliminate the need to run power cords from inside the incubator to the outside, however, the main reason is to eliminate any 50 Hz noise and interference caused by a power adapter connected to the power grid.

6.2.1 Battery

There are several options available concerning the battery technology used. The most used and commercially available options are lithium-ion batteries. A specific lithium-ion battery is most suited for this system, which is a lithium-ion polymer (Li-Po) battery. A summary of the beneficial properties of lithium-ion polymer batteries [20]:

- Comes in varying sizes.
- The properties of polymer electrolyte, such as leakage prevention, fire resistance, and automatic extinction can ensure safety.
- The lamination structure of the electrode and electrolyte has high reliability for impact and vibration.
- The electrode interval can be narrowed by using a polymer electrolyte as a separator. As a result, there will be an increase in capacity.

- The improvement of cyclicity is expected due to the decrease in the inter-facial impedance of the electrode and the electrolyte.

Especially the safety aspect of the Li-Po battery is important, as the system will be used in a sensitive environment. Another important aspect of the battery is the temperature in which it operates. The system will be used in high temperature environments which results in a higher battery capacity during discharging, however, it results in a shorter lifespan [11]. The flexibility in design and size makes it suitable for application in this system, because it helps minimise the size.

For charging a Li-Po battery responsibly a charger IC is needed. Furthermore, to achieve the required voltage to power the development board and sensing board (5 V) a DC-DC converter is needed to boost the voltage level of the Li-Po battery. Subsection 6.2.2 and 6.2.3 give an example of an IC that could be used to achieve this.

6.2.2 DC-DC converter

For the power source to be able to supply the required voltages with the given battery a boost converter must be used. The battery voltage of 3.7 V is boosted to 5.2 V which is enough to power the sensing board. An input voltage of 1.8 – 3.7V [21] is needed for the DC-DC converter to function. What has to be taken into consideration is that boosting the voltage results in a higher flow of current and current consumption from the battery as the power is conserved. The extra current needed for boosting is based on the supply voltage difference and the 96% efficiency [21] of the conversion. Equation 6.1 gives the factor of extra capacity needed.

$$\text{Capacity increase factor} = \frac{V_{out}}{V_{in} \cdot \text{efficiency}} = \frac{5.2}{3.7 \cdot 0.96} = 1.46 \quad (6.1)$$

The current consumption of the converter IC (TPS61090 [21]) is equal to the quiescent current of the IC, this current is $25\mu\text{A}$ on the battery side and $20\mu\text{A}$ on the output side. This makes for a current consumption of $45\mu\text{A}$. This value has to be taken into account when calculating the total required battery capacity.

6.2.3 Charging chip

The battery can be charged through the use of a charger IC (MCP73831 [22]). The purpose of this IC is to charge lithium batteries using an algorithm that supplies a constant current and voltage and terminates the charging when the battery is full to extend battery life. The consumption of the charging IC is not important for the battery capacity of the system, because it only consumes power when connected to the mains.

6.2.4 Battery capacity

Using all the information from chapter 5 and the information in this section the minimum required battery capacity can be determined. As determined in the power analysis the system requires a battery with at least 537.12 mAh. The converter circuit consumes an additional $25 \cdot 10^{-6} \cdot 6 \cdot 24 = 1 \mu\text{Ah}$ which is negligible. The absolute minimum capacity is $537.12 \cdot 10^{-3} \cdot 1.46 = 784.20 \text{ mAh}$.

Li-Po batteries of 3.7 V and 2 Ah are readily available at a decent price point [23]. Suppose this battery would be used it would be interesting to know what the maximum experiment run time can be. Assuming an average current consumption of $I_{avg} = 3.73 \text{ mA}$ and taking into account the capacity increase factor the max run time will be $\frac{2000}{3.73 \cdot 1.46} = 367.2$ hours, or 15.3 days. To be able to measure for this long the sampling rate would have to be decreased. Assuming the amount of measurement points that can be stored is 10668 the sampling rate would have to be 1 sample every 2.1 minutes.

When using the resulting average current $I_{avg} = 1.04 \text{ mA}$ from test case 3, the experiment run time would be $\frac{2000}{1.04 \cdot 1.46} = 1317.2$ hours, or 54.9 days. The maximum sample rate is than 1 sample every 7.4 minutes.

6.3 Power design

Now that the power source is known and all the parts of the system are present an overview of the power management of the system can be made and is shown in figure 6.1. There are 4 fixed voltages present in the system; 3.3, 3.6, 3.7 and 5 V. The development board and sensing board are both supplied 5 V. Inside the power regulators of the development board this 5 V is converted into 3.6 V which powers the micro-controller and temperature sensor, and 3.3 V to power the Bluetooth module. The battery has a voltage of 3.7 V and a capacity of at least 784.20 mAh. V_{DS} is the supply voltage and V_{CG} the bias voltage for the sensor. Both voltages are supplied by the Digital-to-Analog converter in the micro-controller.

The DC-DC conversion is needed to achieve the the right supply voltage for the sensing board and the development board, however this is not very efficient. In a future design a custom power supply designed specifically for this implementation would be the best option.

