Novel sensor structures for enhanced redox cycling

Master thesis M. Odijk
5/2/2007
Report number: 003/bios/2007

Group:

BIOS-Chair

Faculty:

EWI, University of Twente

Committee:

Dr. Ir. Wouter Olthuis

Prof. Dr. Ir. Albert van den Berg

Prof. Dr. Uwe Karst

Dr. V.A.T. Dam van Anh

Contents

Summary

This project is about the redox cycling effect and its use to amplify measured currents in electrochemical experiments. It continues the work of Dr. V.A.T. Dam [1] on redox cycling with facing interdigitated array (IDA) electrodes. Dr. V.A.T. Dam measured a redox cycling (RC) amplification of 60 to 70x. This high amplification factor could not be explained properly by known theory. The first goal of this project is to find a plausible explanation for these results. The second goal is to use the obtained knowledge to design a novel sensor structure to achieve an even higher RC-amplification.

A short introduction to redox cycling is given in the first chapter of this report. In the second chapter basic theory is presented about electrochemistry. The redox cycling effect is explained in detail in this chapter as well. With the presented theory a model is formulated for finite element method (FEM) simulations. The IDA-electrode is simulated and parameters like the facing electrode distance, and the electrode width are varied to study their effect on the RC-amplification. The results found are comparable to other published simulation results in literature. The effect of the electrode height is also studied. A remarkable effect in RC-amplification is found. A special simulation with these elevated electrodes, representing the electrode structure used by Dr. V.A.T. Dam, showed a RC-amplification of 74x. Therefore a plausible explanation is given for the experimental RC-amplification of 60 to 70x found by Dr. V.A.T. Dam. Some novel sensor structures are simulated. The most promising configuration consists of a parallel plate electrode configuration perforated with holes to ensure a good supply of bulk electrolyte due to diffusion of ions to the space between the plates. With this configuration a promising RC-amplification of up to 400x is calculated.

The parallel plate electrode configuration with diffusional bulk supply concept is used for a design which is fabricated in the cleanroom. The fabrication process is described in detail in chapter 4. This novel sensor structure consists of two parallel platinum electrodes placed in close proximity of each other. The spacing between the plates is provided by a silicon oxide sacrificial layer. This sacrificial layer is partly removed to create a space between the two electrodes. this space is used to conduct electrochemistry. One of the two plates is provided with holes to create entries for the electrolyte to reach the space between the two electrodes. The outside of the two electrodes is shielded from the electrolyte. Two different shielding materials have been studied. The first material (silicon nitride) gave problems because it was partly removed during the sacrificial layer etching. The second material (polyimide) showed a better resistivity to the chemicals used to remove the sacrificial layer.

Measurements with the fabricated sensors provided with both shielding materials are presented in chapter 5. Most of the results presented in chapter 5 showed unexpected and unwanted behavior. The shielding material is also responsible for the structural integrity of the device. Since the first shielding layer was partly removed during the sacrificial layer etching, the devices shielded with silicon nitride suffered from shorts between the two electrodes due to bending of one of the electrodes. The devices provided with polyimide as shielding layer suffered from irreversible damage due to a voltage breakdown of the sacrificial layer during the fabrication of the devices. The voltage breakdown resulted in an unstable conducting path between the two electrodes. The problems described here cause the measurement results on the redox cycling amplification to be unreliable. Therefore no real quantitative redox cycling amplification is measured.

In chapter 7 an improved design is recommended to prevent the observed problems during the measurements with the devices.

Parameters

Table 0.1: *Parameters used in this report*

Symbol	Description	Value	Dimension
A	Area		m^2
C	concentration		Mol/m^3
C^*	Initial bulk concentration		Mol/m^3
F	Faradays constant	96485	Coulomb/Mol
i	current		A or Coulomb/s
j	current density		A/m^2
n	Valency of reaction		
N	Flux		Mol/$(m^2 * s)$
k_f	forward reaction rate constant		m/s
k_b	backward reaction rate constant		m/s
k_s	reaction rate constant		m/s
E^o	standard potential		V
E_{nernst}	Nernst potential		V
W_e	Work of electron		N*m
q	Charge of electron	-1.6E-19	Coulomb
E_{appl}	Applied potential		V
D	Diffusion constant		M^2/s
η	Overpotential ($E_{appl} - E^o$)		V
t	time		s
m	mass transfer coefficient		m/s
Φ	Potential difference inside a solution		V
R	Gas constant	8.3145	J/(K*mol)
T	(Absolute) temperature		K
δ	Thickness of the (diffusion) layer		m
α	Transfer coefficient		
$\vec{u}$	Fluid velocity		m/s
	Electrode geometry parameters		
w_g	Width of a gap		m
w_{ee}	Distance between the edges of two gaps		m
h	Distance between two electrodes		m
y_{max}	Position of the so-called semi-infinite bulk condition plane		m
h_g	The height of a gap		m
w_e	The width of an electrode		m
r_0	Radius of a cylinder, disk or spherical electrode		m
m_b	Number of microbands in an interdigitated electrode		
h_e	Height of an electrode		m

1 Introduction

1.1 Preface

A study Electrical Engineering at the University of Twente ends with a graduation project of 25 weeks. The official goals formulated by the faculty reads: *'The student has to solve an extensive, technical problem in an independent and justifiable way. This project must be done under guidance of one of the groups inside the faculty Electrical Engineering.'*

The first time I considered doing my graduation at the BIOS group was after attaining some courses in biomedical signal acquisition. It was the first course inside the biomedical major which was given in a really enthusiastic manner. Furthermore the interdisciplinary character of the group appealed to me. After some meetings with the tutor of the course, Wouter Olthuis, a suitable subject was found. The subject of this graduation project is in the field of electrochemistry. It will continue the work of post-doc V.A.T. Dam on redox cycling at ultra micro electrodes.

Of course, this report would not have been here without the help of other people. Therefore I would like to take a moment to thank some people. The first person on the list, at the same time the most invaluable person for my M.Sc. project is my daily tutor Wouter Olthuis. Although our contact was more weekly than daily I never experienced this as a problem, because even in the most busy weeks there was always the possibility to ask questions. Feedback was always 'strict but equitable' and it is always a pleasure if someone reads your work thorough. The second person who was invaluable for this project for his knowledge and experience in microfabrication was Johan Bomer. Without Johans work I would not have had multiple (working) wafers. Gabriel Sengo also helped on the processing of wafers, for which I am grateful. I also appreciate the support and patience of the Mesa+ staff while I was present inside the cleanroom. Jan Eijkel, thank you for your willingness to answer all kinds of questions and the new ideas that came up during our discussions. Erik Faber helped me with numerous things from discussions to instructions on preparing chemicals. Without his efforts to order reference electrodes and help to find a platinum counter electrode it would probably not have been possible to conduct appropriate measurements! I must thank Jorien van Loon for her time and work on making this report a little bit more readable in terms of English grammar. If you still find sentences which appear 'dutch' to you, this is due to my shortcomings and not her work. I would like to thank Prof. Dr. Uwe Karst, Prof Dr. Ir. Albert van den Berg and Dr. V.A.T. Dam for their willingness to be a member of my M.Sc. committee.

Last but most certainly not least I would like to thank my family for their support. I feel lucky knowing I have such kind and proud parents. Marije, without you I would probably not have enough energy to complete this project. You have the wonderful gift to clear my head from time to time and give me the necessary moments of rest. Our plans can now begin...

1.2 Research goals

Electrochemistry studies chemical reactions in which electrical charges or an electrical current are involved. An example of such reactions is the transformation of a certain ion *'A'* into ion *'B'* inside a solution. This reaction will take place if an electrode with a certain potential is present inside this solution. This transformation of ions from *'A'* to *'B'* can be reversed by a second electrode with the starting ion *'A'* as resulting product. The whole transformation can now start over again and some sort of feedback loop is formed. This loop is called the redox cycling effect. It can be used to amplify measured signals in a chemical way. The higher the number of redox cycles, the higher the amplification of the measured signal will be.

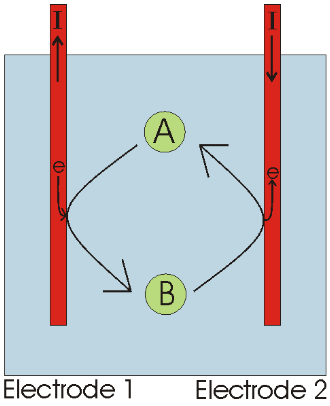


Figure 1.1: *Electrochemical reaction.*

This graduation project will discuss the redox cycling effect. The project continues the work of post-doc V.A.T. Dam [1]. In her research a redox cycling amplification factor of approximately 60 to 70 was observed. These redox cycling results can not be explained by known theory published by A.J. Bard et al. [2], X. Yang and G. Zhang [3] and many others [4–7]. The first goal of this project is to find a plausible explanation for the unusually high redox cycling amplification measured by V.A.T. Dam. To achieve this, finite element method simulations will be used.

With the acquired knowledge it might be possible to break the current world-record in terms of measured redox cycles. Therefore the second goal of this project is to design novel structures to achieve an even higher number of redox cycles. The most promising sensor designs will be fabricated in the cleanroom. Afterward, experiments will be conducted to test if there are differences between the predicted number of redox cycles and measured redox cycles.

Summarized, the following research goals are formulated:

1. Find a plausible explanation for the high number of redox cycles measured by V.A.T. Dam.
2. Design, production and testing of a novel sensor with the goal of gaining even higher redox-cycling amplification factors than the 60 to 70 times observed by V.A.T. Dam.

1.3 Outline

In the following section relevant theory on electrochemistry will be explained. Simulations on electrochemical sensors will be discussed in the succeeding chapter. In this chapter, a novel sensor design will be presented. Processing of this novel sensor will be described in the experimental section. Finally measurements on this fabricated sensor will be presented in the chapter 'results and discussion'.

2 Theory

2.1 Introduction

In this chapter theory will be discussed, which is used later on in simulations on electrochemical sensors. First some basics about electrochemistry will be explained. This basic knowledge is used to explain the redox cycling effect. Finally a small paragraph is devoted to cyclic voltammograms, which is a measurement technique used for measurements on electrochemical sensors.

2.2 Basics of electrochemistry

2.2.1 Introduction to electrochemistry

Electrochemistry deals with the effects of electrical and chemical interrelations. This is mostly studied in chemical reactions where electrical charges or an electrical current play a role. Applications are for example batteries, fuel cells, electrochemical sensors and even some types of displays. Main topics in the field of electrochemistry are electrode reactions and redox reactions in solutions. In the next paragraphs some basic concepts will be explained. This will just be a short summary, a detailed explanation is given in the book 'electrochemical fundamentals' of A. J. Bard [8] and the lecture notes from the course 'Biomedical signal acquisition' given by W. Olthuis [9].

2.2.2 Electrochemical cells

A typical electrochemical cell consists of two electrodes in contact with an electrolyte as in figure 2.1. The overall chemical reaction inside this electrochemical cell is made up of two independent half-reactions. These half-reactions describe the chemical changes at the two electrodes. Each half reaction is influenced by the potential between electrode and electrolyte. An example of such a half-reaction is the reduction of ferrite(III) into ferrite (II). Indicated in the figure by the transformation of O into R.

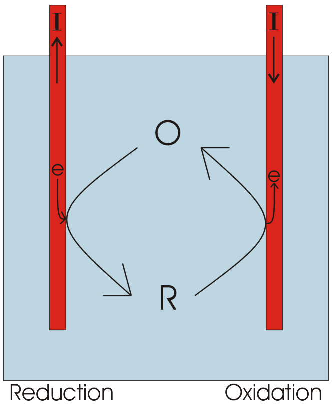


Figure 2.1: *Typical electrochemical cell.*

$$Fe^{3+} + e^- \rightarrow Fe^{2+} \tag{2.1}$$

This reaction can only take place if the electrons in the electrode have enough energy (W_e) to jump on a vacant state (ion) in the electrolyte. This means that the potential of the electrode (E_{appl}) has to be lowered. This is quite obvious if it is considered that the energy of the electrons is given by:

$$We = q \cdot E_{appl},\ (with\ q = -1.6 \cdot 10^{-19} C) \tag{2.2}$$

This results in a flow of electrons from the electrode to the ions in the electrolyte. The reduction current caused by the flow of electrons into the solution is (by convention) in the opposite direction.

When oxidation reactions are considered, the exact opposite will take place. For example the oxidation reaction of ferrite (II) into ferrite (III) is given by:

$$Fe^{2+} \rightarrow Fe^{3+} + e^{-} \tag{2.3}$$

This can only happen if the potential of the electrode is high enough, in that case the energy of the electrons in the electrode will be low enough, leading to more vacant states in the electrode. A ferrite (II) ion in the electrolyte will donate an electron to the electrode resulting in ferrite (III).

In the general form redox reactions are given by:

$$O + n \cdot e^{-} \rightleftharpoons R \tag{2.4}$$

Two other examples of redox reactions are given by the following reactions:

$$Cu_{(s)} \rightleftharpoons Cu^{2+}_{(aq)} + 2e^{-} \tag{2.5}$$

$$Mg_{(s)} \rightleftharpoons Mg^{2+}_{(aq)} + 2e^{-} \tag{2.6}$$

These reactions describe what happens if you put solid copper or magnesium into water. Both metals will produce ions leaving the metal surface. As a result the metal will be negatively charged. After a certain time an equilibrium is established at which the rate of ions leaving the metal surface is equal to the amount of ions recombining with electrons forming solid material again. However, the equilibrium point of both metals is not the same. Magnesium will donate more ions to the solution then copper. As a result, magnesium is more negatively charged than copper. A potential difference will rise between copper and magnesium.

The difference in potential of metals is normally expressed in reference to a normal hydrogen electrode (NHE). The standard potential (E^o) of magnesium in reference to a NHE is for example -2.37 volt and copper has a standard potential of +0.34 volt.

For situations in which the equilibrium is not set, the potential at the electrode is given by the Nernst-equation:

$$E_{Nernst} = E^0 + \frac{2.303 \cdot R \cdot T}{n \cdot F} \log \frac{C_{ox}}{C_{re}} \tag{2.7}$$

In which R is the gas constant (8.31 $J \cdot K^{-1} \cdot mol^{-1}$), T the absolute temperature in Kelvin and F the Faraday constant (96485 Coulomb/Mol). C_{ox} and C_{re} are the concentrations in Mol/m^3 of the oxidized and reduced species. E_{Nernst} is the potential of the (work)electrode with respect to the reference electrode.

2.2.3 Faradaic processes at electrodes

Faradaic processes involve reactions in which charges cross the metal-solution interface. The redox reactions described in the previous paragraph are examples of those processes. The reaction rate of a process is normally defined as as the fraction of the chemical that is formed or removed (in moles or mass units) per unit time per unit volume ($mol \cdot m^{-3} \cdot s^{-1}$). In the special case of reactions happening at electrode surfaces the reaction rate refers to unit interfacial area instead of unit volume. In this case the reaction rate has dimension ($mol \cdot m^{-2} \cdot s^{-1}$). This is equal to a flux of molecules into the solution.

The reaction rate is influenced by a number of factors.

1. Electron transfer at the electrode/solution interface.
2. Mass transfer of the ions in the solution (Explained in detail in paragraph 2.2.5).
3. Chemical reactions preceding or following the electron transfer.
4. Surface reactions such as adsorption, desorption or crystallization.

Most often one of these factors is slowest and therefore the rate-determining process. It is important to notice that the electron transfer is a process happening at a surface, while the mass transfer is a process happening in the bulk of the solution. The first two factors are most important and relevant in the simulations showed in the next chapter, therefore these two factors are treated in detail in the next two paragraphs.

2.2.4 Interfacial electrode reactions as the rate limiting step

Lets have a look at the reaction given by equation 2.4 once more. This bidirectional reaction can be split into a forward and backward form:

$$O + n \cdot e^{-} \xrightarrow{k_f} R \tag{2.8}$$
$$R \xrightarrow{k_b} O + n \cdot e^{-} \tag{2.9}$$

The forward and backward reactions are characterized by the rate constants k_f and k_b respectively. Suppose the forward and backward reaction rates on a single electrode are given by:

$$N_f = k_f \cdot C_{ox}(x = 0, t) \tag{2.10}$$
$$N_b = k_b \cdot C_{re}(x = 0, t) \tag{2.11}$$

Assuming that the electrode is positioned at x=0.

These reaction rate constants at the electrodes are described by the butler-volmer reaction kinetics.

$$k_f = k_s e^{-\alpha(E_{appl} - E^0)\frac{F}{RT}} \tag{2.12}$$
$$k_b = k_s e^{(1-\alpha)(E_{appl} - E^0)\frac{F}{RT}} \tag{2.13}$$

With k_s the standard rate constant and α the transfer coefficient. The standard rate constant is actually the reaction rate constant which can be found if the forward and backward reaction are in equilibrium and the concentrations of C_{ox} and C_{re} are equal. The transfer coefficient is a measure of symmetry between the forward and backward reaction. In symmetrical reactions α has a value of 0.5. $E_{appl} - E^0$ is also known as the overpotential η.

The net reaction rate is then the combination of both reaction rates given by:

$$N_{net} = k_f \cdot C_{ox}(x = 0, t) - k_b \cdot C_{re}(x = 0, t) \tag{2.14}$$

If the overpotential has a large value, one of the terms in equation 2.14 is negligible.

The flux of ions at the electrode surface is directly related to the electrode current via the following relation:

$$N(x=0,t)=\frac{i(t)}{nFA} \tag{2.15}$$

In which N is the flux, i the reduction or oxidation current, A the electrode area, F the Faradaic constant and n the valency of the reaction of interest[1].

The net current can be found by combining equation 2.14 and 2.15 resulting in:

$$i_{net}(t)=n\cdot F\cdot A\cdot(k_f\cdot C_{ox}(x=0,t)-k_b\cdot C_{re}(x=0,t)) \tag{2.16}$$

2.2.5 Mass transport as the rate limiting step

In general three different types of processes are distinguished, which are involved in ion movement.

1. Diffusion - Mass is moving because nature has a desire to distribute particles evenly throughout space.
2. Migration - Mass is moved as a result of a electrical field or potential difference in the electrolyte.
3. Convection - Mass is moved because of a external force. For example stirring in the electrolyte, or electrolyte moving through a channel as a result of pumping.

Mathematically this mass transport is described by the Nernst-Planck equation:

$$\vec{N_j}=\underbrace{-D_j\nabla C_j}_{diffusion}-\underbrace{\frac{n_jF}{RT}D_jC_j\nabla\Phi}_{migration}+\underbrace{C_j\vec{u}}_{convection} \tag{2.17}$$

Conservation of matter dictates that the change in concentration in time is equal to the change in flux over space. This relation is presented here in its one-dimensional form:

$$\frac{\partial C_j(x,t)}{\partial t}=-\frac{\partial N_j(x,t)}{\partial x} \tag{2.18}$$

In most of the applications the effect of migration is negligible since supporting non-electroactive ions are added to the electrolyte. This results in a very good conducting medium, comparable to a metal. Inside this conducting media it is almost impossible to generate significant electric fields or potential differences, because the ions in the solution will move in such a way that this electric field will be compensated almost immediately. Therefore electrolytes with high concentrations of ions can be seen as equipotential media.

If equation 2.18 is combined with the first term of the Nernst-Planck equation, we get Ficks second law of diffusion:

$$\frac{\partial C_j(x,t)}{\partial t}=D_j\frac{\partial^2 C_j(x,t)}{\partial x^2} \tag{2.19}$$

---

[1] n indicates the number of electrons needed to oxidize or reduce one ion in the reaction of interest

The sensors that will be described throughout this thesis can have both macroscopic and microscopic dimensions. It is interesting to know that a sensor working in the macroscopic domain has completely different behavior from a sensor in the microscopic domain. this is especially visible in the relations for the current of such a sensor. Therefore some general current relations of well-known sensor geometries are shown in the following paragraphs.

2.2.6 Macroscopic electrodes

To solve Ficks second law of diffusion for normal macroscopic electrodes, three boundary conditions are necessary:

$$\begin{aligned} C_j(x,t=0) &= C_j^*, \quad Homogeneity\ of\ solution\ at\ the\ start && (2.20)\\ \lim_{x\to\infty} C_j(x,t) &= C_j^*, \quad semi-infinite\ solution && (2.21)\\ C_j(x=0,t) &= 0, \quad for\ t>0;\ All\ ions\ consumed\ by\ the\ half-reaction && (2.22) \end{aligned}$$

C_j^* is the bulk concentration which is present at the beginning of the experiment.

Using Laplace transformation on Ficks second law of diffusion with respect to t, equation 2.19 becomes:

$$s\bar{C}_j(x,s) - C_j(x,0) = D\frac{d^2\bar{C}_j(x,s)}{dx^2} \tag{2.23}$$

The bar above the C_j is used to emphasise that it is a Laplace transformed variable. Using the first boundary condition the following (ordinary) differential equation is formed:

$$\frac{d^2\bar{C}_j(x,s)}{dx^2} - \frac{s}{D}\bar{C}_j(x,s) = -\frac{C_j^*}{D} \tag{2.24}$$

Solving this differential equation leads to:

$$\bar{C}_j(x,s) = \frac{C_j^*}{s} + A'(s)e^{-x\sqrt{s/D}} + B'(s)e^{x\sqrt{s/D}} \tag{2.25}$$

With use of the (Laplace transformed) second boundary condition it can be seen that *B'(s)* has to be zero. The final expression can be found by using the (Laplace transformed) third boundary condition to find an expression for *A'(s)*. The result looks like this:

$$\bar{C}_j(x,s) = \frac{C_j^*}{s} - \frac{C_j^*}{s}e^{-x\sqrt{s/D}} \tag{2.26}$$

The flux of species C_j is equal to the change in concentration over x. From equation 2.15 the relation between flux and current is known leading to:

$$N_j = \frac{i_j(t)}{nFA} = D_j\frac{\partial C_j(x,t)}{\partial x} \quad |x=0 \tag{2.27}$$

Laplace translation of equation 2.27 leads to:

$$\frac{\bar{i_j}(s)}{nFA} = D_j\frac{\partial \bar{C}_j(x,s)}{\partial x} \tag{2.28}$$

If the derivate of equation 2.26 is calculated with respect to x and substituted into 2.28, the result is:

$$\bar{i_j}(s) = \frac{nFAC_j^*\sqrt{D_j}}{\sqrt{s}} \tag{2.29}$$

Transformation back to the time domain gives the final solution which is known as the Cottrell equation:

$$i(t) = nFAC_j^*\sqrt{\frac{D_j}{\pi t}} \tag{2.30}$$

It is also interesting to look at the equation derived by inverse Laplace transform of equation 2.26.

$$C_j(x,t) = C_j^*\left(1 - erfc\left(\frac{x}{2\sqrt{D_j t}}\right)\right) \tag{2.31}$$

Equation 2.31 describes the concentration profile as a function of the position and time. This is shown in figure 2.2 for various times.

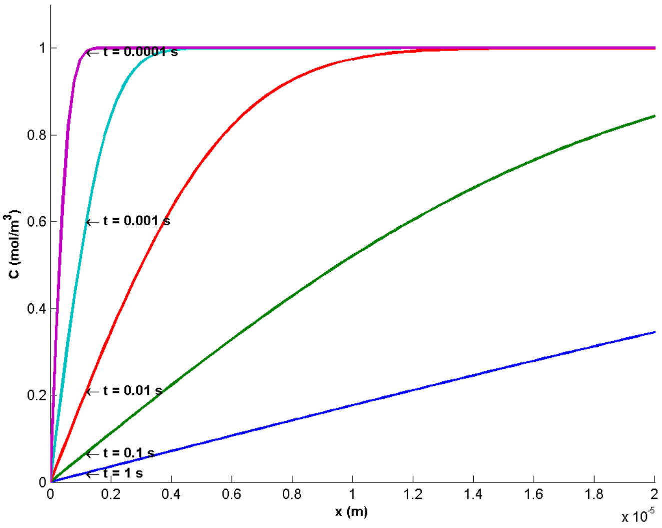


Figure 2.2: *Concentration profiles at various times with* C^*=*1* mol/m^3 *and* D=10^{-9} m^2/s.

From figure 2.2 it is seen that the concentration of species C_j is lower in close proximity of the electrode at x=0. At each time a region exists in which the concentration is lower than the original bulk concentration. This region is called the diffusion layer. This layer grows with time as more of species C_j gets depleted near the electrode.

2.2.7 Ultra Micro Electrodes (UME's)

Ficks second law of diffusion can also be solved for a small, spherical electrode. This geometry is known as an UME (ultra micro electrode). The current is now equal to:

$$i(t) = nFAC_j^* \sqrt{\frac{D_j}{\pi t}} + \frac{nFAD_jC_j^*}{r_0} \tag{2.32}$$

With r_0 the radius of the electrode. Important to notice here is this equation converges for large values of t into a steady state current of:

$$i_{ss}(t) = \frac{nFAD_jC_j^*}{r_0} \tag{2.33}$$

Due to the very small sizes of this electrode, this steady state is reached very soon.

Other configurations like disk, cylindrical or band shaped geometries for UMEs exist which give steady state phenomena as well. The steady state current of these electrodes all have a similar part and a part that is geometry specific. Therefore it is convenient to use the definition of the mass-transfer coefficient m_j. For each configuration the exact mass-transfer coefficients are listed in the book of Bard [8] and the mass-transfer coefficient is also shown in table 2.1 for convenience. Combined with formula 2.34 these mass-transfer coefficients give the steady-state current. As can be seen from the table, the band and cylinder UME are not real steady state electrodes, because the steady state current still has some small time dependence.

$$i_{ss}(t) = nFAC_j^* m_j \tag{2.34}$$

Table 2.1: *mass-transfer coefficient for different types of ultra micro electrodes.*

Band	**Cylinder**	**Disk**	**Sphere**
$\frac{2\pi D_j}{w \cdot ln(64 D_j t / w^2)}$	$\frac{2D_j}{r_0 ln(4 D_j t / r_0^2)}$	$\frac{4D_0}{r_0 \pi}$	$\frac{D_0}{r_0}$

2.3 Redox-cycling effect

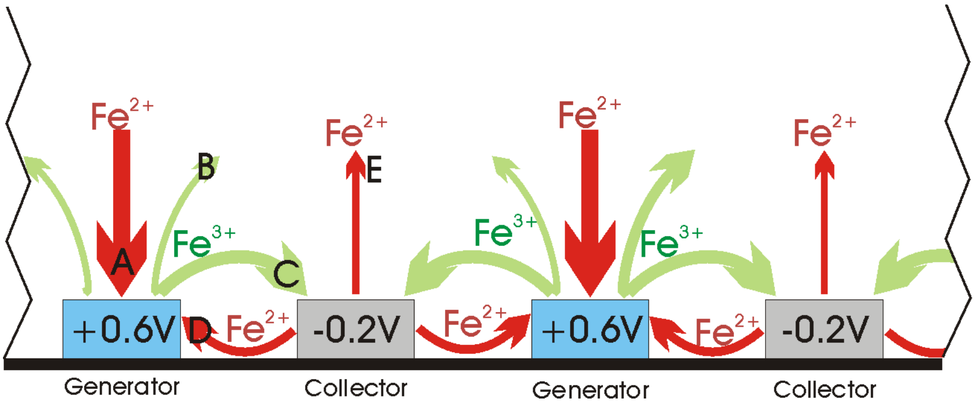


Figure 2.3: *Redox cycling between electrodes.*

Some interesting effects can be seen if two electrodes are placed in close proximity of each other such that their depletion regions overlap. This is illustrated in figure 2.3. In the figure the example of the well-known redox couple of Fe^{2+} and Fe^{3+} is used to explain the concept.

One of the electrodes is placed at a positive potential of +0.6V such that Fe^{2+} ions are oxidized. This electrode is called the generator electrode. The second electrode called the collector electrode is fixed at a potential of -0.2V, such that the oxidized Fe^{3+} ions are reduced into Fe^{2+}. The reduced Fe^{2+} ions can flow back to the generator electrode to be oxidized again. In this way the two electrodes work together creating a sort of feedback loop indicated by the arrows C and D. This loop is known as the redox-cycling effect. It can be used to amplify the measured current with an improved detection limit as a result.

Another possible field of application is to increase the selectivity of the measured signal. Assume that the generator electrode oxidizes two electrochemical active species. If the collector electrode is fixed at a potential at which only one of the two species will reduce again, only the signal of this reduced species will be amplified by means of the redox-cycling effect.

2.3.1 Interdigitated array microelectrode

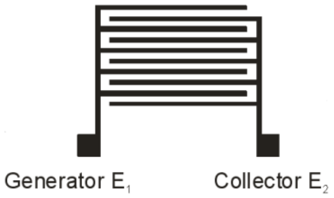


Figure 2.4: *IDA electrode.*

A structure used quite often in combination with the redox-cycling effect is the interdigitated array micro electrode (IDA). A typical IDA is shown in figure 2.4. Figure 2.3 used to explain the redox cycling effect is actually a cross-section of the IDA electrode. In order to get sufficient redox cycling it can be seen from figure 2.3 that it is important for the IDA to use small width's for the electrodes and to place the electrodes in very close proximity of each other. The closer they are, the more ions oxidized by the generator electrode will reach the collector electrode (indicated by arrow C) and less ions will diffuse into the bulk of the solution (indicated by arrow B).

The IDA is composed of a number of small microband electrodes. As stated in section 2.2.7, a microband is a known type of UME. The band UME only needs to have a very small size in one dimension to have a steady state behavior. The microband in the IDA has a small width to make it possible to place the electrodes close to each other. This makes it possible to choose a large value for the length, such that the measured signal will increase.

2.3.2 Collection efficiency and redox cycles

Two important factors determining the performance of an electrode setup using the redox cycling effect are the collection efficiency (CE) and the number of redox cycles (RC). Lets have a look at figure 2.3 once more. The amount of redox cycling depends on the amount of ions reaching the other electrode. Indicated in the figure by arrow C for Fe^{3+} moving from the generator electrode to the collector electrode and arrow D for Fe^{2+} moving in the opposite direction. All ions diffusing into the bulk of the solution (indicated by arrows B and E) will not participate in the redox cycling and can be considered lost. The collection efficiency is defined as the ratio between the amount of ions that reach the electrode versus the total amount of generated ions.

One ion of Fe^{2+} oxidized at the generator electrode will contribute one electron to the oxidation current of the generator electrode (i_g) and one Fe^{3+} to the solution. This Fe^{3+} can either diffuse into the bulk or reach the collector electrode. If it reaches the collector electrode it will be reduced, contributing one electron to the reduction current of the collector electrode (i_c). Following this reasoning it can be seen that the collection efficiency is also equal to the ratio of the collector current divided by the generator current, like it is defined in a paper of Bard [2]:

$$CE = \frac{i_c}{i_g} \tag{2.35}$$

The amount of redox cycles is defined as the amplification factor to the generator current due to the redox cycling effect.

$$RC = \frac{i_{g,rc}}{i_g} \tag{2.36}$$

In which $i_{g,rc}$ represents the generator current with redox cycling and i_g the generator current without redox cycling. This is generally determined in experiments by first measuring the generator current with the collector at open circuit voltage and secondly measuring the generator current with the collector at a potential such that redox cycling will take place.

Equation 2.36 can be explained by using the definition of collection efficiency. Assume that a certain amount x per unit time (mol/s) will be oxidized at the generator electrode. It will contribute with an equal amount of electrons to the generator current (assumed that the valency of the reaction is equal to one). Without redox cycling effect the generator current is simply equal to:

$$i_g = n \cdot F \cdot x \tag{2.37}$$

With the redox cycling effect, a certain amount of ions equal to $CE_1 \cdot x$ will reach the collector electrode, where they will be reduced. Again some ions are lost by diffusing into the bulk, but a certain amount equal to $CE_1 \cdot CE_2 \cdot x$ will reach the generator electrode resulting in additional electrons contributing to the generator current. CE_1 is the collection efficiency from generator to collector electrode, while CE_2 is the collection efficiency from collector to generator electrode. Of course this entire process can repeat itself leading to even more electrons contributing to the generator current. Therefore the generator current with redox cycling effect is equal to:

$$i_{g,rc} = n \cdot F \cdot (x + CE_1 \cdot CE_2 \cdot x + CE_1^2 \cdot CE_2^2 \cdot x + CE_1^3 \cdot CE_2^3 \cdot x + \ldots) \tag{2.38}$$

If equation 2.38 and 2.37 into equation 2.36 are combined the following expression for the amount of redox cycles is formed:

$$RC = 1 + CE_1 \cdot CE_2 + CE_1^2 \cdot CE_2^2 + CE_1^3 \cdot CE_2^3 + \ldots \tag{2.39}$$

For an infinite amount of redox cycles, equation 2.39 becomes:

$$RC = \sum_{n=0}^{\infty} CE_1^n \cdot CE_2^n \tag{2.40}$$

Equation 2.40 is known as the geometric series which can be simplified in the formula given by Niwa [10]:

$$RC = \frac{1}{1 - CE_1 \cdot CE_2} \tag{2.41}$$

CE_1 and CE_2 are the same if the electrodes have the same geometry.

In figure 2.5 the number of redox cycles is shown as a function of the collection efficiency (solid blue line). As shown by this figure that the number of redox cycles quickly increases when the collection efficiency approaches 100%. Also shown in the plot is the value for the RC if a small error of 1% is made with respect to the value for the collection efficiency (dashed red lines). It is important to notice that a small error in the determination of the collection efficiency willlead to large errors in the number of redox cycles. If, for example, a collection efficiency of 90.5% is measured with an error of 1%, the resulting error in the number of redox cycles is 10%. Therefore in most experiments the number of redox cycles is found with use of equation 2.36 instead of 2.41.

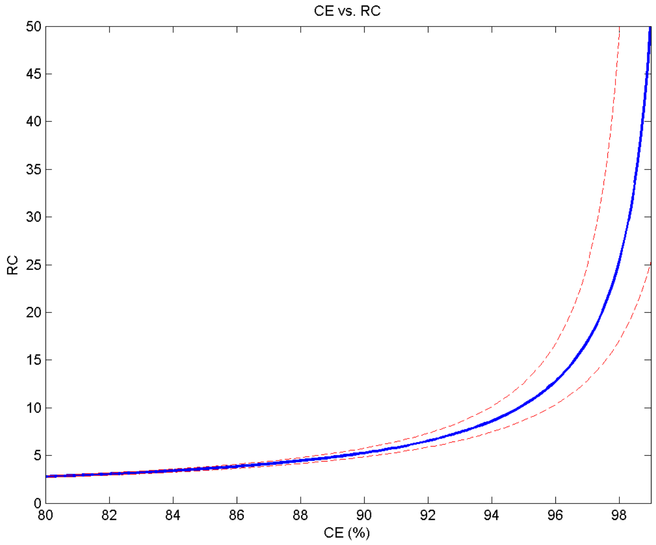


Figure 2.5: *Redox cycles as a function of the collection efficiency ($CE_1 = CE_2$ varied between 80 and 99% solid blue line). Red lines: Redox cycles found with an error of 1% in the determination of the value for CE.*

2.4 Metal-solution interfaces

In paragraph 2.3.1 it was already mentioned that the interdigitated electrode will have higher redox cycling numbers as soon as the gap between generator electrode and collector electrode is decreased. If this gap is decreased down to the nanometer scale, processes at the metal-solution interface will become dominant over bulk processes. As a result Ficks second law of diffusion to describe the mass-transport of ions inside the solution is no longer applicable. From electrodynamics it is known that any excess charges in a metal will be present at the surface. Ions in the solution will counterbalance the charges at the metal surface, giving rise to a double layer at the metal-solution interface. This double layer is typically in the order of 10 to 50 nm. Therefore significant potential differences are present at the metal/solution interface. In paragraph 2.2.5 Ficks second law of diffusion was derived with the assumption that in most applications the effect of migration is negligible since no significant potential differences are present in the bulk of the solution. This assumption is no longer valid if the gap between the generator and collector electrode is decreased to a size smaller than approximately 100nm.

2.5 Redox cycling in literature

The redox cycling effect has been a major subject of research in literature since the pioneering work of Sanderson and Anderson in 1985 [11]. Mathematically oriented work is presented by Aoki and Amatore. Aoki [5] contributed with different expressions describing the steady state current of an interdigitated electrode under different gap and electrode width's, some of which use the Schwarz-Christoffel transformation. He also stated that the expression describing the same steady-state current presented by Seddon [12] was just an approximation giving fairly good results (within 80%) only for $w_g/(w_e + w_g) < 0.5$. Amatore also uses the Schwarz-Christoffel transformation to reduce the IDA geometry into a facing double electrode problem. This is explained in detail in the excellent book edited by Rubinstein [6]. Phillips and Stone [13] calculated and found expressions for the collection efficiency of a two-disc and disc/ring configuration. The results indicate that the two-disc setup is not very promising in terms of collection efficiency. However, the disc/ring configuration shows a collection efficiency in the range of 70%.

One of the first people simulating the interdigitated structure is A.J. Bard [2]. He simulated the interdigitated array electrode structure and checked his results with experimental data. In the work of Bard, the electrode height was neglected. Strutwolf and Williams however, computed the diffusion effects of an IDA as a function of the height of the electrode. Unfortunately they did this in a thin layer cell setup, which is somewhat different from the setup this research intends to use. Jin et al. [14] also simulated the elevated interdigitated array electrode to check the validity of the steady state currents given by Aoki [5]. One of the results they find is that the ions travel more quickly from one electrode to another if the electrode height is increased. Also the steady-state current increases with increasing electrode height. The most recent simulation is shown in the work of Yang and Zhang [3]. They use the finite element modeling program FEMLAB from the company Comsol to simulate the IDA setup. They show that theoretical redox cycling values of up to 1000x times can be achieved when the gap size is decreased to 25nm. Although one might question if the model they use is still valid with such small dimensions, since migration and electrochemical double layer effects might start to play a significant role. One final interesting simulation has been shown by Beriet et al. [15]. He simulated a microdisc array geometry. For small times the current response was equal in shape to a single microdisc electrode. For larger times, when the diffusion layers of the individual electrodes started to overlap, the response was equal to a macroelectrode with an area equal to the sum of the individual electrodes.

The most interesting practical work is shown by the group of Paeschke et al. [16] and Niwa et al. [10]. Both made different interdigitated array electrodes with different electrode widths and gap widths. In both articles incredibly high collection efficiencies in the range of 98% to nearly 100% are shown. The group of Niwa showed redox cycling numbers of approximately 45 times, while Paeschke reaches 14 times. These numbers are so high that they can not be explained with what is known in theoretical literature. In the work of V.T. Dam however, similar results are shown with amplification factors as high as 60 to 70 times. Similar experiments are performed by Morita et al. [17],Tabei et al. [18] and Kurita et al. [19], but in flow channel or cell-like structures. They all focus on a low detection limit under flow conditions. It is shown that the redox cycling decreases under influence of convection. More practical applications with use of interdigitated array electrodes can be found in monitoring blood catecholamine [20], glucose sensing [21] and iron trace determination in ultra pure carbon [22]. Finally Mosbach et al. [23] made a setup which was able to measure very low concentrations of species like biotin and dopamine in a single nano droplet with the use of the redox cycling effect. Very recently M. Zevenbergen et al. [24] published a paper which presents a redox cycling amplification of approximately 400x. One of the goals of this project is to achieve high redox cycling amplifications and possibly break the current world-record, which means that a redox cycling amplification above 400x has to be achieved.

