Technische Universität München
TUM Department of Electrical and Computer Engineering

Professorship of Environmental Sensing and Modeling
Prof. Dr.-Ing. Jia Chen

Bachelor's Thesis

Building a Toxic Gas Measurement Sensor Circuit

Emre Özbas

October 10, 2018

Supervisor:

Prof. Dr.- Ing. Jia Chen

Abstract

Scientific data shows that living in locations with a high density of nitrogen dioxide can cause severe damage to human health and result in a lower quality of life. The researches indicate that main reason for NO_2 emission is mainly caused by burning fossil fuels, industrial processes and especially by motor vehicle exhaust which constitutes approximately 80% of the total NO_2 emission in cities. Therefore it is important to track the air pollutant emission to prevent any harm to human health as well as any financial damage.

In this bachelor thesis, it is aimed to design and build a gas measurement sensor unit which tracks and saves voltage data linearly proportional to air pollutant density. The measurement unit is designed for Alphasense 4-electrode B series sensors and is tested with Alphasense NO2-B43F sensor. The hardware section of the circuit is explained in detail as well as the software segment is documented to enable a faster approach to this matter and realization with minimal problems. The results obtained from the sensor unit elicit that it is possible to track and save data concerning NO_2 measurement within a reasonable margin of error, using minimal resources.

I confirm that this Master's Thesis is my own work and I have documented all sources and material used.

Munich, October 10, 2018

Place, Date

Signature

Table of Contents

APPENDIX

Chapter 1

Introduction

As the world population increases by millions every year, the environmental damage we cause increases dramatically. In densely populated areas -and especially in larger cities- air pollution is a major problem, which does not only have financial consequences but also affects the quality of our lives in many ways. Since air pollution today constitutes a significant problem, there are more than many researches and studies regarding this issue: " . . . the total damage costs of air pollution [is estimated] to be US$ 3.0 trillion in 2010, or 5.6% of Gross World Product (GWP). These losses are equivalent to US$ 430 for every person on the planet."[1] is from just one of the numerous studies made on financial damage caused by air pollution.

In order to assess this problem correctly and take suitable measures to minimize the harm of air pollution, one should first be capable of finding out the cause accurately. Only after an accurate diagnosis can there be a suitable solution and thus a significant outcome. When it comes to air pollution, one of the best ways to detect the cause is to make density measurements of air pollutants in various locations with electrochemical sensors sensitive to specific gases. However there are some requirements that must be fulfilled: "To adequately characterize air quality (AQ), measurements must be fast (real-time), scalable, and reliable (with known accuracy, precision, and stability over time)."[2] The more accurate and fast the sensors get, the more expensive the gas measurement station will be. Since it is important to make measurements in multiple locations to create a pool of air pollutant density data and thus getting a better understanding of the environmental damage, a collective of stations are needed, which increases the total cost dramatically. For this reason, a low-cost gas measurement station could be a simpler and cheaper solution.

The goal of this bachelor thesis is to design and build a low-cost circuit suited for Alphasense NO2-B43F sensor, which is sensitive to NO_2 concentration in the air. This circuit is capable of receiving real-time information concerning NO_2 concentration and storing the data into an SD-Card. In addition to that documenting the steps well and describing the final project in detail also plays a very important part, since it enables to create the same sensor station more efficiently, which means with less time and resources spent.

In this writing researches on means of measuring NO_2 concentrations in different locations are reviewed and possible structures of gas measurement stations are examined. The results are then shown and the challenges of designing and building NO_2 measurement stations are explained. Lastly, some possible future enhancements to the latest design are discussed and the documents considering the hardware design as well as the source code are provided.

Chapter 2

Preparation

2.1 Initial Researches

Some research on the working principle of Alphasense NO_2 electrochemical sensor had to be made prior to the initialization of the mechanical and electrical sections of this project. The paper [1] on previous experiments conducted in Boston, United States of America was very helpful to get a first idea about how I could start building my project designed for NO_2 density measurements. In another paper about a research conducted in Zurich, Switzerland, [3] it is emphasized that it is quite expensive to build a gas pollutant measurement station sensitive to a certain gas and thus low-cost sensor stations play a very important role in collecting environmental data.