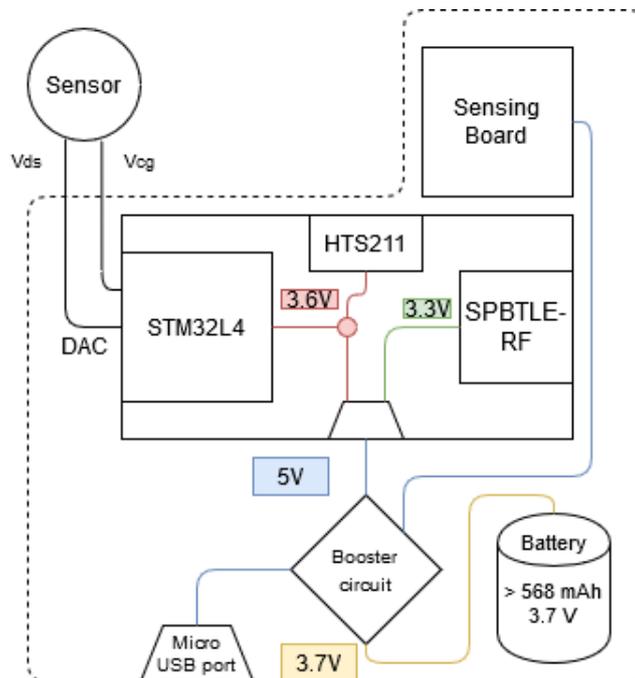


Figure 6.1: Overview of supply voltages and power source

6.4 Heat

With the test cases described in Section 5.4 it is shown that there are several methods for manipulating the current consumption of the system. This naturally results in different amounts of power dissipated by the system as heat into the surrounding environment. In what way this relates to a temperature increase in an enclosure depends on many factors of which some are heat transfer coefficients, enclosure volume, and enclosure material. This is too complex of a problem to solve for this project.

Chapter 7

Discussion

The main result of this project is the current consumption of the system for different implementations of the measurement routine. After calculating the current consumption for the individual parts of the system an estimation was made for two test cases in which was tested what the effect of changing the amount of data transfer occurrences is. Decided was to include a third test case to test the effect of switching off the instrumentation amplifier in between sampling, as the first two test cases did not show significantly different results. This resulted in the following:

- The biggest contributor to the current consumption of the system is the instrumentation amplifier.
- Changing the amount of time the instrumentation amplifier is switched on has a significant impact on the power consumed and dissipated. The average current consumption of the system is 3.73 mA when the instrumentation amplifier is switched on for the entire duration of the experiment. The average current consumption of the system is 1.04 mA when the instrumentation amplifier is switched on only during sampling decreasing the consumption dramatically.
- The amount of data transfer occurrences does not contribute significantly to changes in current consumption.

Besides these results the battery technology has been selected to be lithium-ion polymer and the minimum battery capacity required for 6 days of continuous operation has been determined for the case in which the instrumentation amplifier is always switched on. This capacity is 784.20 mAh. For the case in which the in-amp is switched off in between sampling this required capacity will be significantly less. The maximum sampling rate for these cases has been determined to be 1 sample every 2.1 minutes for the former case and 1 sample every 7.4 minutes for the latter.

These results are estimations based on simulations of the designed system and on the available manufacturer specifications. Besides that, due to time constraints and the fact that no physical demonstrator was allowed, the analyses performed don't go into much detail. The results seem very reasonable and some conclusions could be drawn from them, however, in order to validate the results the system would have to be implemented, in both hardware and software, and tested.

It is important to note that all current consumption analyses are performed with the output buffers of the DACs enabled. The analyses were already completed at the time it was discovered that the buffers were not required. Disabling the buffers will have an impact on the consumption of the MCU. However, the analysis results still hold as disabling the buffers would only have increased the relative contribution of the instrumentation amplifier to the current consumption.

Chapter 8

Conclusion

In conclusion, during this project a design for the analyser has been developed together with a measurement routine for acquiring data from the FGFET sensor correctly. A power consumption analysis has been performed on the system by simulating the current consumption of individual components. Using these results the consumption of the different measurement routines has been determined and compared. In the end different methods of affecting the consumption have been shown. The estimated consumption results were used to determine the required battery capacity to be able to meet requirements.

8.1 Recommendations

There are a couple of recommendations that would benefit a future version of this analyser:

1. A lot can be gained by developing a dedicated PCB for the analyser. Some unnecessary power conversion is done in the current design due to the development board used, this could be optimised for.
2. Because whether the operational amplifier is switched on during the course of the entire experiment or it is switched off in between sampling has such a large effect on power consumption, it is recommended to look into possibilities to switch off this component in between sampling.
3. Calibration could be included to compensate for temperature and humidity differences during sampling.

8.2 Future work

Before the system can be implemented there is still some work to be done. An in-depth analysis needs to be made on ADC and DAC noise performance. Also the effects of high temperature on ADC and DAC performance must be mapped out. In order to gain better understanding in the thermal properties of the system more accurate power consumption simulations and measurements as well as proper heat flow simulations should be performed.