2.6 Cyclic voltammetry

Cyclic voltammetry is a common technique for measuring electrochemical reactions of electroactive species. A cyclic voltammogram measures the current(-time) response of an electrochemical cell in response to a linear sweep of potentials applied to the work electrode. This concept is illustrated in figure 2.6. The potential applied to the work electrode is varied between V1 and V2 (left). The resulting current response measured from the work electrode of a typical electrochemical cell is shown in in the right hand side of figure 2.6. At the start of the linear sweep, only non-faradaic currents are measured. As the potential increases, electrochemical reactions will occur. The concentration of electroactive ions at the surface of the electrode will decrease. Therefore the flux of ions to the electrode surface will increase, resulting in a higher current. If the concentration of electroactive species approaches zero, mass-transport will reach its maximum value (indicated by (E_p) in the figure). Finally depletion-effects will occur resulting in a decrease in current. Basically the same story holds if the potential sweep is reversed. A cyclic voltammogram consists not only of a sweep from V1 to V2, but also includes the reverse sweep. The speed at which the potentials are varied is called the scan-rate.

Figure 2.7 shows a typical CV-diagram of the redox-couple ferrocyanide/ferricyanide. This couple is a commonly used couple for cyclic voltammetry measurements for its known behavior. The ferro/ferricyanide-couple is also used in the paper of V.A.T. Dam [1] and X. Yang and G. Zhang [3] and in this paper for electrochemical measurements on a novel sensor structure (See chapter 5). The reversible chemical reaction of this couple can be described by:

$$[Fe(CN)_6]^{3-} + e^- \rightleftharpoons [Fe(CN)_6]^{4-} \tag{2.42}$$

Cyclic voltammograms can be characterized by a number of parameters. These parameters are also illustrated in figure 2.7 and will be discussed here. The peaks of a CV-diagram are always centered around the standard potential E^o. Therefore the following relation is valid for a CV of

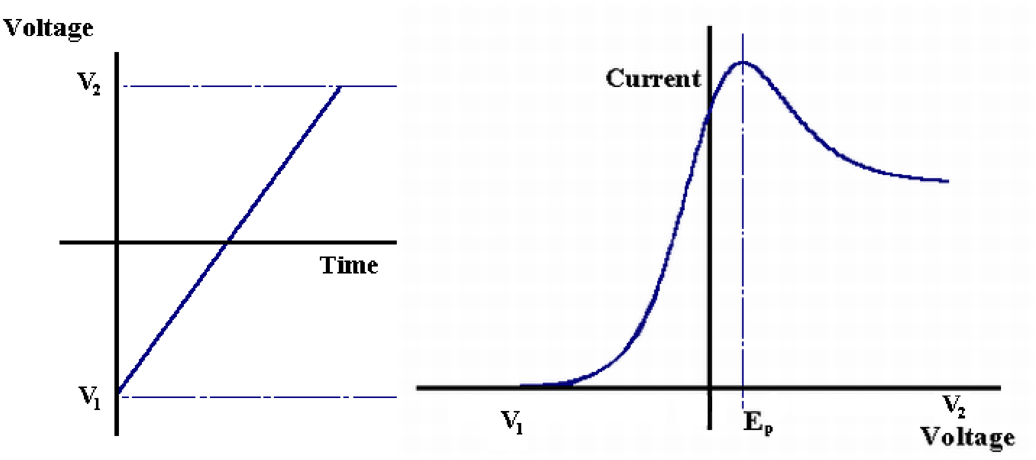


Figure 2.6: *Left: Linear potential sweep from V1 to V2. Right: Typical current response of an electrochemical cell to the linear potential sweep.*

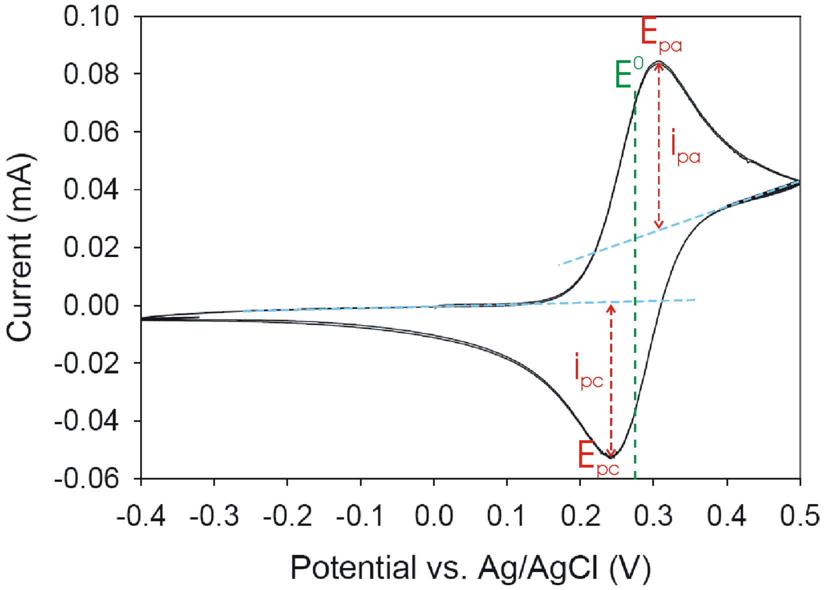


Figure 2.7: *Typical CV-diagram for the ferrocyanide/ferricyanide redox couple (From the paper of X. Yang and G. Zhang [3]).*

a reversible redox-couple:

$$E^0 = (E_{pa} + E_{pc})/2 \tag{2.43}$$

The standard potential is always fixed for a certain redox-couple, as described in section 2.2.2. For the ferrocyanide/ferricyanide-couple the standard potential is 0.265V with respect to a Silver/Silverchloride reference electrode according to X. Yang and G. Zhang [3]. The difference in peak potential of an ideal reversible redox-couple is always around $\Delta E = E_{pa} - E_{pc}$ =59mV. Therefore the position of the peaks with respect to the standard potential is fixed in normal cases. The peak currents, indicated by i_{pc} and i_{pa}, are dependent on factors like the scanrate, the bulk-concentration, the size of the electrode surface and the valency of the reaction of interest. The ratio of i_{pa} to i_{pc} should be close to one, although chemical reactions coupled to the electrode process can significantly alter the ratio of peak currents.

One phenomenon which can alter the shape of the CV-diagram significantly is called IR-drop or ohmic potential drop. In order to understand this effect, it is necessary to look more closely to

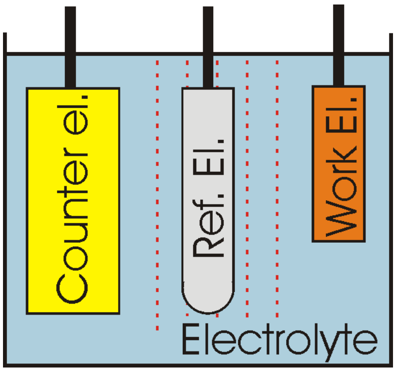


Figure 2.8: *Typical three electrode setup used in cyclic voltammetry.*

the electrode placement in the electrolyte. A typical three electrode setup is shown in figure 2.8. Current flows from the work electrode to the (platinum) counter electrode. In the middle a non-polarizable reference electrode is placed. Suppose the electrolyte has a certain ohmic resistance R_Ω. As a result of the resistance, a potential distribution will occur over the electrolyte from work electrode to counter electrode. Some approximate equipotential lines are indicated in figure 2.8 by the dashed red lines. The following relation is valid for the real (chemical) potential of the work electrode:

$$E_{w,real} = E_{w,appl} - i \cdot R_\Omega \tag{2.44}$$

The difference between real potential and applied potential is caused by the second term, which is called the IR-drop. The IR-drop will increase with higher currents or a higher electrolyte resistance. The major effect of the IR-drop in cyclic voltammetry is a shift in peak potential and an increase in peak separation. The effects of the IR-drop can be minimized by adding a supporting electrolyte of non-redox-active ions to the solution, resulting in a lower electrolyte resistivity. Another precaution which can decrease the IR-drop is a careful placement of the electrodes. The IR-drop will be lower if the reference electrode is placed in close proximity of the work electrode. However, it must not be placed too close, since that will introduce a shielding error.

The IR-drop effects will become more evident as the scan rate is increased. Suppose the metal-solution interface is characterized by the equivalent schematic shown in figure 2.9. R_Ω represents the ohmic resistance of the electrolyte, C represents the electric double layer capitance. For a capacitor and resistance in series, the following equation for the potential can be derived:

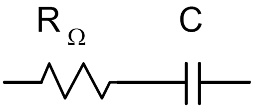


Figure 2.9: *Equivalent schematic.*

$$E_{w,real} = E_{w,appl}(1 - e^{(\frac{-t}{R_\Omega C})}) \tag{2.45}$$

For high scan rates (low t) or a large RC-constant, $e^{(\frac{-t}{R_\Omega C})}$ will become close to 1. This will introduce significant errors in E_{real} with respect to E_{appl}. Therefore it is best to decrease the scan rate if an IR-drop is suspected.

3 Simulations

3.1 Introduction

In this chapter the results of different simulation studies will be presented. First a description is given about the method and assumptions used in the simulations. Secondly results are shown of simulations of the interdigitated electrode as it was described in the previous section 2.3.1. These results include simulations on the IDA to check the validity of the used assumptions. Also simulations will be presented which vary some parameters on the IDA electrode to see their effect on the number of redox cycles. Finally some novel structures are discussed with the goal of gaining a higher number of redox cycles than the traditional IDA electrode.

3.2 Method

3.2.1 Software

The solutions are computed in the program Multiphysics version 3.2 (formerly known as Femlab) from the company Comsol. This program can be used as a module inside the program Matlab (version 7.1), therefore making it possible to do some more advanced simulations and to perform automated parameter variation studies. The Matlab code produced to perform the simulations are included in appendix D.

A finite element method (FEM) program like Comsol Multiphysics needs a number of steps to calculate a solution.

1. **Define a geometry**
   To decrease computing time, it is often recommended to simplify a three-dimensional problem into a two-dimensional problem when possible. Furthermore it is important to look at axes of symmetry. For the case of the IDA electrode for example, it is possible to simplify the model into a single set of one collector and one generator electrode.

2. **Define domain equations**
   The domain equations are a set of differential equations describing the physical phenomena occurring in the domain specified by the geometry.

3. **Define boundary conditions**
   To solve the domain equations it is necessary to define boundary conditions.

4. **Mesh element generation**
   Finally it is necessary to define a correct mesh of elements inside the geometry.

The domain equations, boundary conditions and definition of mesh elements will be described in more detail in the following three paragraphs. The definitions of the geometries will be given at the introduction of each simulation.

3.2.2 Domain

The domain equations describe what happens inside the specified geometry. For this thesis it means formulating equations describing the behavior of (electroactive) ions inside an electrolyte. The theory of mass transport will be used here as it was explained in section 2.2.5.

It will be assumed that there is no migration and convection in the solution [2]. Therefore the redox species obey only Ficks second law of diffusion:

$$\frac{\partial C_j}{\partial t} = D_j \nabla^2 C_j \tag{3.1}$$

These assumptions are only valid as long as the concentration of ionic species is high enough such that the electrolyte can be seen as an equipotential media (like a metal). This can be achieved by adding supporting ions to the solution.

Unless mentioned otherwise, some constants which are listed in table 3.1 are used throughout all simulations described in this chapter. For both the reduced species and the oxidized species the diffusion constant is considered the same with a value of $7 \cdot 10^{-10}\ m^2/s$. The initial bulk concentration C^* and the potentials of the electrodes are listed in table 3.1 as well. The generator is set at a potential of -0.2V, while the collector is set at a potential of 0.7V. The standard potential is equal to 0.265V. As a result the generator will reduce species C_{ox} into C_{re} and the exact opposite will take place at the collector electrode. All these values are the same as listed in the paper of Yang and Zhang [3]. Therefore fair comparisons between their simulation setup and the one described here can be made.

Table 3.1: *Constants used in simulations*

D_{re}	7E-10 m^2/s
D_{ox}	7E-10 m^2/s
C^*_{re}	0 mol/m^3
C^*_{ox}	5 mol/m^3
k_s	1.5E-4 m/s
R	8.31 J/K
T	290 K
F	96485 C/mol
E^0	0.265 V
E_g	-0.2 V
E_c	0.7 V
α	0.5

3.2.3 Boundary conditions

Boundary conditions define what happens at the edge of the geometry. For the electrodes the theory on faradaic processes at metal/solution interfaces will be used. This has already been explained in detail in section 2.2.4. The definitions for planes of symmetry and insulating surfaces will be described in this paragraph as well.

- **Generator electrode**
  At the position of the generator the flux of ions at the electrode surface is given by:

$$N_{red} = \vec{n} \cdot (D \nabla C_{red}) = k_{f,g} \cdot C_{ox} - k_{b,g} \cdot C_{red} \tag{3.2}$$

$$N_{ox} = \vec{n} \cdot (D \nabla C_{ox}) = -k_{f,g} \cdot C_{ox} + k_{b,g} \cdot C_{red} \tag{3.3}$$

  With $\vec{n}$ the normal vector of the plane of interest and the reaction rate constants $k_{f,g}$ and $k_{b,g}$ given by:

$$k_{f,g} = k_s \cdot e^{-\alpha \cdot F \cdot (E_g - E^o)/(R \cdot T)} \tag{3.4}$$

$$k_{b,g} = k_s \cdot e^{(1-\alpha) \cdot F \cdot (E_g - E^o)/(R \cdot T)} \tag{3.5}$$

---

[2] Except for the simulation described in paragraph 3.4.2 which does have convection, but an alternative domain description is included there.

- **Collector electrode**
  Via the same reasoning the flux of ions at the position of the collector electrode is equal to:

$$N_{ox} = \vec{n}\cdot(D\nabla C_{ox}) = k_{b,c}\cdot C_{red} - k_{f,c}\cdot C_{ox} \tag{3.6}$$

$$N_{red} = \vec{n}\cdot(D\nabla C_{red}) = -k_{b,c}\cdot C_{red} + k_{f,c}\cdot C_{ox} \tag{3.7}$$

  With the reaction rate constants given by:

$$k_{f,c} = k_s\cdot e^{-\alpha\cdot F\cdot(E_c-E^o)/(R\cdot T)} \tag{3.8}$$

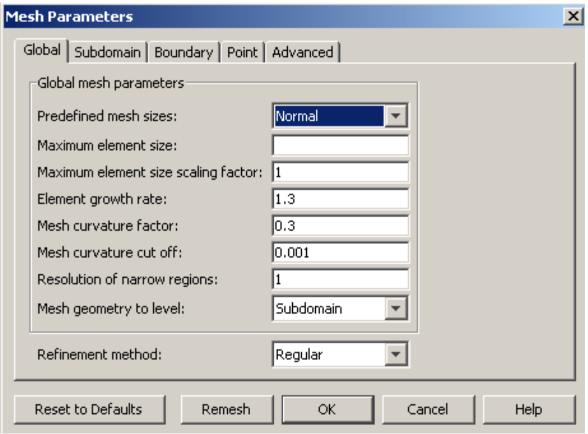
$$k_{b,c} = k_s\cdot e^{(1-\alpha)\cdot F\cdot(E_c-E^o)/(R\cdot T)} \tag{3.9}$$

- **Planes of symmetry or insulation**
  At the position of planes of symmetry or boundaries that insulate, a boundary condition is chosen which is equivalent to zero flux through the plane or boundary.

$$N_{ins} = \vec{n}\cdot(-D_j\nabla C_j) = 0 \tag{3.10}$$

- **Semi-infinite bulk condition**
  The reduced and oxidized species concentration is kept at the initial bulk concentration at a plane far away from the electrodes, thus creating a semi-infinite bulk condition. This condition is valid under certain conditions which are further explained in paragraph 3.3.4.

3.2.4 Mesh definition



Global mesh parameters

Predefined mesh sizes: Normal
Maximum element size:
Maximum element size scaling factor: 1
Element growth rate: 1.3
Mesh curvature factor: 0.3
Mesh curvature cut off: 0.001
Resolution of narrow regions: 1
Mesh geometry to level: Subdomain
Refinement method: Regular

Figure 3.1: *Dialog box to set the mesh parameters.*

Comsol Multiphysics has a lot of options which can be set to generate a mesh of elements, like it can be seen in the dialog box shown in figure 3.1. Most of the parameters like the *Mesh curvature cut off* and the *mesh curvature factor* define how meshes are generated around curved edges. These parameters will be left at their default values since no curved edges are used throughout this chapter. The most important parameter which has been changed to fit the simulations is the *maximum element size*. This parameter can both be controlled for the whole domain as well as to specify the maximum element size at the boundaries.

In section 2.2.6 an expression is found for the concentration profile around a macroscopic electrode. This concentration profile is shown in figure 2.2. As can be seen from this figure the concentration profile around an electrode is quite different from that of the bulk due to the diffusion layer. It makes sense to define a finer grid of elements in close proximity of the electrodes to make it possible to calculate the diffusion layer with sufficient resolution. This is accomplished inside the program by defining a very small maximum element size at the boundaries representing the electrodes. Another parameter called the *Element growth rate* lets the elements grow with increasing distance from the electrode boundaries. Unless mentioned otherwise, the *maximum element size* at the electrode boundaries is set at 10nm, while the maximum element size for the whole domain is equal to 0.5μm. The element growth rate is equal to 1.3, which means that the elements grow with a factor of 30%. In paragraph 3.3.3 the effects of the number of mesh elements will be studied in more detail.

3.2.5 Post-processing

The following equation describes the relation between the current from the electrodes and the flux at the electrode:

$$N = \frac{i}{n_j F A} = \vec{n} \cdot (D_j \nabla C_j) \tag{3.11}$$

In most simulations this flux is not equal over the entire surface (A) of the electrode. To get correct values it is necessary to integrate the flux over the entire electrode surface. In 2D simulations the result is a current with dimensions A/m calculated using the following relation:

$$i = F \int_0^{w_e} D_j \nabla C_j dx \tag{3.12}$$

Afterward the collection efficiency and the number of redox cycles are calculated using equation 2.35 and 2.41.

3.3 Interdigitated array electrode results

3.3.1 Geometry

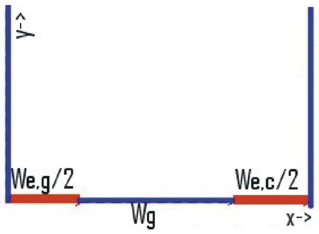


Figure 3.2: *Geometry of the IDA.*

The geometry used to simulate the IDA electrode is shown in figure 3.2. This figure is actually zoomed into the electrode area. The height of the vertical walls in the simulation are equal to[3] 50μm. In the real-world situation two vertical planes of symmetry can be found at the middle of the electrodes at which no net flux occurs. Therefore the figure shows only half of one generator and half of one collector electrode. The size of the lines representing the generator and collector electrode are indicated in the figure by $w_{e,g}/2$ and $w_{e,c}/2$. The size of the line representing the gap width between the electrodes is indicated by w_g. In the rest of section 3.3 the electrode width (denoted by w_e) is referring to the real world electrode, meaning that the lines representing the electrodes are actually half that size.

3.3.2 Concentration profile and current of the IDA electrode

A typical concentration profile at the interdigitated electrode is shown in figure 3.3. The concentration profile of both the oxidized species (3.3a) and the reduced species (3.3b) is shown. The arrows indicate the amount of diffusive flux. Although these kind of pictures can not be used to quantify the number of redox cycles it is interesting to see that redox cycling is taking place. A lot of oxidized species is diffusing into the direction of the generator electrode, while a lot of reduced species is diffusing into the direction of the collector electrode. Furthermore a depletion region of C_{ox} can be seen in close proximity of the generator electrode and a diffusion area where C_{re} is higher than the bulk in close proximity of the collector electrode.

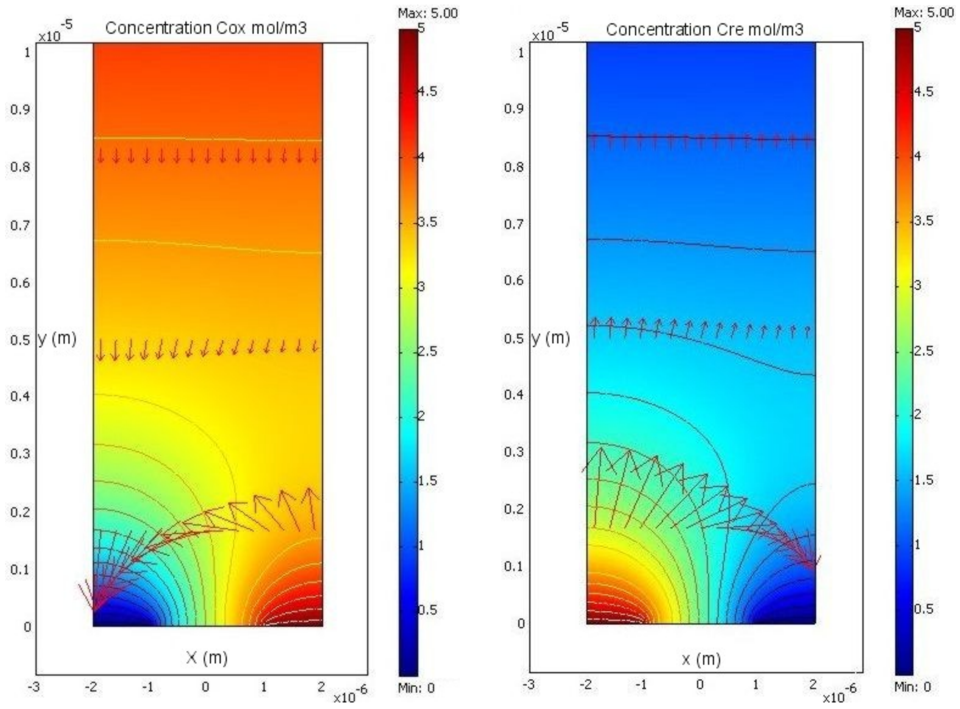
3.3.3 Mesh convergence study

In the paper of Yang and Zhang [3] it was shown that simulations in Multiphysics are sensitive to a correct mesh. A convergence study for the current at the generator electrode as a function of the mesh element size has been conducted on the geometry shown in figure 3.2. The electrode width and gap width were kept at 2μm. The position of the semi-infinite bulk boundary was set at $y = 50\mu$m. The maximum size of the mesh elements in the whole domain was set at 0.5 μm, while the maximum element size at the electrodes was varied between 1 nm and 0.1μm. A mesh was generated and the total number of mesh elements was noted. A typical generated mesh looks like the one shown in figure 3.4a. The results are shown in figure 3.4b and table A.1 of the appendix.

In figure 3.4a it can be seen that the number of mesh elements is indeed larger close to the electrodes. It is also shown that indeed no elements larger than 0.5μm are generated by the program. From figure 3.4b it is shown that the current of both the generator and collector current varies with different numbers of mesh elements. The generator current is larger than that of the collector electrode. Due to the unequal bulk concentrations of oxidized and reduced species. With use of equation 2.35 the number of redox cycles can already be calculated for this

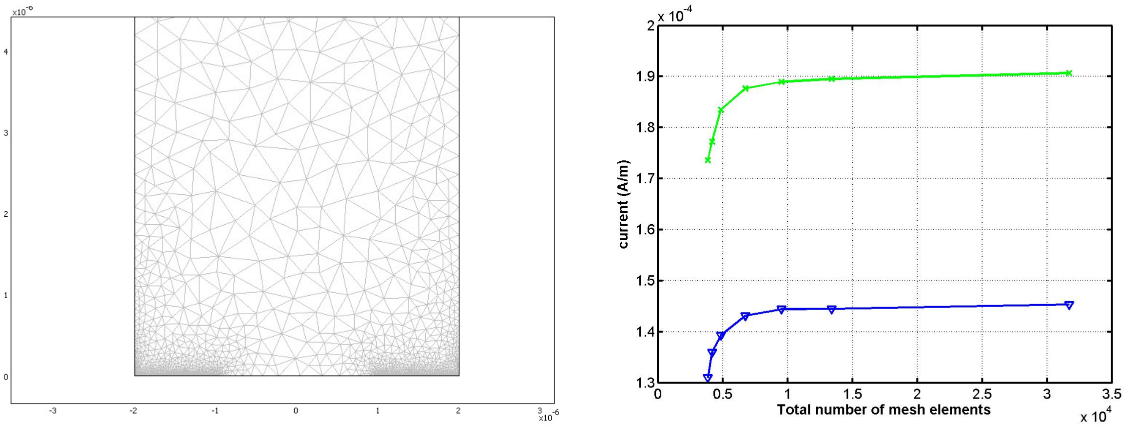
---

[3] Except for the simulation described in section 3.3.4 where this height is studied.



(a) Concentration C_{ox} (b) Concentration C_{re}

Figure 3.3: *Concentration profiles in a typical IDA simulation.*



(a) A typical mesh generated for the IDA geometry with the max. element size of the domain at $5 \cdot 10^{-7}$m and the max. element size of the electrode boundaries at 10^{-8}m. w_g and w_e are both 2μm.

(b) Current at the generator [x] and collector [∇] electrode vs. the nr. of mesh elements

Figure 3.4: *Mesh convergence study*

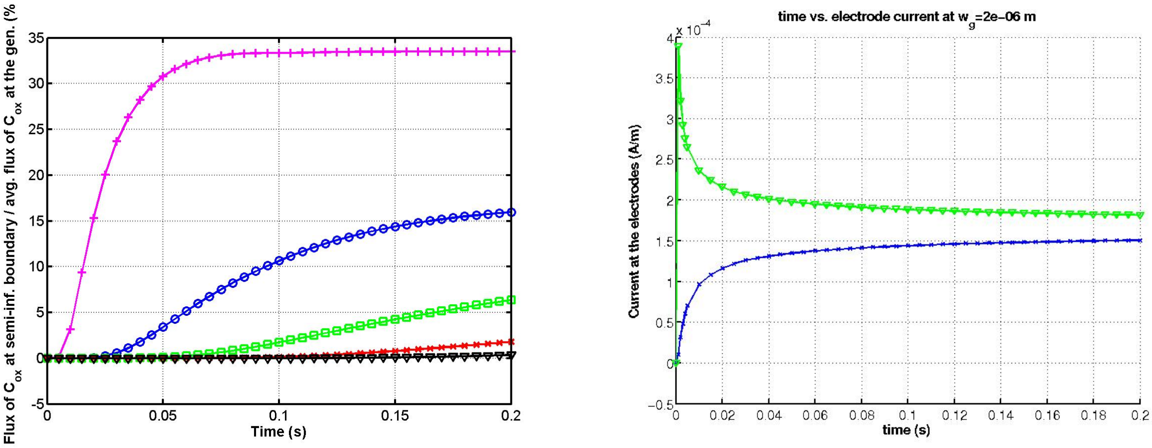
particular geometry. This is not done here for now, because these results are discussed later in more detail in section 3.3.5.

From these results it can be concluded that the current converges to steady values as soon as the number of mesh elements is higher than 10000, which corresponds to a maximum element size of 5nm at the electrodes. This is similar to the results found by Yang and Zhang [3]. Since the geometry is changed in later simulations on both the IDA electrode and other novel structures the number of mesh elements is less relevant. It is more important that there is sufficient resolution in close proximity of the electrodes where the largest differences in concentration can be seen. Therefore the maximum element size at the electrodes is kept at 5nm in future simulations unless otherwise mentioned. Although it is important to keep in mind that a similar convergence study needs to be conducted for every new geometry.

3.3.4 Steady-state analysis

In all models a certain plane far away from the electrodes is set equal to the initial bulk concentration, thus creating a semi-infinite bulk boundary (SIBB) condition. The electrodes under investigation are however not real steady-state electrodes like is stated in section 2.2.7 about ultra micro electrodes. Therefore it is necessary to choose a certain time at which the current of the electrode is assumed to be at quasi steady state. If this time is chosen to be too large the SIBB condition might not be valid anymore since ions generated at the electrodes have enough time to reach this boundary by diffusion.

A small study has been conducted in which the distance of the SIBB and the simulation time is varied while maintaining the rest of the geometry equal to the configuration of figure 3.2 with gap and electrode width's of 2μm. The results are shown in table A.2 and figure 3.5a. The percentages in the figure are calculated by dividing the total flux of ions of one species at the SIBB with the average flux of ions of that same species at the generator electrode, which is estimated as $2 \cdot 10^{-9}\ mol \cdot m^{-1} s^{-1}$.



(a) Flux at the semi-infinite boundary divided by the estimated flux at the electrodes ($2 \cdot 10^{-9}\ mol \cdot m^{-1} s^{-1}$) vs. time for y=10$\mu$m [+], y=20$\mu$m [o], y=30$\mu$m [⊟], y=40$\mu$m [x] and y=50$\mu$m [∇]

(b) Current at the generator [∇] and collector [x] electrode vs. time with $w_g = 2\mu$m, $w_e = 2\mu$m and $y_{max} = 50\mu$m.

Figure 3.5: *Steady-state analysis*

We see five lines representing the results from five simulations with different positions for the SIBB layer. As expected, the ions reach the SIBB faster if the SIBB is placed closer to the electrodes. The line corresponding to a position of y=10μm for the SIBB shows an area at which the flux is not changing any more for increasing time. This might be explained by the observation that an equilibrium is reached at which a fully developed concentration profile is seen between the SIBB and the electrodes. For larger distances between the electrodes and the SIBB the flux at the SIBB is negligible compared to the flux at the generator electrode.

Figure 3.5b shows the current at the IDA electrode with parameters $w_g = 2\mu$m, $w_e = 2\mu$m and $y_{max} = 50\mu$m. We see that both the collector current and the generator current converge into a quasi-steady state. At a time close to t=0 the current of the generator is approximately $4 \cdot 10^{-4}$ A/m, while the current of the collector is zero. This can be explained by looking at the initial bulk concentrations for both species at the start of the simulation. The generator has plenty of oxidized ions to reduce at the start while the collector electrode has to wait for reduced ions produced at the generator electrode before it can oxidize them again.

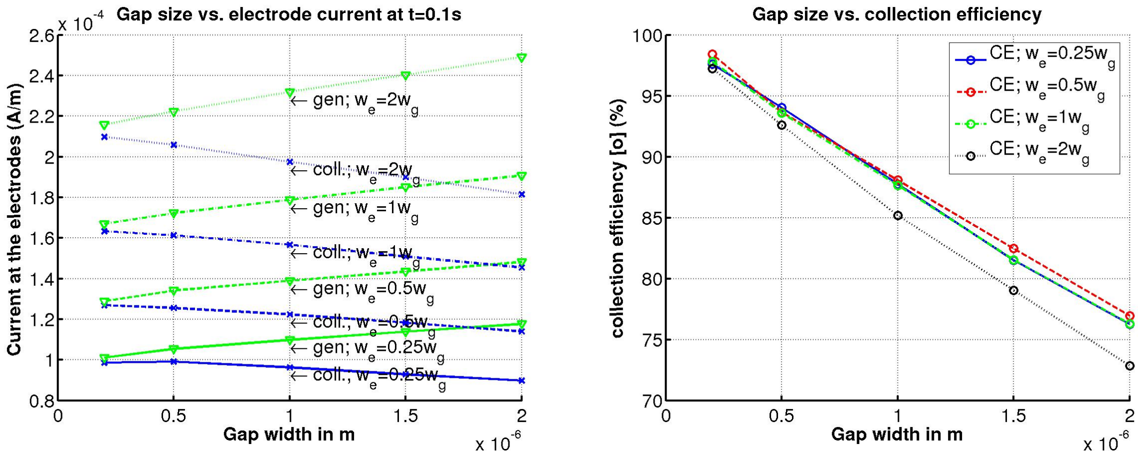
From this study it can be concluded that it is important to choose the right semi-infinite boundary distance in combination with the right simulation time. In further simulations a distance of 50μm is used in combination with a simulation time of 0.1 s unless mentioned otherwise. The current of both the generator and collector electrode is already (quasi) steady-state at this time, while the flux of ions at the semi-infinite boundary is very close to zero.

3.3.5 IDA with different sizes

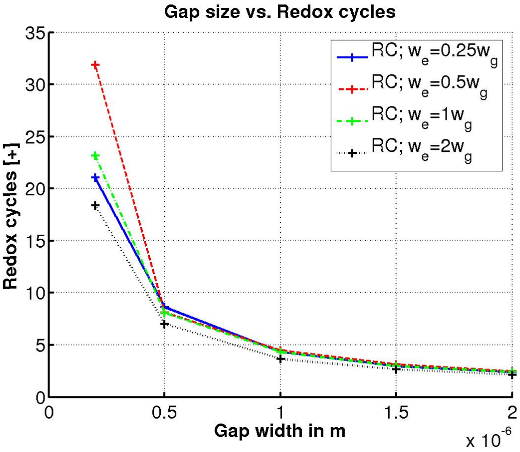
From previous simulations shown in literature [2, 3] it is known that the collection efficiency will increase with decreasing gap size between the generator and collector electrode. To confirm once more the validity of the model, the geometry shown in figure 3.2 is simulated with different gap and electrode widths. The results are listed in table A.3. As stated before, the number of mesh elements is kept at enough elements to ensure a converged result for the current. Since the electrode size and gap width was decreased up to 200nm, it was necessary to decrease the maximum mesh element size at the electrodes in this simulation down to 1nm. In the paper of Yang and Zhang [3] the gap size is decreased down to 25nm. Since double layers at the electrode surfaces have a typical thickness up to 50 nanometers, migration might start to play a significant role at gap width's smaller than 200nm.

In figure 3.6c four pairs of generator and collector currents are visible, each with a different ratio of w_e/w_g. The smaller this ratio, the smaller the electrode size. Therefore the electrode currents are smaller with a lower ratio. It can be seen that the ratio between a pair of collector and generator currents is getting closer to one at lower gap widths. Since the ratio between collector and generator current is equal to the collection efficiency according to equation 2.35, this is also visible in figure 3.6a where the collection efficiency is shown. The collection efficiency is getting closer to 100% with decreasing gap width. It is interesting to notice that the differences between the lines corresponding to $w_e = 0.25w_g$, $w_e = 0.5w_g$ and $w_e = w_g$ are small. Apparently the gap width is a more important factor determining the CE than the electrode width. Figure 3.6b shows the number of redox cycles. The highest number of redox cycles for the IDA are reached with the smallest gap width of 200nm and a ratio for w_e/w_g of 0.5.

From figure 3.6a it can be concluded that the collection efficiency is indeed increasing with decreasing gap width. The results found at $W_g = 1\mu m$ and $W_e = 1\mu m$ from this simulation (87.4%) are comparable with the results from Bard [2] (85%) and Yang [3] (87%). The small deviations between these simulations might be explained because of small differences between the choice for the position of the semi-infinite bulk boundary condition (50μm), the choice for the simulation time (0.1 s), the choice for parameters like the diffusion constant and bulk concentration and the number of mesh elements.



(a) Generator (∇), collector (x) current (A/m) (b) Collection efficiency (%)



(c) Redox cycles

Figure 3.6: *IDA electrode size study*

3.3.6 Elevated IDA electrode

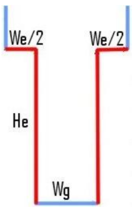


Figure 3.7: *Geometry of the elevated IDA.*

The next simulation involves the influence of the electrode height on the collection efficiency. The geometry is again the same as the previous simulations with the steady-state analysis and the mesh element study, so the electrode width and the gap width are both 2μm. Only this time the electrode is elevated as shown in figure 3.7. The electrode height is varied between 100nm and 7μm. The results are shown in table A.4 and figure 3.8.

Figure 3.8a shows the concentration profile C_{ox}. There is clearly some increase in redox cycling because the generated species are somewhat enclosed in the trench formed by the space between the elevated electrodes. From figure 3.8b it can be seen that the electrode current increases with larger electrode heights. This is actually what is to be expected since the electrode surface is enlarged. It is interesting to notice that the absolute difference between the generator and collector current remains constant with varied electrode heights. Since the current itself is increasing while the difference between generator and collector current remains the same, this results in an increased collection efficiency. This is also visible from figure 3.8c. The increase in collection efficiency leads to an increased number of redox cycles, like is shown in figure 3.8d. It appears as if the number of redox cycles has a linear relation to the electrode height.

An increased current with increased electrode height is also noticed in a paper published by J. Min and A.J. Baeumner [25]. Unfortunately this paper is focused on the signal to noise ratio and does not mention the collection efficiency or redox cycling. From these results it can be concluded that an increased electrode height leads to an increased number of redox cycles. This might actually be the explanation for the unusually high experimental results found by V.A.T. Dam [1] and Niwa [10]. The results of V.A.T. Dam are studied more in detail in the next paragraph.

3.3.7 The IDA electrode used by V.A.T. Dam

Figure 3.9 shows a sem-picture of the device used by V.A.T. Dam [1]. By using the scale printed in the figure, parameters of the geometry are estimated[4] at w_g=1μm and w_e=2μm. The trench is formed using deep reactive ion etching. Its depth is approximately 7μm according to the original paper. The metal is added by sputtering the trench under an angle of 45^o. It is therefore likely that the trench in the experimental setup was somewhat tapered leading to a smaller gap width at the bottom of the trench.

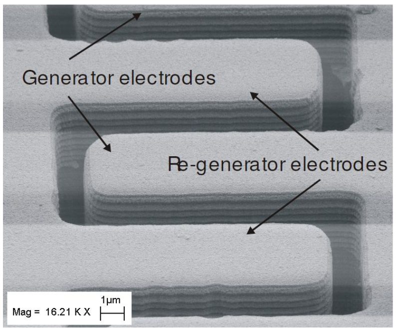
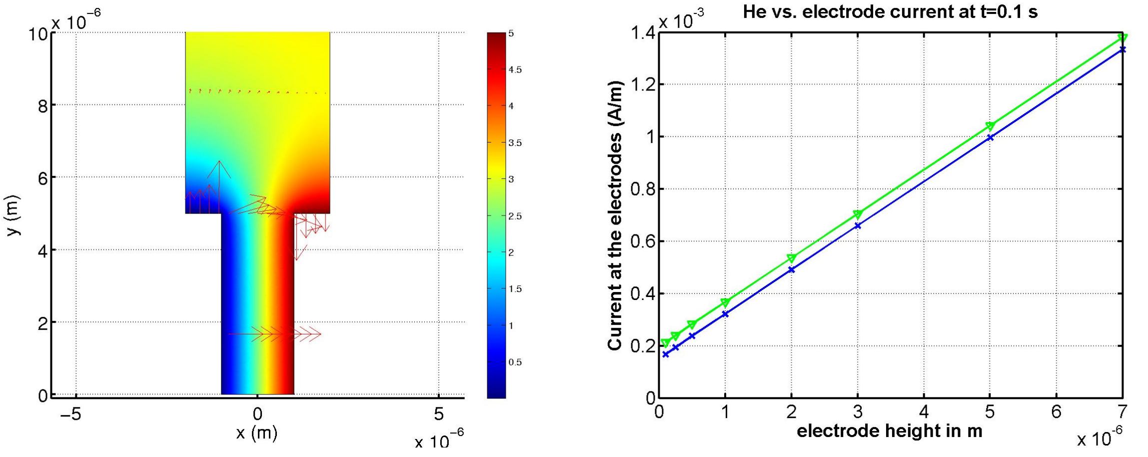


Figure 3.9: *SEM picture of the IDA.*

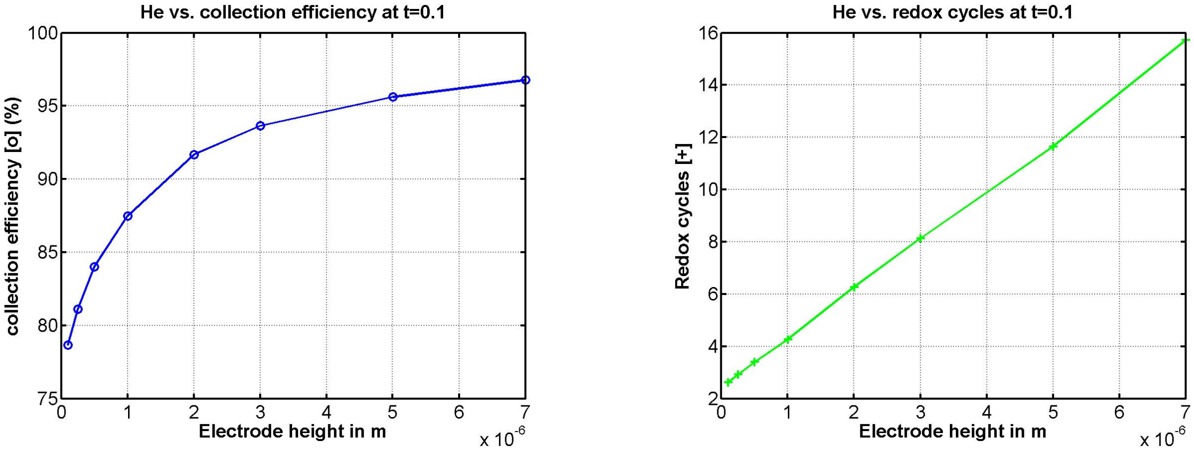
A simulation has been conducted with use of the above described parameters for w_g and w_e applied to the geometry shown in figure 3.8a. The only difference is that a tapered trench was used. The trench was kept at a width of 1μm at the top and a width of 0.5μm at the bottom. The results show a generator current of 3.507E-3 A/m and a collector current of 3.483E-3 A/m leading to a collection efficiency of 99%. This is equivalent to a redox cycling number of approximately 74 times. This result is very close to the 60 to 70 times found by V.A.T. Dam.