My goal was to build a functioning circuit, to get meaningful data from it and to document the entire process of my project well, and the researches and the papers I read gave me even more motivation about how topical air pollution is and thus how important it is to try to build a low-cost air pollutant density measuring station and to collect useful environmental data.

2.2 Getting a Better Understanding

I had to read the application notes on the Alphasense webpage to get a better understanding of the inner structure as well as the pinout of the sensor and for this purpose, I began to study the documentation about the toxic sensors [4]. As I started to understand which electrode of the sensor was responsible for which purpose, I began to get an idea of how I could build my own circuit, which would be able to supply enough current to the sensor and output voltage, linearly proportional to the concentration of the air pollutant, which in other words is the ppb level of NO_2 in the air.

Afterward, I started to read the documentation from Alphasense [5], which gave me a starting point for the circuit. In diagram **PotCircuitDiagram** you can see a circuit design for a three-electrode sensor. I was actually working with the sensor NO2-B43F, which is a four-electrode sensor, but this circuit schematic was nonetheless a good point to start building and testing the circuit.

Chapter 3

Putting Into Practice

3.1 Building the Initial Circuit

3.1.1 Hardware

3.1.1.1 Circuit Design

In order to understand the circuit fully, we must first study the structure and the pinout of the NO2-B43F sensor. There are 4 electrodes of the electrochemical sensor: Working electrode, the reference electrode, the counter electrode, and the auxiliary electrode. The reference electrode holds the potential of the working electrode stable at a certain level, which is equal to the potential of the reference electrode itself. This potential must be fixed to ensure a stable outcome from the sensor and the circuit. The counter electrode must be able to supply enough current to the working electrode so that the current through the counter and working electrode can be translated and amplified to create the output voltage.

The A4 and B4 sensors have two sensing electrodes: The working electrode and the auxiliary electrode. The main purpose of the working electrode is to react to the $\mathrm{NO_2}$ in the air and thus create a current flow proportional to the gas concentration. In other words, the working electrode responds to gas concentration whereas the auxiliary electrode does not respond to gas. The idea is to be able to correct for zero drifts using the auxiliary electrode output. It is thus recommended that at the beginning both electrode outputs are recorded (Working electrode and auxiliary electrode) rather than applying a correction directly.

The circuit, in general, comprises of operational amplifiers, a couple of capacitors to reduce noise from the sensor and various resistors to get the desired gain from the operational amplifiers. In Figure **PotCircuitDiagram** the detailed circuit schematic is given. On the left-hand side of the circuit, the NO2-B43F sensor is connected via 3 electrodes to the circuit. This schematic is for 3 electrode sensors like previously mentioned, so the auxiliary electrode is left out. The power connections of the operational amplifiers are also not shown, however they must be provided a regulated and high enough voltage to encompass the maxima of the output voltage. In addition to that, the power supply must be rated with high enough current to be able to supply the required current drawn by the counter electrode and the operational amplifiers themselves.

The circuit consists of 2 stages of amplifiers. The first stage is the control circuit, whose main objective is to supply the counter electrode with enough current so that the current required by the working electrode is met. The potential at the reference electrode, namely the reference voltage is connected to the inverting input of the operational amplifier. It is important that near to zero current is drawn from the reference electrode, so an op-amp with minimal input bias current is recommended. [5]

Since the current control part of the circuit can already supply the counter electrode with enough current, the next step is to build the sensing part of the circuit, namely the current measuring stage. In this stage, the current through the counter and working electrode flows through the sensing resistor R_{Load}, which in return creates a voltage on the inverting input of the second operational amplifier IC1. As it is mentioned before this current through R_{Load} is linearly proportional to the gas concentration in the air. As a reminder, our main goal is to measure the amplitude of this current created by the working electrode, which gives us information about the concentration of NO_2. The voltage difference between the inputs of IC1 is amplified (multiplied) by a very large number and is created on the output of IC1. However, this high voltage is fed back to the inverting input over the resistor R4, which increases the voltage on the inverting input and thus reduces the voltage difference between the inputs of the operational amplifier. As a result, the output voltage is reduced and so is the influence of the output on the inverting input voltage. This pendulum saturates at a specific voltage level, which is equal to the input voltage multiplied by a constant determined by the resistors R_{Load} and R4. In this specific case the operational amplifier is being operated in inverting configuration, since the input voltage is connected to the inverting input of the operational. ampflifier. Therefore the constant i.e. the gain of the amplifier is equal to the following:

$$A = \frac{V_{Out}}{V_{In}} = -\frac{R_4}{R_{Load}}$$

In conclusion, the control stage supplies the required current by the counter electrode to the sensor, which is created by the potential difference between the counter and working electrodes. The current then flows through the sensing resistor R_{Load} creating a voltage on the inverting input of IC1. This input voltage is then multiplied by the gain and in the end, creates the output voltage.