Bibliography

- [1] J. van den Eijnden-van Raaij M. Mastrangeli, S. Millet. Organ-on-chip in development: Towards a roadmap for organs-on-chip. *ALTEX*, pp. 650-668, 2019. Available: 10.14573/altex.1908271.
- [2] Poster or Oral from TU Delft. Cmos-compatible charge-modulated fgfet sensor for organ-on-chip applications. *Biosensors and Bioreactors*, 2020.
- [3] H. Aydogmus et al. in preparation (2020).
- [4] Y. Wesseling J. Noordhoek. Portable parameter analyser for organs-on-chip: Calibration & gui. Bachelor's thesis, Delft University of Technology, June 2020.
- [5] M. van der Maas K. dos Reis Vezo. Portable parameter analyser for organs-on-chip: Charge sensor model. Bachelor's thesis, Delft University of Technology, June 2020.
- [6] Hewlett Packard. *Operation Manual, HP 4145B Semiconductor Parameter Analyzer*, April 1989. Manual Part No. 04145-90310.
- [7] M. P. Flynn J. Fredenburg. Adc trends and impact on sar adc architecture and analysis. *IEEE Custom Integrated Circuits Conference (CICC)*, 2015. DOI: 10.1109/CICC.2015.7338380.
- [8] K. Mafinezhad W. A. Serdijn M. Saberi, R. Lotfi. Analysis of power consumption and linearity in capacitive digital-to-analog converters used in successive approximation adcs. *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS*, 58(8), 2011. pp. 1736-1748.
- [9] M. L. Petersen K. Stentoft. Electronics in harsh environments - product verification and validation. *Annual Reliability and Maintainability Symposium*, 2018.
- [10] M. Stendahl Jellesen R. Ambat H. Conseil, V. Chakravarthy Gudla. Humidity build-up in a typical electronic enclosure exposed to cycling conditions and effect on corrosion reliability. *IEEE Transactions on components, packaging and manufacturing technology*, 6(9), 2016. pp. 1379 - 1388, DOI: 10.1109/TCPMT.2016.2590779.
- [11] M. Pecht F. Leng, C. Ming Tan. Effect of temperature on the aging rate of li ion battery operating above room temperature. *Sci. Rep.* 5, 12967, 2015. DOI: 10.1038/srep12967.
- [12] Microchip. *16-Bit I/O Expander with Serial Interface*, 2005. Rev. C.
- [13] STMicroelectronics. *Discovery kit for IoT node, multi-channel communication with STM32L4*, October 2019. Rev. 5.
- [14] STMicroelectronics. *Datasheet, STM32L475xx, Ultra-low-power Arm® Cortex®-M4 32-bit MCU+FPU, 100DMIPS, up to 1MB Flash, 128 KB SRAM, USB OTG FS, analog, audio*, March 2019. DS10969 Rev. 5.
- [15] STMicroelectronics. *SPBTLE-RF, Very low power network processor module for Bluetooth® low energy v4.1*, October 2017. DocID027851 Rev. 10.

-
- [16] STMicroelectronics. *Capacitive digital sensor for relative humidity and temperature*, August 2016. Rev. 4.
- [17] STMicroelectronics. *Application note, How to get the best ADC accuracy in STM32 microcontrollers*, November 2019. AN2834 Rev. 4.
- [18] STMicroelectronics. *Application note, EEPROM emulation techniques and software for STM32L4 and STM32L4+ Series microcontrollers*, September 2018. AN4894 Rev. 2.
- [19] STMicroelectronics. *Application note, Optimizing power and performance with STM32L4 Series microcontrollers*, December 2019. AN4746 Rev. 3.
- [20] A. Kozawa M. Yoshio, R.j. Brodd. *lithium-Ion Batteries*, chapter 21, pages 413–426. Springer Science+Business Media. LLC, 2009.
- [21] Texas Instruments. *TPS6109x Synchronous Boost Converter With 2-A Switch*, June 2003. Rev. C.
- [22] Microchip. *Miniature Single-Cell, Fully Integrated Li-Ion, Li-Polymer Charge Management Controllers*, november 2005. Rev. H.
- [23] RS Online. Rs pro, 3.7v, lithium polymer rechargeable battery, 2ah. <https://uk.rs-online.com/web/p/speciality-size-rechargeable-batteries/1251266/>.