---

[4] The picture is taken at an angle. To do the best measurements it is advisable to measure in the horizontal direction.



(a) Concentration C_{ox} (mol/m^3); $h_e = 5\mu m$ (b) Generator (∇), collector (x) current (A/m)



(c) Collection efficiency (%) (d) Redox cycles

Figure 3.8: *Elevated IDA electrode study.*

3.4 Novel sensor structures

Some novel structures for redox cycling are considered. The first structure under study is the parallel plate electrode. This structure is interesting because it has in principle a collection efficiency of 100%, assuming that no ions can leave the area between the plates. Secondly, more advanced models are studied which all derive from the basic model of the parallel plate electrode.

3.4.1 Parallel plate electrode

The parallel plate electrode is the first and most simple novel structure. The geometry consists of a closed space like a channel or chamber. Two opposite parallel planes are used as the generator and collector electrode like is shown in figure 3.10. The most important parameter is the distance between the plates (h). The simplicity of this geometry makes it possible to calculate the results by hand.

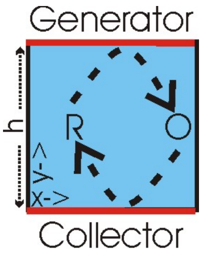


Figure 3.10: *Parallel plate setup*

Let's look again at the Nernst-Planck equations for diffusion:

$$N_j = -D_j \cdot \frac{\partial C_j}{\partial y} \tag{3.13}$$

Basically the same boundary conditions as in the previous cases are valid. After a certain amount of time the diffusion layers of both electrodes will be as large as the height of the channel and a steady-state situation is reached. There will be a fully developed, linear concentration profile between the two parallel plates in this steady-state situation. In that case equation 3.13 can be simplified into:

$$N_j = -D_j \cdot \frac{C_j^* - 0}{h} = \frac{-D_j}{h} \cdot C_j^* \tag{3.14}$$

From this equation it can be concluded that the flux (N) is linearly dependent on the initial bulk concentration. The only problem with this setup is the effort that has to be taken to put the electrolyte inside the space between the electrodes.

3.4.2 Parallel plate with convection

If we take another look at the parallel plate setup from the previous paragraph, one can think of a pump at the entrance to insert a certain amount of electrolyte inside the channel. This pump can run continuously leading to contribution of convection to the total flux of electro active species. Equation 3.13 now gets an added convection term:

$$N_j = -D_j \cdot \frac{\partial C_j}{\partial y} + C_j \cdot \vec{u}(x, y) \tag{3.15}$$

This case with a convection due to pumping is simulated. The geometry used is shown in figure 3.11. The first micrometer of the channel indicated by w_{ins} is chosen to be a insulation to prevent strange simulation errors, since the inlet boundary is set at bulk concentration which might be in conflict with diffusion of species generated at the electrodes toward the inlet. The width of the electrodes (w_e) is set at an arbitrary value of 9 μm. The height of the channel (h) is 1 μm.

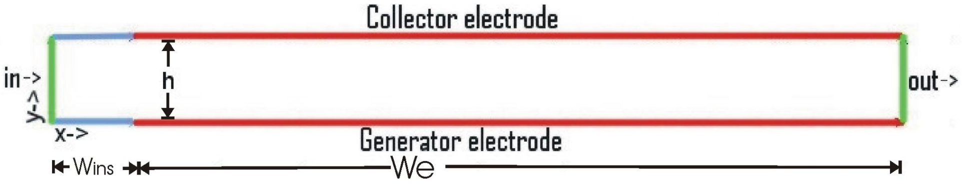


Figure 3.11: *Geometry of the parallel plate setup with continuous flow*

The boundary conditions at the electrodes are the same as explained in section 3.2.3 for the previous simulations with the IDA. The inlet of the channel is set at the initial bulk concentrations as listed in table 3.1. The outlet of the channel is set at a so-called convective flux boundary condition with an analytical expression describing laminar flow between two parallel plates:

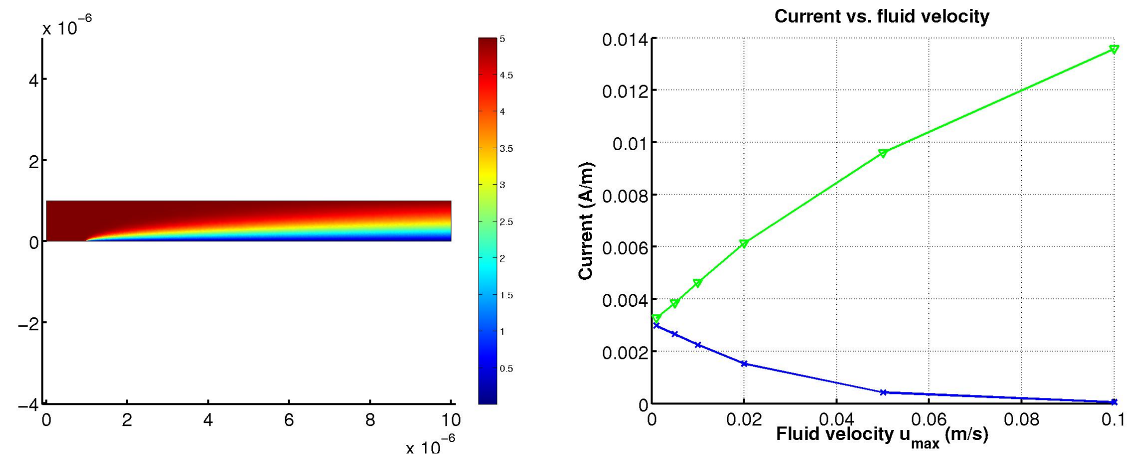
$$\vec{u} = (u_{max} \cdot \left[1 - \left(\frac{y - 0.5h}{0.5h}\right)^2\right], 0) \tag{3.16}$$

A steady-state simulation is used, in contrary to the previous time-transient simulations of the IDA electrode. This means that all concentration profiles are fully developed. The maximum mesh element size is set at 5nm at the electrodes. u_{max} is varied between 0.001 and 0.1 m/s. The results are shown in table A.5 and figure 3.12.

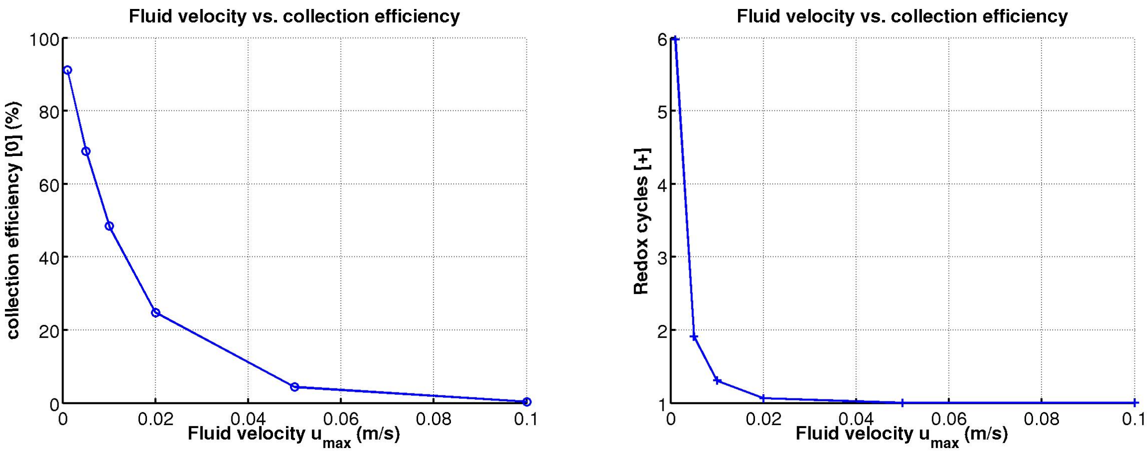
The concentration profile of C_{re} with a value of 0.05 m/s for u_{max} is visible in figure 3.12a. It can be seen that the initial bulk concentration is increasing toward the outlet of the channel. Virtually none of the generated ions reach the collector electrode at the top of the channel. For this particular geometry and flow velocity this might lead to a very small current for the collector electrode. This is indeed visible in figure 3.12b. The collector current is decreasing with increasing flow velocity. This is actually quite easy to explain, since the ions spend less time in the space between the two electrodes.

It is interesting to notice that the generator current is actually increasing with higher values for u_{max}. This is due to the increased contribution of convection to the total flux at the generator electrode. Since the generator current is increasing while the collector current is decreasing, the collection efficiency is also lowered with higher values for the fluid velocity. The collection efficiency is shown in figure 3.12c. The parallel plate with continuous flow performs really bad in terms of collection efficiency for all simulations with a u_{max} above 0.001 m/s. As a result the number of redox cycles is also very low, as can be seen in figure 3.12d.

In this simulation an arbitrary value of 9μm is chosen for the width of the electrodes. However, the decrease in collection efficiency can be compensated by increasing the channel length. Similar results are found in the work of Morita et al. [17], Tabei et al. [18] and Kurita et al. [19].



(a) Concentration C_{re} with $u_{max} = 0.05\ m/s$ (b) Generator (∇), collector (x) current (A/m)



(c) Collection efficiency (%) (d) Redox cycles

Figure 3.12: *Parallel plate with convection study.*

3.4.3 Parallel plate with passive bulk supply

The parallel plate configuration is ideal in terms of redox cycling, since the collection efficiency is equal to 100%. The drawback of this setup is the need for a pumping mechanism. Therefore the next simulation consist of a similar parallel plate setup. Only this time some holes are created through the top electrode. The bulk concentration will reach the area between the electodes via the holes.

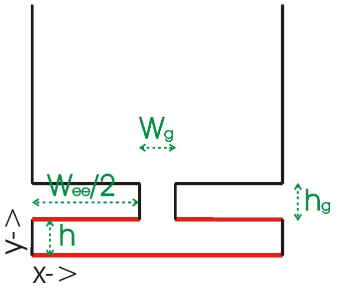


Figure 3.13: *Geometry of the parallel-plate setup with holes through the plates.*

The geometry of interest is shown in figure 3.13. Because of computing time a 2D approximation with only a single hole is simulated. This hole is present in the top electrode. It is possible to add a hole in the bottom electrode. However, a hole in the bottom electrode complicates the fabrication of the sensor with advanced back-end etching processes. A number of parameters can be varied like the diameter of the holes in the top electrode (w_g), the distance between the edges of two holes (w_{ee}), the distance between the electrodes (h) and the thickness of the insulating layer covering the top electrode (h_g). A few simulation parameters are the same as in the IDA electrode simulations. The maximum mesh element size is again 5nm at the electrodes. The simulation time is also 0.1s and the distance between the semi-infinite bulk boundary condition and the bottom electrode is 50μm.

Three different studies are shown in this section. First a study will be presented to find the best way to calculate the collection efficiency and the number of redox cycles for this particular sensor. Secondly a parameter study is shown to find the highest number of redox cycles. Finally a small study will be carried out to find the response of the sensor to variations in the bulk concentration.

Calculating the collection efficiency and the number of redox cycles

In previous simulations the collection efficiency was calculated by dividing the current at the collector electrode by the current of the generator electrode. Due to unequal area's of the electrodes in this simulation this is not a correct measure for the collection efficiency. There are two approaches to find the collection efficiency and the redox cycling number when dealing with unequal electrode areas.

The first approach is to do two simulations. One with the collector electrode switched on and one simulation with the collector electrode switched off. Meaning the collector electrode is set to a fixed potential in the first case and it is set to open circuit voltage in the second case. With both simulations the current at the generator electrode is calculated. The ratio between these two currents is equal to the number of redox cycles. The collection efficiency is determined indirectly by equation 2.41 of Niwa, assuming CE_1 and CE_2 are the same. In reality they are not, but the result is an averaged value for the collection efficiency.

The second approach is also calculated by conducting two simulations. In the first simulation the collector electrode is set to a oxidation potential, while the generator is set to a reduction potential. In the second simulation the applied voltages on the electrodes are switched in place. So the generator electrode becomes the collector electrode and the collector electrode becomes the generator electrode. In both simulations a collection efficiency is calculated by dividing the collector current with the generator current. The number of redox cycles can now be calculated using the already mentioned formula of Niwa (2.41).

A small study has been conducted to compare these two approaches. The results are listed in table 3.2. The geometry used in this study has the following dimensions: $w_{ee} = 4\mu m$, $w_g = 1\mu m$, $h = 1\mu m$, $h_g = 0.1\mu m$ and $y_{max} = 5\mu m$.

Table 3.2: *Collector and generator currents calculated for the two methods for determining the number of redox cycles and the collection efficiency.*

	Coll. off, Bottom plate=coll. Top plate=gen.	**Coll. on, Bottom plate =coll. Top plate =gen.**	**Coll. off, Bottom plate=gen. Top plate=coll.**	**Coll. on, Bottom plate=gen. Top plate=coll.**
i_g (A/m)	9.9E-5	1.6E-3	7.05E-5	1.5328E-3
i_c (A/m)		1.49E-3		1.5728E-3
$CE = i_c/i_g$		CE_1=93.16%		CE_2=102.61%

If we use the generator currents listed in the first two columns and use equation 2.36 to calculate the number of redox cycles we get 16 as a result. The same calculation with the electrodes switched in place (column 3 and 4) leads to 21.7 redox cycles. If we use the second approach with use of both collection efficiencies CE_1 and CE_2 in combination with equation 2.41 we find 22.68 redox cycles.

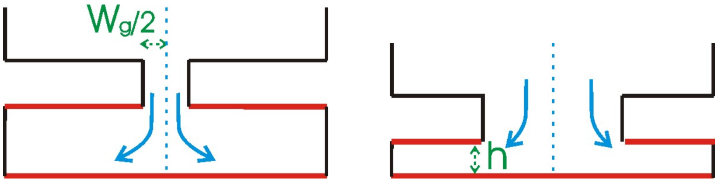
From this small study it can be seen that the number of redox cycles calculated using the first method is dependent on which electrode is chosen to be the generator or collector. This is not a desirable effect. Secondly the way in which the collection efficiency is calculated in the first method is not a good one, since it does not follow the definition given by Niwa. Therefore the second approach is used in the rest of this paragraph.

Parameter study

The next simulation involves a parameter study of this type of sensor. The geometry parameters like the gap width (w_g), the plate distance (h) and the distance between the gaps (w_{ee}) are varied. The gap height (h_g) determined by the thickness of the insulating layer covering the top electrode is set at 500nm. From experiments it is found that the gap height has little influence on the performance of the device. The results are shown in figure 3.15. We see a clear increase in the number of redox cycles as soon as the edge-to-edge distance between the holes is increased. This is actually what might be expected since the geometry is more and more identical to the parallel plate setup without holes.

A similar increase in the number of redox cycles can be seen if the distance between the plates is decreased. If the plate distance is much smaller than the gap width, the first parameter becomes the dominant factor determining the inlet area through which the ions diffuse into the work space of the sensor. Figure 3.14 is included to clarify this concept. Lets have a look at the line with parameters $w_g = 4\mu m$, h=0.5μm and the line with parameters $w_g = 1\mu m$, h=1μm from figure 3.15. The first line has the plate distance (h) as dominant factor (fig. 3.14b). The second line has the gap width (w_g) as the dominant factor (fig. 3.14a). Since one gap has to provide ions for both the right hand side and the left hand side of the sensor, it is important to take half of the gap width as a measure for the sensor inlet area. The sensor inlet area is in both cases determined by a dominant factor of 0.5μm. That actually explains why the two lines mentioned have similar values. As a general rule of thumb it can be said that the ratio between the inlet area and the sensor area determines the redox cycling number and the collection efficiency.

Figure 3.15 predicts redox cycling numbers up to approximately 400x for the device discussed in this subsection. The linear trend indicates that even higher number of redox cycles can be achieved by increasing the gap distance (w_{ee}) or decreasing the plate distance (h). Therefore the parallel plate with passive bulk supply structure looks like a promising candidate for achieving even higher numbers of redox cycling than the 60 to 70 times measured by V.A.T. Dam.



(a) w_g as dominant factor.

(b) h as dominant factor.

Figure 3.14: *Dominant inlet factors determining the performance of the device.*

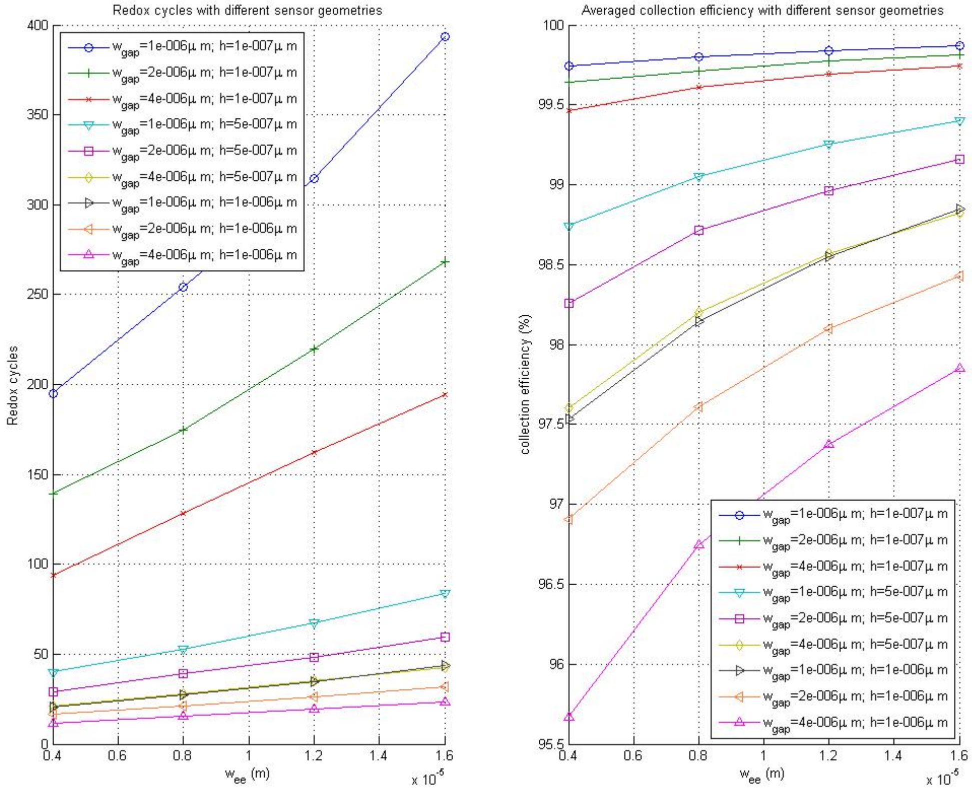


Figure 3.15: *Collection efficiency and redox cycles with different gap sizes (w_{gap}), electrode distances (h) and edge-to-edge distances (w_{ee}) of the holes.*

Bulk variation study

Another interesting simulation is the variation of the semi-infinite bulk concentration in time. Normally the semi-infinite bulk boundary condition is set at the bulk concentration (C^*) throughout the simulation. In the steady-state analysis it was already shown that for longer simulation times a (fully-developed) concentration profile between the sensor and the semi-infinite bulk boundary is formed. The current found at the electrode is completely dictated by the upper bulk boundary condition and time seems frozen since no changes appear in the concentration profile. In this simulation the bulk concentration at the upper boundary is lowered from the original C^* to zero in five steps. The position of this semi-infinite bulk boundary is still 50μm. The simulation time is increased to form a steady-state concentration profile between each step. The result for the generator electrode current is shown in figure 3.16. The geometry used in this simulation has parameters w_g=1μm, w_{ee}=16μm, h=1μm and h_g=0.5μm. The bottom electrode is used as generator, as a consequence the top electrode is the collector.

These results show that a new equilibrium is set within approximately 16 seconds. It is important to notice that this time needed to set a new steady state is mostly defined by the diffusion time from the position of the semi-infinite bulk layer to the sensor. Unfortunately it is not possible to adjust the simulation model in such a way that part of the concentration outside the sensor changes in time by a user-defined function throughout the simulation. This means it is not possible to calculate the response time of the sensor when changes occur in the bulk concentration just outside the sensor work area. However, the result with the varied concentration at the semi-infinite bulk boundary indicate that the time needed for the sensor to reach a new steady state is at least less than 16 seconds for this particular geometry. Similar studies with other geometries are conducted, but the differences between them were minimal. This can also be explained by the contribution of the diffusion time from semi-infinite bulk boundary to the sensor, which is larger than the diffusion time inside the sensor.

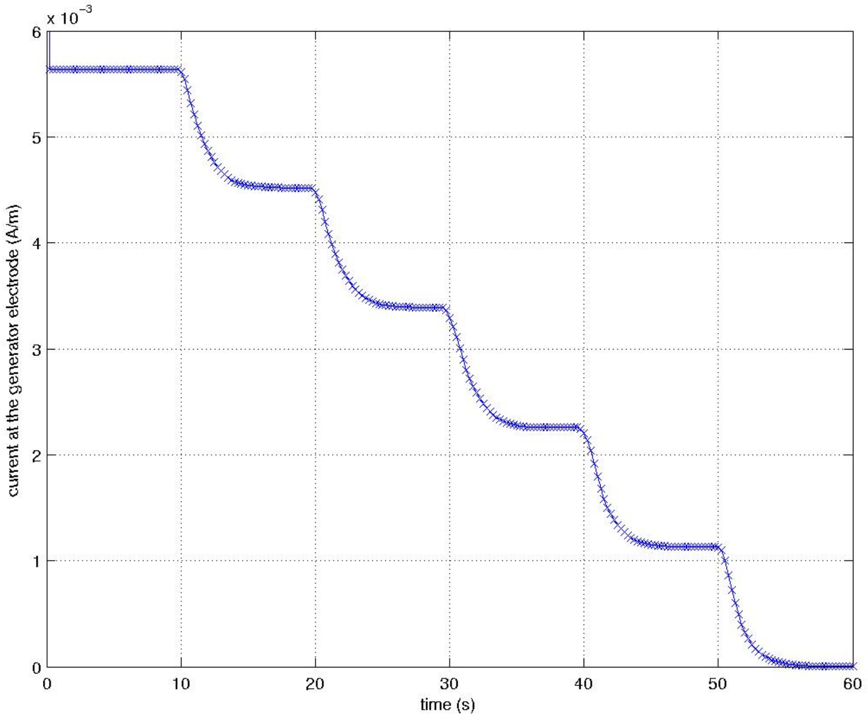


Figure 3.16: *Current of the generator electrode (A/m).*

3.4.4 Cyclic Voltammograms of the parallel plate sensor with passive bulk supply

In figure 2.7 of section 2.6 a typical CV-measurement of the ferrocyanide/ferricyanide couple is shown. This is a measurement with a single electrode. The novel sensor structure with two parallel plates discussed here (see figure 3.13) has two work electrodes placed at close proximity of each other. Therefore it is possible to do advanced cyclic voltammetry experiments. Three different types of experiments have been studied.

1. The first experiment applies a potential sweep to a single electrode.
2. With the second experiment a CV measurement is calculated with the two electrodes connected together. As a result a potential sweep is applied to both electrodes simultaneously.
3. With the third experiment a potential sweep is applied to one electrode, while the other is set to a fixed potential of +0.7V. The second electrode with the fixed potential will generate oxidized ions. It is expected that the current of the first electrode will have a higher reduction current for negative potentials due to a higher availability of oxidized ions and an amplification of the current due to redox cycling. Redox cycling amplification will increase the current from the second electrode with the fixed potential as well.

The potential sweep used in all three experiments starts at the standard potential of E^0=0.265V. The potential is increased to +0.8V and from there it is reversed to -0.3V. Finally the potential is switched back to 0.265V. The sweeprate is equal to 20mV/s. The simulation time needed to conduct this sweep is therefore equal to $2 * (0.8 + 0.3)/0.02 = 110s$. Because of the increased simulation time, the semi-infinite bulk boundary (SIBB) has to be positioned further away from the electrodes to maintain correct results (see section 3.3.4 for detailed explanation). In all CV-experiments discussed here, the SIBB is placed 1mm above the bottom electrode. With this placement the flux of ions at the SIBB is still negligible compared to the flux of ions to the electrodes at the end of the simulation. Like in previous simulations, the maximum size of the mesh elements is set to 5nm at the electrodes. The distance between the plates (h) and the thickness of the insulating layer covering the top electrode (h_g) are both set to 500nm. The hole to hole distance (w_{ee}) is set to relative small width of 2μm to keep the computing time and necessary memory within reasonable limits. In previous experiments the bulk concentration of reduced species was set to zero. This caused sharp current peaks at the start of the simulations due to a difference in Nernst potential of the solution at the electrode surface and the applied electrode potential at t=0s. Therefore both the bulk concentration of the oxidized and reduced species are set to 5 mol/m^3, resulting in a Nerst potential of the solution equal to the standard potential (E^0).

Single cyclic voltammetry experiment

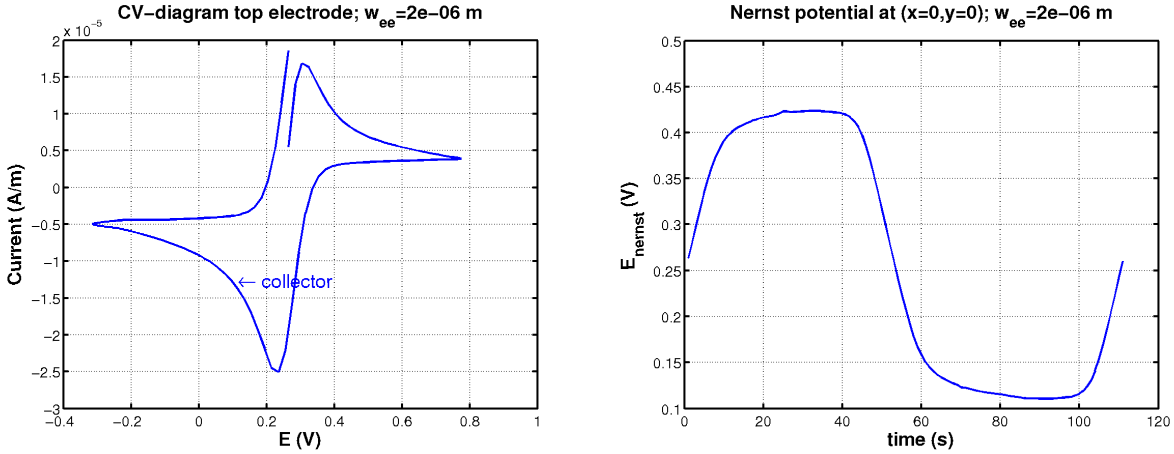
With the first experiment a potential sweep is applied to the top electrode, while the bottom electrode is removed. This is accomplished by changing the boundary condition of the bottom electrode. The bottom electrode is considered to be an insulating boundary with zero flux[5]:

$$N_{bottom} = \vec{n} \cdot (-D_j \nabla C_j) = 0 \tag{3.17}$$

---

[5] Normally the potential of an electrode is equal over its entire surface. Every potential difference which might be present in the solution at the surface of the electrode is soon compensated by the electrode inducing electrochemical reactions until a new equipotential situation is established. The insulating boundary condition chosen for the bottom electrode does not take this effect into account.

The potential at the surface of the bottom boundary is calculated using the Nernst equation 2.7, with $E^0 = 0.265V$ and the concentration of oxidized and reduced species located at the middle of the bottom boundary (x=0, y=0).



(a) CV diagram calculated for the top electrode, while the bottom electrode is removed.

(b) Nernst potential calculated from concentrations in the middle of the bottom boundary.

Figure 3.17: *CV-experiment applied to the collector electrode while the generator electrode is removed.*

Figure 3.17 shows the results of the first experiment. The CV calculated from the top electrode is shown in figure 3.17a. If this CV is compared to the measured CV from the ferrocyanide/-ferricyanide redox couple shown in figure 2.7, it is seen that the peak separation distance and standard potential are the same. Generation of reduced species during the experiment cause an increased current around +0.2V at the end of the potential sweep. The calculated Nerst potential during the CV-experiment is shown in figure 3.17b. It can be seen that the calculated potential follows the applied potential at the top electrode, although difference between the highest and lowest potential is smaller than that of the applied potential sweep.

Double simultaneous sweep CV experiment

With the second experiment the same potential sweep is applied to both the top and bottom electrode simultaneously. The results are shown in figure 3.18. The peak separation distance and standard potential of both the bottom electrode (generator) and top electrode (collector) are the same as seen in figure 3.17. The peak currents from bottom electrode are larger than that of the top electrode. This can be explained by the difference in surface area of both electrodes. Compared to the peak current from the first experiment shown in figure 3.17, the current of each individual electrode of this experiment is smaller. However, it is interesting to notice that the sum of both the bottom and top electrode currents is almost equal to the current of the single CV experiment. Apparently the larger surface area of both bottom and top electrode does not produce larger currents than that of the single CV experiment. This is probably due to the limited supply of ions from the bulk through the hole in the top electrode.

Single top electrode CV experiment with fixed bottom electrode potential

With the third experiment the potential of the bottom electrode is fixed to +0.7V, while the potential of the top electrode is varied. The results of this simulation are shown in figure 3.19. The most remarkable difference between the results shown here and the results shown in figure 3.17 and 3.18 is the difference in shape of the CV. The CV-diagrams from the previous experiments showed a clear hysteresis due to local depletion of ions. This hysteresis is not seen in figure 3.19. This is due to a fast recombination of ions due to redox cycling at the bottom electrode. As a result depletion of ions is not visible anymore and the shape of the CV appears

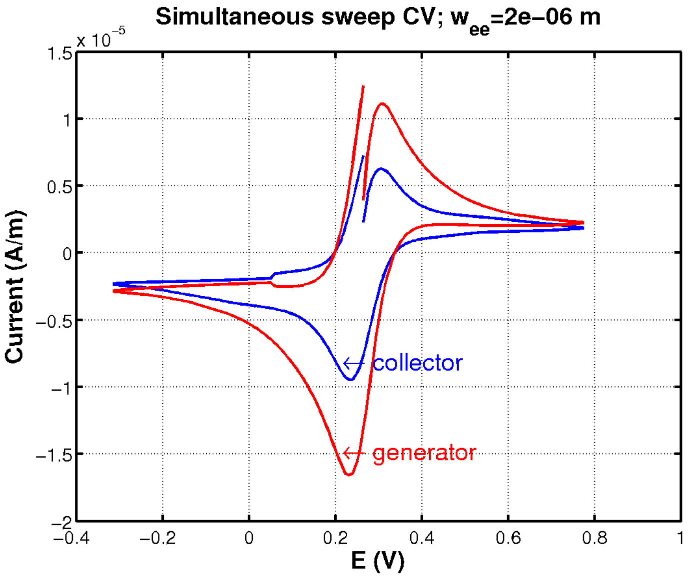


Figure 3.18: *CV-experiment applied to bottom electrode (indicated as generator) and top electrode (indicated as collector) simultaneously.*

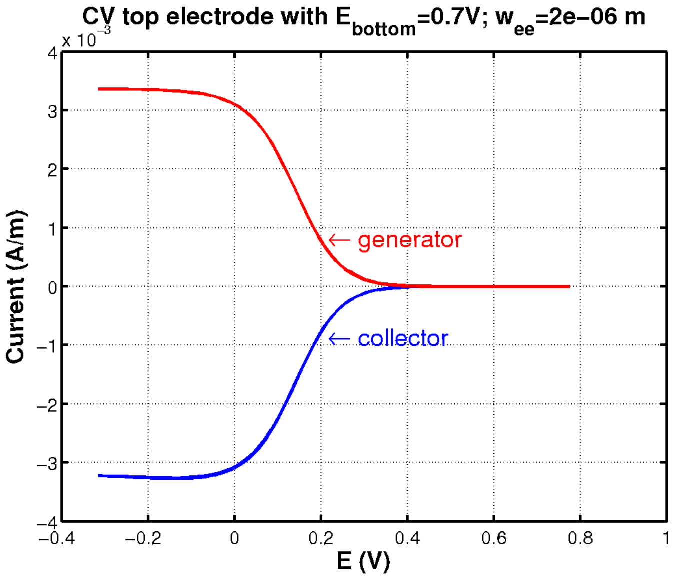


Figure 3.19: *CV-experiment applied to the top electrode (indicated as collector) with the bottom electrode (indicated as generator) potential fixed to +0.7V.*

like the CV measured from an ultra micro electrode. As expected, it is noticed that the difference in current between the bottom and top electrode is very small. The peak currents for negative potentials applied to the top electrode are approximately 3.4mA/m which is much higher than the peak current observed in figure 3.17 (0.025mA/m). This is also due to the redox cycling amplification.

3.5 Conclusion

In this section FEM-simulations on the interdigitated array electrode and novel sensor structures have been presented. A method is developed to ensure that results from the simulations correspond with the associated real-world experiments they represent. Within this method suitable domain equations, boundary conditions and guidelines for simulation time and a mesh with sufficient resolution have been formulated. This method is tested with the IDA-geometry with flat electrodes to check its validity. The results in terms of collection efficiency for a $W_g = 1\mu m$, $W_e = 1\mu m$ structure presented in this report (87.4%) are comparable with results from a paper of X. yang and G. Zhang [3] (87%) and a paper of A.J. Bard et al. [2] (85%).

Simulations on IDA-electrodes in which the electrode height is varied, showed an increase in redox cycling for larger heights. The special case with an estimated geometry derived from the devices used by V.A.T. Dam is simulated. The result shows 74x redox cycling, which is very close to the 60 to 70 times found by V.A.T. Dam. Therefore a plausible explanation is given for the experimental results of V.A.T. Dam.

Some novel sensor structures are studied of which the parallel plate configuration was the most promising in terms of redox cycling. The parallel plate configuration needs a supply mechanism to provide electrolyte into the space between the two plates. Two supply mechanisms have been studied for their effect on the number of redox cycles. A supply mechanism based on convection and a supply based on diffusion are considered. The latter showed better results in terms of redox cycling. This novel sensor structure showed up to 400x redox cycling, depending on the dimensions of the device.

The second research goal formulated in this report states: 'Design, production and testing of a novel sensor with the goal of gaining even higher redox-cycling amplification factors than the 60 to 70 times observed by V.A.T. Dam'. The sensor design with the parallel plates and passive bulk supply is chosen as most suitable candidate to achieve this second goal. In the next chapter the fabrication of this device will be described.

4 Experimental

In the previous chapter a novel sensor structure was studied with promising results with respect to redox cycling. In this chapter the fabrication of this novel sensor is described. Afterward the measurement setup and the measurement protocol will be given.

4.1 Device fabrication

The parallel plate with passive bulk supply sensor consists of two plates with a small space between them. The most common way to fabricate this space is to use sacrificial layer etching. The general process starts with fabricating a sandwich of layers containing one bottom electrode, a sacrificial layer , a top electrode and a final insulating layer. The term 'insulating layer' will be used more often in the following chapters and refers to the upper-most layer.

Once this sandwich is completed, holes are etched through the layers. Finally the sacrificial layer is removed and the sensor is assembled. At first a sensor with silicon nitride as insulating layer was fabricated, but during the sacrificial layer etching it became clear that this sensor was not working well. As an alternative, aluminum oxide and polyimide are considered as insulating layer. A small test indicates that polyimide is the most suitable candidate as a replacement of the silicon nitride, because the aluminum oxide layer start to bend if the sacrificial layer is removed. A picture of both the aluminum oxide and the polyimide test is shown in figure [B.5,B.6] in appendix B. The sensor with the polyimide is fabricated in a slightly different way. All the steps involved with the fabrication process of the sensor with silicon nitride are listed in figure 4.1.1. It was decided to include this second-best insulating material in order to discuss interesting measurements from this wafer in the results and discussion section. The fabrication process of the sensor with polyimide is shown in figure 4.1.2. The mask layout will be presented in succeeding section 4.1.3.

4.1.1 Process steps of the sensor with silicon nitride

1. Wafer selection. This is a standard p-type wafer with orientation <1-0-0> and 5-10Ωcm resistivity.
2. Thermal growth of a $1\mu m$ thick SiO_2 layer.
3. Three metals are sputtered on the substrate. The first layer is tantalum, because it adheres well on the SiO_2 underground, contrary to platinum. This first layer is 10nm thick. The second layer is made of platinum. This will be the actual bottom electrode once the device is finished, its thickness is 100nm. The third layer (10nm) made of titanium is added to ensure good adhesion of the next layer onto the substrate. Titanium is chosen, because it is more easy to remove together with the next SiO_2 layer on a later stage in the process compared to tantalum.

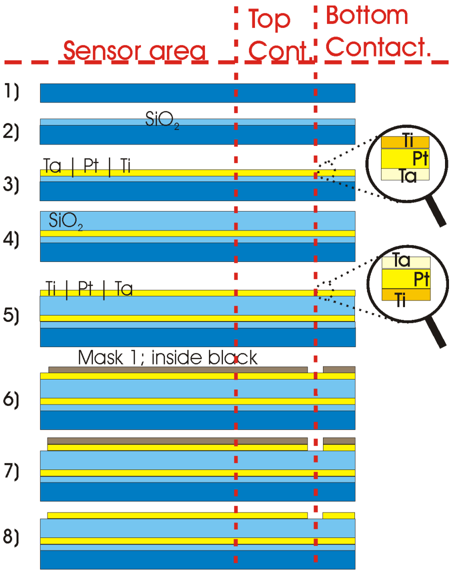


Figure 4.1.1a: Processing steps.

4. SiO_2 is grown onto the substrate with a chemical vapor deposition technique. The thickness of this layer is varied on three different wafers. On the first wafer a thickness of 200nm is deposited. The second layer is 500nm thick, and the third has a layer thickness of 1μm.

5. The top electrode is sputtered onto the substrate. titanium (10nm), platinum (100nm) and tantalum (10nm) are added in this order. Like the bottom electrode, titanium and tantalum are added to ensure good adhesion between the layers.

6. A photolithography step is carried out with a pattern used to etch the top electrode into the desired shape in the next step.

7. The excess metal from the top electrode is removed using an ion beam etch technique. The excess metal is removed to decrease capacitive effects between the two electrodes and to prevent shorts at the edge of the sensor when the wafer is diced.

8. The photoresist is removed by exposing the substrate to an oxygen plasma.

9. Low-stress SiN is added using a chemical vapor deposition method. The estimated thickness is 500nm. This is the insulating layer covering the top electrode to ensure that the top electrode is not in contact with the bulk electrolyte during measurements.