3.1.1.2 Microcontroller

For this project, I decided to use an Arduino Uno board, since it is inexpensive and easy to use. I used a 12 Volts 1 Amp wall adapter for the power supply of the Arduino board. The Arduino board has 5V and 3.3V voltage regulator onboard. Since this circuit does not consist any component which requires high current, the voltage regulation from 12 Volts down to 5 Volts does not dissipate any excessive heat.

I put all the parts depicted in the circuit design together on a breadboard. For the power supply needed for the operational amplifiers I simply used the $5V$ and GND power supply lines of the

Arduino Uno board. The output of IC1 is connected to the analog input $A0$ on the board, which makes the voltage level readings possible.

3.1.2 Software

The Arduino Uno board is simply an I/O device with several outputs and input pins. Through programming, via the Arduino Integrated Development Environment (IDE) these pins can be accessed and thus works as an interface between the circuit and the computer. It is capable of reading the voltage connected to its input, sending it to the computer via serial communication. These values sent from the Arduino are then displayed on the computer screen via the Arduino Integrated Development Environment (IDE). But in order to let the Arduino board know which data to send and at which frequency, we first have to program it using the same Arduino IDE.

Arduino can supply voltage and read voltage inputs up to 5 Volts. Analog inputs of the Arduino divide the continuous voltage range from 0 Volts to the analog reference voltage into 1024 discrete steps while digital outputs with Pulse Width Modulation (PWM) feature can supply 1024 different levels of voltage from 0 Volts up to a maximum of 5 Volts. Digital outputs without this feature can only supply either 0 Volts (LOW) or 5 Volts (HIGH). For this purpose, I connected the output pin of the operational amplifier in the last stage to one of the analog inputs on the Arduino. This way I could read the voltage level on the output of the operational amplifier at a high frequency (9600 bits per second) and thus get enough data to plot the output signal with sufficient resolution.

For this reason, I wrote a simple Arduino code using the Arduino IDE **sourceCode**. This code consists of two standard main functions, namely $setup()$ and $loop()$. The $setup()$ function is called once when the sketch starts and runs only once after each powerup or reset of the Arduino board. Afterward, the $loop$ function is called and loops consecutively as the name suggests. In $setup()$ the serial communication speed, at which the microcontroller Atmega328p of the Arduino Uno is going to communicate with the computer, set with the function $Serial.begin()$ at 9600 bits per second. In $loop$ the function $Serial.print()$ is then used to print the voltage values on the serial monitor or on the serial plotter. For testing purposes, it is recommended to use the serial monitor embedded in the Arduino IDE, since it is sufficient to show the voltage values at the analog pin on the computer. Alternatively, the serial plotter, again embedded in the Arduino IDE, can be used to plot the data and thus to create a sample-voltage graph. At the end of the code, I added the $delay()$ function, which "[p]auses the program for the amount of time (in milliseconds) specified as parameter." [6]. In the code, I have set the parameter equal to 50 resulting in a 50 milliseconds delay before acquiring the next value. This way it becomes more pleasant to read the data printed on the serial monitor.

3.1.3 Discussion

This design was a success in terms of testing the sensor and the circuit. The values printed on the serial monitor showed us that the sensor reacted to the $\mathrm{NO_2}$ concentration in the air. However, it was lacking protection against electrical noise and the values were not similar to the nominal values which were determined with the help of the Individual Sensor Board (ISB) from Alphasense. Most importantly the auxiliary electrode was left out, which is why a correction for zero drifts was impossible with this circuit design alone. An improvement of the circuit was needed for better results, namely for the output data to get closer to the nominal values of the ISB circuit. Therefore I started to study and eventually build the circuit explained in the next section.