Appendices

Appendix A

In-amp current consumption simulation

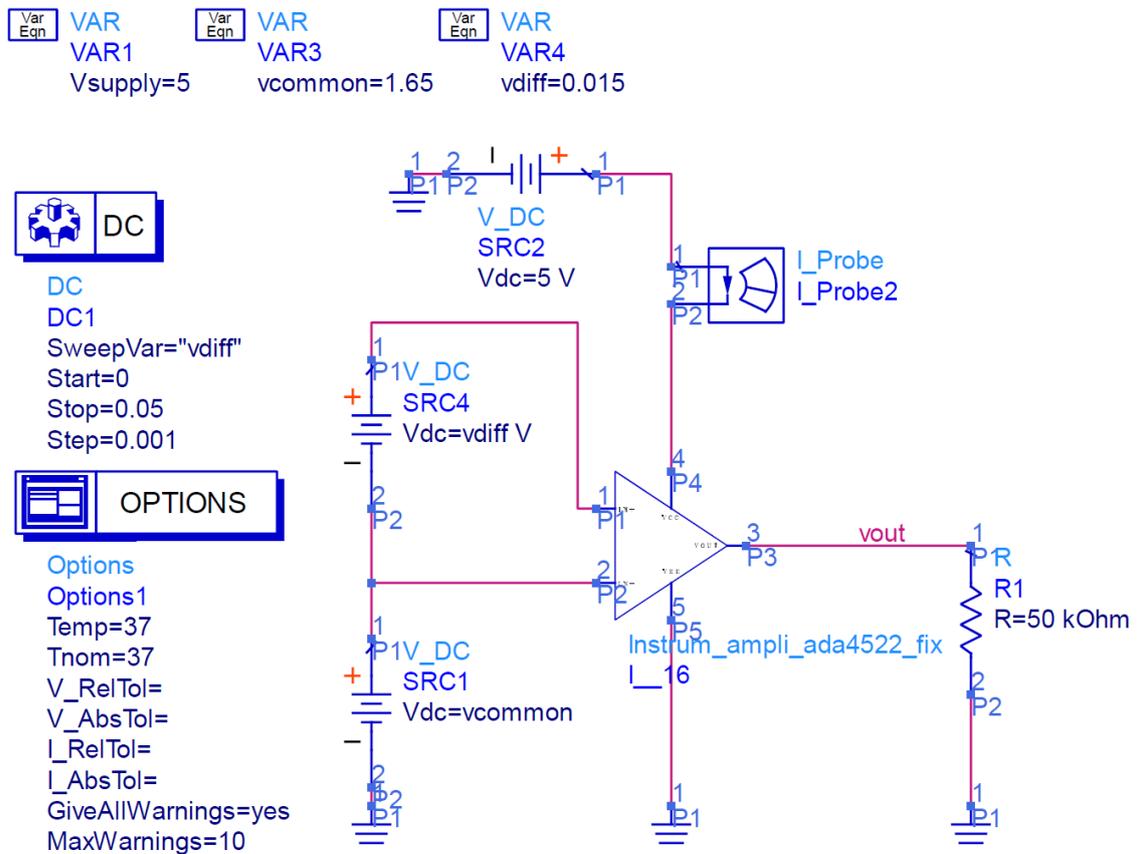


Figure A.1: Simulation schematics and parameters for the in-amp current consumption and output voltage simulation.

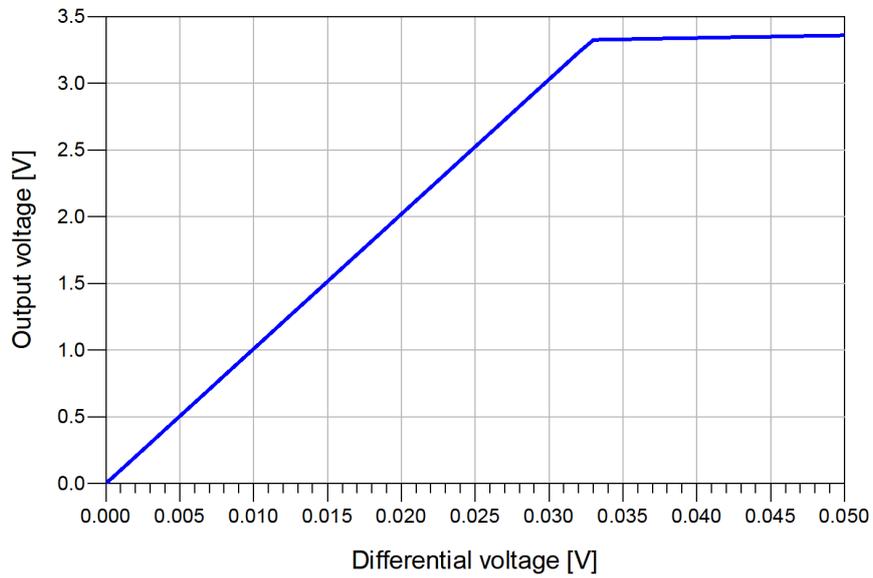


Figure A.2: In-amp output voltage plotted against the differential input voltage, with $V_{common} = 1.65$ V.

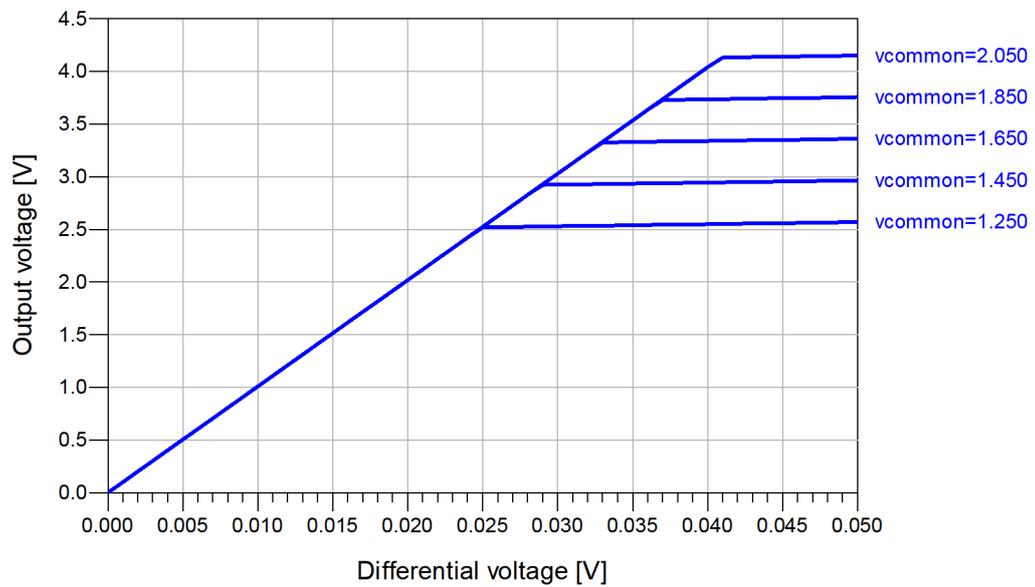


Figure A.3: In-amp output voltage plotted against the differential input voltage for different V_{common} .