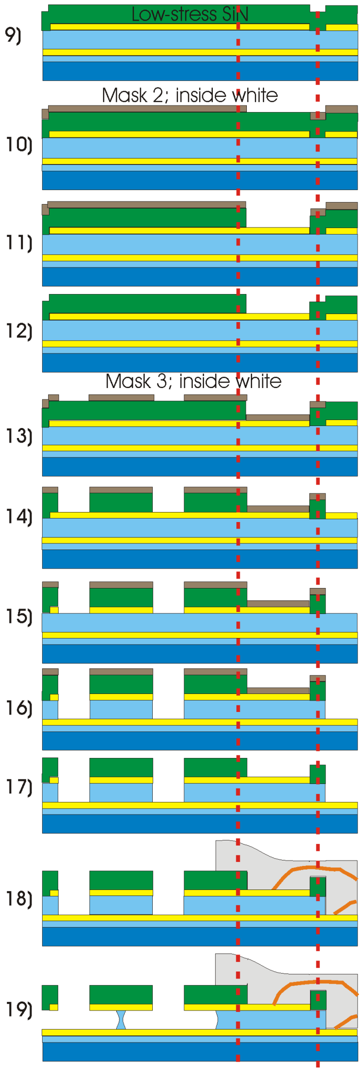


Figure 4.1.1b.

10. A second photolithography is conducted to clear the top electrode contact pad from the SiN in the next step.

11. The SiN on top of the top electrode contact pad is removed using reactive ion etching.

12. The photoresist is removed with an oxygen plasma.

13. A third photolithography step covers the whole wafer except for the locations of the diffusion holes in the top electrode and the contact pad for the bottom electrode.

14. The SiN layer is removed with reactive etching.

15. The top electrode material is removed with ion beam etching[6].

16. Finally the SiO_2 layer is removed with reactive ion etching.

---

[6] It might be possible to etch the holes already in step 7 and the silicon nitride in step 11, resulting in only two masks needed to create the sensor. However, this makes alignment of mask 1 and 2 very critical. Therefore the etching of all layers at the position of the holes is carried out in step 14, 15 and 16 successively.

17. Afterward the photoresist layer is removed by exposing the wafer to an oxygen plasma.
18. The wafer is diced into small individual sensors. These sensors are glued onto a small printed circuit board with epoxy. The electrical connection between PCB and sensors is made with small wires provided by a wire bonder. Afterward the sensor is completely covered with epoxy except for the sensor area.
19. In the final step the SiO_2 layer is partly removed by sacrificial layer etching in small time-intervals using a BHF solution. It is not possible to do this wet etching while the sensor is still on the wafer, since a number of sensors with different sizes are present. This results in different etch times for each sensor. The progress of the sacrificial layer etching is monitored using a visual check under the microscope. The residual parts of the sacrificial layer act as pillars to provide structural integrity of the sensor.

4.1.2 Process steps of the sensor with polyimide

The first 8 steps are the same as in the process of the sensor with silicon nitride (see 4.1.1a). Therefore the description of this process starts at step 9.

9. Polyimide is mixed with a solvent to decrease the viscosity. As a result the layer thickness is decreased when the mixture is spinned onto the wafer. The spinning of the polyimide is carried out at 6000 rpm for 1 min., resulting in an approximate layer thickness of 900nm. Polyimide is a polymer which is made sensitive to light. It can be processed in the same way as photoresist. If it is exposed to UV-light using a mask, the exposed areas can be removed. For our purpose this means we need inverted versions of mask 2 and 3 used in the process of the sensor with silicon nitride.
10. To prevent the need for two new masks, a 200nm thick aluminum layer is sputtered on top of the polyimide to act as mask for the polyimide etch in step 13.
11. A photolithography step is conducted to pattern the aluminum etch mask.
12. The aluminum is removed at the position of the top electrode pad using wet etching.
13. Afterward the polyimide is removed using reactive ion etching with an oxygen plasma. A small amount of CHF_3 is added to the plasma to decrease the amount of residue that remains after etching.

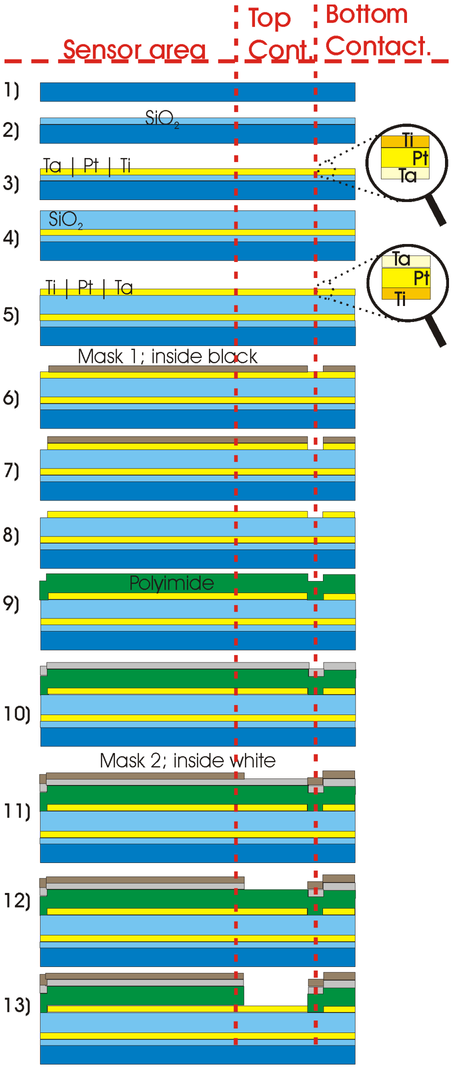


Figure 4.1.2a: Processing steps.

14. The photoresist is removed during this oxygen plasma etching as well.
15. The aluminum mask is now completely removed using wet etching.
16. A new layer of 300nm aluminum is sputtered onto the substrate to act as a new mask. This aluminum mask is slightly thicker than the previous one, since it is used during the ion beam etch process described in step 20.
17. A photoresist layer is added on top of the aluminum for the patterning of holes in the sensor area and to clear the bottom contact pad.
18. The pattern of the photoresist is copied onto the aluminum mask using wet etching.
19. Once more the polyimide is etched using reactive ion etching in an oxygen plasma. The photoresist layer is removed at the same time.
20. The platinum top contact is perforated using ion beam etching.
21. Finally the silicon oxide is removed using reactive ion etching.
22. The aluminum mask is removed using a wet etch.

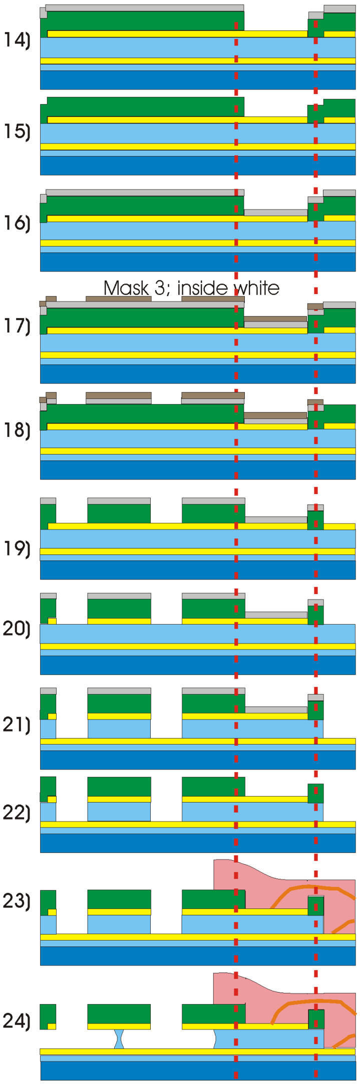


Figure 4.1.2b: Processing steps.

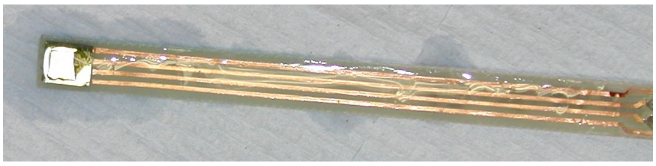


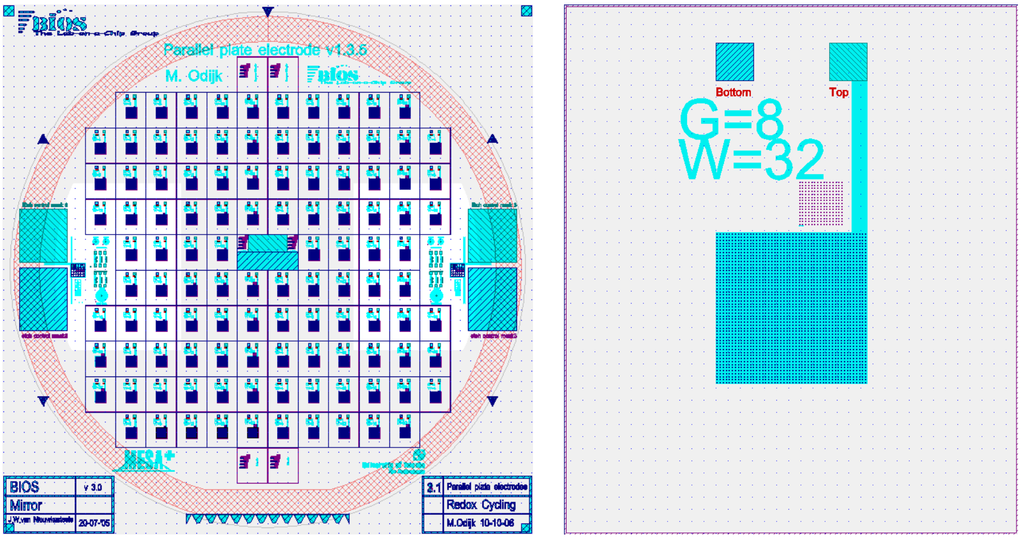
Figure 4.3: *Final result after assembly of the sensor. The total structure of sensor and PCB is 0.8 • 9.5 cm in size (w • l).*

23. Afterward the wafer is diced, the sensors are glued onto a small printed circuit board using epoxy. Wires are provided between the PCB and the sensor using a wire bonder. These wires and the contact pads of the sensor are covered with epoxy to insulate them from the solution during measurements. The final result can be seen in figure 4.3. The small reflecting square is a piece of the wafer. The actual sensor area is not visible, because it is too small.
24. As a final step the sacrificial layer is etched in small intervals using BHF. The etch progress is monitored under the microscope. The sacrificial layer etching process is described in more detail in paragraph 4.1.4.

4.1.3 Masks

The masks used for the processing of the parallel plate electrodes are shown in figure 4.4. In figure 4.4a, the overall mask layout for the entire wafer is shown. A zoomed view on a single sensor is shown in figure 4.4b. The aqua colored areas of mask 1 indicate where the platinum of the top electrode will remain after completion of the sensor. The second mask (green) indicates the removal of the insulating layer. These green areas are only visible at the top electrode. The third mask, linked to the color purple, shows where the removal of the holes and the contact pad for the bottom electrode will take place. The work area of the sensor is the large square in the middle of the sensor. It is 2mm by 2mm in size, resulting in a work area of approximately 4 mm^2. The small square grid of purple dots above the sensor is used to determine the progress of the sacrificial layer etching with a microscope. Because of the absence of the top metal electrode layer at this small grid, it is possible to look through the insulating layer covering the sacrificial layer. The two squares at the top of the sensor are the pads at which small wires are bonded to provide contact with the printed circuit board. The mask layout of the single sensor shown here has holes with a diameter of 8μm spaced at 32μm from neighboring holes, measured from the edge of the holes. Throughout the wafer these two parameters are varied. Besides the 8μm diameter holes, 4, 2 and 1μm holes are drawn. The spacing of these holes is varied with 4, 8, 12, 16 and 32μm.

At each wafer some space is reserved for four special test structures which are solely used to measure the etch rate. These test structures contain all the special grids which are present above the work area of each sensor, used to determine the sacrificial layer etch progress. An example of such a test structure is shown in figure B.4 in appendix B. A special study has been conducted to measure the etch rate of the sacrificial layer. This special study is discussed in the next paragraph.



(a) Overall mask layout (b) Single sensor mask layout

Figure 4.4: *Mask layout (blue=mask1, purple=mask2, green=mask3).*

4.1.4 Sacrificial layer etching

This paragraph will treat the efforts to determine the etch rate of the sacrificial layer. The special test structures mentioned in the previous paragraph and shown in figure B.4 will be used for this etch rate study. The etching of individual sensors will be treated in section 5 were the measurement results are discussed.

A standard BHF solution is used for the sacrificial layer etching. The temperature of the BHF solution is measured during the etch process. The temperature is not controlled making the etch rate slightly unpredictable, although it is within acceptable margins. The temperature was typically between 18 and 20°C during the sacrificial layer etching. Within this temperature range the observed etch rate was predictable and reproducible.

The etch results of the wafer with silicon nitride were bad. Silicon nitride was chosen as insulating layer because of its low etch rate in BHF. However, measurements with the microscope revealed that its etch rate was so fast that the silicon nitride layer was removed completely after approximately 400s. This is also visible in figure 4.5. This series of pictures is measured from the test structure. The pictures shown here are measured at small time-intervals. At 420s the letters indicating the specific grid parameters start to disappear. These letters are normally covered with the insulating layer, which should prevent these letters from degradating. Also the color of the wafer changes with increased etch time, indicating a decrease in layer thickness of the silicon nitride.

From literature [26] it is known that the etch rate of PECVD grown silicon nitride is between 8.2 and 60 nm/min depending on the specific growth conditions. The measurements shown here indicate an etch rate of approximately $500nm \cdot 60s/400s = 75$ nm/min. However, it is not assured that the silicon nitride is really 500nm since it is not measured during the fabrication process in the cleanroom. Even with the lowest etch rate known in literature, a 500nm thick silicon nitride layer will be removed if it is etched for $500/8.2 = 61\ min$. Experiments mentioned in the next paragraph will show that an etch for 1 hour is necessary in some cases. The silicon nitride is also important for the structural integrity of the sensor, therefore it must not be weakened too much during the etch process. In retrospect it can be said that silicon nitride was a bad choice as insulating layer.

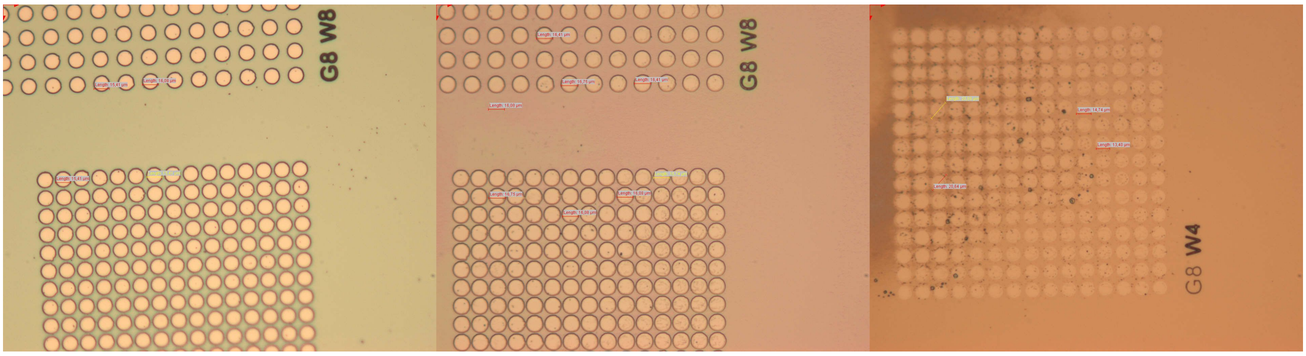


Figure 4.5: *Image of the test structure with SiN as insulating layer under the micoscope at a magnification of 20x for 0s, 180s, 420s etching in BHF.*

The sensors with polyimide as insulating layer showed better results during the sacrificial layer etching. The polyimide layer clearly has a better chemical resistivity to BHF. During the etch process circles underneath the polyimide arise, indicating the removal of the sacrificial layer around the holes which are etched through the polyimide in a previous process step. With these devices it was possible to measure the etch rate for the sacrificial layer. The special test structure without the top electrode was therefore etched in small steps. After each step the difference in diameter between the hole in the polyimide and the hole underneath the polyimide was measured under the microscope.

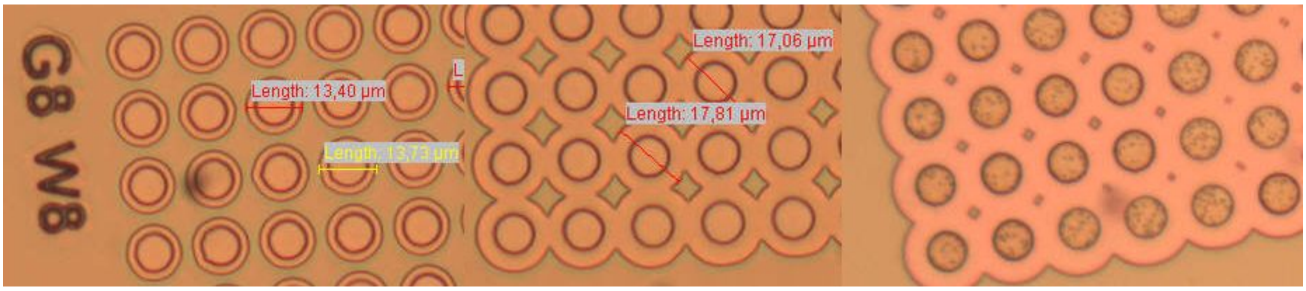
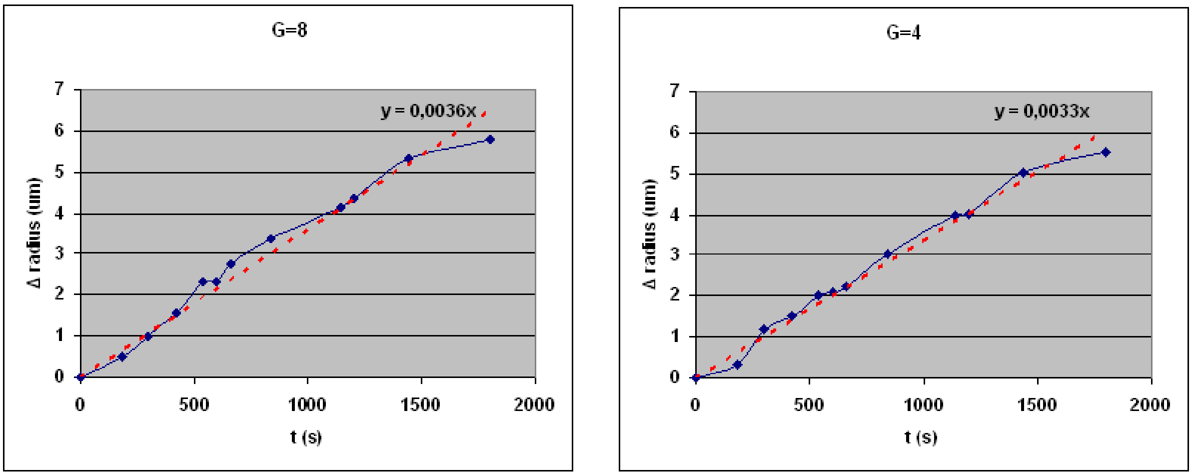


Figure 4.6: *Image of the test structure with polyimide as insulating layer under the microscope at a magnification of 20x for 600s, 1200s and 1800s etching in BHF.*

Figure 4.6 shows a series of three images of the test structure after etching 600s, 1200s and 1800s. It can be seen that the space underneath the polyimide is growing with increased etch time. Figure 4.7 shows the difference between radius of the hole in the polyimide and the hole in the sacrificial layer as a function of the etch time for two different hole sizes.



(a) Etch rate for the holes with a diameter of $8\mu m$.

(b) Etch rate for the holes with a diameter of $4\mu m$.

Figure 4.7: *Etch rate for the sacrificial layer in a buffered HF solution.*

Figure 4.7a shows the measurements for a test structure with holes of 8μm, while the measurements for the test structures with holes of 4μm are shown in figure 4.7b. A trendline and corresponding equation is added to both figures. The etch rate for structures with a hole diameter of 8μm is equal to 0.0036μm/s and for a structure with holes of 4μm it is equal to 0.0033μm/s. This corresponds to an etch rate of 0.216μm/min and 0.198μm/min. for the 8μm and 4μm structures respectively. From the figures it can be seen that the etch rate is more or less linear, although the etch rate seems to deviate from the trendline for larger etch times. The etch rate is expected to be an logarithmic function, since for larger etch times diffusion of the reaction products will become more important. Apparently the chemical reaction between the silicon oxide and the BHF is the rate determining step for smaller etch times and not the diffusion of chemicals to and from the silicon oxide/BHF solution interface. This assumption is confirmed by the small differences between the etch rate for the 8μm and 4μm holes. Although the etch rate for the 8μm holes is slightly higher, the difference is not significant.

4.1.5 Additional processing details

From the finalized sensors some important characteristics can be mentioned. On each wafer all holes with a diameter of 1μm are missing. This is due to the absence of the 1μm holes on the third mask. 2μm is apparently the minimum size of what the laser mask writer can handle. Even with the 2μm diameter, some holes are missing. An interference pattern is visible in the 2μm grids. This interference pattern is also shown in figure C.1 of appendix C. It is assumed that this interference pattern is originating from the laser mask writer's resolution and the grid of holes on the mask.

The test structures and sensors with polyimide as insulating layer had a large amount of debris at the bottom of the holes after the plasma etch of the sacrificial layer[7]. This debris could be removed from the 4μm and 8μm holes by ultrasonic cleaning for 5min. The removal of this debris was more difficult for the 2μm holes. The remaining debris could cause an obstruction for the sacrificial layer etching. This resulted in some holes which were not opened after etching.

While diced, the wafers are under a lot of mechanical stress and become filled with particles. During the dicing of the silicon nitride sensors the edge around the sensor area got damaged. This damage is shown in figure C.2 of appendix C. Especially around the corners some of the layers got removed by the friction of the rotating disk cutter with the wafer. The usual way to prevent this problem is to cover the top side of the wafer with ultraviolet light sensitive foil. This foil protects the wafer from dirt and some of the stress. Afterward the wafer with its foil is exposed under an UV-light source. The foil stops to stick onto the wafer so it can be removed with ease. Learned from the experience with the silicon nitride sensors, the wafers with the polyimide were covered with this protection foil before dicing. This gave a problem afterward, because the foil did not stop sticking to the polyimide after the UV-light treatment. In attempts to remove the foil, several other layers were removed as well. An unintentional so-called 'sticking-tape' peel-off experiment was done. As a result it was noticed that the adhesion of the bottom platinum electrode to the wafer was not very strong. A real peel-off experiment was done on the wafer with silicon nitride afterward and it was shown that the adhesion of the bottom platinum electrode to the wafer was not good throughout all produced wafers.

Some experiments were conducted to remove the foil from the polyimide-type wafers with all kinds of solvents, but with no success. Eventually, the best way to remove the foil was to anneal the sensors for approximately 40s at 290^oC. It is suspected that the bond between the foil and the polyimide starts to weaken at a higher temperature. After some time the UV-foil melts and therefore it starts to bend. As a result the foil removes itself from the sensors. Visual inspection under a microscope indicated that the polyimide was still intact after this unusual annealing technique to remove the UV-foil.

---

[7] The processing step mentioned here is shown as step 21 of figure 4.1.2b

4.2 Measurement setup

In figure 4.8 the measurement setup used for the electrochemical measurements in the lab is visible. The blue box on the left is the potentiostat. It is a two channel bi-stat from the company Bio-Logic. The thick, black wires from the bi-stat run through a small hole into the metal locker on the right. This locker is used as a Faraday cage to shield the experiments from external electrical fields. At the end of the wires two small blue boxes are provided with the actual measurement leads. The bi-stat is connected to a computer via an ethernet connection.

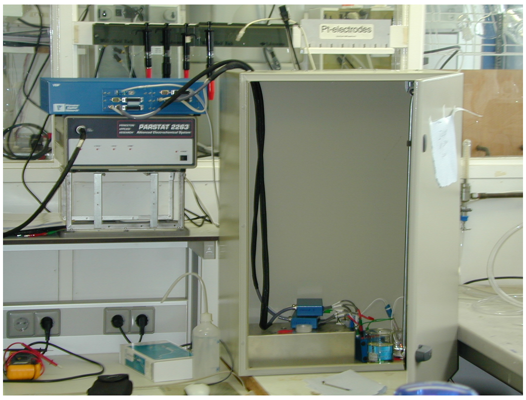


Figure 4.8: *The measurement setup in the lab of the BIOS group.*

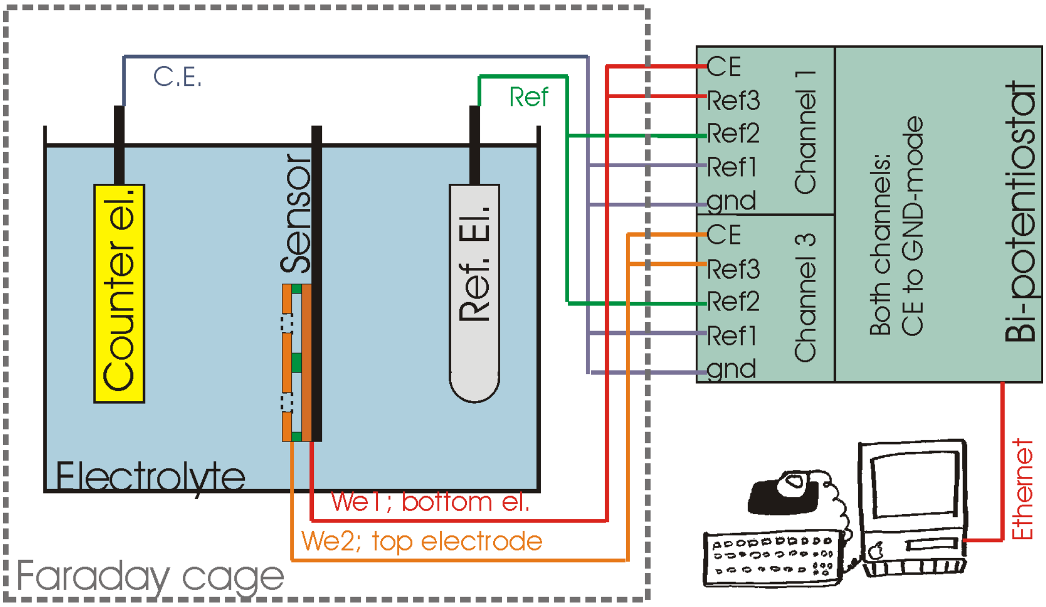


Figure 4.9: *Schematic representation of the measurement setup.*

The schematic representation of the measurement setup is shown in figure 4.9. On the left the Faraday cage is visible. Inside the cage, the electrochemical cell is shown. The box on the right represents the bi-potentiostat. The measurement channels are set at the so-called *CE to ground* mode. This mode makes it possible to measure with two work electrodes in a single electrolyte. The wiring is as follows: the bottom electrode of the sensor is connected to leads CE and ref3 of channel 1. The top electrode is connected to leads CE and ref3 of channel 3. A single platinum counter electrode is connected to leads ref1 and ground of both channel 1 and 3. One reference electrode is connected to leads ref2 of both channels. This reference electrode is of silver/silverchloride type, filled with a saturated KCl solution bought from Radiometer Analytical (electrode type REF201). De bi-potentiostat buffers data during measurements and sends this data to a computer running the program EC-Lab (v9.21) via an ethernet connection. Final processing of the data is done with Matlab. The Matlab-scripts are included in appendix D.11.

4.3 Measurement protocol

At the beginning of the measurements a stock electrolyte is made containing 0.01 molar potassium ferrocyanide and 0.5 molar of potassium nitrate. The potassium nitrate is added to act as supporting electrolyte. As explained in section 2.6, the ferrocyanide acts as redox-active species, while the potassium nitrate is added to decrease the electrolyte resistivity and effects of migration. This stock solution is stored in a dark bottle at room temperature. For each series of measurements 40ml of electrolyte is taken from the stock solution to use for the measurements. The platinum counter electrode, the reference electrode and the sensor are rinsed with de-ionized water before they are placed in the electrolyte to prevent contamination of the measurement solution with other ions.

Two measurement protocols are programmed within EC-lab. The second protocol is actually an extended version of the first protocol. The protocols consist of several small measurement techniques which are performed sequentially by the program. Between each technique the potential is switched to open circuit voltage (OCV). The sensor structure of interest contains two work electrodes. Therefore advanced cyclic voltammetry measurements are possible.

The following techniques are conducted with the extended protocol respectively:

1. A cyclic voltammogram is measured consisting of a single sweep from OCV to 0.8V, reversed to -0.4V and back to OCV with a scanrate of 20mV/s on both channels synchronously. Because of the synchronous sweep, the electrodes can be considered as a single combined electrode effectively. A slow scan-rate of 20mV/s is chosen to prevent strange measurements caused by (double layer) charging effects.

2. A second cyclic voltammogram is measured on channel 1 with the same parameters as in technique 1. Channel 3 is kept at OCV, while its potential is measured. The second (top) electrode can be considered as a sensor measuring the local potential in close proximity of the first (bottom) electrode.

3. A third cyclic voltammogram is measured on channel 1 with the same parameters as in technique 1, while the potential at channel 3 is set to +0.3V. This specific potential of +0.3V is chosen, because it is close to the standard potential of the ferrocyanide/ferricyanide redox couple. Therefore its contribution to the current due to redox cycling of channel 1 should be limited.

4. Technique 2 is repeated but this time the cyclic voltammogram is measured on channel 3, while the first channel is kept at OCV.

5. Also the third technique is repeated but with the cyclic voltammetry measurement on channel 3, while channel 1 is set at a fixed potential of 0.3V.

6. With the final technique a series of fixed potentials are provided at both channels while the current response is measured. This technique is used to derive the collection and redox cycling amplification factor.

Figure 4.10 shows a typical measurement of the potential on both channels using the extended measurement protocol. The first 5 techniques are applied to test the correct behavior of the total sensor structure and each of its work electrodes.

From 685s the sixth technique with a series of fixed potentials is started. This technique is used to derive the collection efficiency and the redox cycling amplification factor. Throughout the

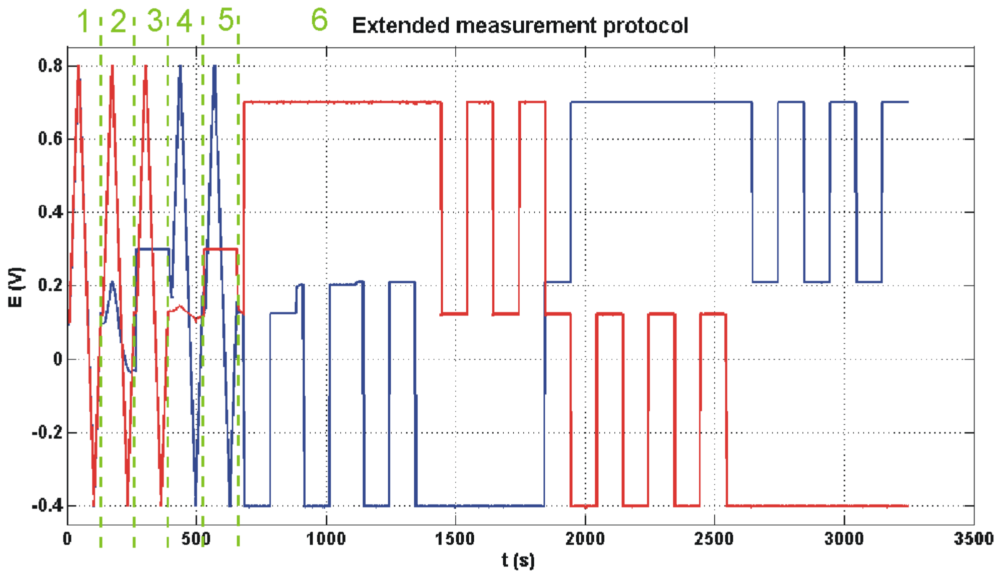


Figure 4.10: *Potential on channel 1 (red) and channel 2 (blue) for a measurement with the extended protocol.*

measurement the current of both channels is measured. First channel 1 is fixed at +0.7V for 600s, while the potential on channel 3 is switched between OCV and -0.4V in small blocks of 100s. The difference in current on the first channel for the case with channel 3 at OCV and the case with channel 3 at -0.4V is a measure for the redox cycling amplification factor on channel 1. The redox cycling amplification and collection efficiency can be calculated from the measured currents using the already explained equations 2.36 and 2.35 mentioned in section 2.3.2.

Secondly channel 3 is fixed at -0.4V for 500s, while the potential of channel 1 is switched between OCV and +0.7V. From this second set of switched potentials the redox cycling amplification for channel 3 can be calculated. After a small pause at which both channels are at OCV, this principle of determining the redox cycling amplification is repeated in the third and fourth block of switched potentials. The redox cycling amplification is measured once more but with the potentials reversed. Meaning; channel 1 is switched between OCV and -0.4 and channel 3 is switched between OCV and +0.7V.

The simple measurement protocol is the same as the extended version, but with the first, third and fifth technique removed.

5 Results and discussion

5.1 Introduction

Three different wafers have been fabricated. Each wafer contains a series of 112 sensors. The first wafer is fabricated according to the process described in section 4.1.1. Therefore it has silicon nitride as an insulating layer[8]. This wafer has a sacrificial layer thickness of approximately 450nm. The second wafer is fabricated with polyimide as insulating layer as described in section 4.1.2. This second wafer has a sacrificial layer thickness of 450nm. The third wafer is also of the polyimide-type. However, this wafer has a sacrificial layer thickness of approximately 900nm.

In this section, the measurements with the fabricated sensors are presented. First the sacrificial layer etching and measurement results for the sensors made of silicon nitride are discussed. Secondly the polyimide sensors are treated.

5.2 Silicon nitride sensors

The measurements on the silicon nitride sensors will be illustrated with measurements on a single sensor, because this sensor shows typical behavior of what is seen in several measurements with different sensors. This single sensor has holes designed with a diameter of 8μm spaced at 32μm from each other. Measurements with the microscope revealed that the actual size of the holes is around 11.4μm. The spacing from edge to edge between those holes is 28.6μm.

5.2.1 Etching of the sensor

The sensor is etched in small time intervals. After each interval the result is recorded with a microscope and the sensor is used for an electrochemical experiment. The electrochemical measurements are shown in the next paragraph. Figure 5.1 shows a series of pictures recorded after 0s, 200s, 300s and 600s of etching. From this slide of pictures it can be seen that the color on the wafer changes with increased etch time. After 600s almost no holes are visible in the test structure. As mentioned in the section about the sacrificial layer etching (4.1.4) the text indicating the specific design parameters (G8 W32) disappears after 300s, although this is not shown here. These two observations indicate that the silicon nitride layer is completely removed after approximately 300 to 400s.

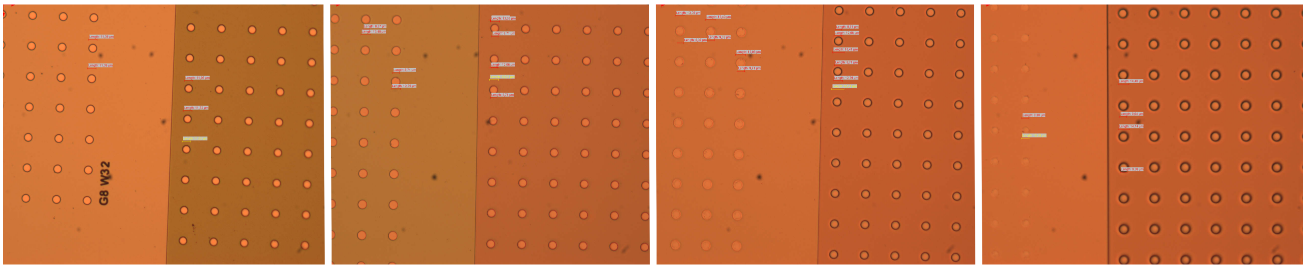


Figure 5.1: *Photos from the sensor and its test structure for each times of 0s, 200s, 300s and 600s etching at a magnification of 20x. The holes on the left on each photo are from the test structure (without top electrode) and the holes on the right are from the sensor area.*

---

[8] Please note that the term insulating layer is referring only to the upper-most layer which should insulate the top electrode from the electrolyte.

Not so clearly visible is the growth of local dents around the holes in the sensor area. To illustrate this effect a picture of a different sensor which is etched for 1500s is shown in figure 5.2. It is assumed that the platinum top electrode starts to bends toward the bottom electrode when the sacrificial layer underneath the top electrode is removed. A difference in focus of the microscope image exists between the outer rings and the inner rings from the holes, confirming the assumption of dents. In the original design the silicon nitride layer was meant as supporting layer to prevent the top electrode from bending. Unfortunately this silicon nitride is removed during the etch process.

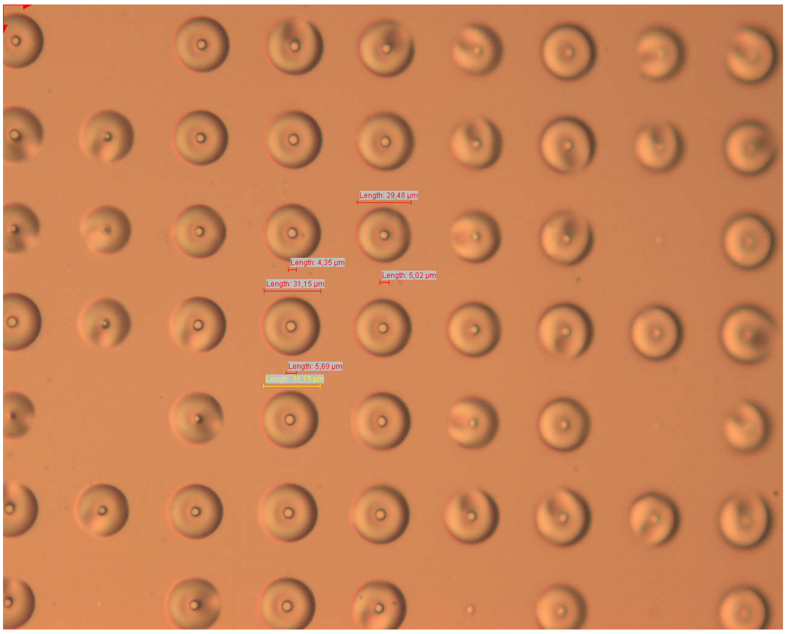
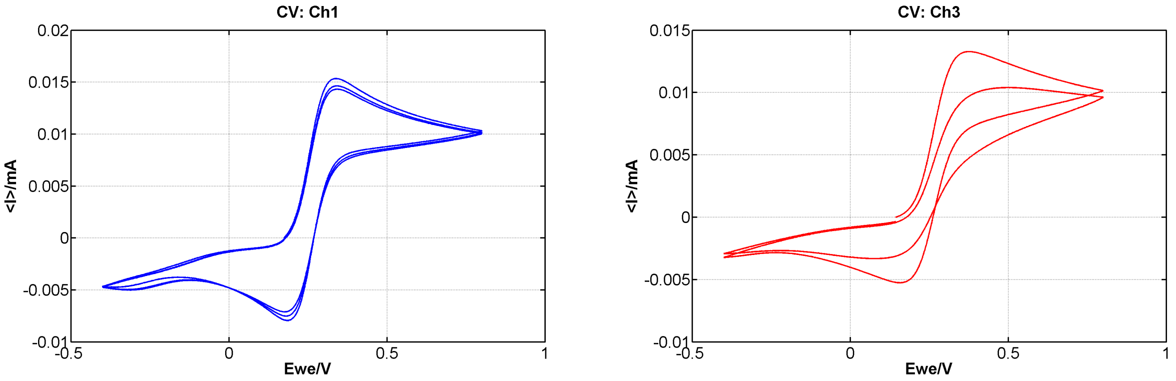


Figure 5.2: *Photo from a sensor which is etched for 1500s in BHF at a magnification of 20x. The specific design parameters are:* $W_g = 2\mu m$ *and* $W_{ee} = 32\mu m$.