3.2 Realising the Last Circuit

3.2.1 Hardware

LastCircuitSchematic The circuit in general consists of 3 operational amplifiers, a couple of capacitors to reduce noise from the power supply and various resistors to get the required output voltage to input voltage ratio from each operational amplifier. Unlike the previous circuit design, the auxiliary electrode is included in this schematic, which will be helpful in the future to make zero drift corrections through programming and thus better results outputted from the circuit. Additionally, the subcircuit for the low-dropout DC linear voltage regulator $MCP1702T25$ is also depicted in detail. Lastly, the power supply subcircuit of the operational amplifiers is included.

3.2.1.1 Main Circuit

This design consists of two branches which lead to two different outputs. The working electrode of the sensor leads to the first branch and thus the first output $O/P1$, while the auxiliary electrode leads to the second output $O/P2$. However, both branches are identical copies of each other. The main reason to build two identical copies of a branch is to separate the working and the auxiliary electrode from each other in order to avoid any interference between the two electrodes. Like mentioned before, the auxiliary electrode does not react to $\mathrm{NO_2}$ and thus the voltage level at $O/P2$ must not change depending on the $\mathrm{NO_2}$ density in the air. Additionally, it is to notice that two different operational amplifier ICs, namely two separate $LT6011$ chips are used in order to separate the two branches better. The operational amplifiers $U4A$ and $U4B$ constitutes one $LT6011$, while $U1A$ and $U1B$ constitutes the second IC. The last operational amplifier $U2$ is the IC $TLV2211$ which consists of only one operational amplifier. Although the two different ICs have similar features, $TLV2211$ is more efficient in power consumption and spatial terms since the $LT6011$ has a bigger packaging than the $TLV2211$ and consumes more energy. [7] [8]

The first stage of the circuit is the same as the initial design explained in the previous section. The only difference is that the capacitor $C2$ in the initial design is left out in this schematic, which is not necessary for this operational amplifier. [5] Since this stage is explained in detail in the previous section, it is not needed to explain it again.

After the first stage, the circuit divides into 2 branches through the working and the auxiliary electrodes of the sensor. Like mentioned before these branches are identical, so it is more meaningful to explain only the branch starting from the working electrode in detail as the same logic can be applied to the branch starting from the auxiliary electrode leading to $O/P2$.

$V_{mid}1$ is directly connected to the output of the voltage regulator and has a potential of 2.5 Volts, which is explained in the next section in detail. The current flowing through $R15$ creates a voltage difference between its two ends, namely the working electrode and the inverting input of $U4B$. Since the potential of the working electrode is equal to the reference electrode and thus to 2.5 Volts, the voltage on the inverting input is higher than 2.5 Volts and thus is also higher than $V_{mid}1$, which is connected to the non-inverting input of the operational amplifier. As the $\mathrm{NO_2}$ concentration gets higher, the current flowing through $R15$ becomes greater. For this reason the voltage difference between the two inputs of $U4C$ changes linearly proportional to the ppb level of $\mathrm{NO_2}$. This voltage difference gets amplified through $U4B$. The amplified voltage created at the output of $U4A$ goes into the inverting input of the last operational amplifier on this branch $U2$. The voltage level on the non-inverting input of $U2$ is determined by the voltage divider circuit consisting of $R29$, $R30$, $R8$, $R7A$ and $R9$. For this reason, the voltage level of the non-inverting input is between 0 and 2.5 Volts. The voltage difference between the inputs of $U2$ is then multiplied by the gain of the operational amplifier, which is determined by the resistors $R6A$, $R4$ and $R3$. The same formula used for the earlier stage can be applied here, while the input resistor R_{in} is $R6A$ and the value of the feedback resistor R_f is equal to the sum of the values of $R4$ and $R3$, as they are connected in series. The voltage outputted from $U2$ is approximately equal to the output voltage since the resistor $R1$ has a much greater value than $R3$ and has a negligible effect on the output voltage. $R1$ and the capacitor $C3$ are against noise at the output of the circuit $O/P1$.

In conclusion, the working principle of the main circuit can be summarized as follows: The voltage output at $O/P1$ is the current created through the voltage difference between the working and the counter electrode translated to a voltage with the help of resistors and amplified through 2 stages of operational amplifiers.