Appendix B

Power Consumption Calculator

The Power Consumption Calculator (PCC) included in the STM32CubeIDE software from ST-Microelectronics can be used to accurately calculate the current consumption for the STM32L4 micro-controller. Figure B.1 shows the interface of the PCC. A sequence of steps can be configured and the current consumption for these steps is plotted in the consumption profile. The step current is shown in the 'User Sequence Table'.

The steps are used to determine the measurement routine state current consumption. Figures B.2, B.3, B.4, and B.5, show the step settings for the initialisation, sampling, data transfer, and idle state respectively. The configurable settings per step are:

- Power mode
- Power range (if applicable)
- Memory fetch type
- Supply voltage V_{DD}
- Voltage source
- CPU frequency
- Clock configuration
- Clock source frequency
- Step duration
- Additional consumption

The resulting total step consumption, step consumption without peripherals and step consumption for the peripherals is shown under 'Results'. Under 'Peripheral selection' the required peripherals can be enabled. These show up under 'Enabled peripherals'.

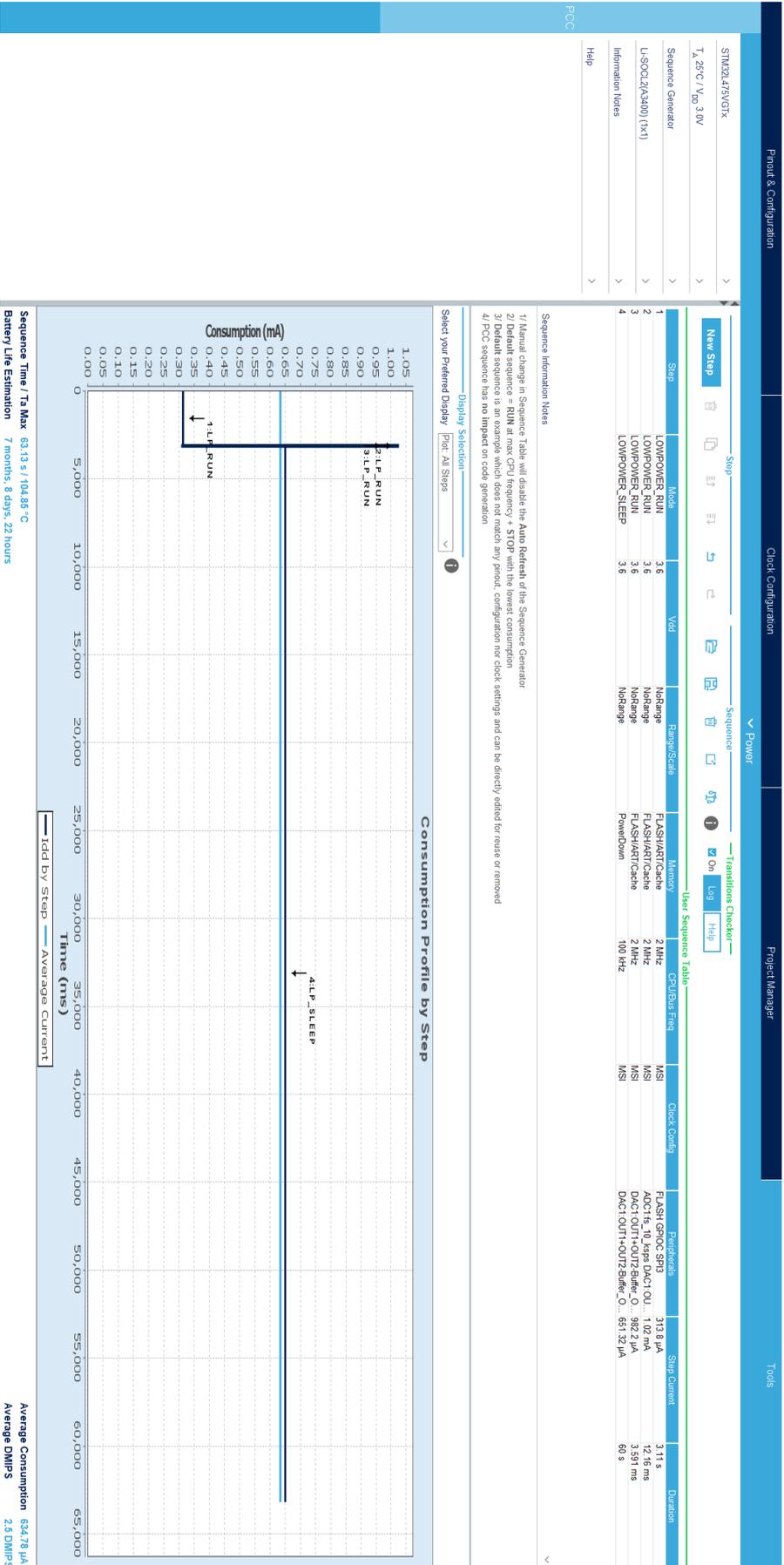


Figure B.1: PCC interface

MX Edit Step
✕

Reset Step Settings
Enable All IPs
Disable All IPs
Enable IPs from Pinout

Power/Memory

Power Mode:

Power Range:

Memory Fetch Type:

V_{DD}:

Voltage Source:

Clocks

CPU Frequency:

Interpolation Ranges:

User Choice (Hz):

Clock Configuration:

Clock Source Frequency:

Optional Settings

Step Duration: s

Additional Consumption: mA

Results

Step Consumption:

Without Peripherals:

Peripherals Part:

T_A Max (°C):

Peripherals Selection

Enabled Peripherals

FLASH
GPIOC
SPI3

Peripherals

- ADC1
 - fs_10_ksp
 - fs_1_Msp
 - fs_5_Msp
- ADC2
 - fs_10_ksp
 - fs_1_Msp
 - fs_5_Msp
- ADC3
 - fs_10_ksp
 - fs_1_Msp
 - fs_5_Msp
- AHB_APB1_Bridge
- AHB_APB2_Bridge
- CAN1
- CRC
- DAC1
 - OUT1+OUT2-Buffer
 - OUT1+OUT2-Buffer
 - OUT1+OUT2-Buffer
 - OUT1-Buffer_OFF-
 - OUT1-Buffer_ON-M
 - OUT1-Buffer_ON-V

OK
Cancel

Figure B.2: Step configuration for the initialisation state.

MX Edit Step
✕

Reset Step Settings
Enable All IPs
Disable All IPs
Enable IPs from Pinout

Power/Memory

Power Mode:

Power Range:

Memory Fetch Type:

V_{DD}:

Voltage Source:

Clocks

CPU Frequency:

Interpolation Ranges:

User Choice (Hz):

Clock Configuration:

Clock Source Frequency:

Optional Settings

Step Duration: ms

Additional Consumption: mA

Results

Step Consumption:

Without Peripherals:

Peripherals Part:

Ta Max (°C):

Peripherals Selection

Peripherals

- ADC1
 - fs_10_ksp
 - fs_1_Msp
 - fs_5_Msp
- ADC2
 - fs_10_ksp
 - fs_1_Msp
 - fs_5_Msp
- ADC3
 - fs_10_ksp
 - fs_1_Msp
 - fs_5_Msp
- AHB_APB1_Bridge
- AHB_APB2_Bridge
- CAN1
- CRC
- DAC1
 - OUT1+OUT2-Buffer
 - OUT1+OUT2-Buffer
 - OUT1+OUT2-Buffer
 - OUT1-Buffer_OFF-
 - OUT1-Buffer_ON-M
 - OUT1-Buffer_ON-V

Enabled Peripherals

ADC1

DAC1

FLASH

GPIOA

GPIOB

GPIOC

I2C1

RTC

SPI3

TIM1

OK

Cancel

Figure B.3: Step configuration for the sampling state.

The screenshot displays the 'MX Edit Step' configuration window, which is divided into several sections for setting up a step configuration for a data transfer state.

Reset Step Settings: Enable All IPs, Disable All IPs, Enable IPs from Pinout

Power/Memory:

- Power Mode: LOWPOWER_RUN
- Power Range: NoRange
- Memory Fetch Type: FLASH/ART/Cache
- V_{DD}: 3.6
- Voltage Source: Battery

Clocks:

- CPU Frequency: 2 MHz
- Interpolation Ranges: (empty)
- User Choice (Hz): (empty)
- Clock Configuration: MSI
- Clock Source Frequency: 2 MHz

Optional Settings:

- Step Duration: 3.591 ms
- Additional Consumption: 0 mA

Results:

- Step Consumption: 982.2 μ A
- Without Peripherals: 275 μ A
- Peripherals Part: 707.2 μ A (A: 645 μ A - D: 62.2 μ A)
- Ta Max (°C): 104.85

Peripherals Selection:

- ADC1: fs_10_kmps, fs_1_Mmps, fs_5_Mmps
- ADC2: fs_10_kmps, fs_1_Mmps, fs_5_Mmps
- ADC3: fs_10_kmps, fs_1_Mmps, fs_5_Mmps
- AHB_APB1_Bridge
- AHB_APB2_Bridge
- CAN1
- CRC
- DAC1: OUT1+OUT2-Buffer, OUT1+OUT2-Buffer (checked), OUT1+OUT2-Buffer, OUT1-Buffer_OFF-, OUT1-Buffer_ON-M, OUT1-Buffer_ON-V

Enabled Peripherals: DAC1, FLASH, GPIOA, GPIOC, I2C1, RTC, SPI3, TIM1

Warnings: (empty)

Buttons: OK, Cancel

Figure B.4: Step configuration for the data transfer state.