5.2.2 Measurement results

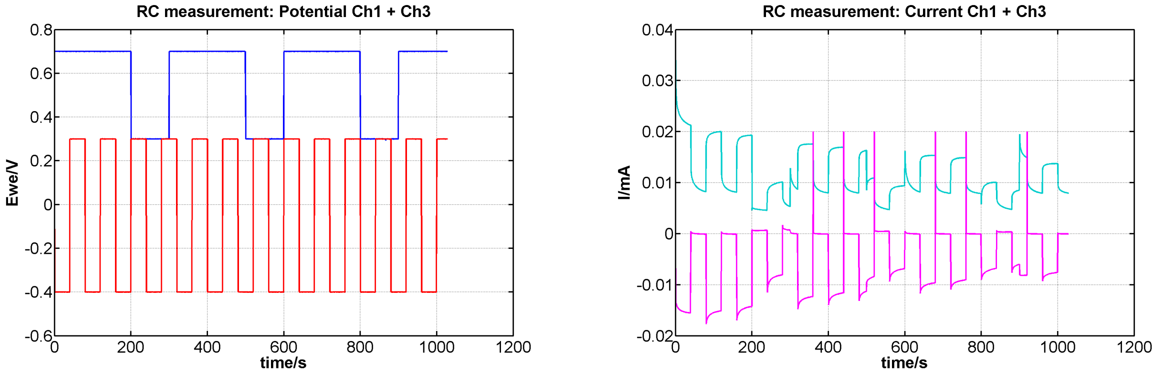
As mentioned in section 4.3 the simple measurement protocol consists of 3 measurement techniques which are executed sequentially. First a cyclic voltammogram is recorded on channel 1, which is connected to the bottom electrode. Secondly a cyclic voltammogram is recorded on channel 3 which is connected to the top electrode. Thirdly a series of potentials is applied to both electrodes to measure the redox amplification factor. During the CV-measurements, the 'inactive' electrode is left at OCV. All CV measurements presented in this section have the same settings, meaning the potential is switched from OCV to +0.8V, reversed to -0.4V and back to OCV. The scanrate is 20mV/s. All potentials mentioned in this section are compared to a silver/silverchloride reference electrode saturated with potassium chloride (offset potential 0.197V with respect to a normal hydrogen electrode).

The sensor that is discussed here is etched in small intervals. The series of measurement techniques is applied after each interval. Figure 5.3 shows these measurements before the sacrificial layer etching. Figure 5.3a shows a cyclic voltammogram of two cycles measured from the bottom electrode, while the top electrode is kept at OCV. 5.3b shows a CV-diagram of two cycles measured from the top electrode, while the bottom electrode is set to OCV. The CV from the top electrode shows a difference between the first cycle and the second cycle. It is suspected that local depletion of ions caused the decrease in current noticed in the second cycle. The current of the bottom electrode is slightly higher, which is due to the larger surface area of the bottom electrode. It is suspected that a small part of the top electrode is already cleared during the reactive ion etch process described as step 16 in section 4.1.1, resulting in a larger work area of the top electrode. The redox cycling amplification measurement on both electrodes is shown in figures 5.3c and 5.3d. Figure 5.3c shows the applied potentials, while the current response of both channels is shown in figure 5.3d.



(a) CV measured from the bottom electrode.

(b) CV measured from the top electrode.



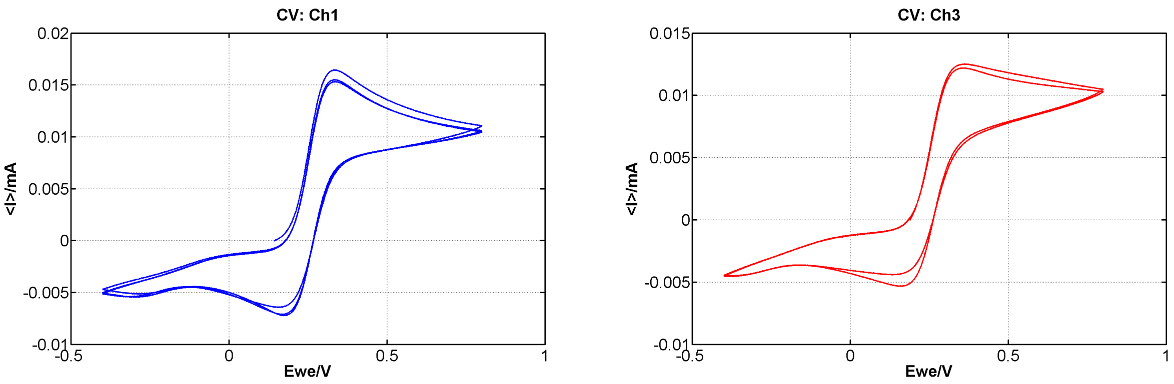
(c) Potentials applied to the bottom (blue) and top (red) electrode during the RC-measurement.

(d) Current response measured from bottom (aqua) and top (pink) electrode during the RC-measurement.

Figure 5.3: *CV and RC measurements before sacrificial layer etching. Parameters:* $W_g = 8\mu m$ *and* $W_{ee} = 32\mu m$.

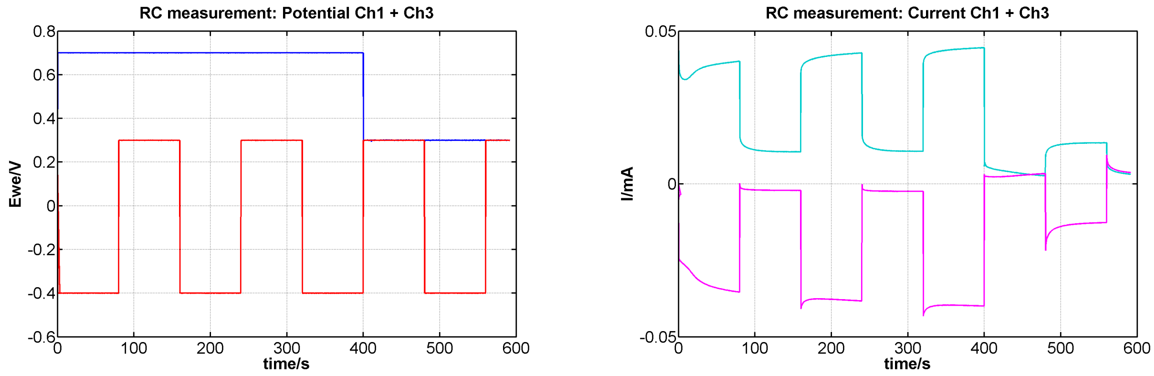
From these figures it is shown that the device is working as expected before the sacrificial layer etching. Even some redox cycling is measured. The current of the bottom electrode is equal to approximately 8μA if the top electrode is set at a potential of 0.3V. If the top electrode is switched to a negative potential of -0.4V the current of the bottom electrode increases to approximately 15 μA due to redox cycling. This means the redox cycling amplification is around $15mA/8mA \approx 2x$.

If the same sensor is etched for 200s in BHF we measure the results shown in figure 5.4. The CV measured from the bottom electrode is shown in figure 5.4a, while the CV measured from the top electrode is shown in figure 5.4b. If the CV-diagrams from this measurement are compared to the measurements shown in figure 5.3, no significant change is observed. The peak currents and peak potentials are in the same order of magnitude. Apparently no significant change in work area of the electrodes has occurred due to 200s etching in BHF. Figures 5.4c and 5.4d show the applied potentials (c) and the current response (d) of both electrodes observed in the redox



(a) CV measured from the bottom electrode.

(b) CV measured from the top electrode.



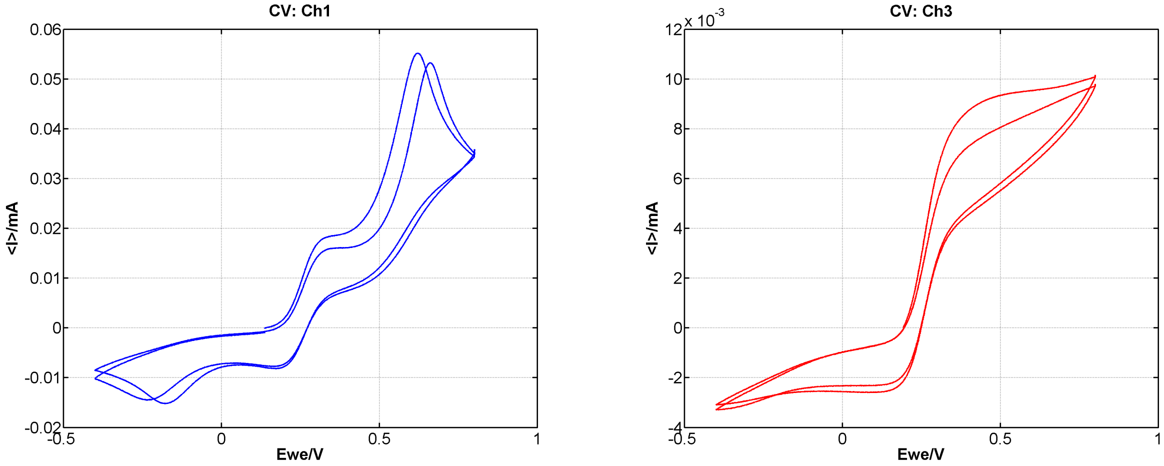
(c) Potentials applied to the bottom (blue) and top (red) electrode during the RC-measurement.

(d) Current response measured from bottom (aqua) and top (pink) electrode during the RC-measurement.

Figure 5.4: *CV and RC measurements after 200s sacrificial layer etching. Parameters:* $W_g = 8\mu m$ *and* $W_{ee} = 32\mu m$.

cycling measurement. The current of the bottom electrode in the RC measurement is equal to 0.01 mA, when a potential of 0.3V is applied to the top electrode. If the potential of the top electrode is switched to -0.4V, the current of the bottom electrode increases to 0.04 mA. The resulting redox amplification is around 4x, which is more compared to the measurement before the sacrificial layer etching. The same measurements are conducted after 300s and 400s etching, with similar results.

Remarkable results are found if the sacrificial layer is etched for 600s. The associated measurements after 600s etching are shown in figure 5.5. It appears like the cyclic voltammogram of the bottom electrode (figure 5.5a) is measured in an electrolyte with two redox-couples. As an example; CV-diagrams with similar shapes are found if the potential sweep range is increased such that hydrogen ions (H+)in the water are reduced and hydrogen is formed or hydroxide (OH-) ions are oxidized into oxygen. Control measurements with other electrodes indicated however that the electrolyte was not contaminated. The potential sweep is also small enough to keep away from the electrolysis of water, which will take place at -0.83V or +1.23V with respect to a normal hydrogen electrode. The CV-diagram of the top electrode (figure 5.5b) does not show the same behavior as the bottom electrode. The measurement of the top electrode does show a steeper slope in the curve from the maximum applied potentials (-0.4V and 0.8V) back to the standard potential (0.265V). As mentioned in section 2.6, a normal CV-diagram shows a decrease in current after the peak-values due to local depletion of redox-active ions. These peak-values are normally located approximately ±29.5mV from the standard potential. Therefore it



(a) CV measured from the bottom electrode.

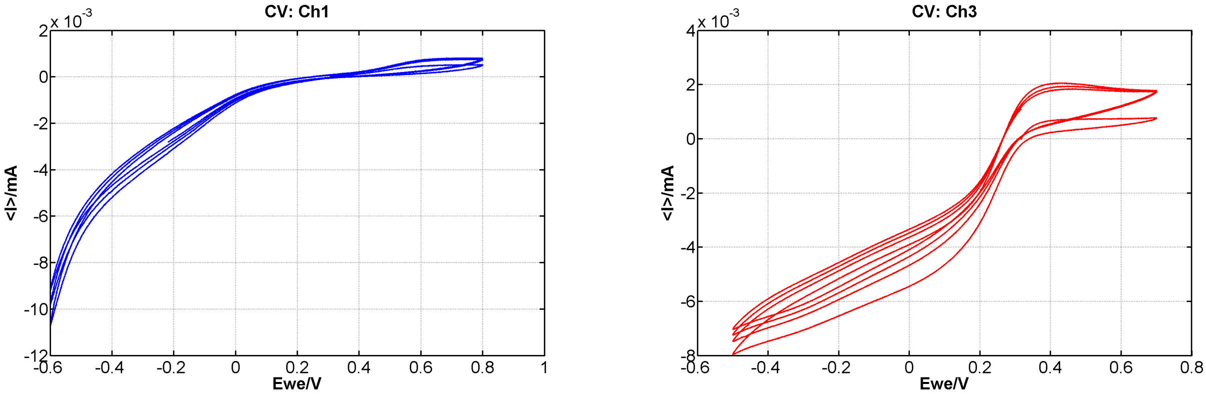
(b) CV measured from the top electrode.

Figure 5.5: *CV measurements after 600s sacrificial layer etching. Parameters:* $W_g = 8\mu m$ *and* $W_{ee} = 32\mu m$.

can be concluded that the CV measured from the top electrode is distorted as well, although the shape of the CV is still recognizable. Similar distortions on the cyclic voltammograms are measured on multiple other devices.

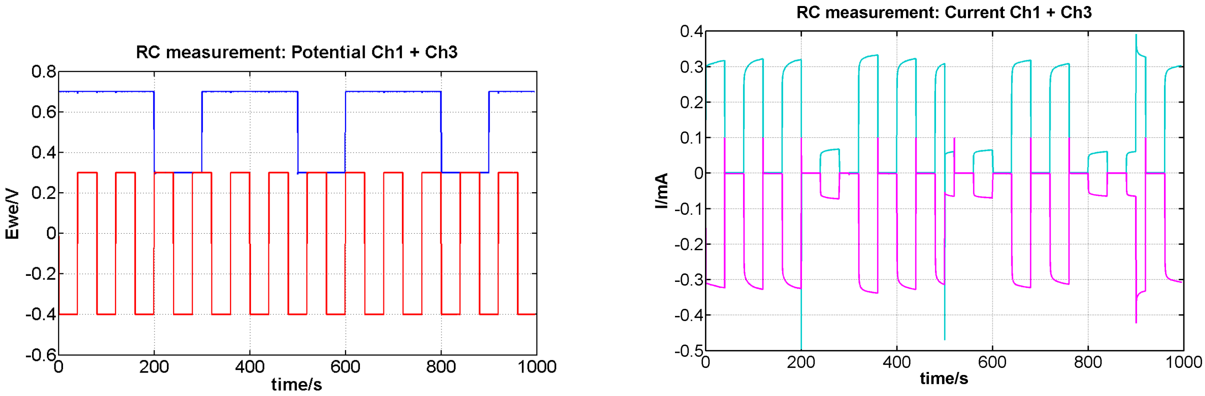
From figures 5.1 and 5.2 it is known that the top electrode starts to bend for longer etch times. That might be an explanation for the distorted CV-diagrams if the sensors are etched for a period longer than approximately 500s. The extra peaks in the CV-diagram of the bottom electrode might be caused by encapsulated BHF underneath collapsed top electrodes. An other more likely explanation is that the top electrode and the bottom electrode might form a (bad) conducting connection. The bottom electrode starts to pass its potential to the top electrode and uses the top electrode to conduct electrochemistry. Since the bottom electrode is shielded from the electrolyte by the collapsed top electrode this behavior is not measured on the CV-diagram of the top electrode.

The last measurement on a silicon nitride-type sensor which will be treated here is shown in figure 5.6. This sensor has a hole diameter of $2\mu m$ and a hole-to-hole distance of 32μm. The CV measured from bottom electrode is shown in figure 5.6a, while the CV measured from the top electrode is shown in figure 5.6b. The applied potentials and corresponding current responses used to determine the redox cycling are shown in figures 5.6c and 5.6d respectively. At a first glance the RC-measurement looks impressive. If the top electrode is switched to a positive potential of 0.3V, the current measured at the bottom electrode is around 1.25μA. This current increases to 0.43mA if the top electrode is switched to a potential of -0.4V. That is an impressive increase of the current of 340x. The CV-measurements might indicate that this increase in current is not only due to redox cycling amplification. The increase might as well be caused by a short. The potential difference between the two electrodes will lead to an added current if a connection exists. The CV-diagram of the bottom electrode is no longer recognizable as a typical CV-diagram, while the CV-diagram measured at the top electrode looks distorted in the same manner as figure 5.5b. The peak currents are slightly smaller than the peak currents shown from the $w_g = 8\mu m$, $w_{ee} = 32\mu m$ sensor. This can be explained by the smaller hole diameter of the device discussed here.



(a) CV measured from the bottom electrode.

(b) CV measured from the top electrode. CV measured from the bottom electrode.



(c) Potentials applied to the bottom (blue) and top (red) electrode during the RC-measurement.

(d) Current response measured from bottom (aqua) and top (pink) electrode during the RC-measurement.

Figure 5.6: *CV and RC measurements after 1500s sacrificial layer etching. Parameters:* $W_g = 2\mu m$ *and* $W_{ee} = 32\mu m$.

5.3 Polyimide sensors

Two wafers were fabricated with polyimide as insulating layer to prevent direct contact of the top electrode with the electrolyte. The first wafer has a sacrificial layer thickness of 450nm, while the second wafer has a sacrificial layer thickness of 900nm. Almost all sensors from the first wafer had a short between the two work electrodes. Therefore the electrochemical measurements presented in this section are from the second wafer with a sacrificial layer thickness of 900nm.

5.3.1 Etching of the sensors

A typical etch of the sensors covered with polyimide is shown in figure 5.7. The sensor shown here is etched for 1800s. On the left the test structure is visible, while on the right the sensor area is shown. The holes in the polyimide and the top electrode have a diameter of 8.7μm and are spaced 11.3μm from each other. The space underneath the polyimide has a diameter of 20.4μm, resulting in a difference in radius of 5.85μm.

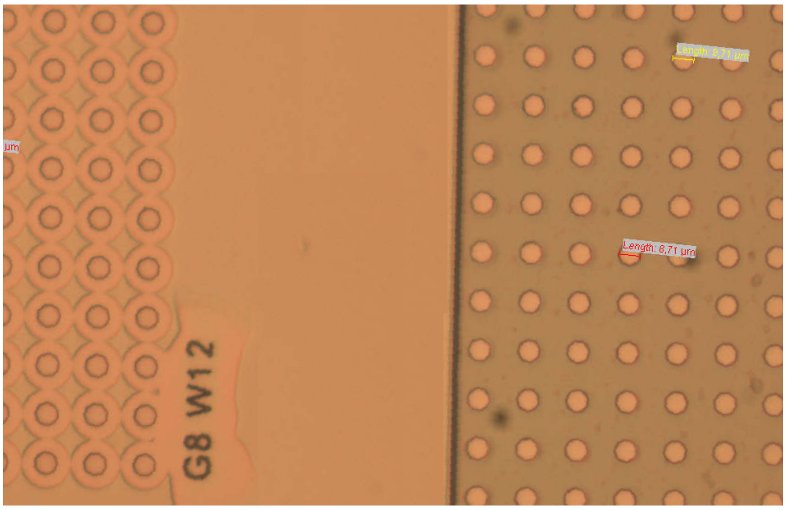


Figure 5.7: *Photo from a sensor which is etched for 1800s in BHF at a magnification of 20x. The specific design parameters are:* $W_g = 8\mu m$ *and* $W_{ee} = 12\mu m$.

If this same sensor is etched for 2100s it is noticed that the top electrode starts to come off, even if there are still small pillars visible in the test structure. For the sensor shown in figure 5.7 this detached top electrode is shown in figure 5.8. Notice the rippled surface of the top electrode. The pillars observed in the test structure of the device shown in figure 5.8 are approximately 5.5x5.5μm in size. In general the exact size of these pillars at which the top electrode starts to detach is dependent of the specific grid dimensions. However, the detachment of the top electrode is not studied in great detail, because efforts were aimed to preserve as much sensors as possible. The poor adhesion of the top electrode to these pillars might be explained by a small difference in etch rate between the sacrificial silicon oxide layer and the titanium which is present as adhesion layer between the silicon oxide and the platinum electrode.

5.3.2 Measurement results

The polyimide sensors showed some typical behavior which was not expected at forehand. Three trends were observed in measurements on multiple different sensors. In this section the measurements on a single sensor with design parameters $W_g = 8\mu m$ and $W_{ee} = 32\mu m$ will be used as

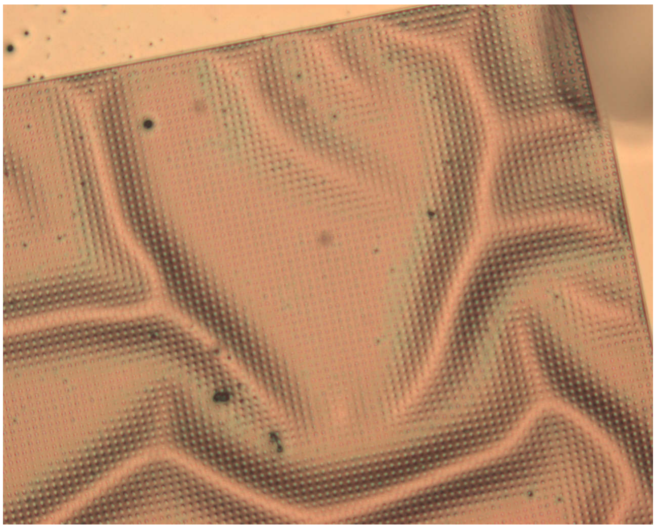
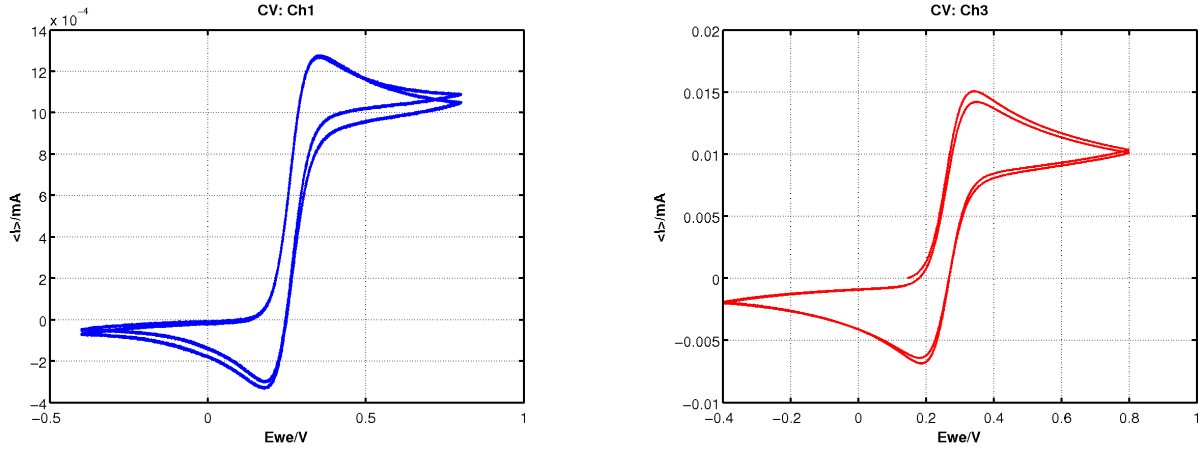


Figure 5.8: *Photo from a sensor with detached top electrode which is etched for 2100s in BHF at a magnification of 5x. The specific design parameters are:* $W_g = 8\mu m$ *and* $W_{ee} = 12\mu m$.

guideline to illustrate typical measurements on sensors with polyimide as insulating top layer. If necessary, references will be given to measurements included in the appendix to emphasise that the trends are observed with other sensors as well. The 8μm hole diameter refers to the size of the holes on the original mask. Measurements with the microscope indicated that the real diameter of the holes of this particular sensor is 8.71μm. All CV measurements presented in this section have the same scan settings, meaning the potential is switched from OCV to +0.8v, reversed to -0.4V and back to OCV. The scanrate is 20mV/s.

The measurements on the $W_g = 8\mu m$ and $W_{ee} = 32\mu m$ sensor before etching in BHF are shown in figure 5.9. The CV diagram measured on the bottom electrode is shown in 5.9a (channel 1). During this measurement the top electrode is kept at open circuit voltage (OCV). Figure 5.9b shows the CV-diagram of the top electrode (channel 3) while the bottom electrode is at OCV. From these measurements the first typical trend is visible. The peak current of the top electrode is slightly higher then the peak current of the bottom electrode. This observation can



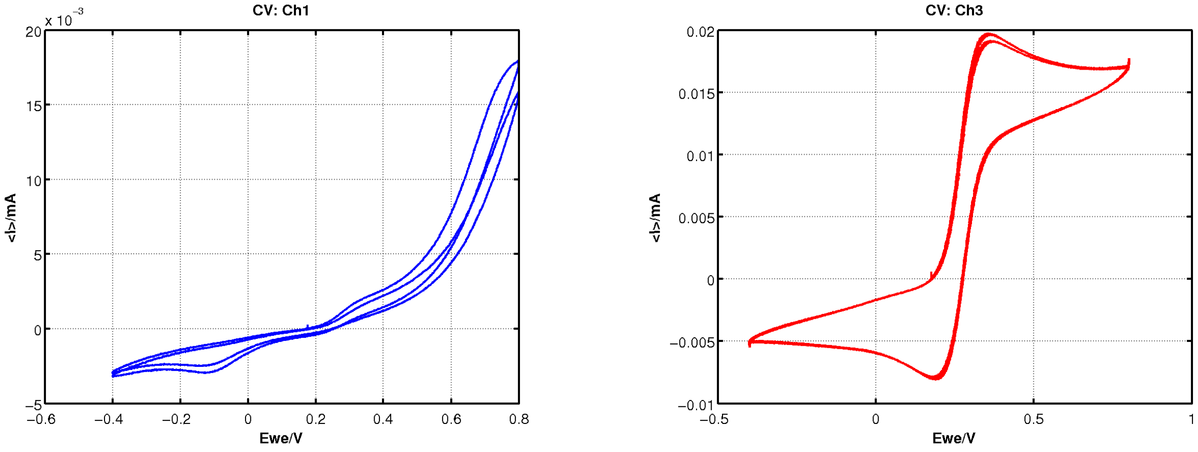
(a) CV-measurement of the bottom electrode (CH1), while the top electrode is at OCV.

(b) CV-measurement of the top electrode (CH3), while the bottom electrode is at OCV.

Figure 5.9: *CV measurements before sacrificial layer etching on a polyimide-type sensor. Parameters:* $W_g = 8\mu m$ *and* $W_{ee} = 32\mu m$.

be seen in the measurements shown in figure 5.11 and 5.12 as well. From the difference in work area it is expected that the current of the bottom electrode is higher than the current of the top electrode, especially in measurements before the sacrificial layer etching. This strange behavior might be due to an incomplete coverage of the top electrode by the polyimide. The photo shown in figure C.3 in the appendix might confirm this thesis. From this figure the local detachment of the polyimide around the holes is visible. Another explanation for the lower bottom electrode current might be given by the debris present in the holes. This dirt was already mentioned in section 4.1.4. This dirt was also observed within the holes of the sensor area. Perhaps the debris covered the bottom electrode, resulting in a decreased work area of the bottom electrode. However, the difference in current was still visible after cleaning and etching in BHF, long after the removal of the dirt. The debris-theory is therefore less likely as an explanation to the lower current of the bottom electrode. Besides this difference in current the shape of both CV-diagrams seems to be in order.

The same $W_g = 8\mu m$, $W_{ee} = 32\mu m$ sensor is etched for one hour. Like the photo shown in figure 5.7, the sensor area does not look bend as a result of the sacrificial layer etching. Photos from this sensor are included in appendix C (figure C.4 and figure C.5). The difference in radius of the hole in the polyimide and the space underneath the PI is 11.7μm.



(a) CV-measurement of the bottom electrode (CH1), while the top electrode is at OCV

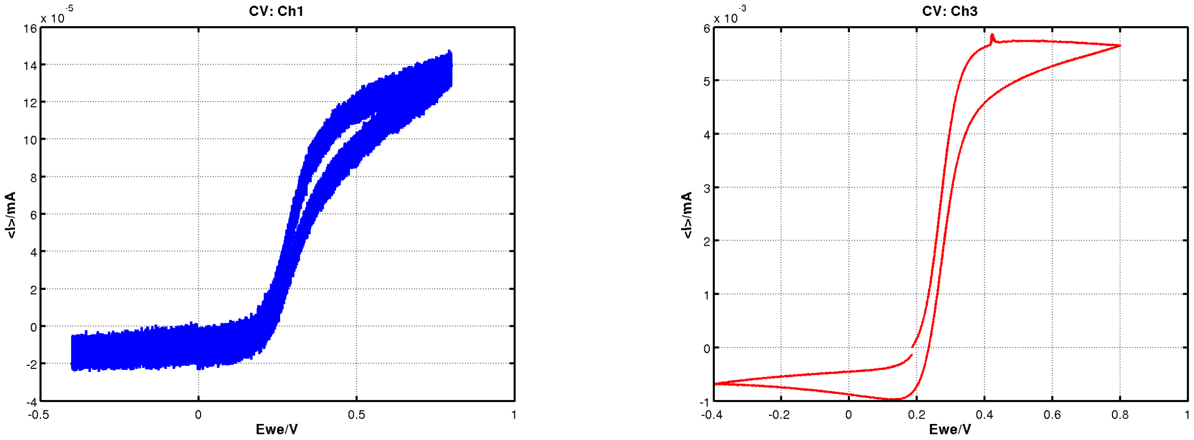
(b) CV-measurement of the top electrode (CH3), while the bottom electrode is at OCV

Figure 5.10: *CV measurements after 1 hour sacrificial layer etching on a polyimide-type sensor. Parameters:* $W_g = 8\mu m$ *and* $W_{ee} = 32\mu m$.

The CV-measurements after the sacrificial layer etching are shown in figures 5.10a and 5.10b. A nicely shaped CV is measured from the top electrode. The peak current of the top electrode is increased compared to the CV shown in figure 5.9b. The second trend is shown in CV-diagram of the bottom electrode. A similar distorted CV-diagram on the bottom electrode was shown before in figure 5.5 in the section discussing the silicon nitride sensor measurements. The most likely explanation for the CV from figure 5.5 was a short between the top and bottom electrode due to bending of the top electrode. Images from the microscope shown in figure 5.7 and C.5 indicate that the top electrode is not damaged or bend. A series of experiments have been conducted to find a plausible explanation for this strange behavior of the bottom electrode.

Figure 5.11 shows the first measurement from a series of measurements on a polyimide sensor with parameters $W_g = 2\mu m$ and $W_{ee} = 32\mu m$. Two CV's with identical settings are started simultaneously on both electrodes before any sacrificial layer etching. Both CV diagrams have a recognizable shape, although the measurement on the bottom electrode shows some noise.

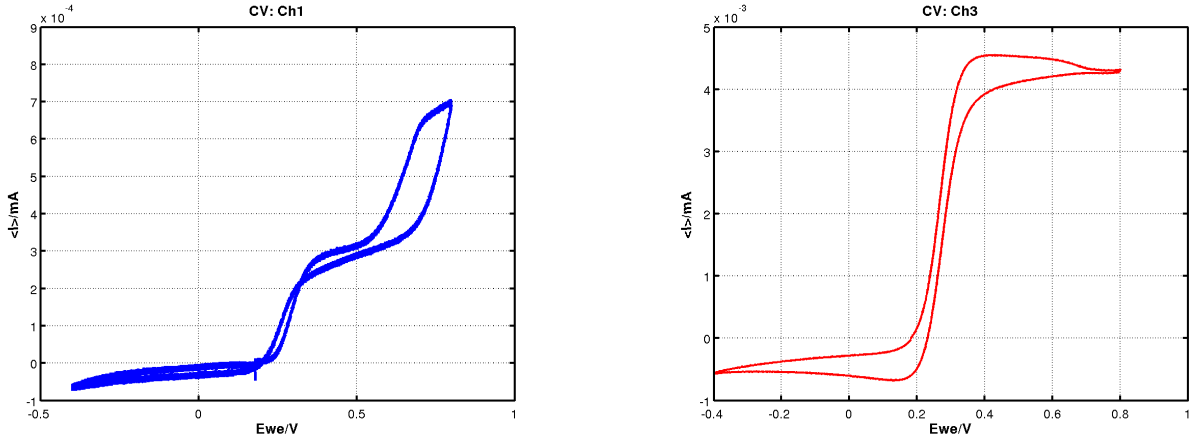
It is assumed that this noise is due to the small order of magnitude of the measured current. Compared to the measurements shown in figure 5.9 and 5.10, the peak current of both the top electrode and the bottom electrode is smaller. This smaller peak current is a result of the smaller work area of the electrode, since the holes of this sensor have a diameter of only $2\mu m$. The difference in magnitude of the currents from bottom and top electrode is even more visible from this measurement. In section 4.1.4 it was already mentioned that debris were present in the holes of the polyimide-type sensors. This debris might decrease the work area of the bottom electrode even further.



(a) CV-measurement of the bottom electrode (CH1), with a simultaneous CV measurement on the top electrode.

(b) CV-measurement of the top electrode (CH3), with a simultaneous CV measurement on the bottom electrode.

Figure 5.11: *Simultaneous CV measurements before sacrificial layer etching on a polyimide-type sensor. Parameters:* $W_g = 2\mu m$ *and* $W_{ee} = 32\mu m$.



(a) CV-measurement of the bottom electrode (CH1), with a simultaneous CV measurement on the top electrode.

(b) CV-measurement of the top electrode (CH3), with a simultaneous CV measurement on the bottom electrode.

Figure 5.12: *Simultaneous CV measurements after 300s sacrificial layer etching on a polyimide-type sensor. Parameters:* $W_g = 2\mu m$ *and* $W_{ee} = 32\mu m$.

This $W_g = 2\mu m$, $W_{ee} = 32\mu m$ sensor is now etched for 300s in BHF. Afterward the same experiment is repeated. The result is shown in figure 5.12. The CV on the bottom electrode shows the previously mentioned abnormal behavior. However, it is interesting to notice that the CV of the top electrode shows a small decrease on the moment of increase in current on the CV of the bottom electrode. It seems like the top electrode is providing current to the bottom electrode. This can occur if both electrodes form a short. This short has a bigger impact on the bottom electrode, because of its small current. It is therefore less visible in the CV of the top electrode since its current around +0.6V is much larger. To test this short theory, both CV's are added. The combined result is shown in figure 5.13. The decrease in current of the top electrode and the increase of current of the bottom electrode seem to be in the same order of magnitude.

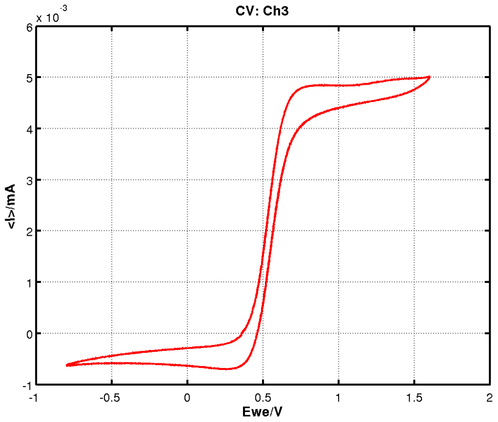
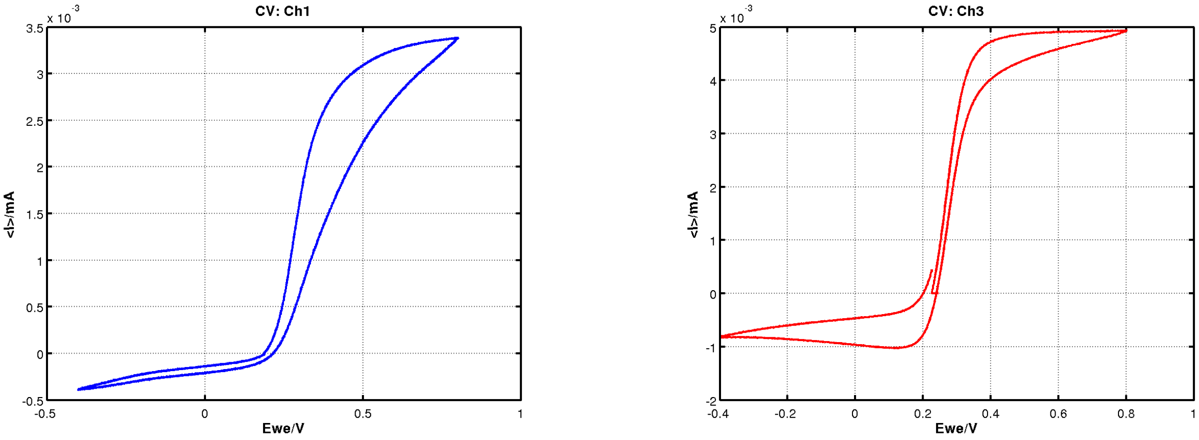


Figure 5.13: *Combination of the two CV-diagrams shown in figure 5.12.*

Figure 5.14 shows another measurement on the $W_g = 2\mu m$, $W_{ee} = 32\mu m$ sensor. In this experiment a CV is measured on one of the electrodes, while the other is kept at OCV. As always the CV on the top electrode (5.14b) looks fine, although the current is slightly increased compared to the measurements shown in figure 5.12b. However, the CV shown in figure 5.14a looks like the combination of a CV-diagram and the measurement on a diode.



(a) CV-measurement of the bottom electrode (CH1), while the top electrode is at OCV

(b) CV-measurement of the top electrode (CH3), while the bottom electrode is at OCV

Figure 5.14: *CV measurements on a single electrode, while the other electrode is at OCV. Parameters:* $W_g = 2\mu m$ *and* $W_{ee} = 32\mu m$*, 300s etching.*

The unusual CV-measurement shown in figure 5.14a might be explained by the equivalent schematic shown in figure 5.15. Suppose an electrical connection (indicated as a resistor in the figure) is formed sometime during the CV measurement on the bottom electrode. The potential of the top electrode is measured with the very high-impedance measurement channel 3 of the bi-stat. Due to the electrical connection, the potential of the top electrode will follow the potential applied to the bottom electrode. Therefore the top electrode will start to conduct electrochemistry. The electrons that will be generated due to this electrochemistry have only the newly formed electrical connection as pathway to leave the top electrode. As a result, the measured current of the bottom electrode will increase. Of course this is not a complete explanation. It does not explain why the increase in current shown in figure 5.14a is only observed for positive potentials for example.

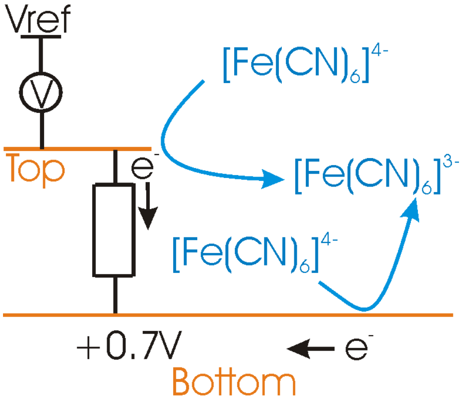


Figure 5.15: *Equivalent schematic as possible explanation for the unexpected CV measurements.*

Figure 5.15 suggests that the increase in current is due to electrochemical reactions occurring at the top electrode. It might be interesting to see the results of a CV measurement from the bottom electrode, while the top electrode is set to a potential close to the standard potential of the used redox-couple. If a short exists, an even larger increase in current might be observed as soon as the potential difference between top and bottom electrode is increased. This idea is illustrated by the equivalent schematic shown in figure 5.16.