The voltage levels at $O/P1$ and $O/P2$ are supplied to the analog inputs of the Arduino board and then gets printed on the computer. Since the voltage at $O/P1$ is a translation of the working electrode and the voltage at $O/P2$ is a translation of the auxiliary electrode, both outputs can be used at the same time for noise reduction and zero drift correction which leads to more reliable results than the data outputted using the initial design.

3.2.1.2 Voltage Regulator Subcircuit

On the bottom left corner, the voltage regulator subcircuit is to be observed. This part of the schematic consists of a low-dropout DC linear voltage regulator $MCP1702T25$, a Zener diode with 6.8V Zener voltage, an electrolytic capacitor, and various resistors and ceramic capacitors. The $MCP1702T25$ receives the input voltage at its V_{in} pin and translates it to a regulated 2.5V potential level at V_{out}. The last pin depicted as Adj in the schematic is simply connected to GND. The input voltage at V_{in} is also regulated with an external voltage regulator $L7806CV$, which converts the 12V outputted from the wall adapter down to 6 Volts. The Zener diode $D2$ cuts off any excess voltage higher than 6.8 Volts by shorting V_{in} with GND. Therefore it is dismissable since the 6 Volts input voltage is outputted from the $L7806CV$ voltage regulator and is less than 6.8 Volts at all times. The resistor $R16$ is connected to an unregulated voltage supply, and therefore is also not necessary in our circuit. On the output side of the voltage regulator the resistors $R21$, $R19$, $R27$ and $R28$ constitutes a voltage divider circuit which creates the voltage level depicted as $V_{off}1$. $Vpot1$, $Vpot2$, $Vce1$ and $Vmid1$ are all connected directly to $Vout$ and thus have the same 2.5V potential. The capacitors $C5$, $C12$ and $C13$ cuts out the AC part of the input and output voltages and thus reduces the overall noise ratio at the outputs $O/P1$ and $O/P2$.

3.2.1.3 Power Supply of the Operational Amplifiers

The operational amplifiers depicted as $U1C$ and $U4C$ are connected to the 6V regulated voltage level outputted from the $L7806CV$. The capacitors $C2$ and $C8$ are in parallel to the ICs and thus shorts the AC component of the input voltage while staying open for the DC section. This results in a more stabilized input voltage for the operational amplifier ICs and therefore reduces noise overall in the circuit. The single operational amplifier IC $TLV2211$ is also connected in parallel to the two $LT6011$ ICs, which is not depicted in this circuit schematic.

3.2.1.4 SD Card Module

An SD card module has been added to the circuit in order to save the concentration data supplied by the circuit. Six connections between the SD card module and the Arduino board are needed for a successful communication between the two. The V_cc pin is connected to the $5V$ or the $3V3$ power supply line of the Arduino board, while the GND pins of both module and the board are connected together to ensure a common ground. The $MOSI$ pin is connected to the eleventh pin, the $MISO$ pin is connected to the twelfth pin and the SCK pin is connected to the thirteenth pin of the Arduino board. The CS pin is connected to the fourth digital I/O pin of the Arduino board. Lastly, there is an LED connected to the fifth digital I/O pin of the board which indicated the status of the SD card module. The details of saving data into the SD card are explained further in the next section.

3.2.2 Software

The previous software for the initial circuit design is developed and some new features are added to this code. The recent Arduino code can not only show the voltage levels at $O/P1$ and $O/P2$ on the serial monitor or the serial plotter but also saves every incoming data into an SD card provided with a time stamp.

First of all the programming libraries needed for providing a time stamp and saving the data into an SD card are included at the beginning of the Arduino code; namely $TimeLib.h$, $SPI.h$ and $SD.h$. Afterward comes the declaration of the two constants $TIME\_HEADER$ and $TIME\_REQUEST$. The latter is set to 7 which is the ASCII bell character. The former is set to the character *T*. For this reason, whenever a time stamp is provided to the Arduino board via serial communication, the header *T* must be also provided at the start of the UNIX timestamp, which will then be recognized by the Arduino microcontroller as a time stamp and attached to the incoming data from the circuit. After a time stamp has been provided to the microcontroller, the process of providing a time stamp for every succeeding incoming data is done automatically, since the microcontroller can track time beginning from the initialization of the sketch. For this reason, it is only needed once to provide a time stamp to the microcontroller via serial monitor. The parameter $chipSelect$ is set to *4* since the CS pin of the module is connected to the fourth pin of the board.