MX Edit Step
✕

Reset Step Settings
Enable All IPs
Disable All IPs
Enable IPs from Pinout

Power/Memory

Power Mode: LOWPOWER_SLEEP

Power Range: NoRange

Flash Status: PowerDown

V_{DD}: 3.6

Voltage Source: Battery

Clocks

Bus Frequency: 100 kHz

Interpolation Ranges:

User Choice (Hz):

Clock Configuration: MSI

Clock Source Frequency: 100 kHz

Optional Settings

Step Duration: 60 s

Additional Consumption: 0 mA

Results

Step Consumption: 651.32 μA

Without Peripherals: 18.8 μA

Peripherals Part: 632.52 μA (A: 630 μA - D: 2.52 μA)

Ta Max (°C): 104.9

Peripherals Selection

Peripherals

- ADC1
 - fs_10_kmps
 - fs_1_Mmps
 - fs_5_Mmps
- ADC2
 - fs_10_kmps
 - fs_1_Mmps
 - fs_5_Mmps
- ADC3
 - fs_10_kmps
 - fs_1_Mmps
 - fs_5_Mmps
- AHB_APB1_Bridge
- AHB_APB2_Bridge
- Bus-Matrix
- CAN1
- CRC
- DAC1
 - OUT1+OUT2-Buffer
 - OUT1+OUT2-Buffer
 - OUT1+OUT2-Buffer
 - OUT1-Buffer_OFF-
 - OUT1-Buffer_ON-M

Enabled Peripherals

DAC1

GPIOA

GPIOB

GPIOC

RTC

SPI3

TIM1

OK
Cancel

Figure B.5: Step configuration for the idle state.

Appendix C

BlueNRG current consumption estimator

In this appendix the simulation results of the estimation of the current consumed during Bluetooth operations are presented. Figure C.1 shows the current consumed for only setting up the connection. Figure C.2 shows the total transfer of one package containing 12 bytes of measured data. Figure C.3 shows the total transfer of the initialisation package containing 5 bytes of control data.