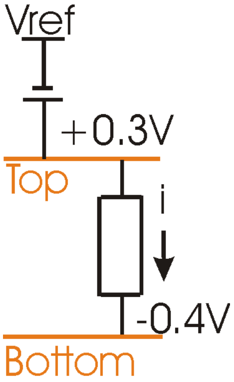
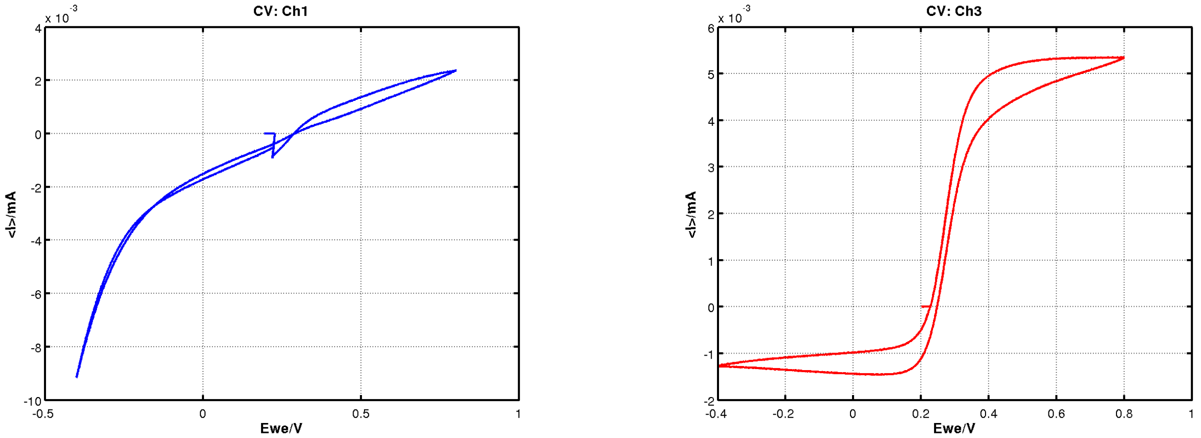


Figure 5.16: *Equivalent schematic as possible explanation for the unexpected CV measurements.*

The CV-measurement on bottom electrode with the top electrode at a potential of +0.3V is shown in figure 5.17a. The same experiment, with the CV conducted to the top electrode is shown in figure 5.17b. Another diode-characteristic can be derived from the measurements on the bottom electrode (fig. 5.17a), although the increase in current is now seen at negative potentials. the peak current measured from the top electrode has increased compared to the measurements shown in figure 5.14b and 5.12b. These diode-like measurements have been observed on several other devices shown in figure C.6 and C.7 in appendix C.



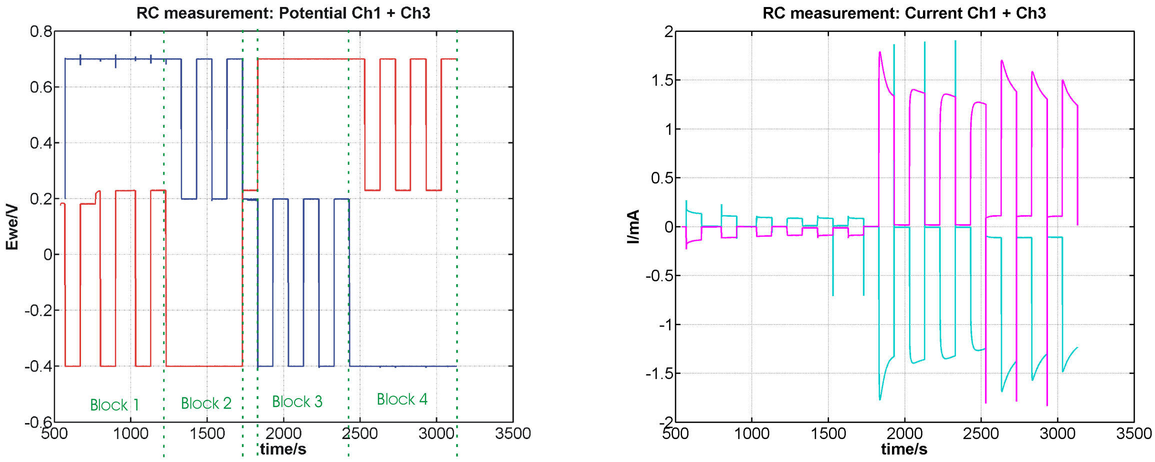
(a) CV-measurement of the bottom electrode (CH1), while the top electrode is at a fixed potential of +0.3V

(b) CV-measurement of the top electrode (CH3), while the bottom electrode is at a fixed potential of +0.3V

Figure 5.17: *CV measurements on a single electrode, while the other electrode is fixed at +0.3V. Parameters:* $W_g = 2\mu m$ *and* $W_{ee} = 32\mu m$*, 300s etching.*

This strange behavior of the sensors with polyimide, might be explained by a breakdown of the remaining sacrificial silicon oxide layer. It is suspected that the silicon oxide had a breakdown during the processing of the wafer. During the ion beam etch process it was noticed that the neutralizer showed abnormal behavior. The neutralizer is responsible for the neutralization of the charged argon-ions that are fired at the wafer surface. Therefore it is likely that the top electrode was charged during the etch of the top electrode with this ion-beam etch process. If too much charge builds up at the top electrode, the potential difference between top and bottom may be to high and charge will find a way through the silicon nitride to the bottom electrode. Once a breakdown has occurred in silicon oxide, it is known that breakdowns will occur more often at lower voltages. The result of a breakdown during the electrochemical measurements presented here is that bottom and top electrode will form a short at a certain potential difference. Especially figure 5.12 and 5.13 confirm a short between the two electrodes. It was already mentioned that all measurements presented here are measured from the wafer with a sacrificial layer of 900nm thick. Almost all sensors with a sacrificial layer thickness of 450nm were shorted. The breakdown-theory might be an explanation for this shorted sensors as well, since a breakdown will occur sooner if the sacrificial layer thickness is decreased. Ions diffusing into the sacrificial layer might initiate these breakdowns even more.

To conclude this section one final measurement is shown on our previously mentioned sensor with parameters $W_g = 8\mu m$ and $W_{ee} = 32\mu m$ which was etched for one hour in BHF. Figure 5.18 shows a redox cycling amplification measurement. The applied potential is shown in figure 5.18a, while the current response is shown in figure 5.18b. In the first block the potential at the bottom electrode is fixed at 0.7V, while the potential at the top electrode is switched between OCV and -0.4V. The current from the bottom electrode is equal to approximately 0.11mA if the potential at the top electrode is equal to -0.4V. It decreases to 0.8μA if the potential of the top electrode is switched to OCV. This means a difference of $0.14/0.0008 = 175x$. In the second block the potential of the top electrode is fixed to -0.4V, while the bottom electrode is switched between OCV and +0.7V. The current response measured on the top electrode is between 0.087mA and 0.012mA, which is a factor of 7.2x. It is remarkable to see that the currents involved with the third and fourth block are much larger. It can only be assumed that some sort of diode-effect due to a voltage breakdown has occurred in this measurement as well, resulting in the larger current for the third and fourth block. The currents observed from the bottom electrode in the third block are between 1.3722mA and 0.0053mA, which is a difference of 259x. For the fourth block the measured currents from the top electrode are between 1.36mA and 0.1074mA, resulting in a difference of 12.7x. The significant differences between these factors also indicate that the measured amplification factors are not only the result of redox cycling.



(a) RC measurement: Applied potentials to bottom (blue) and top (red) electrode.

(b) RC measurement: Current response from bottom (aqua) and top (pink) electrode.

Figure 5.18: *Redox amplification measurements after 1 hour sacrificial layer etching on a polyimide-type sensor. Parameters:* $W_g = 8\mu m$ *and* $W_{ee} = 32\mu m$.

The same redox cycling amplification measurement has been repeated for a $W_g = 8\mu m$, $W_{ee} = 12\mu m$ sensor which is etched for 1800s and a $W_g = 8\mu m$, $W_{ee} = 8\mu m$ sensor which is etched for 1200s. The corresponding figures showing the actual measurements are included in appendix C (figure C.8 and figure C.9). The maximum recorded redox cycling amplification is shown in figure 5.19 for these three sensors. The line shows a linear trend as predicted in the simulation chapter (see figure 3.15). However, the strange behavior of the sensors with respect to the breakdown and current difference measured from the CV-diagrams indicate that these results are not only due to redox cycling. Therefore, the data shown in figure 5.19 does not have much importance.

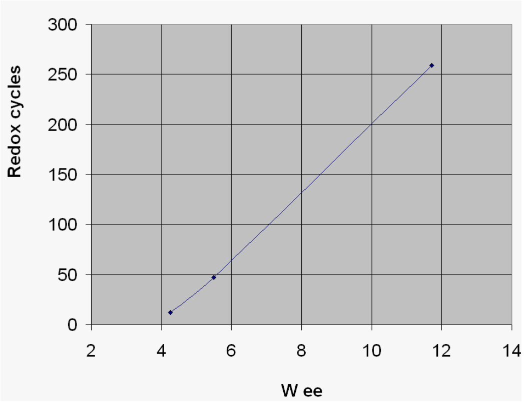


Figure 5.19: *Redox amplification measurements on three sensors with different parameters.*

5.4 Conclusion

In this section measurements on two different types of wafers were discussed. The first type, with the silicon nitride as insulating layer, had problems due to the removal of the silicon nitride during the sacrificial layer etching. The silicon nitride was meant as supporting layer to prevent bending of the top electrode. However, the silicon nitride layer was weakened too much during the etch, causing the top layer to move toward the bottom layer. As a result multiple shorts were measured on these devices.

The second type of wafer was fabricated with polyimide as insulating layer. The polyimide proofed to be a better candidate as insulating layer due to its high chemical resistivity to BHF. However, problems during the processing of this wafer resulted in a sacrificial layer which had voltage breakdowns. These breakdowns caused unpredictable behavior and unwanted electrical connections between the top and bottom work electrode.

Some redox cycling has been measured, but the aim to gain high amplification factors above 60 to 70x is not achieved. In chapter 7 some recommendations will be presented which will hopefully prevent the problems encountered so far.

6 Conclusion

In the introduction two goals were formulated for this project:

1. Find a plausible explanation for the high number of redox cycles measured by V.A.T. Dam.
2. Design, production and testing of a novel sensor with the goal of gaining even higher redox-cycling amplification factors than the 60 to 70 times observed by V.A.T. Dam.

The simulations of the sensor with the same geometry as the IDA-electrode used by V.A.T. Dam predict a redox cycling amplification of 74x. This is close to the 60 to 70x found by V.A.T. Dam. This high number of redox cycles is due to the enclosed space between the two electrodes formed by the trench. Therefore a plausible explanation for the results of V.A.T. Dam is found.

Simulations of the novel sensor structure with passive bulk supply predicted redox cycles up to 400x with certain specific design parameters ($w_{ee} = 16\mu m$, $w_g = 1\mu m$, $h = 0.1\mu m$). These promising theoretical results have not been achieved in practical measurements. Problems with a wrong choice as insulating layer, a bad adhesion of the insulating layer to the top electrode and a sacrificial silicon oxide layer with irreversible damage due to voltage breakdowns gave disappointing results. Therefore it can be concluded that the second goal of this thesis is not achieved.

7 Recommendations for further research

7.1 Improved sensor design

The most obvious recommendation is the improvement of the sensor fabrication. The practical results were disappointing due to a number of problems with the fabricated devices. Four suggestions for improvement of the devices were formulated during discussions with Johan Bomer and Jan Eijkel.

1. Polyimide will be maintained as insulating top layer for its chemical resistivity to BHF. Therefore it is very important that no UV-foil will be used during the dicing of the wafer. The problems with the removal of this foil do not even out the advantages of protection against dicing particles.

2. All sensors containing polyimide showed a higher current from the top electrode than the current from the bottom electrode. It is believed that this higher current is due to bad adhesion of the polyimide around the holes in the top electrode. To prevent this behavior in the future, a small layer of aluminum oxide may be provided between the top electrode and the polyimide to improve the electrochemical insulation of the top electrode to the bulk of the solution. It is known that aluminum oxide layers can induce a lot of stress. Therefore it is important that only a thin layer will be used.

3. The cyclic voltammetry measurements showed a lot of unexpected behavior of the sensors. It is believed that this is caused by a breakdown of the remaining sacrificial silicon oxide layer. Probably the first breakdown was induced by the ion-beam etch (IBE) process or the reactive ion etch (RIE) process. Once a breakdown has occurred, it will occur again at lower voltages due to irreversible damage. It is important to prevent the first breakdown during the processing, since a breakdown is less likely to occur during measurements because of the low voltages involved. The breakdown of the sacrificial silicon oxide layer can be prevented by a temporary short of bottom and top electrode during the processing. Therefore a new mask and processing diagram is designed which provides this short during the cleanroom processing. The short is removed during the dicing process.

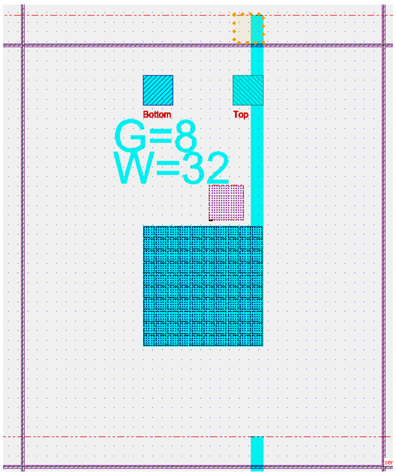


Figure 7.1: *Single sensor mask layout (blue=mask1, green=mask2, purple=mask3).*

This new mask design is shown in figure 7.1. The basic idea is to use the green mask two times. The first time it will be used to create a hole in the sacrificial layer at the position of the orange dotted square. The second time it will be used to create a contact pad for the top electrode. Mask 2 is extended with a path to the orange dotted square to provide a short from bottom electrode to top electrode. This path is diced at the end of the processing.

4. Another solution to prevent a voltage breakdown is to enhance the insulating properties of the sacrificial layer. In a paper of F. Heer [27] it is shown that a sandwich of silicon oxide/silicon nitride/silicon oxide (ONO) has better insulating properties. It might be interesting to fabricate one wafer with this ONO layer to study its effect on the sensor properties. From the etching experiments it is known that silicon nitride also etches quite fast in BHF. Although differences between the etch rate of silicon oxide and silicon nitride might lead to new problems during the processing.

7.2 Selectivity measurements

In the work of V.A.T. Dam [1] measurements to study the selectivity of the IDA-electrode are presented. Figure 7.2 shows three CV-measurements from the work of V.A.T. Dam. Two potential ranges are indicated in the figure by E_1 and E_2. Suppose a potential of +0.7V is applied to the generator electrode, while the collector is set to a potential of +0.4V (E_2 range). All the ferrocyanide will remain in its oxidized state and only the current measured from the dopamine will amplify due to redox cycling. This method can be used to increase the selectivity of the sensor. The same method can be used to measure $K_4[Fe(CN)_6]$ selectively. The generator is now set to +0.4V, while the collector is set to -0.2V (E_1 range). The dopamine will now remain in its reduced state and only the current of the $K_4[Fe(CN)_6]$ will amplify due to redox cycling.

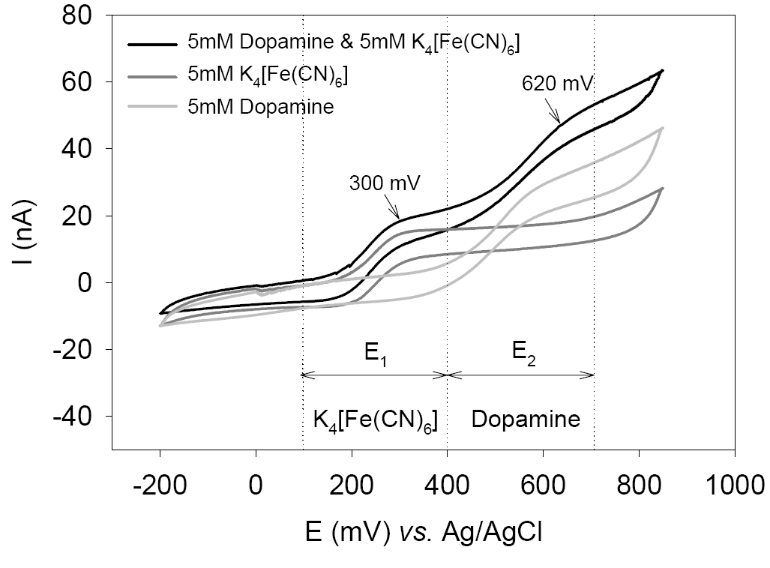


Figure 7.2: *CV measurements on 5mM dopamine, 5mM $K_4[Fe(CN)_6]$ and a mixture of both chemicals.*

With some small adjustments to the model, the selectivity measurement can be simulated. It might be interesting to simulate the selectivity measurements on the IDA-electrode and check the results with the experimental work of V.A.T. Dam. Also, the selectivity measurements can be repeated with the sensor design presented in this project once a good functioning sensor is fabricated. It is expected that the selectivity will increase with higher redox cycling amplification.

8 Bibliography

References

[1] V. A. T. Dam, W. Olthuis, and A. van den Berg. Redox cycling with facing interdigitated array electrodes as a method for selective detection of redox species. *The Analyst*, 1:1, 2007. DOI: 10.1039/b616667a.

[2] Allen J. Bard, Joseph A. Crayston, Gregg P. Kittlesen, Theresa Varco Shea, and Mark S. Wrighton. Digital simulation of the measured electrochemical response of reversible redox couples at microelectrode arrays: Consequences arising from closely spaced ultramicroelectrodes. *Analytical Chemistry*, 58:2321–2331, 1986.

[3] Xiaoling Yang and Guigen Zhang. Diffusion-controlled redox cycling at nanoscale interdigitated electrodes. *Comsol Proceedings and user presentations CD*, 1:1–6, 2005.

[4] Koichi Aoki, Masao Morita, Osamu Niwa, and Hisao Tabei. Quantitative analysis of reversible diffusion-controlled currents of redox soluble species at interdigitated array electrodes under steady-state conditions. *J. Electroanal. Chem.*, 256:269–282, 1988.

[5] Koichi Aoki. Approximate models of interdigitated array electrodes for evaluating steady-state currents. *J. Electroanal. Chem.*, 284:35–42, 1990.

[6] I. Rubinstein. *Physical Electrochemistry - principles, methods and applications.* Number 0-8247-9452-4 (ISBN). Marcel Dekker, 1995. Ch4. Electrochemistry at Ultramicroelectrodes by C. Amatore; Ch3. Digital simulations with the Fast Implicit Finite Difference algorithm by M. Rudolph.

[7] Yuzuru Iwasaki and Masao Morita. Electrochemical measurements with interdigitated array microelectrodes. *Current separations*, 14:1–8, 1995.

[8] Allen J. Bard. *Electrochemical methods - fundamentals and applications.* Number 0-471-04372-9 (ISBN). Wiley, 2nd edition, 2001.

[9] W. Olthuis. Lecture notes from the course biomedical signal acquisition. 2005.

[10] Osamu Niwa, Masao Morita, and Hisao Tabei. Electrochemical behavior of reversible redox species at interdigitated array electrodes with different geometries: Consideration of redox cycling and collection efficiency. *Analytical Chemistry*, 62:447–452, 1990.

[11] Douglas G. Sanderson and Larry B. Anderson. Filar electrodes: Steady-state currents and spectroelectrochemistry at twin interdigitated electrodes. *Analytical Chemistry*, 57:2388–2393, 1985.

[12] B.J. Seddon, H.H. Girault, and M.J. Eddowes. Interdigitated microband electrodes: chronoamperometry and steady state currents. *J. Electroanal. Chem.*, 266:227–238, 1989.

[13] Christopher G. Phillips and H.A. Stone. Theoretical calculation of collection efficiencies for collector-generator microelectrode systems. *Journal of Electroanalytical Chemistry*, 437:157–165, 1997.

[14] Baokang Jin, Weijun Qian, Zuxun Zhang, and Hansheng Shi. Application of the finite analytic numerical method. part 1. diffusion problems on coplanar and elevated interdigitated microarray band electrodes. *Journal of Electroanalytical Chemistry*, 411:29–36, 1996.

[15] C. Beriet, R. Ferrigno, and H.H. Girault. Simulation of the chronoamperometric response of a regular array of micro-disc electrodes. *Journal of Electroanalytical Chemistry*, 486:56–64, 2000.

[16] M. Paeschke, U. Wollenberger, C. Kohler, T. Lisec, U. Schnakenberg, and R. Hintsche. Properties of interdigital electrode arrays with different geometries. *Analytica Chimica Acta*, 305:126–136, 1995.

[17] Masao Morita, Osamu Niwa, and Tsutomu Horiuchi. Interdigitated array microelectrodes as electrochemical sensors. *Electrochemica Acta*, 42(20-22):3177–3183, 1997.

[18] Hisao Tabei, Masaya Takahashi, Satoshl Hoshino, Osamu Niwa, and Tsutomu Horiuchi. Subfemtomole detection of catecholamine with interdigitated array carbon microelectrodes in hplc. *Analytical Chemistry*, 66(20):3500–3502, 1994.

[19] Ryoji Kurita, Hisao Tabei, Zhiming Liu, Tsutomu Horiuchi, and Osamu Niwa. Fabrication and electrochemical properties of an interdigitated array electrode in a microfabricated wall-jet cell. *Sensors and actuators B*, 71:82–89, 2000.

[20] Katsuyoshi Hayashi, Yuzuru Iwasaki, Ryoji Kurita, Kenji Sunagawa, Osamu Niwa, and Akiyuki Tate. The highly sensitive detection of catecholamines using a microfluidic device integrated with an enzyme-modified pre-reactor for interferent elimination and an interdigitated array electrode. *Journal of Electroanalytical Chemistry*, 579:215–222, 2005.

[21] Peng Jin, Akira Yamaguchi, Fumika Asari Oi, shigeki Matsuo, Jiubin Tan, and Hiroaki Misawa. Glucose sensing based on interdigitated array microelectrode. *Analytical sciences*, 17:841–846, 2001.

[22] D. Bustin, S. Mesaros, P. Tomcik, M. Rievaj, and V. Tvarozek. Application of redox cycling enhanced current at an interdigitated array electrode for iron-trace determination in ultrapure spectral carbon. *Analytica chemica acta*, 305:121–125, 1995.

[23] Marcus Mosbach, Thomas Laurell, Johan Nilsson, Elisabeth Csoregi, and Wolfgang Schuhmann. A miniaturised electrochemical affinity assay based on a wall-free sample droplet and micro-dispensing of the redox-labelled binding partner. *Biosensors & Bioelectronics*, 16:611ï£¡620, 2001.

[24] Marcel A. G. Zevenbergen, Diego Krapf, Marc R. Zuiddam, and Serge G. Lemay. Mesoscopic concentration fluctuations in a fluidic nanocavity detected by redox cycling. *Nano Letters*, 0(0):1, 2006. in press.

[25] Junhong Min and Antje J. Baeumner. Characterization and optimization of interdigitated ultramicroelectrode arrays as electrochemical biosensor transducers. *Electroanalysis*, 16(9):724–729, 2004.

[26] Kirt R. Williams and Matthew Wasilik. Etch rates for micromachining processing - part ii. *Journal of microelectromechnical systems*, 12(6):761–778, 2003.

[27] F. Heer, W. Franks, A. Blau, S. Taschini, C. Ziegler, A. Hierlemann, and H. Baltes. Cmos microelectrode array for the monitoring of electrogenic cells. *Biosensors and Bioelectronics*, 20:358–366, 2004. not printed, not read thorough.

A Simulation results

A.1 Mesh convergence study

Table A.1: *Current at the generator/collector electrode as a function of the number of mesh elements.*

Max. element size (m)	**Mesh elements**	**Current col. [x] (A/m)**	**Current gen. [∇] (A/m)**
1.00e-007	3.86e+003	1.31e-004	1.74e-004
5.00e-008	4.18e+003	1.36e-004	1.77e-004
2.50e-008	4.88e+003	1.39e-004	1.84e-004
1.00e-008	6.77e+003	1.43e-004	1.88e-004
5.00e-009	9.53e+003	1.44e-004	1.89e-004
3.00e-009	1.34e+004	1.44e-004	1.90e-004
1.00e-009	3.17e+004	1.45e-004	1.91e-004

A.2 Semi-infinite bulk boundary condition study

Table A.2: *Current at the semi infinite boundary as a function of time and bulk height*

	y_{bulk}=**10um [+]**	y_{bulk}=**20um [o]**	y_{bulk}=**30um [s]**	y_{bulk}=**40um [x]**	y_{bulk}=**50um [v]**
t=0 s	-3.26e-024	-8.33e-025	-4.78e-024	3.32e-023	-1.81e-025
t=0.005 s	2.44e-012	1.15e-019	-4.69e-023	-3.22e-023	-4.17e-023
t=0.01 s	6.32e-011	2.79e-015	3.49e-020	2.46e-023	-3.29e-023
t=0.015 s	1.87e-010	1.48e-013	6.55e-017	1.37e-020	-3.66e-023
t=0.02 s	3.06e-010	1.29e-012	1.31e-015	7.63e-019	1.16e-022
t=0.025 s	4.01e-010	4.87e-012	1.29e-014	1.51e-017	5.17e-021
t=0.03 s	4.74e-010	1.19e-011	6.96e-014	2.67e-016	6.68e-020
t=0.035 s	5.26e-010	2.23e-011	2.30e-013	1.41e-015	5.37e-019
t=0.04 s	5.64e-010	3.58e-011	5.54e-013	4.65e-015	8.25e-018
t=0.045 s	5.94e-010	5.14e-011	1.12e-012	1.23e-014	3.70e-017
t=0.05 s	6.16e-010	6.82e-011	2.01e-012	2.92e-014	1.49e-016
t=0.055 s	6.32e-010	8.54e-011	3.28e-012	6.14e-014	4.01e-016
t=0.06 s	6.43e-010	1.03e-010	4.99e-012	1.17e-013	1.05e-015
t=0.065 s	6.52e-010	1.20e-010	7.16e-012	2.04e-013	2.31e-015
t=0.07 s	6.57e-010	1.35e-010	9.83e-012	3.36e-013	4.84e-015
t=0.075 s	6.61e-010	1.50e-010	1.30e-011	5.21e-013	9.64e-015
t=0.08 s	6.64e-010	1.64e-010	1.66e-011	7.73e-013	1.87e-014
t=0.085 s	6.66e-010	1.78e-010	2.07e-011	1.11e-012	3.29e-014
t=0.09 s	6.67e-010	1.90e-010	2.51e-011	1.54e-012	5.35e-014
t=0.095 s	6.67e-010	2.02e-010	2.98e-011	2.09e-012	8.42e-014
t=0.1 s	6.67e-010	2.13e-010	3.47e-011	2.75e-012	1.26e-013
t=0.105 s	6.67e-010	2.23e-010	3.97e-011	3.55e-012	1.81e-013
t=0.11 s	6.68e-010	2.33e-010	4.47e-011	4.47e-012	2.52e-013
t=0.115 s	6.68e-010	2.42e-010	4.98e-011	5.53e-012	3.44e-013
t=0.12 s	6.69e-010	2.50e-010	5.49e-011	6.71e-012	4.58e-013
t=0.125 s	6.69e-010	2.58e-010	6.00e-011	7.98e-012	5.97e-013
t=0.13 s	6.69e-010	2.65e-010	6.51e-011	9.36e-012	7.64e-013
t=0.135 s	6.69e-010	2.71e-010	7.02e-011	1.08e-011	9.63e-013
t=0.14 s	6.70e-010	2.77e-010	7.51e-011	1.24e-011	1.20e-012
t=0.145 s	6.70e-010	2.83e-010	8.00e-011	1.40e-011	1.46e-012
t=0.15 s	6.70e-010	2.87e-010	8.48e-011	1.58e-011	1.77e-012
t=0.155 s	6.70e-010	2.92e-010	8.96e-011	1.76e-011	2.11e-012
t=0.16 s	6.70e-010	2.96e-010	9.42e-011	1.95e-011	2.50e-012
t=0.165 s	6.70e-010	3.00e-010	9.88e-011	2.14e-011	2.93e-012
t=0.17 s	6.70e-010	3.04e-010	1.03e-010	2.34e-011	3.40e-012
t=0.175 s	6.70e-010	3.07e-010	1.08e-010	2.54e-011	3.91e-012
t=0.18 s	6.70e-010	3.10e-010	1.12e-010	2.75e-011	4.46e-012
t=0.185 s	6.70e-010	3.13e-010	1.16e-010	2.96e-011	5.06e-012
t=0.19 s	6.70e-010	3.15e-010	1.20e-010	3.18e-011	5.69e-012
t=0.195 s	6.70e-010	3.17e-010	1.24e-010	3.39e-011	6.37e-012
t=0.2 s	6.70e-010	3.19e-010	1.28e-010	3.61e-011	7.07e-012

A.3 IDA with varied gap and electrode width

Table A.3: *Collection efficiency, redox cycles and electrode current with different IDA geometries*

We/2 (m)	**Wg (m)**	**Current col. [x] (A/m)**	**Current gen. [∇](A/m)**	**CE [o] (%)**	**RC [+]**
2.50e-08	2.00e-07	9.87e-05	1.01e-04	9.76e+01	2.10e+01
6.25e-08	5.00e-07	9.91e-05	1.05e-04	9.40e+01	8.62e+00
1.25e-07	1.00e-06	9.63e-05	1.10e-04	8.77e+01	4.34e+00
1.88e-07	1.50e-06	9.29e-05	1.14e-04	8.15e+01	2.98e+00
2.50e-07	2.00e-06	8.97e-05	1.18e-04	7.62e+01	2.39e+00
5.00e-08	2.00e-07	1.27e-04	1.29e-04	9.84e+01	3.18e+01
1.25e-07	5.00e-07	1.26e-04	1.34e-04	9.36e+01	8.11e+00
2.50e-07	1.00e-06	1.22e-04	1.39e-04	8.81e+01	4.46e+00
3.75e-07	1.50e-06	1.18e-04	1.43e-04	8.25e+01	3.13e+00
5.00e-07	2.00e-06	1.14e-04	1.48e-04	7.69e+01	2.45e+00
1.00e-07	2.00e-07	1.63e-04	1.67e-04	9.78e+01	2.32e+01
2.50e-07	5.00e-07	1.61e-04	1.72e-04	9.36e+01	8.07e+00
5.00e-07	1.00e-06	1.57e-04	1.79e-04	8.77e+01	4.32e+00
7.50e-07	1.50e-06	1.51e-04	1.85e-04	8.15e+01	2.98e+00
1.00e-06	2.00e-06	1.45e-04	1.91e-04	7.63e+01	2.39e+00

A.4 IDA with elevated electrodes

Table A.4: *Collection efficiency of different electrode heights*

He (m)	**Current col. [x] (A/m)**	**Current gen. [∇] (A/m)**	**CE [o](%)**	**RC [+]**
1.00e-007	1.67e-004	2.13e-004	7.87e+001	2.62e+000
2.50e-007	1.95e-004	2.40e-004	8.11e+001	2.92e+000
5.00e-007	2.38e-004	2.83e-004	8.40e+001	3.40e+000
1.00e-006	3.22e-004	3.68e-004	8.75e+001	4.26e+000
2.00e-006	4.91e-004	5.36e-004	9.17e+001	6.27e+000
3.00e-006	6.60e-004	7.05e-004	9.36e+001	8.13e+000
5.00e-006	9.97e-004	1.04e-003	9.56e+001	1.16e+001
7.00e-006	1.34e-003	1.38e-003	9.68e+001	1.57e+001

A.5 Parallel plate electrodes with convection

Table A.5: *Collection efficiency and electrode current at different fluid velocities*

u_{max} **(m/s)**	**Current col. [x](A/m)**	**Current gen. [∇](A/m)**	**CE [o](%)**	**RC [+]**
1.00e-03	2.98e-03	3.27e-03	9.13e+01	6.03e+00
5.00e-03	2.66e-03	3.84e-03	6.92e+01	1.92e+00
1.00e-02	2.24e-03	4.62e-03	4.86e+01	1.31e+00
2.00e-02	1.52e-03	6.12e-03	2.49e+01	1.07e+00
5.00e-02	4.21e-04	9.58e-03	4.40e+00	1.00e+00
1.00e-01	4.95e-05	1.35e-02	3.65e-01	1.00e+00

A.6 Parallel plate electrodes with passive bulk supply

Table A.6: *Collection efficiency with different edge-to-edge distances between the holes*

W_{total} **p/unit**	**Flux gen. | coll. on [v](mol/(m*s))**	**Flux gen. | coll. off [x](mol/(m*s))**	**CE [o](%)**	**RC [+]**
2.00e-006	-5.99e-009	4.47e-010	9.62e+001	1.34e+001
4.00e-006	-1.30e-008	7.53e-010	9.71e+001	1.73e+001
8.00e-006	-2.70e-008	1.04e-009	9.81e+001	2.60e+001
1.60e-005	-5.50e-008	1.20e-009	9.89e+001	4.57e+001
3.20e-005	-1.11e-007	1.24e-009	9.94e+001	8.93e+001

B Processing steps

This is version 1.3 made on 27-9-2007.

B.1 Introduction to the parallel plate sensor

The parallel plate sensor is developed with the goal of oxidizing and reducing electrochemical active species. The redox cycling effect is used to chemically amplify the measured signal to have a small detection limit and better SNR. The sensor consists out of two parallel plates acting as electrodes. The top electrode has small holes in it to make it possible for the electrolyte to flow in between the parallel plates. The diameter of these holes, the distance between the holes and the distance between the two parallel plates are the major parameters that influence the performance of the device.

B.2 Mask layout

B.2.1 Overall mask layout

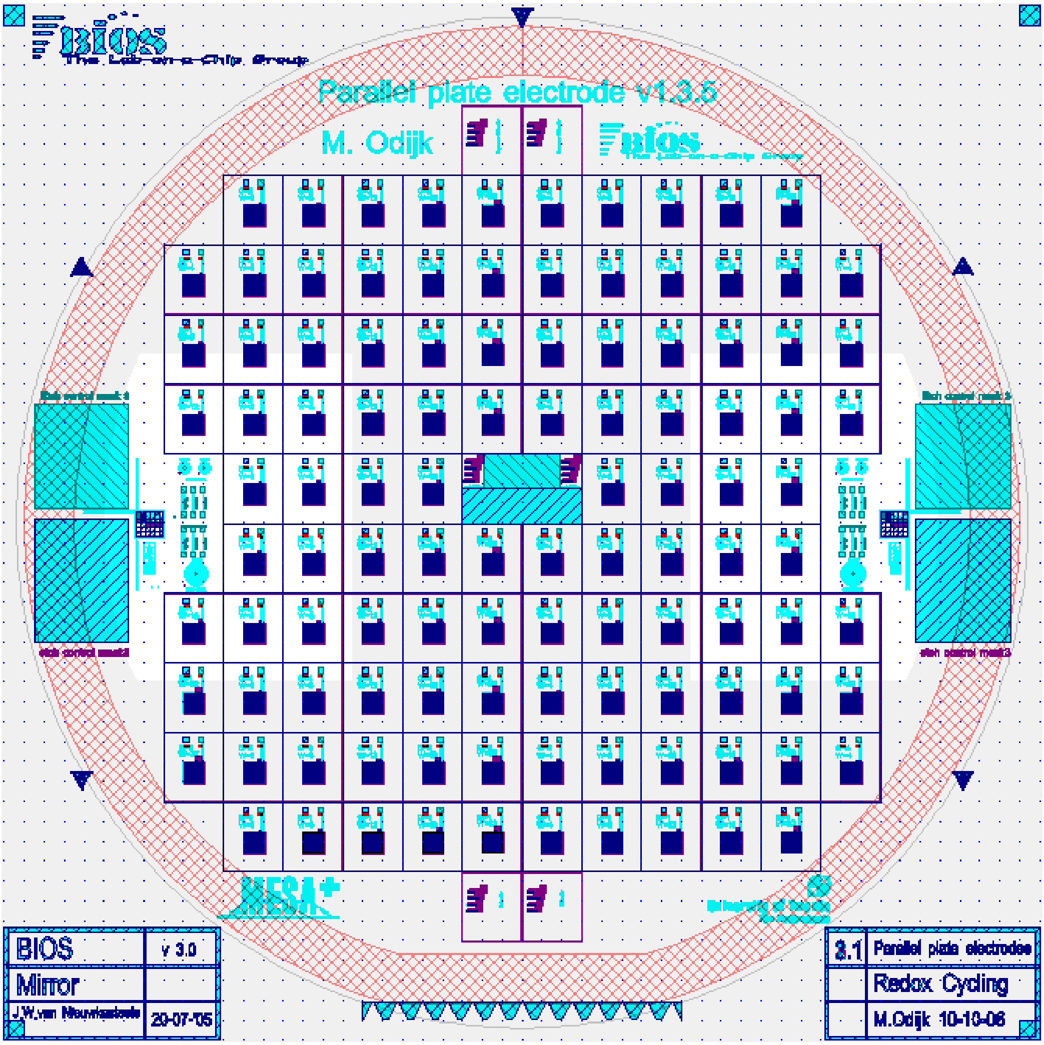


Figure B.1: *Overall mask layout (blue=mask1, purple=mask2, green=mask3).*

B.2.2 Single sensor mask layout

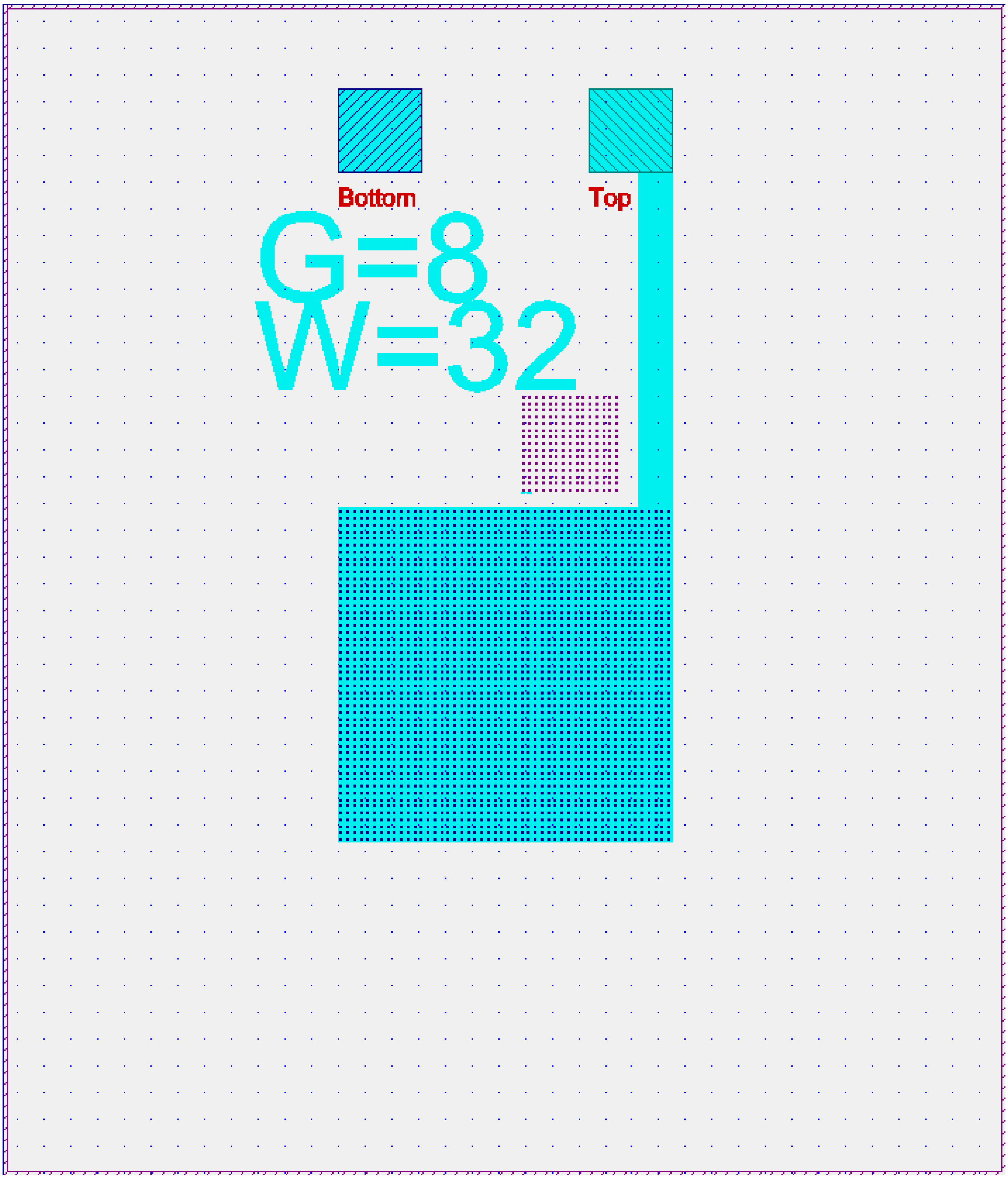


Figure B.2: *Single sensor mask layout (blue=mask1, green=mask2, purple=mask3).*

B.2.3 Test structures mask layout

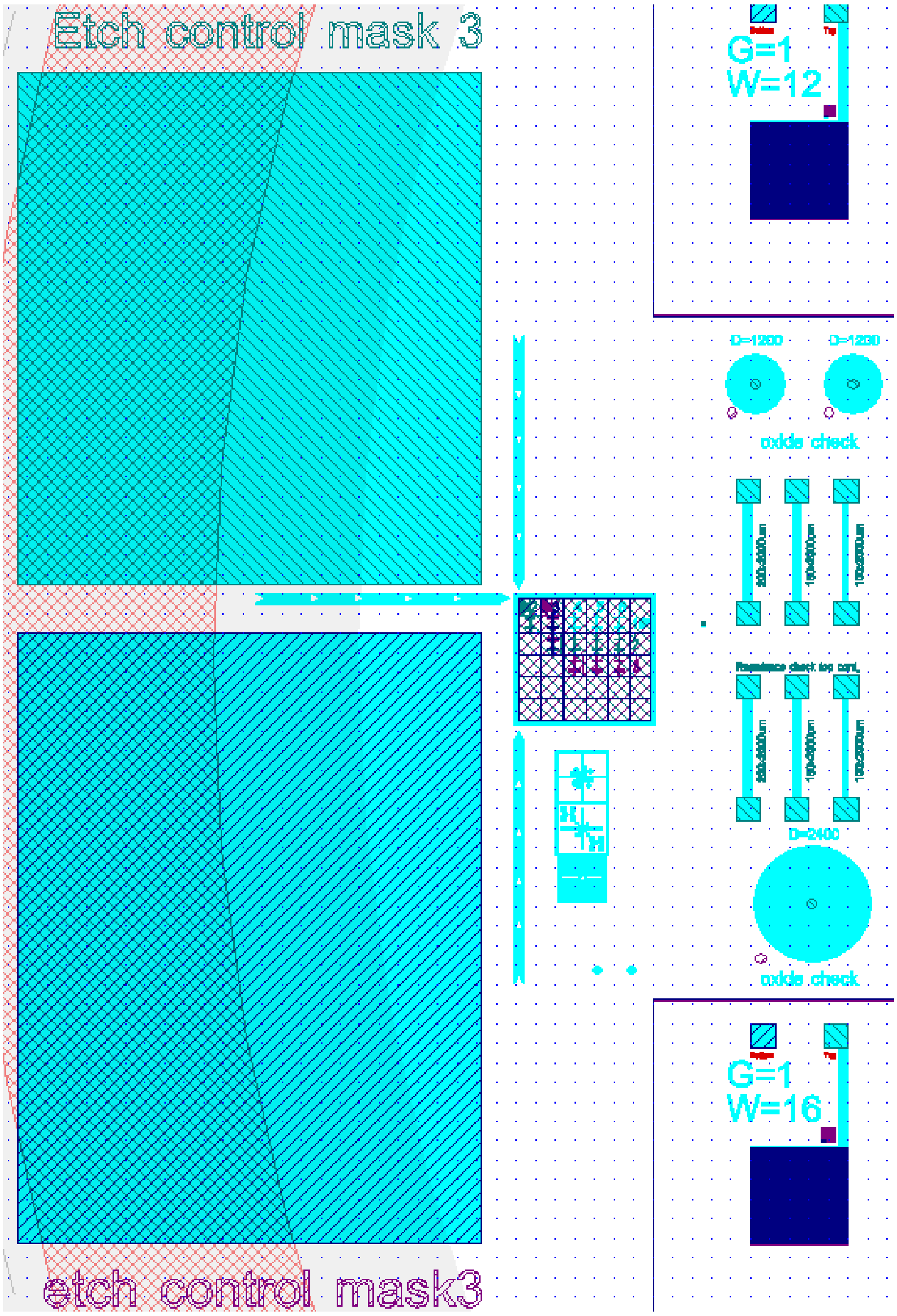


Figure B.3: *Test structure mask layout (blue=mask1, purple=mask2, green=mask3).*