In the $setup()$ function the serial communication speed is set to 9600 bits per second. Since the incoming voltage levels are in millivolts, it is recommended to set the analog reference (which is set to 5 Volts as default) to the virtual ground of the circuit at 2.5 Volts (in reference to the common ground). For this purpose the function $analogReference()$ is called and its input parameter is set to $EXTERNAL$. The analog reference pin $AREF$ of the Arduino board is thus connected to the 2.5V power supply line of the circuit, which sets the analog reference voltage to the same voltage level. The resolution of the data is therefore doubled since the reference voltage is halved.

The function $pinMode()$ is called in order to set the fifth pin of the board as $OUTPUT$, which drives the LED indicating the status of the module. A series of functions are called for the initialization of the SD card module: $setSyncProvider()$ sets the external time provider. [9] The function $processSyncMessage()$ sets the internal clock of the Arduino microcontroller to the time received on the serial port. This is done by receiving the time stamp sent via serial monitor and decoding the message. $Serial.find()$ searches for the letter *T* in the message and decodes the rest of the message into time information using the function $Serial.parseInt()$. After the message is decoded and the time information is acquired, the internal clock of the microcontroller is set to the received time. Therefore the clock gets synchronized with the real-time and the microcontroller can attach time to the incoming data. Lastly the function $SD.begin()$ initializes the SD library and the module [10], which returns a variable in boolean. If the initialization is a success, $true$ is returned and the message "Card Initialized" is printed on the serial monitor; and otherwise $false$ is returned and the message "Card failed, or not present" is printed.

In $loop()$ function the serial communication is checked with the function $Serial.available()$ which returns a boolean. If $true$ is returned, the $processSyncMessage()$ is called and the time is set to the received UNIX time stamp. Afterward, the status of the time synchronization is checked by calling the function $timeStatus()$ which indicates if the time has been set. [9] If the synchronization is flawless the fifth pin is set to $HIGH$ and the indicator LED is powered on. Otherwise, the LED turns off and indicates an error concerning the synchronization. Afterward, a string called $dataString$ is created. This string is then set to a series of strings concatenated together, which contain data of the values of two outputs $O/P1$ and $O/P2$ as well as the date and time at which the data was received. After that the $datalog.txt$ file contained in the SD card is opened by calling $SD.open$ and setting its mode to $FILE\_WRITE$. This enables writing incoming data from the circuit onto the $datalog.txt$ file in the SD card. If the $datalog.txt$ is available for writing the string $dataString$ is written into the $datalog.txt$ file by calling the $dataFile.println$ function and is closed after $dataFile.close$ is called. After the save is completed the $dataString$ is printed on the serial monitor for testing purposes. If the file is unavailable the message "Error opening datalog.txt" is printed on the serial monitor. Lastly, a $delay()$ function with its input parameter set to 1000 pauses the code for 1000 milliseconds, namely for one second. After the $delay()$ function is executed and exited the $loop()$ function starts again from the top and saves the next incoming data provided with time information.

3.2.3 Discussion

Since the auxiliary electrode was used in this circuit unlike the initial circuit, this latest design was a success in terms of outputting approximate results as the ISB circuit. The circuit has more resistor-capacitor pairs at the inputs and outputs of each operational amplifier, which also results in less noise contained in the output signal.

Chapter 4

Results

4.1 Initial Results

4.2 Final Results

After the circuit was finished, I started to test the circuit

Chapter 5

Challenges

5.1 Converting SMD to THT

After preparing an orders list according to the previously mentioned circuit design taken from Alphasense [5] and receiving these components, my first challenge was very clear: Transforming the surface-mount device (SMD) parts into through-hole components. The operational amplifiers used in the circuit (LT6011) were only available to order in SMD form. Since I needed the SMD components to verify the circuit design, I could not simply design a printed circuit board and solder the components on the board without testing the circuit design with a breadboard first. Otherwise, everything would be inalterable and the tiniest change in circuit schematic would lead to a whole new circuit design and thus a printed circuit board from scratch. That is why I started to build an adapter for making the SMD modules breadboard compatible.