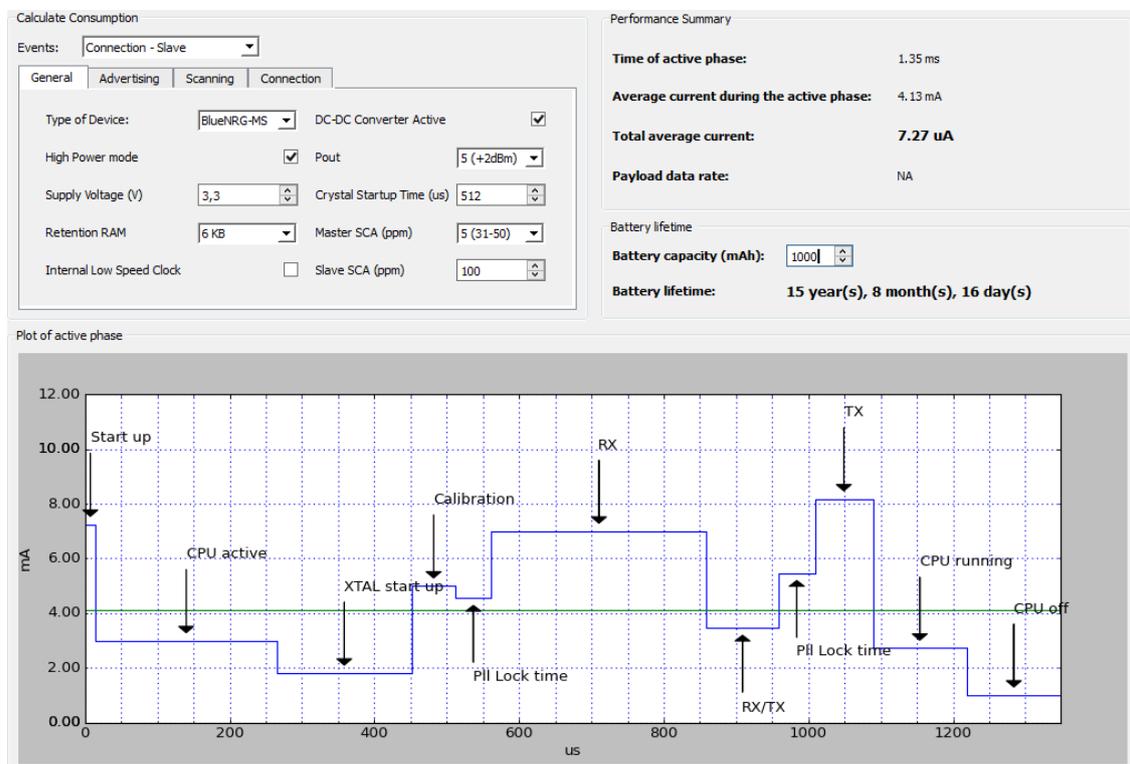


Figure C.1: Simulation of a Bluetooth connection

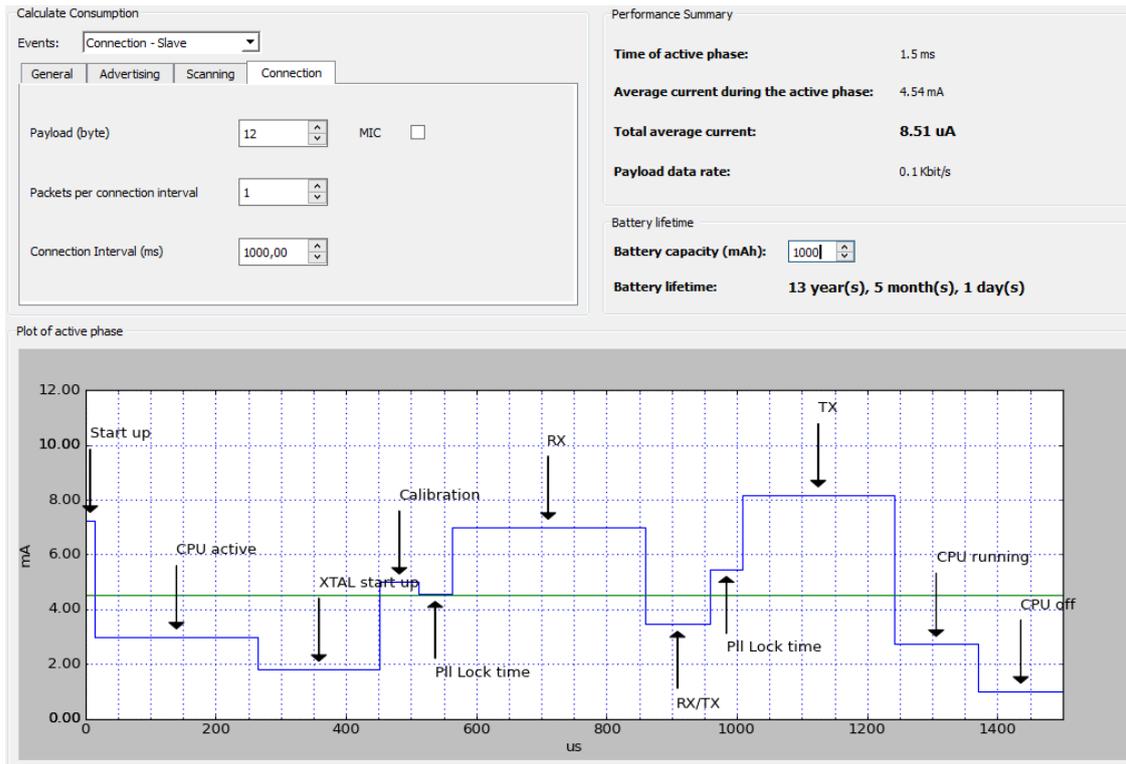


Figure C.2: Simulation of a Bluetooth data transfer (1 package, 8 bytes)

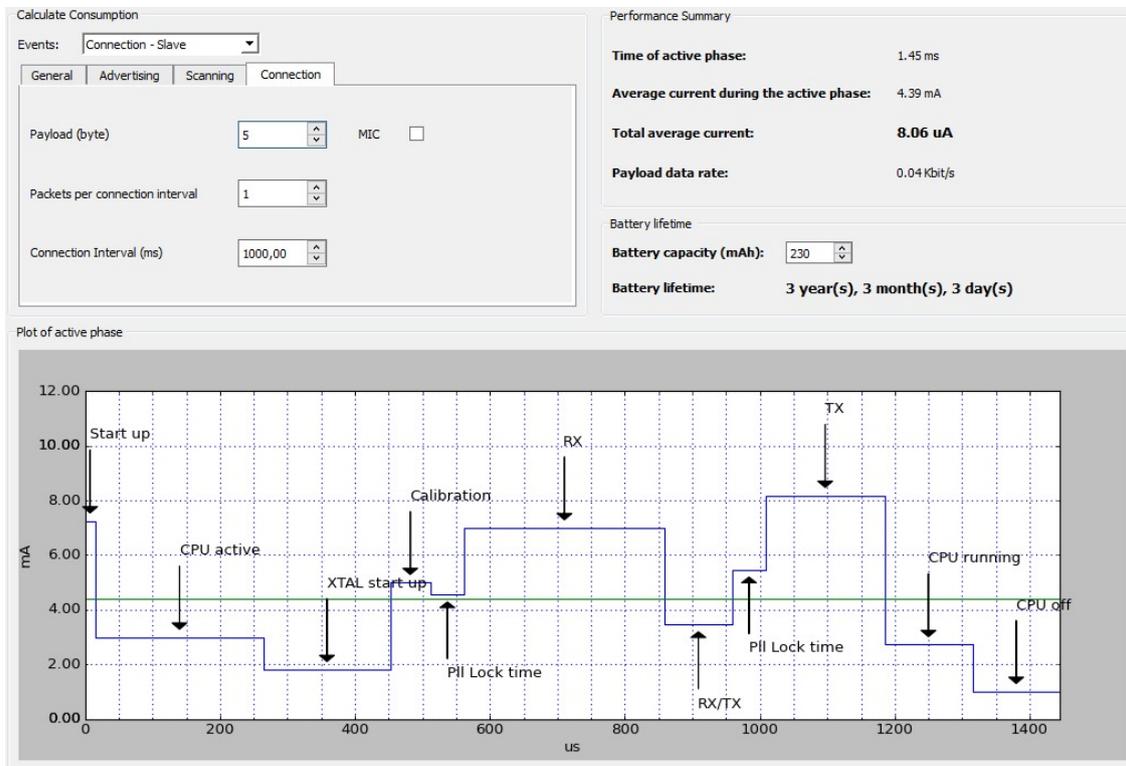


Figure C.3: Simulation of a Bluetooth data transfer (1 package, 5 bytes)