B.2.4 Etch test structures mask layout

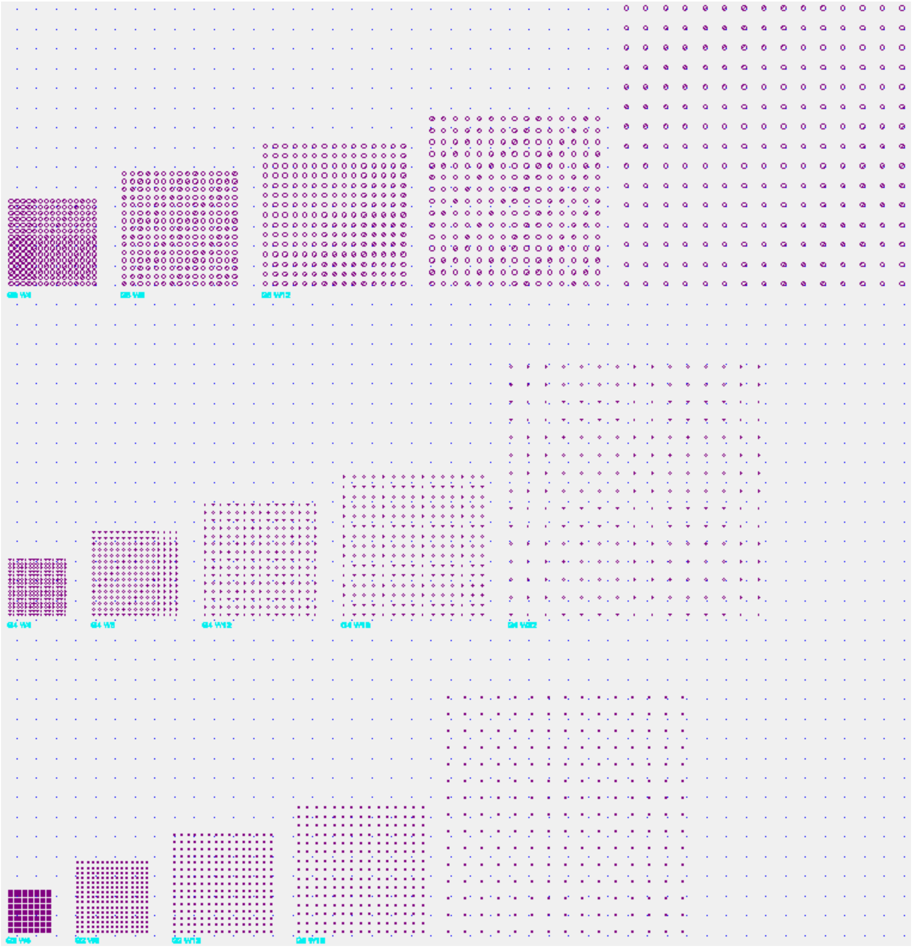
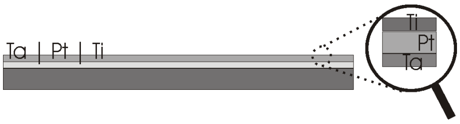
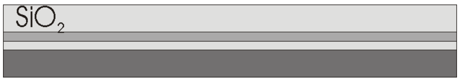
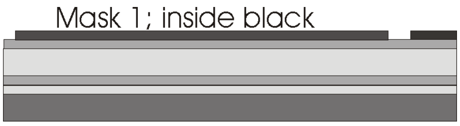
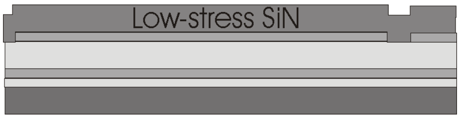
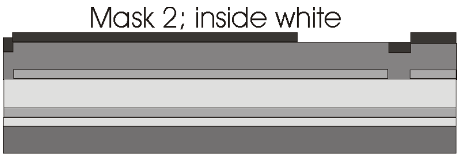


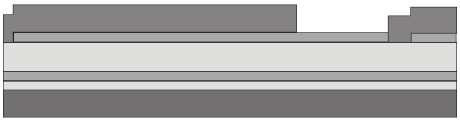
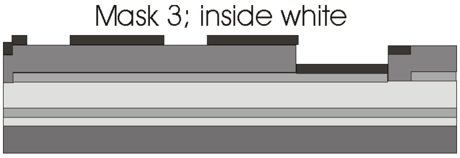
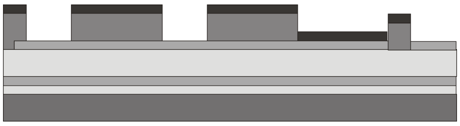
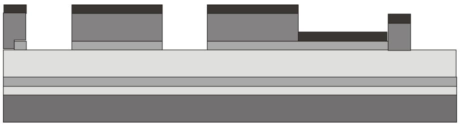
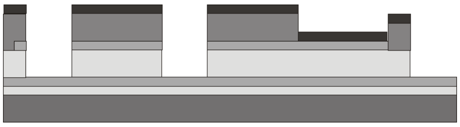
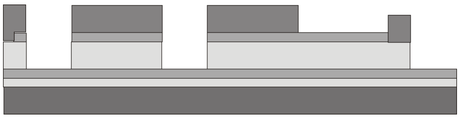
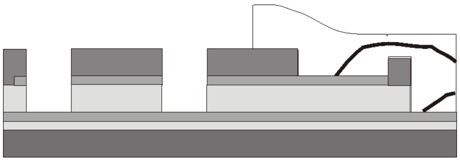
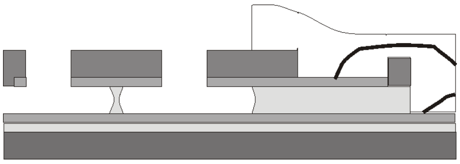
Figure B.4: *Test structure mask layout (blue=mask1, purple=mask2, green=mask3).*

B.3 Process outline sensor with SiN

These are the process steps involved with the manufacture of the parallel plate sensor.

Table B.1: *Process outline for the sensor with SiN.*

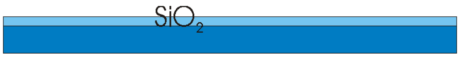
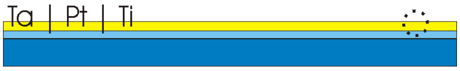
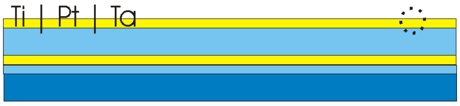
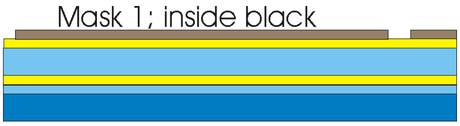
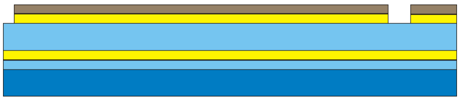
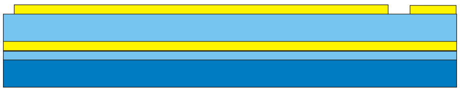
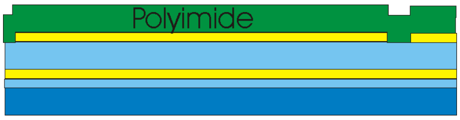
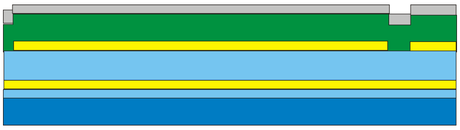
	description	**Layer view**
A	New wafer selection and cleaning. Standard p-type wafer; orientation <1-0-0>; 5-10Ωcm; 5 wafers, 3 with different sacrificial layer thickness and 2 without adhesion layer Ti.	
B	Thermal growth of SiO2 layer (1μm). 3hours at 1150^{o}C	
C	Sputtering of bottom electrode. First Ta (10nm) as adhesion layer (1:10 min); secondly Pt (100nm) as electrode material (5:20 min) and Ti (10nm) as 3th layer for adhesion (1:15min.).	
D	Growth of SiO2 layer by a PECVD process. Layer thickness of 200nm, 500nm, 1um with 3 different wafers. 200nm->3:12min; 500nm->8:00min; 1000nm->16:00min. A target of 500nm is 443nm result. P=800mTorr.	
E	Sputtering of top electrode. First Ti (10nm) as adhesion layer; secondly Pt (100nm) as electrode material and Ta (10nm) as 3th layer for adhesion.	
F	Photo lithography step to create the shape of the top electrode with mask 1. Resist-type=907/17.	
G	Etch of top electrode with ion-beam (IBE). Beam current 30mA, 500V, angle 20^o, +/- 30min.	
H	Remove photoresist with oxygenplasma. 3:30 min at 75W/10mT & 20 O2.	
I	Growth of low-stress SiN layer (500nm) by a PECVD process.	
J	Photo lithography step to clear the top contact with mask 2. Resist-type=907/17.	

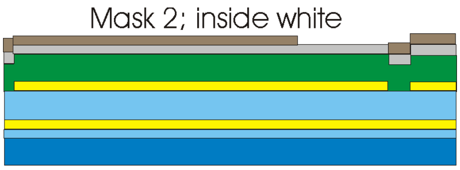
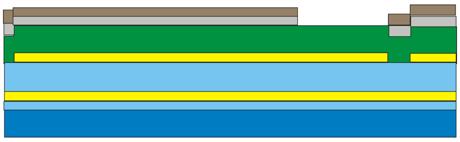
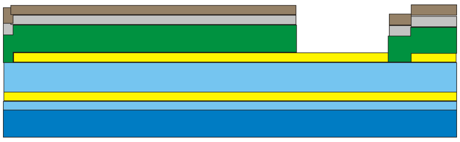
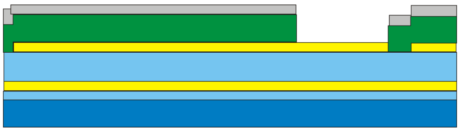
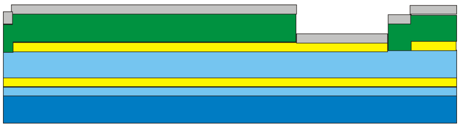
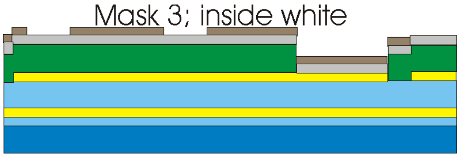
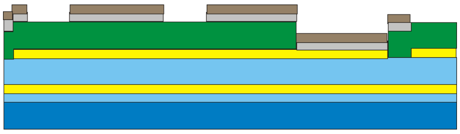
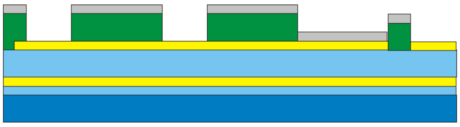
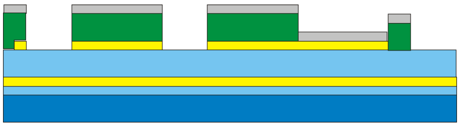
K	Etch of SiN with plasma. 5:30 min at 75W/10mTorr & 25/5 CHF3/O2	
L	Remove photoresist with oxygenplasma. 3:30 min at 75W/10mT & 20 O2.	
M	Photolithography step to etch holes and clear bottom electrode with mask 3. Resist-type=907/17 (min. 2um thick) spin at 2000RPM (recipe 2). 20s exposure at the mask aligner. Developing: 1min. beaker 1, 30 sec. beaker 2.	
N	Etch of SiN with RIE. Machine = PlasmaTherm 790 or Adixen.	
O	Remove Pt with ion-beam etching (IBE). Beam current 30mA, 500V, angle 20^o, +/- 30min.	
P	Remove SiO2 with reactive ion etching (RIE). Machine=PlasmaTherm 790 or Adixen.	
Q	Remove photoresist with oxygenplasma. 3:30 min at 75W/10mT & 20 O2.	
R	Sensor assembly, meaning dicing, wire bonding and covering it with epoxy.	
S	Etch the sacrificial SiO2 layer and remove top Ti adhesion layer with BHF in small steps. Test the ratio SiO2/water by a capacity measurement during each etch step.	

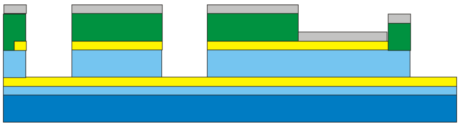
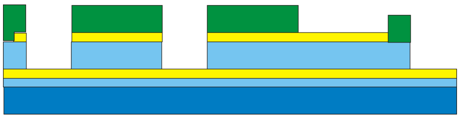
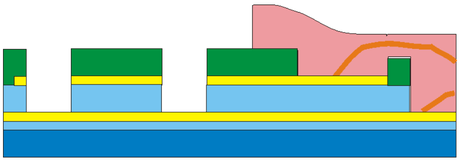
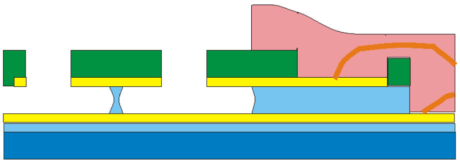
B.4 Process outline Polyimide

These are the steps involved with the manufacture of the polyimide P.P. sensor.

Table B.2: *Process outline for the sensor with Polyimide.*

	description	**Layer view**
A	New wafer selection and cleaning. Standard p-type wafer; orientation <1-0-0>; 5-10Ωcm; 1 wafer 500nm SiO2 without adhesion layer Ti, 1 with 1um SiO2.	
B	Thermal growth of SiO2 layer (1μm). 3hours at 1150^{o}C	
C	Sputtering of bottom electrode. First Ta (10nm) as adhesion layer (1:10 min); secondly Pt (100nm) as electrode material (5:20 min) and Ti (10nm) as 3th layer for adhesion (1:15min.).	
D	Growth of SiO2 layer by a PECVD process. Layer thickness of 200nm, 500nm, 1um with 3 different wafers. 200nm->3:12min; 500nm->8:00min; 1000nm->16:00min. A target of 500nm is 443nm result. P=800mTorr.	
E	Sputtering of top electrode. First Ti (10nm) as adhesion layer; secondly Pt (100nm) as electrode material and Ta (10nm) as 3th layer for adhesion.	
F	Photo lithography step to create the shape of the top electrode with mask 1. Resist-type=907/17.	
G	Etch of top electrode with ion-beam (IBE). Beam current 30mA, 500V, angle 20^o, +/- 30min.	
H	Remove photoresist with oxygenplasma. 3:30 min at 75W/10mT & 20sccm O2.	
I	Adding a polyimide layer by spinning at 6000 rpm for 1 min. The polyimide is mixed 3:1 with a solvent. Baking for a few min. at 110 C. Flood lighting for 10s. Baking at 300 C for 90 min. in the vacuumoven.	
J	200nm Al is sputtered on top of the polyimide to act as mask in the Oxford PL400. 800W for 2min.	

K	Photo lithography step to clear the top contact with mask 2. Resist-type=907/17. 5s lighting at the mask aligner.	
L	Remove the Al. layer to create a mask for the next step. About 1 min. in the Al-etch (wet). First do a pre-cleaning in UV-ozon to make the resist hydrophyllic, then make the wafer wet with DI and etch.	
M	Etch of the polyimide to clear the top contact with an RIE-process. Parameters: 20 / 2 sccm O2/CHF3[9] 3min. 100W, 10mTorr.	
N	Remove photoresist with oxygenplasma. 75W/10mT & 20sccm O2.	
O	Remove Al with wet etching.	
P	300nm Al is sputtered on top of the polyimide to act as mask in the Oxford PL400. 800W for 3min.	
Q	Photo lithography step to create the holes in the sensor and to clear the bottom contact. Resist-type=907/12. 5s lighting at the mask aligner. Development: 45s in beaker 1; 30s in beaker 2.	
R	Wet etch of the Al to create a mask for the next steps. First do a pre-cleaning in UV-ozon to make the resist hydrophyllic, then make the wafer wet with DI and etch.	
S	Etch of the polyimide to clear the top contact with an RIE-process. 20/2 O2/CHF3 3min. Photoresist is removed along the way. Power,pressure: 100W, 10mTorr.	
T	Remove Pt with ion-beam etching (IBE). Beam current 30mA, 500V, angle 20^o, +/- 30min.	

U	Do a short ultrasonic cleaning step to remove dirt inside the holes. Remove SiO2 with reactive ion etching (RIE). Machine=PlasmaTherm 790 or Adixen. Parameters: 10mTorr, 75W, 25sccm CHF3, 3sccm O2 for approximately 12m with 500nm SiO2.	
V	Wet etch of Al to remove the mask.	
W	Sensor assembly, meaning dicing, wire bonding and covering it with epoxy.	
X	Etch the sacrificial SiO2 layer and remove top Ti adhesion layer with BHF (17%) in small steps. Test the ratio SiO2/water by a visual check and perhaps a capacity measurement during each etch step.	

B.5 Specific design parameters

On each wafer a total of 112 sensors is placed. The sensors differ in the way the holes in the top electrode are aligned. This is illustrated in table B.3. All dimensions in um. G means gap diameter, W means edge to edge distance of the gaps.

Table B.3: *Sensor layout on the wafer.*

	G=2 W=4	G=2 W=8	G=2 W=12	G=2 W=16	G=2 W=32	G=2 W=4	G=2 W=8	G=2 W=12	G=2 W=16	G=2 W=32	
G=1 G=4	G=4 W=4	G=4 W=8	G=4 W=12	G=4 W=16	G=4 W=32	G=4 W=4	G=4 W=8	G=4 W=12	G=4 W=16	G=4 W=32	G=1 W=4
G=1 W=8	G=8 W=4	G=8 W=8	G=8 W=12	G=8 W=16	G=8 W=32	G=8 W=4	G=8 W=8	G=8 W=12	G=8 W=16	G=8 W=32	G=1 W=8
G=1 W=12	G=2 W=4	G=2 W=8	G=2 W=12	G=2 W=16	G=2 W=32	G=2 W=4	G=2 W=8	G=2 W=12	G=2 W=16	G=2 W=32	G=1 W=12
	G=4 W=4	G=4 W=8	G=4 W=12	G=4 W=16	Etch control	Etch control	G=4 W=8	G=4 W=12	G=4 W=16	G=4 W=32	
	G=8 W=4	G=8 W=8	G=8 W=12	G=8 W=16	G=8 W=32	G=8 W=4	G=8 W=8	G=8 W=12	G=8 W=16	G=8 W=32	
G=1 W=16	G=2 W=4	G=2 W=8	G=2 W=12	G=2 W=16	G=2 W=32	G=2 W=4	G=2 W=8	G=2 W=12	G=2 W=16	G=2 W=32	G=1 W=16
G=1 W=32	G=4 W=4	G=4 W=8	G=4 W=12	G=4 W=16	G=4 W=32	G=4 W=4	G=4 W=8	G=4 W=12	G=4 W=16	G=4 W=32	G=1 W=32
G=4 W=32	G=8 W=4	G=8 W=8	G=8 W=12	G=8 W=16	G=8 W=32	G=8 W=4	G=8 W=8	G=8 W=12	G=8 W=16	G=8 W=32	G=4 W=4
	G=1 W=4	G=1 W=8	G=1 W=12	G=1 W=16	G=1 W=32	G=1 W=4	G=1 W=8	G=1 W=12	G=1 W=16	G=1 W=32	

B.6 Polyimide and aluminumoxide as replacement for silicon nitride as insulating layer

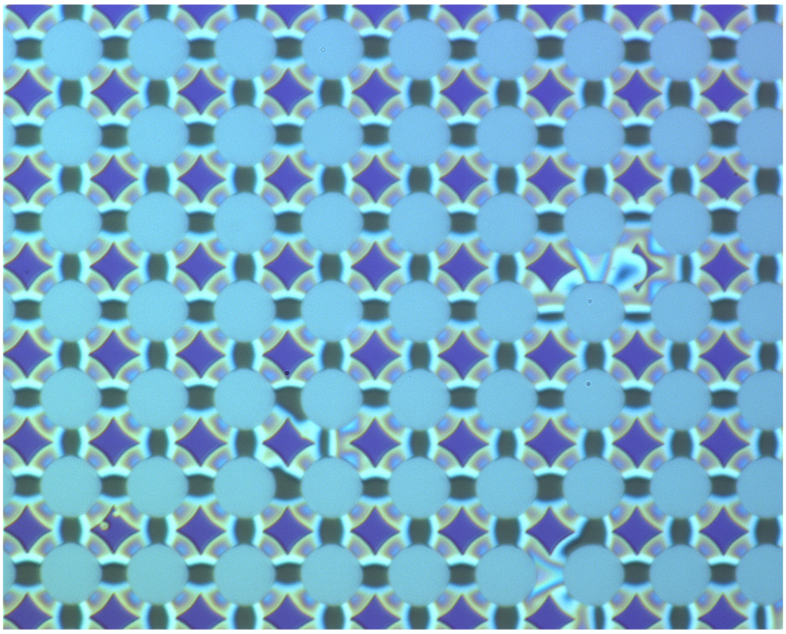


Figure B.5: *This is a picture of a test to see if aluminumoxide is a suitable replacement of silicon nitride. It is a sandwich of thermically grown SiO2 (200nm) and reactive sputtered aluminiumoxide (55nm) on top of that. The purple pillars are the remains of the SiO2 after etching for 30min. in BHF. The black areas indicate the collapse of the AlOx, touching the wafer.*

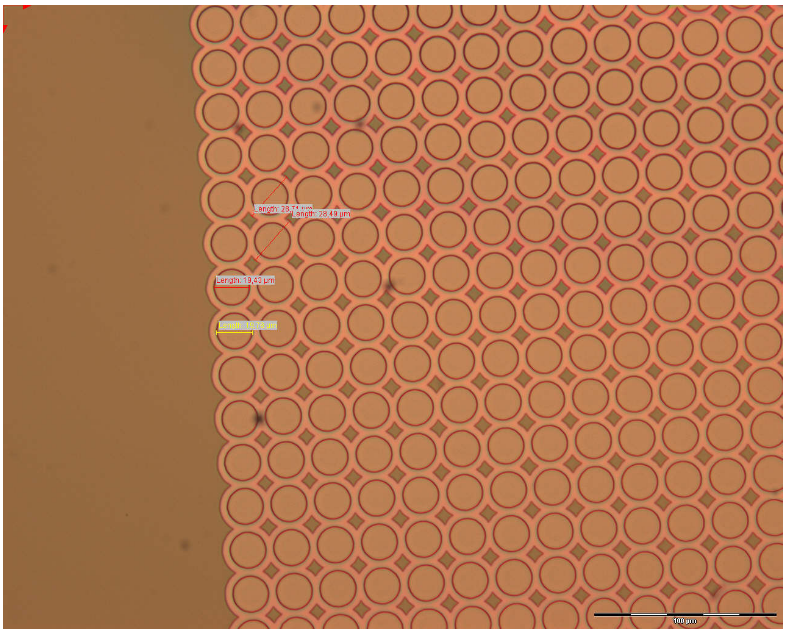


Figure B.6: *This is a picture of a test to see if polyimide is a suitable replacement of silicon nitride. It is a sandwich of thermically grown SiO2 (200nm) and a layer of polyimide (900nm) on top of that. The dark pillars are the remains of the SiO2 after etching for 45min. in BHF.*

C Measurements

C.1 Interference pattern in the 2um diameter hole structures

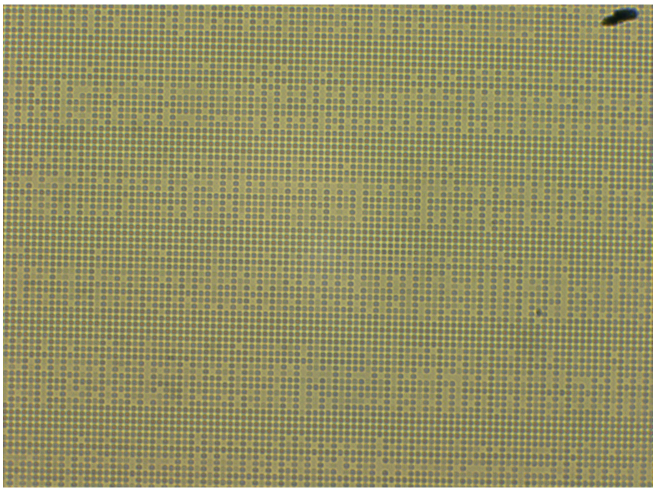


Figure C.1: *Interference pattern visible in the* $W_g = 2\mu m$ *grid at a magnification of 5x.*

C.2 Damage to the SiN sensors after dicing

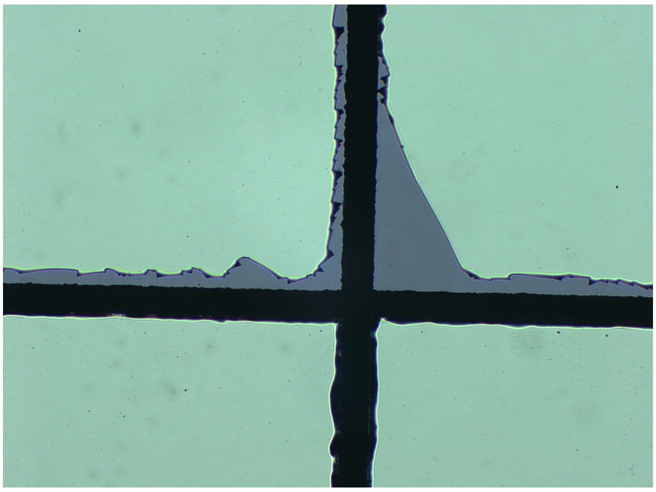


Figure C.2: *Damage to the corners of a SiN sensor after dicing. The black line is the cut, the blue colored areas are probably from the thermically grown SiO2. The magnification is 5x.*

C.3 Detached polyimide around the holes

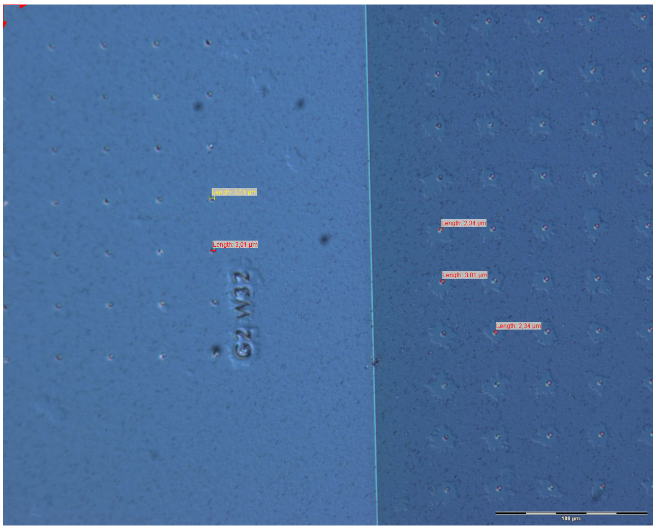


Figure C.3: *Photo which shows some detachment of the polyimide around the holes before etching. Parameters of the sensors are:* $W_g = 2\mu m$, $W_{ee} = 32\mu m$. *This photo is recorded with a 'differential interference contrast'-technique at a magnification of 20x.*

C.4 Photos from polyimide sensors after sacrificial layer etching

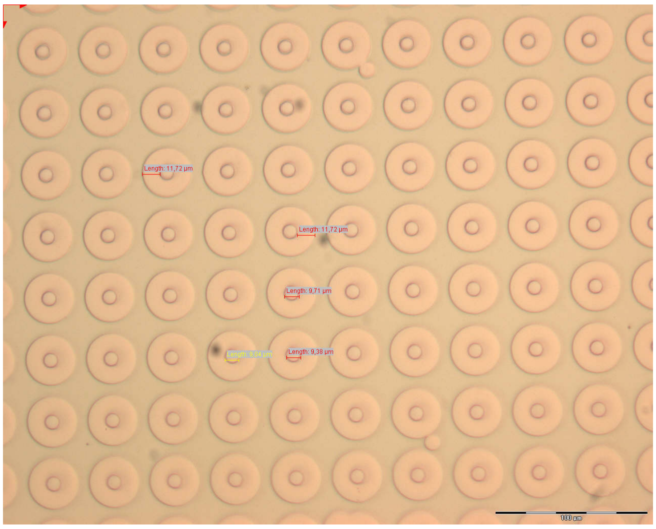


Figure C.4: *Photo from the test structure after 3600s sacrificial layer etching and extensive electrochemical measurements at a magnification of 20x. Parameters of the sensors are:* $W_g = 8\mu m$, $W_{ee} = 32\mu m$.

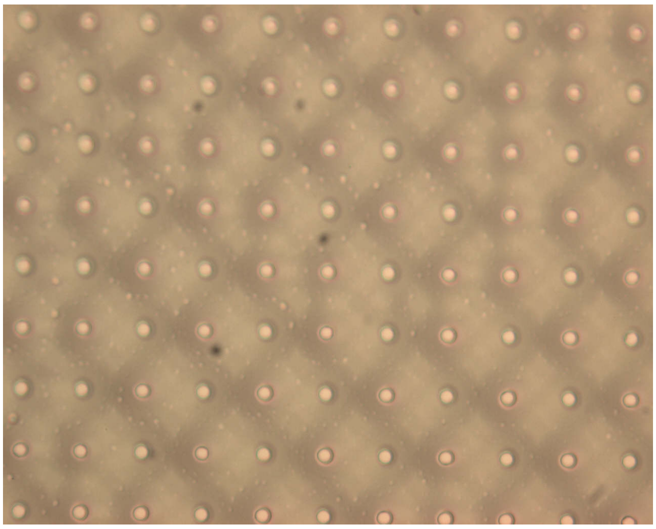


Figure C.5: *Photo from the sensor area after 3600s sacrificial layer etching at a magnification of 20x. Parameters of the sensors are:* $W_g = 8\mu m$, $W_{ee} = 32\mu m$.

C.5 Diode-like measurements on PI-sensors

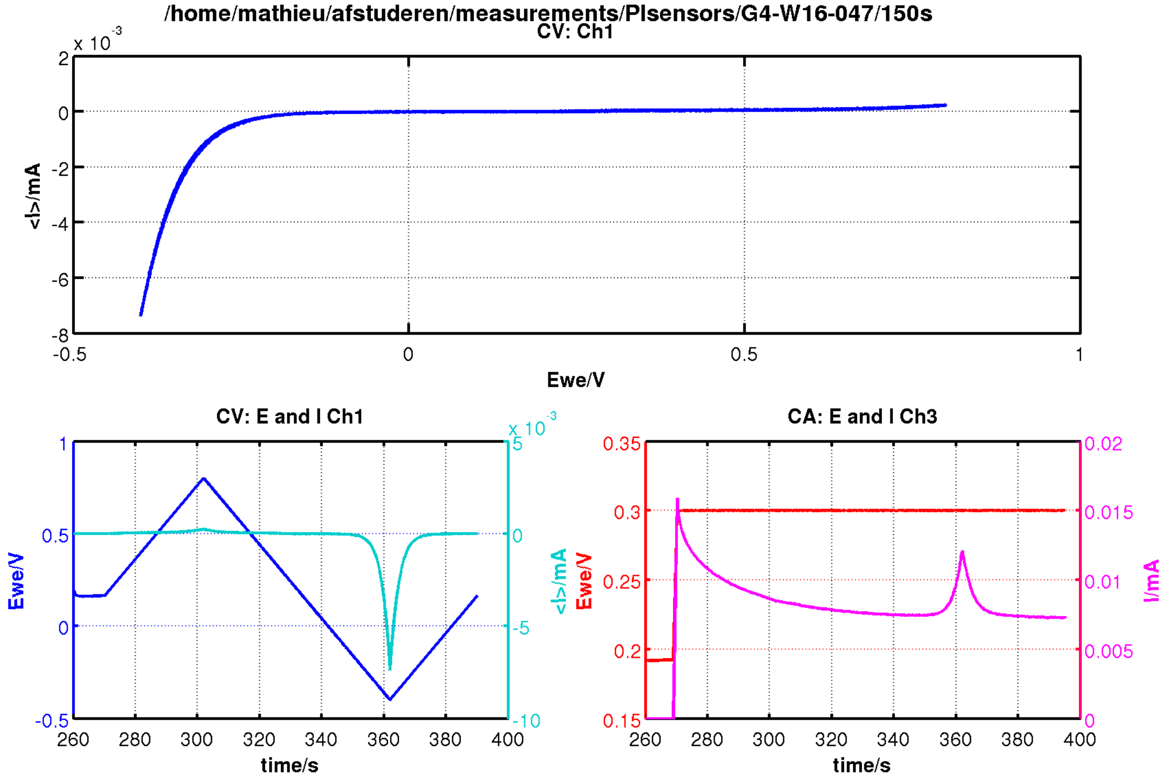


Figure C.6: *Measurement on a sensor with parameters:* $W_g = 4\mu m$*,* $W_{ee} = 16\mu m$ *after 150s of etching. The measurement shows a CV on the bottom electrode (ch1), while the top electrode (ch3) is kept at a potential of +0.3V.*

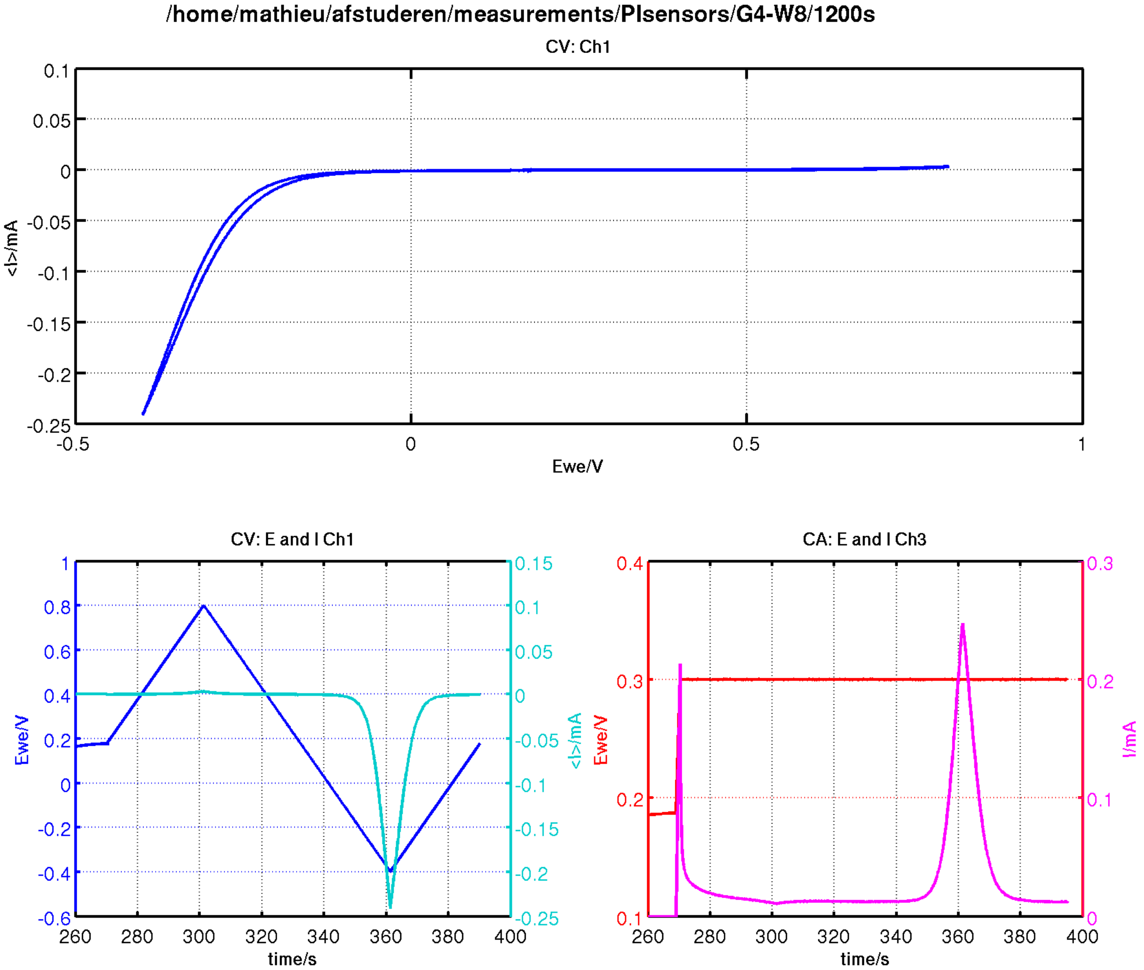


Figure C.7: *Measurement on a sensor with parameters: $W_g = 4\mu m$, $W_{ee} = 8\mu m$ after 1200s of etching. The measurement shows a CV on the bottom electrode (ch1), while the top electrode (ch3) is kept at a potential of $+0.3V$.*

Redox cycling amplification measurements on PI-sensors RC-values indicated in the figure are guessed values based on a programming algorithm. More precise values are indicated in the measurement section.