For this purpose, I used a perfboard and male sockets. I first split the male sockets into two 4-pin male sockets, since the LT6011 ICs have an 8-pin structure, namely 4 pins on both sides. Afterward, I soldered these sockets parallel to each other with a reasonable distance in between so that there is enough room for the IC to fit between the two sockets. Here it is quite important to first solder the sockets on the side of the perfboard with copper rings around the holes with the longer part of the sockets laying on this side. The IC will be fixed on the other side of the perfboard because the distance between any two pins of the IC is much smaller than the distance between two holes of the perfboard and this would thus lead to a short between the pins. Afterward, I placed the IC between the sockets and anchored it on the perfboard by gluing it with hot glue. This makes it easier to make the connections between the sockets and the IC itself since it stays fixed during soldering. It can be tricky to solder one pin of the IC with one pin of the socket since there is no such surface e.g copper between those two on which the solder can stick. This makes the soldering process much harder since the solder tends to stay on one the IC or the socket, thus not binding them together. It is therefore recommended to make the distance between the two sockets as small as possible so that the IC and the socket stay as close as possible. This makes the soldering much easier since the distance between the two parts is smaller and a connection between the two can be made with enough solder applied. Another option would be to use tiny wires between the parts. This way the solder can stick onto this wire and make the connection instead of just creating a solder bubble on one pin. Here it is recommended to fix the wire by gluing it on the perfboard or use a third-hand soldering stand to stabilize the wire and the other parts.

5.2 Initial Design

5.2.1 Inconsistencies Between Design and Sensor

The circuit schematic taken from Alphasense webpage [5] illustrates a design for a 3-electrode sensor, but what we wanted to build was a circuit for the 4-electrode sensor NO2-B43F. This led to some problems concerning the correctness of the output data. The three electrodes of the sensor in the schematic, namely the working, counter and reference electrodes, have the same purposes like the three electrodes of the NO2-B43F. However, according to this design, the auxiliary electrode was not used at all, which is the fourth electrode of our sensor. This electrode is used for noise cancellation and zero drift correction. Not using the auxiliary electrode led to an output with high noise ratio, because firstly the electrode was floating (not connected to any stable potential) and secondly it could not be used for corrections in the output voltage.

5.2.1.1 Testing the Sensors

Since we did not have any containers of NO_2 in the lab, I had to find another way to test the response of both sensors (the one on the ISB circuit and the one connected to my circuit) to a change of NO_2 concentration in the air. So I fixed the sensors in close proximity and parallel to each other onto a portable board in order to have the same environmental conditions for both of the sensors. Afterward, I fanned the sensors with the sensing areas of the sensors facing directly to the fan. Normally gas flow parallel to the sensing area of the sensor must be supplied to the sensors, since the sensors are sensitive to air flow perpendicular to the sensing area which causes a change in the outputted signal although the NO_2 concentration in the air does not change. For this reason, the sensors were fanned in order to create a perpendicular gas flow and thus create a change in the output signals of both sensors, of course only for testing purposes. This way I could compare the reaction velocity and amplitude of the output signals in millivolts.

For the reasons explained in the previous section the outputted voltage values were different compared to the values gathered from the ISB circuit. The two circuit boards were tested in the same environment and for both circuits, the same power supply unit was used. For that reason, the values should have been close to each other. I tried to solve this problem by changing the software and adapting it to suppress the noise consisted of the output signal. For this reason, I added another function to the Arduino code in order to make the Arduino print the average voltage level over a specific time interval instead of printing the values directly. This resulted in better values with lower SNR, but the response velocity and amplitude were different compared to the values from the ISB circuit. Additionally, this was not a befitting solution since the values coming from my circuit were faulty and a correction in the hardware was needed.

5.3 Final Design

5.3.1 Critical parameters

It is of utmost importance to pay attention to the resistor values in the circuit schematic and the resistors used in the actual circuit. A little difference in these values creates a large spread between the desired output voltage and the actual output since most of the resistors in this circuit design play a role in determining the gain of the operational amplifiers. An inaccuracy in resistor value changes the gain of the operational amplifier and consequently the output voltage gets multiplied with that error. If the resistor inaccuracy is in earlier stages of the circuit, the error at the output side gets even higher since the error gets amplified by the second operational amplifier in the last amplifying stage. It is recommended to check the resistors' values preferably with a multimeter before mounting it on the circuit, even if the circuit is being built on a breadboard and will be modifiable, since there are a lot of resistors in this circuit design and it can be quite complicated to detect where the problem lies after the circuit is completed. For this reason, it is quite important to check every part individually before putting them into the circuit.

5.3.1.1 Testing the Sensors

The results from my circuit were higher than the output values of the ISB circuit. For this reason, I firstly subtracted a constant number from the voltage values. As the outputs were stable, meaning that the sensors were not fanned and the test was conducted indoors with little to no change in the NO_2 concentration in the air, the values were approximately equal to each other. However after fanning the sensors the difference in the response amplitude and response velocity was noticeable.

Chapter 6

Outlook

6.1 Conclusion

In this bachelor thesis, an initial design was tested and problems were detected and documented for facilitating future realizations. Afterward, the voltage data outputted from this sensor unit was saved and plotted for better visualization and documentation purposes. These values indicate an unacceptable margin of error compared to the ISB circuit from Alphasense since the auxiliary electrode was not being used in this sensor design and thus not protected against electrical and electromagnetic noise. Therefore a new circuit schematic was built and tested with the same electrochemical sensor. Since the auxiliary electrode was not left out in the later design, the circuit was a success and could supply similar results as the ISB board. Additionally with the help of low noise voltage regulators and the use of more resistor-capacitor pairs as noise filters the circuit was more resistant to electrical noise and resulted in an output signal with a higher signal-to-noise ratio.

6.2 Possible Future Enhancements

Directly measuring the output voltage gives us, in theory, the raw data of the NO_2 concentration, since the output voltage increases or decreases linearly as the ppb level of NO_2 changes. However, the data will depend on environmental elements -especially from humidity and temperature- and thus be somewhat inconsistent. The raw data obtained directly from the circuit must be calibrated against such noises in order to get a better approximation of real concentration values. This can be achieved by altering the circuit and making it more resistant to noise. There are two possible ways of dealing with such noises: Passive and active filtering. The former is achieved by adding passive elements like resistors, capacitors, and inductors. After detecting the peak frequency of the noise contained in the raw signal (possibly with the help of an oscilloscope by applying Fourier Transformation on the signal and acquiring the frequency spectrum as well as peak frequency of the noise content) it can be eliminated by adding high and low-pass filters. The latter option requires an outside power source, hence the name "active", which makes it a more expensive option than the former, however, this can amplify the desired frequency while suppressing the noise, which makes the signal-to-noise ratio greater. Both options can be useful and serve our purpose of noise cancellation well.

Additionally, data calibration against humidity and temperature could be achieved by integrating a humidity and temperature sensor into the circuit board and measuring their effect on the output voltage and then subtracting these elements from the raw data itself. As a result, the circuit would be more resistant against humidity and temperature changes in the environment and the outputted values would be even more dependent on the NO_2 concentration rather than other environmental effects.

APPENDIX

Appendix A

Some Appendix

text text text text

Appendix B

Source Code

text text text text

Bibliography

[1] Guy Hutton. Air pollution. *Global Damage Costs of Air Pollution from 1900 to 2050*, 2011.

[2] David K. Lewis Gregory R. Magoon Timothy B. Onasch Michael L. Kaminsky Douglas R. Worsnop John T. Jayne Eben S. Cross, Leah R. Williams. Use of electrochemical sensors for measurement of air pollution: correcting interference response and validating measurements. Technical report, Massachusetts Institute of Technology, Cambridge, MA 02139, USA, 2017.

[3] Christoph Hueglin Michael Mueller, Jonas Meyer. Design of an ozone and nitrogen dioxide sensor unit and its long-term operation within a sensor network in the city of zurich. 2017.

[4] Alphasense, Sensor Technology House, 300 Avenue West, Skyline 120, Great Notley, Essex CM77 7AA. *How Electrochemical Gas Sensors Work*.

[5] Alphasense, Sensor Technology House, 300 Avenue West, Skyline 120, Great Notley, Essex CM77 7AA. *Designing A Potentiostatic Circuit*, March 2009.

[6] Arduino. *Delay - Arduino Reference*.

[7] *TLV2211, TLV2211Y Advanced LinCMOS Rail-To-Rail Micropower Single Operational Amplifiers*.

[8] *LT6011/ LT6012 Dual/Quad Rail-to-Rail Output Precision Op Amp*.

[9] *Arduino Time Library*.

[10] *Arduino SD Library*.