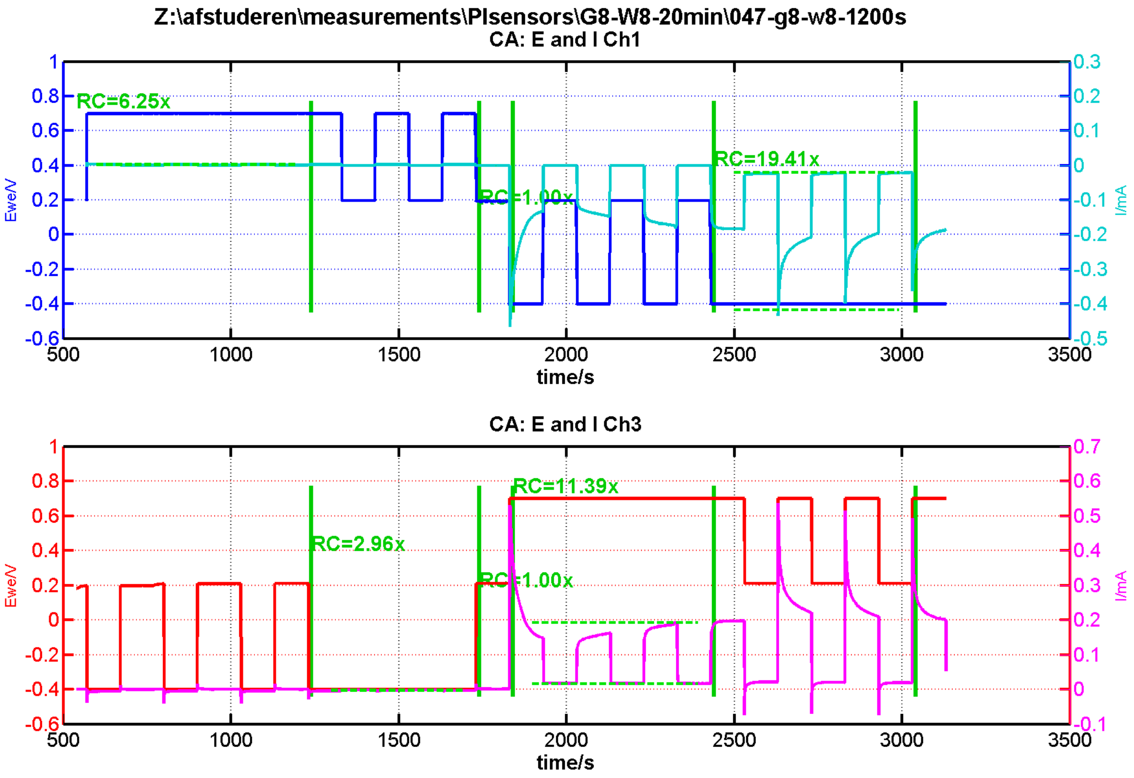


Figure C.8: *RC-measurement on a sensor with parameters:* $W_g = 8\mu m$, $W_{ee} = 8\mu m$ *after 1200s of etching.*

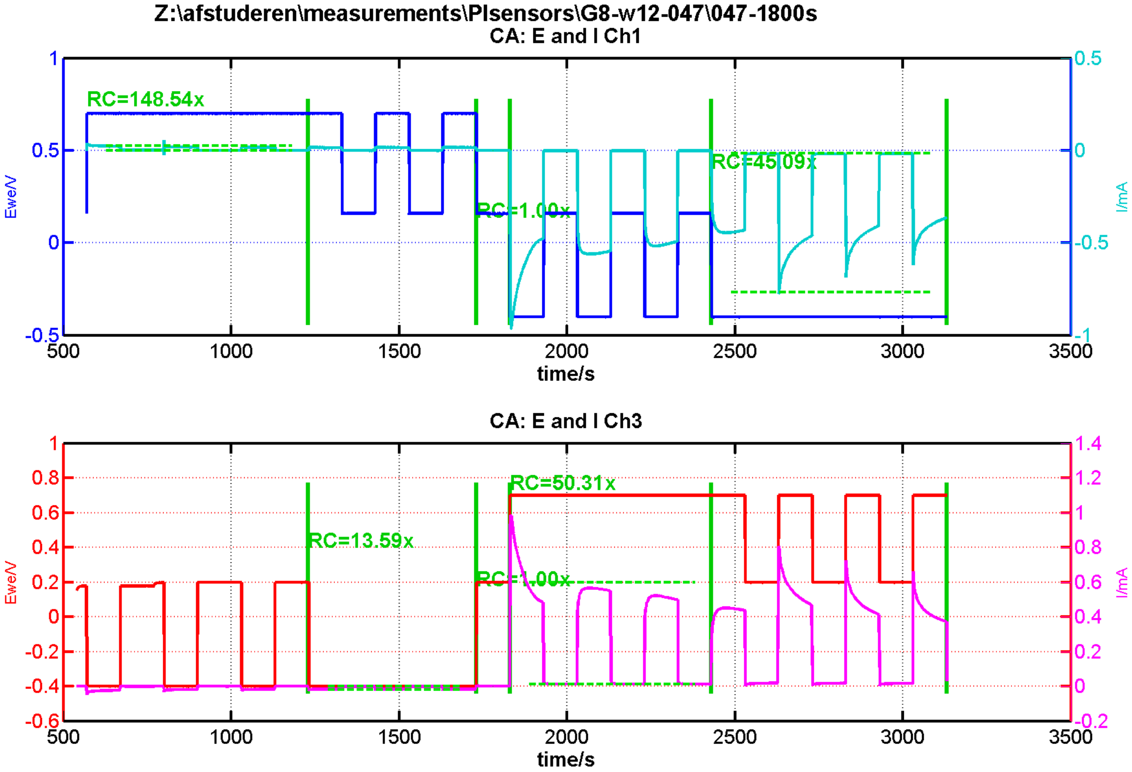


Figure C.9: *RC-measurement on a sensor with parameters:* $W_g = 8\mu m$, $W_{ee} = 12\mu m$ *after 1800s of etching.*

D Scripts

D.1 IDA with varied mesh script

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%First set some basic parameters and empty variables
clear;
scrsz = get(0,'ScreenSize');
F=96485; %faraday constant
ymax=5E-5;
w=2;
w=w*1E-6;

mesh=[1E-7, 5E-8 2.5E-8 1E-8 5E-9 3E-9 1E-9];
domain_mesh=0.5E-6;

mesh_res=zeros(1,size(mesh,2));
time=[0:0.01:0.1];
t=size(time,2); %corresponds to 0.1 s
flux_coll=zeros(size(time,2),size(mesh,2));
flux_gen=zeros(size(time,2),size(mesh,2));
coll_eff=zeros(size(time,2),size(mesh,2));
redox_cycles=zeros(size(time,2),size(mesh,2));

%clear fem structure
flclear fem

% COMSOL version
clear vrsn
vrsn.name = 'COMSOL␣3.2';
vrsn.ext = '';
vrsn.major = 0;
vrsn.build = 222;
vrsn.rcs = 'Name:';
vrsn.date = 'Date:2005/09/0118:02:30';
fem.version = vrsn;

% Constants inside model
fem.const = {'c0','5', ...
   'D','7E-10', ...
   'F','96485.34', ...
   'R','8.314472', ...
   'T','290', ...
   'Estd','0.265', ...
   'alpha','0.5', ...
   'Ec','0.7', ...
   'Eg','-0.2', ...
   'ks','1.5E-4'};

%iterate through different mesh setups
for i=1:size(mesh,2)
    display(i) %give some feedback to user
    % set Geometry
    wg=w;
    we=w/2;
    clear draw
    clear g4
    clear g5
    clear g1
    g1=rect2(2*wg,ymax,'Pos',[-wg 0],'Base','corner');
    draw.s.objs = {g1};
    draw.s.name = {'R1'};
    draw.s.tags = {'g1'};
    g4=point2(-we,0);
    g5=point2(we,0);
    draw.p.objs = {g4,g5};
    draw.p.name = {'PT1','PT2'};
    draw.p.tags = {'g4','g5'};
    fem.draw = draw;
    fem.geom = geomcsg(fem);

    % Initialize mesh; use mesh setup from top of file
    fem.mesh=meshinit(fem, ...
                        'hmaxedg',[2,mesh(i),5,mesh(i)], ...
                        'hmaxsub',[1,domain_mesh]);
    mesh_res(i)=size(fem.mesh.t,2);
    % (Default values are not included)

    % Application mode 1
    clear appl
    appl.mode.class = 'Diffusion';
    appl.dim = {'Cox','Cre'};
    appl.module = 'CHEM';
    appl.gporder = 4;
    appl.cporder = 2;
    appl.assignsuffix = '_chdi';
    clear prop
    prop.analysis='static';
    appl.prop = prop;
    clear bnd
    bnd.type = {{'N0';'N0'},{'N';'N'},{'C';'C'},{'N';'N'}};
    bnd.c0 = {0,0,'c0',0};
    bnd.N = {0,{'-kfg*Cox+kbg*Cre';'kfg*Cox-kbg*Cre'},0,{'kbc*Cre-kfc*Cox';'-kbc*Cre+kfc*Cox'}}; %sorted in groups
    bnd.ind = [1,2,3,1,4,1];
```

```
    appl.bnd = bnd;
    clear equ
    equ.D = {{'D';'D'}};
    equ.init = {{'c0';0}};
    equ.ind = [1];
    appl.equ = equ;
    fem.appl{1} = appl;
    fem.border = 1;
    fem.outform = 'general';
    fem.units = 'SI';

    % Subdomain expressions
    clear equ
    equ.ind = [1];
    equ.dim = {'Cox','Cre'};
    equ.expr = {'kfg','ks*exp(-alpha*F*(Eg-Estd)/(R*T))', ...
      'kbc','ks*exp((1-alpha)*F*(Ec-Estd)/(R*T))',...
      'kfc','ks*exp(-alpha*F*(Ec-Estd)/(R*T))',...
      'kbg','ks*exp((1-alpha)*F*(Eg-Estd)/(R*T))'};
    fem.equ = equ;

    % Descriptions
    clear descr
    descr.const= {'D','Diffusion␣constant','R','Gas␣constant␣J/K*mol',...
        'F','Faradays␣constant␣C/mol','c0','Bulk␣concentration␣Oxidized␣species',...
        'T','Temp.␣K','Estd','Standard␣potenial␣from␣Yang'};
    fem.descr = descr;

    % Multiphysics
    fem=µltiphysics(fem);

    % Extend mesh
    fem.xmesh=meshextend(fem);

    % Solve problem
    fem.sol=femtime(fem, ...
                    'solcomp',{'Cox','Cre'}, ...
                    'outcomp',{'Cox','Cre'}, ...
                    'tlist',time, ...
                    'tout','tlist');

    % Save current fem structure for restart purposes
    fem0=fem;

    %calculate flux at the electrode boundaries for all timesteps
    for j=1:size(time,2)
        % Integrate
        flux_coll(j,i)=F*postint(fem,'ndflux_Cox_chdi', ...
                   'dl',[5], ...
                   'edim',1, ...
                   'solnum',j);

        % Integrate
        flux_gen(j,i)=F*postint(fem,'ndflux_Cox_chdi', ...
                   'dl',[2], ...
                   'edim',1, ...
                   'solnum',j);
        coll_eff(j,i)=-flux_gen(j,i)/flux_coll(j,i)*100;
        redox_cycles(j,i)=1/(1-(coll_eff(j,i)/100)^2);
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% POST PROCESSING
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% PLOT OF ELECTRODE FLUX VS. mesh
H1=figure(1);
set(H1 ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
plot(mesh_res,-flux_coll(t,:),'Marker','v','color','b','LineWidth',2,'MarkerSize',8);
hold on;
plot(mesh_res,flux_gen(t,:),'Marker','x','color','g','LineWidth',2,'MarkerSize',8);
hold off;
grid on;
xlabel('Total␣number␣of␣mesh␣elements','FontWeight','bold','FontSize',14);
ylabel('current␣(A/m)','FontWeight','bold','FontSize',14);
set(gca,'linewidth',2,'fontsize',14);
loc='Z:\afstuderen\πcs\sim-ida-varied-mesh\electrode-flux-vs-mesh.jpg';
%loc='/home/mathieu/afstuderen/πcs/sim-ida-varied-mesh/electrode-flux-vs-mesh.jpg';
saveas(H1,loc,'jpg');

%% save data into latex table
clear columnLabels;
columnLabels= {'Max.␣element␣size␣(m)','Mesh␣elements','Current␣col.␣[x]␣(A/m)',...
    'Current␣gen.␣[∇] (A/m)'};
clear matrix;
matrix(:,1)=mesh;
matrix(:,2)=mesh_res;
matrix(:,3)=-flux_coll(t,:);
matrix(:,4)=flux_gen(t,:);

location='Z:/afstuderen/eindverslag/data/mesh-study.tex';
%location='/home/mathieu/afstuderen/eindverslag/data/mesh-study.tex';
matrix2latex(matrix, location,...
    'columnLabels',columnLabels, 'alignment', 'c', 'format', '%10.2e',...
    'size', 'tiny','caption',...
    'Current␣at␣the␣generator/collector␣electrode␣as␣a␣function␣of␣the␣number␣of␣mesh␣elements.',...
    'label','tbl:flux-at-gen-vs-mesh');
```

D.2 IDA with varied bulk height

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%First set some basic parameters and empty variables
clear;
F=96485; %faraday constant
scrsz = get(0,'ScreenSize');
y_max=[1E-5 2E-5 3E-5 4E-5 5E-5];
w=2;
w=w*1E-6;
time=[0:0.005:0.2];
t=size(time,2); %corresponds to 0.1 s
flux_inf=zeros(size(time,2),size(y_max,2));
flux_gen=zeros(size(time,2),size(y_max,2));
flclear fem

% COMSOL version
clear vrsn
vrsn.name = 'COMSOL␣3.2';
vrsn.ext = '';
vrsn.major = 0;
vrsn.build = 222;
vrsn.rcs = 'Name:';
vrsn.date = 'Date: 2005/09/0118 : 02 : 30';
fem.version = vrsn;

% model Constants
fem.const = {'c0','5', ...
  'D','7E-10', ...
  'F','96485.34', ...
  'R','8.314472', ...
  'T','290', ...
  'Estd','0.265', ...
  'alpha','0.5', ...
  'Ec','0.7', ...
  'Eg','-0.2', ...
  'ks','1.5E-4'};

%iterate through different bulk heights
for i=1:size(y_max,2)
    display(i)
    % build Geometry
    wg=w;
    we=w/2;
    ymax=y_max(i);
    clear draw
    clear g4
    clear g5
    clear g1
    g1=rect2(2*wg,ymax,'Pos',[-wg 0],'Base','corner');
    draw.s.objs = {g1};
    draw.s.name = {'R1'};
    draw.s.tags = {'g1'};
    g4=point2(-we,0);
    g5=point2(we,0);
    draw.p.objs = {g4,g5};
    draw.p.name = {'PT1','PT2'};
    draw.p.tags = {'g4','g5'};
    fem.draw = draw;
    fem.geom = geomcsg(fem);

    % Initialize mesh
    mesh=5E-9;
    fem.mesh=meshinit(fem, ...
                        'hmaxedg',[2,mesh,5,mesh], ...
                        'hmaxsub',[1,0.5e-6]);

    % (Default values are not included)

    % Application mode 1
    clear appl
    appl.mode.class = 'Diffusion';
    appl.dim = {'Cox','Cre'};
    appl.module = 'CHEM';
    appl.gporder = 4;
    appl.cporder = 2;
    appl.assignsuffix = '_chdi';
    clear prop
    prop.analysis='static';
    appl.prop = prop;
    clear bnd
    bnd.type = {{'N0';'N0'},{'N';'N'},{'C';'C'},{'N';'N'}};
    bnd.c0 = {0,0,'c0',0};
    bnd.N = {0,{'-kfg*Cox+kbg*Cre';'kfg*Cox-kbg*Cre'},0,{'kbc*Cre-kfc*Cox';'-kbc*Cre+kfc*Cox'}}; %sorted in groups
    bnd.ind = [1,2,3,1,4,1];
    appl.bnd = bnd;
    clear equ
    equ.D = {{'D';'D'}};
    equ.init = {{'c0';0}};
    equ.ind = [1];
    appl.equ = equ;
    fem.appl{1} = appl;
    fem.border = 1;
    fem.outform = 'general';
    fem.units = 'SI';

    % Subdomain expressions
```

```
    clear equ
    equ.ind = [1];
    equ.dim = {'Cox','Cre'};
    equ.expr = {'kfg','ks*exp(-alpha*F*(Eg-Estd)/(R*T))', ...
       'kbc','ks*exp((1-alpha)*F*(Ec-Estd)/(R*T))',...
       'kfc','ks*exp(-alpha*F*(Ec-Estd)/(R*T))',...
       'kbg','ks*exp((1-alpha)*F*(Eg-Estd)/(R*T))'};
    fem.equ = equ;

    % Descriptions
    clear descr
    descr.const= {'D','Diffusion␣constant','R','Gas␣constant␣J/K*mol',...
        'F','Faradays␣constant␣C/mol','c0','Bulk␣concentration␣Oxidized␣species',...
        'T','Temp.␣K','Estd','Standard␣potenial␣from␣Yang'};
    fem.descr = descr;

    % Multiphysics
    fem=μltiphysics(fem);

    % Extend mesh
    fem.xmesh=meshextend(fem);

    % Solve problem
    fem.sol=femtime(fem, ...
                    'solcomp',{'Cox','Cre'}, ...
                    'outcomp',{'Cox','Cre'}, ...
                    'tlist',time, ...
                    'tout','tlist');

    % Save current fem structure for restart purposes
    fem0=fem;
%
%     % Plot solution and save it to disk
%     H=figure(i);
%     set(H ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
%     postplot(fem, ...
%              'tridata',{'Cox','cont','internal'}, ...
%              'trimap','jet(1024)', ...
%              'contdata',{'Cox','cont','internal'}, ...
%              'contlevels',20, ...
%              'contmap','cool(1024)', ...
%              'arrowdata',{'Coxx','Coxy'}, ...
%              'arrowxspacing',15, ...
%              'arrowyspacing',15, ...
%              'arrowtype','arrow', ...
%              'arrowstyle','proportional', ...
%              'arrowcolor',[1.0,0.0,0.0], ...
%              'solnum',t, ...
%              'title',['Time=',num2str(time(t)),'; Surface+contour: Conc., Cox [mol/m^3]; Arrow: Conc. grad., Cox [mol/m^4]'], ...
%              'refine',3, ...
%              'axis',[-5E-6,5E-6,-1E-7,ymax,-1,1]);
%     xlabel('x (m)');
%     ylabel('y (m)');
%     loc=['Z:\afstuderen\πcs\sim-ida-varied-ymax\ymax-',num2str(ymax),'.jpg'];
%     loc=['/home/mathieu/odie/afstuderen/πcs/sim-ida-varied-ymax/ymax-',num2str(ymax),'.jpg'];
%     saveas(H,loc,'jpg');

    %integrate normal diffusive flux over electrode boundaries
    for j=1:size(time,2)
        % Integrate
        flux_inf(j,i)=postint(fem,'ndflux_Cox_chdi', ...
                    'dl',[3], ...
                    'edim',1, ...
                    'solnum',j);
        flux_gen(j,i)=postint(fem,'ndflux_Cox_chdi', ...
                    'dl',[2], ...
                    'edim',1, ...
                    'solnum',j);
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% POST PROCESSING
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% PLOT OF FLUX at infinite boundary VS. ymax
H1=figure(size(y_max,2)+1);
set(H1 ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
color(1,:)=[1 0 1];
color(2,:)=[0 0 1];
color(3,:)=[0 1 0];
color(4,:)=[1 0 0];
color(5,:)=[0 0 0];
marker(1)='+';
marker(2)='o';
marker(3)='s';
marker(4)='x';
marker(5)='v';
average_flux=mean(mean(flux_gen(2,end,:)))
average_flux=2E-9
for i=1:size(y_max,2)
    plot(time,-flux_inf(:,i)/average_flux*100,'Color',color(i,:),'Marker',marker(i),...
        'LineWidth',2,'MarkerSize',8);
    hold on;
end
hold off;
grid on;
xlabel('Time␣(s)','FontWeight','bold','FontSize',14);
ylabel('Flux␣of␣C_{ox}␣at␣semi-inf.␣boundary␣/␣avg.␣flux␣of␣C_{ox}␣at␣the␣gen.␣(%)','FontWeight','bold','FontSize',14);
set(gca,'linewidth',2,'fontsize',14);
loc2=['Z:\afstuderen\πcs\sim-ida-varied-ymax\flux-at-infinite-boundary-vs-ymax.jpg'];
%loc2=['/home/mathieu/odie/afstuderen/πcs/sim-ida-varied-ymax/flux-at-infinite-boundary-vs-ymax.jpg']
saveas(H1,loc2,'jpg');
```

```
%% save data to latex table
clear rowLabels;
clear columnLabels;
for j=1:size(time,2)
    rowLabels(j) = {['t=',num2str(time(j)),'␣s']};
end
for i=1:size(y_max,2)
    columnLabels(i) = {['$y_{bulk}$=',num2str(1E6*y_max(i)),'um␣[',marker(i),']']};
end
loc3='Z:/afstuderen/eindverslag/data/flux-inf.tex';
%loc3='/home/mathieu/odie/afstuderen/eindverslag/data/flux-inf.tex'
matrix2latex(-flux_inf, loc3,...
    'rowLabels', rowLabels, 'columnLabels',columnLabels, 'alignment', 'c',...
    'format', '%10.2e', 'size', 'tiny','caption',...
    'Current␣at␣the␣semi␣infinite␣boundary␣as␣a␣function␣of␣time␣and␣bulk␣height',...
    'label','tbl:flux-at-infty');
```

D.3 IDA with varied gap width and electrode width

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%% First set some basic parameters and empty variables
clear;
F=96485;
scrsz=get(0,'ScreenSize');

ymax=5E-5;
w=[0.2 0.5 1 1.5 2];
%w=2;
w=w*1E-6;
ratio=[0.25 0.5 1 2];
%ratio=1;
time=[0:0.001:0.004];
time=cat(2,time,[0.005:0.005:0.2]);
t=find(time==0.1); %corresponds to 0.2s
flux_coll=zeros(size(time,2),size(w,2),size(ratio,2));
flux_gen=zeros(size(time,2),size(w,2),size(ratio,2));
coll_eff=zeros(size(time,2),size(w,2),size(ratio,2));
redox_cycles=zeros(size(time,2),size(w,2),size(ratio,2));

%clear fem structure
flclear fem

%% COMSOL version
clear vrsn
vrsn.name = 'COMSOL␣3.2';
vrsn.ext = '';
vrsn.major = 0;
vrsn.build = 222;
vrsn.rcs = 'Name:';
vrsn.date = 'Date:2005/09/0118:02:30';
fem.version = vrsn;

% set model Constants
fem.const = {'c0','5', ...
   'D','7E-10', ...
   'F','96485.34', ...
   'R','8.314472', ...
   'T','290', ...
   'Estd','0.265', ...
   'alpha','0.5', ...
   'Ec','0.7', ...
   'Eg','-0.2', ...
   'ks','1.5E-4'};

%iterate through different gap and electrode widths
for i=1:size(w,2)
    for k=1:size(ratio,2)
        display(i) %give some feedback to user
        % set Geometry
        wg=w(i);
        we=w(i)/2*ratio(k);
        clear draw
        clear g4
        clear g5
        clear g1
        g1=rect2(wg+2*we,ymax,'Pos',[-wg/2-we 0],'Base','corner');
        draw.s.objs = {g1};
        draw.s.name = {'R1'};
        draw.s.tags = {'g1'};
        g4=point2(-wg/2,0);
        g5=point2(wg/2,0);
        draw.p.objs = {g4,g5};
        draw.p.name = {'PT1','PT2'};
        draw.p.tags = {'g4','g5'};
        fem.draw = draw;
        fem.geom = geomcsg(fem);

        % Initialize mesh, use variable mesh size at electrodes to ensure
        % more than 10000 elements in the different models for accuracy.
        mesh=1E-9;
        fem.mesh=meshinit(fem, ...
                          'hmaxedg',[2,mesh,5,mesh], ...
                          'hmaxsub',[1,0.5e-6]);

        % (Default values are not included)

        % Application mode 1
        clear appl
        appl.mode.class = 'Diffusion';
        appl.dim = {'Cox','Cre'};
        appl.module = 'CHEM';
        appl.gporder = 4;
        appl.cporder = 2;
        appl.assignsuffix = '_chdi';
        clear prop
        prop.analysis='static';
        appl.prop = prop;
        clear bnd
        bnd.type = {{'N0';'N0'},{'N';'N'},{'C';'C'},{'N';'N'}};
        bnd.c0 = {0,0,'c0',0};
        bnd.N = {0,{'-kfg*Cox+kbg*Cre';'kfg*Cox-kbg*Cre'},0,{'kbc*Cre-kfc*Cox';'-kbc*Cre+kfc*Cox'}}; %sorted in groups
        bnd.ind = [1,2,3,1,4,1];
        appl.bnd = bnd;
        clear equ
```

```
        equ.D = {{'D';'D'}};
        equ.init = {{'c0';0}};
        equ.ind = [1];
        appl.equ = equ;
        fem.appl{1} = appl;
        fem.border = 1;
        fem.outform = 'general';
        fem.units = 'SI';

        % Subdomain expressions
        clear equ
        equ.ind = [1];
        equ.dim = {'Cox','Cre'};
        equ.expr = {'kfg','ks*exp(-alpha*F*(Eg-Estd)/(R*T))', ...
          'kbc','ks*exp((1-alpha)*F*(Ec-Estd)/(R*T))',...
          'kfc','ks*exp(-alpha*F*(Ec-Estd)/(R*T))',...
          'kbg','ks*exp((1-alpha)*F*(Eg-Estd)/(R*T))'};
        fem.equ = equ;

        % Descriptions
        clear descr
        descr.const= {'D','Diffusion␣constant','R','Gas␣constant␣J/K*mol',...
            'F','Faradays␣constant␣C/mol','c0','Bulk␣concentration␣Oxidized␣species',...
            'T','Temp.␣K','Estd','Standard␣potenial␣from␣Yang'};
        fem.descr = descr;

        % Multiphysics
        fem=μltiphysics(fem);

        % Extend mesh
        fem.xmesh=meshextend(fem);

        % Solve problem
        fem.sol=femtime(fem, ...
                        'solcomp',{'Cox','Cre'}, ...
                        'outcomp',{'Cox','Cre'}, ...
                        'tlist',time, ...
                        'tout','tlist');

        % Save current fem structure for restart purposes
        fem0=fem;

        % Plot solution and save it to disk
%         H=figure(i);
%         set(H ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
%         postplot(fem, ...
%                 'tridata',{'Cox','cont','internal'}, ...
%                 'trimap','jet(1024)', ...
%                 'contdata',{'Cox','cont','internal'}, ...
%                 'contlevels',20, ...
%                 'contmap','cool(1024)', ...
%                 'arrowdata',{'Coxx','Coxy'}, ...
%                 'arrowxspacing',15, ...
%                 'arrowyspacing',15, ...
%                 'arrowtype','arrow', ...
%                 'arrowstyle','proportional', ...
%                 'arrowcolor',[1.0,0.0,0.0], ...
%                 'solnum',t, ...
%                 'title',['Time=',num2str(time(t)),'; Surface+contour: Conc., Cox [mol/m^3]; Arrow: Conc. grad., Cox [mol/m^4]'], ...
%                 'refine',3, ...
%                 'axis',[-5E-6,5E-6,-1E-7,1E-5,-1,1]);
%         xlabel('x (m)');
%         ylabel('y (m)');
%        %loc1=['Z:\afstuderen\πcs\sim-ida-varied-wg\wg-',num2str(w(i)),'.jpg'];
%         loc1=['/home/mathieu/odie/afstuderen/πcs/sim-ida-varied-wg/wg-',num2str(w(i)),'ratio -',num2str(ratio(k)),'.jpg'];
%         saveas(H,loc1,'jpg');

        %integrate flux at electrode boundaries for each timestep
        for j=1:size(time,2)
            % Integrate
            flux_coll(j,i,k)=F*postint(fem,'ndflux_Cox_chdi', ...
                          'dl',[5], ...
                          'edim',1, ...
                          'solnum',j);

            % Integrate
            flux_gen(j,i,k)=F*postint(fem,'ndflux_Cox_chdi', ...
                          'dl',[2], ...
                          'edim',1, ...
                          'solnum',j);
            coll_eff(j,i,k)=-flux_coll(j,i,k)/flux_gen(j,i,k)*100;
            redox_cycles(j,i,k)=1/(1-(coll_eff(j,i,k)/100)^2);
        end
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% POST PROCESSING
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% PLOT OF ELECTRODE FLUX VS. wg
linest(1)={'-'};
linest(2)={'--'};
linest(3)={'-.'};
linest(4)={':'};

col='brgk';


Ha=figure(size(w,2)+1);
set(Ha ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
%subplot(1,2,2);
```

```
hold on;
m=1;
clear lgnd;
for k=1:size(ratio,2)
    plot(w,−flux_coll(t,:,k),'Marker','x','color','b','LineStyle',char(linest(k)),'LineWidth',2,'MarkerSize',8);
    text(w(3),−flux_coll(t,3,k),['\leftarrow␣coll.;␣w_e=',num2str(ratio(k)),'w_g'],'FontSize',18,'VerticalAlignment','top');

    plot(w,flux_gen(t,:,k),'Marker','v','color','g','LineStyle',char(linest(k)),'LineWidth',2,'MarkerSize',8);
    text(w(3),flux_gen(t,3,k),['\leftarrow␣gen;␣w_e=',num2str(ratio(k)),'w_g'],'FontSize',18,'VerticalAlignment','top');
end
hold off;
set(gca,'linewidth',2,'FontSize',18);
grid on;
xlabel('Gap␣width␣in␣m','FontWeight','bold','FontSize',18);
ylabel('Current␣at␣the␣electrodes␣(A/m)','FontWeight','bold','FontSize',18);
title(['Gap␣size␣vs.␣electrode␣current␣at␣t=',num2str(time(t)),'s'],'FontWeight','bold','FontSize',18);
%lg=legend(char(lgnd));
%loc1=['Z:\afstuderen\πcs\sim—ida—varied—wg\electrode—flux—vs—w.jpg'];
loc1=['/home/mathieu/odie/afstuderen/πcs/sim—ida—varied—wg/electrode—flux—vs—w.jpg'];
saveas(Ha,loc1,'jpg');

%% PLOT OF COLLECTION EFFICIENCY VS. wg
Hb=figure(size(w,2)+2);
set(Hb ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
hold on;
m=1;
clear lgnd;
for k=1:size(ratio,2)
    lgnd(m)={['CE;␣w_e=',num2str(ratio(k)),'w_g']};
    plot(w,coll_eff(t,:,k),'Marker','o','LineStyle',char(linest(k)),'color',col(k),'LineWidth',2,'MarkerSize',8);
    m=m+1;
end
hold off;
set(gca,'linewidth',2,'FontSize',18);
legend(char(lgnd));
xlabel('Gap␣width␣in␣m','FontWeight','bold','FontSize',18);
ylabel('collection␣efficiency␣[o]␣(%)','FontWeight','bold','FontSize',18);
grid on;
title('Gap␣size␣vs.␣collection␣efficiency','FontWeight','bold','FontSize',18);
%loc2=['Z:\afstuderen\πcs\sim—ida—varied—wg\coll—eff—vs—w.jpg'];
loc2=['/home/mathieu/odie/afstuderen/πcs/sim—ida—varied—wg/coll—eff—vs—w.jpg'];
saveas(Hb,loc2,'jpg');

%% PLOT OF redox cycles VS. wg
Hc=figure(size(w,2)+3);
set(Hc ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);

hold on;
m=1;
clear lgnd;
for k=1:size(ratio,2)
    lgnd(m)={['RC;␣w_e=',num2str(ratio(k)),'w_g']};
    plot(w,redox_cycles(t,:,k),'Marker','+','LineStyle',char(linest(k)),'color',col(k),'LineWidth',2,'MarkerSize',8);
    m=m+1;
end
hold off;
set(gca,'linewidth',2,'FontSize',18);
legend(char(lgnd));

xlabel('Gap␣width␣in␣m','FontWeight','bold','FontSize',18);
ylabel('Redox␣cycles␣[+]','FontWeight','bold','FontSize',18);
grid on;
title('Gap␣size␣vs.␣Redox␣cycles','FontWeight','bold','FontSize',18);
%loc3=['Z:\afstuderen\πcs\sim—ida—varied—wg\redox—cycles—vs—w.jpg'];
loc3=['/home/mathieu/odie/afstuderen/πcs/sim—ida—varied—wg/redox—cycles—vs—w.jpg'];
saveas(Hc,loc3,'jpg');

%% PLOT OF ELECTRODE FLUX VS. t
ww=size(w,2);
Hd=figure(size(w,2)+4);
set(Hd ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);

hold on;
%m=1;
%clear lgnd;
k=3;
%for k=1:size(ratio,2)
    %lgnd(m)={['Current collector; w_e=',num2str(ratio(k)),'w_g']};
    plot(time,−flux_coll(:,ww,k),'marker','x','LineStyle',char(linest(k)),'color','b','LineWidth',2,'MarkerSize',8);
    %m=m+1;

    %lgnd(m)={['Current generator; w_e=',num2str(ratio(k)),'w_g']};
    plot(time,flux_gen(:,ww,k),'marker','v','LineStyle',char(linest(k)),'color','g','LineWidth',2,'MarkerSize',8);
    %m=m+1;
%end
hold off;
%legend(char(lgnd));
set(gca,'linewidth',2,'FontSize',18);
grid on;
%ylim([0 2.9E −4]);
xlabel('time␣(s)','FontWeight','bold','FontSize',18);
ylabel('Current␣at␣the␣electrodes␣(A/m)','FontWeight','bold','FontSize',18);
title(['time␣vs.␣electrode␣current␣at␣w_g=',num2str(w(ww)),'␣m'],'FontWeight','bold','FontSize',18);
%loc4=['Z:\afstuderen\πcs\sim—ida—varied—wg\electrode—flux—vs—t.jpg'];
loc4=['/home/mathieu/odie/afstuderen/πcs/sim—ida—varied—wg/electrode—flux—vs—t.jpg'];
saveas(Hd,loc4,'jpg');

%% save data to latex table
clear columnLabels;
columnLabels= {'We/2␣(m)','Wg␣(m)','Current␣col.␣[x]␣(A/m)','Current␣gen.␣[∇](A/m)',...
    'CE␣[o]␣(\%)','RC␣[+]'};
```

```
clear matrix;
matrix(:,1)=w/2*ratio(1);
matrix(:,2)=w;
matrix(:,3)=-flux_coll(t,:,1);
matrix(:,4)=flux_gen(t,:,1);
matrix(:,5)=coll_eff(t,:,1);
matrix(:,6)=redox_cycles(t,:,1);
matrix2(:,1)=w/2*ratio(2);
matrix2(:,2)=w;
matrix2(:,3)=-flux_coll(t,:,2);
matrix2(:,4)=flux_gen(t,:,2);
matrix2(:,5)=coll_eff(t,:,2);
matrix2(:,6)=redox_cycles(t,:,2);
matrix3(:,1)=w/2*ratio(3);
matrix3(:,2)=w;
matrix3(:,3)=-flux_coll(t,:,3);
matrix3(:,4)=flux_gen(t,:,3);
matrix3(:,5)=coll_eff(t,:,3);
matrix3(:,6)=redox_cycles(t,:,3);

matrix=cat(1,matrix,matrix2);
matrix=cat(1,matrix,matrix3);


%loc4='Z:/afstuderen/eindverslag/data/wg-study.tex';
loc4='/home/mathieu/odie/afstuderen/eindverslag/data/wg-study.tex';
matrix2latex(matrix, loc4,...
    'columnLabels',columnLabels, 'alignment', 'c', 'format', '%10.2e',...
    'size', 'tiny','caption',...
    'Collection␣efficiency,␣redox␣cycles␣and␣electrode␣current␣with␣different␣IDA␣geometries',...
    'label','tbl:wg-vs-coll-eff-IDA');

%% Save workspace
save '/home/mathieu/odie/afstuderen/scripts/resultStorage/varied-wg';
%save 'Z:\afstuderen\scripts\resultStorage\varied-wg';
```

D.4 IDA with varied electrode height

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%% set some general parameters or variables
clear;
scrsz = get(0,'ScreenSize');
F=96485;
ymax=5E-5;
w=2;
w=w*1E-6;
h=[0.1 0.25 0.5 1 2 3 5 7];
%h=[5];
h=h*1E-6;
time=[0:0.01:0.2];
t=size(time,2); %corresponds to 0.2s
flux_coll=zeros(size(time,2),size(h,2));
flux_gen=zeros(size(time,2),size(h,2));
coll_eff=zeros(size(time,2),size(h,2));
redox_cycles=zeros(size(time,2),size(h,2));

%clear fem structure
flclear fem

%% COMSOL version
clear vrsn
vrsn.name = 'COMSOL␣3.2';
vrsn.ext = '';
vrsn.major = 0;
vrsn.build = 222;
vrsn.rcs = 'Name:';
vrsn.date = 'Date: 2005/09/0118:02:30';
fem.version = vrsn;

% Constants
fem.const = {'c0','5', ...
  'D','7E-10', ...
  'F','96485.34', ...
  'R','8.314472', ...
  'T','290', ...
  'Estd','0.265', ...
  'alpha','0.5', ...
  'Ec','0.7', ...
  'Eg','-0.2', ...
  'ks','1.5E-4'};

%iterate through electrode heights
for i=1:size(h,2)
    display(i)
    % Geometry
    wg=w;
    we=w/2;
    clear draw
    clear g1
    c1 = line1([-we we],[0 0]);
    c2 = line1([-we,-we],[0,h(i)]);
    c3 = line1([we,we],[0,h(i)]);
    c4 = line1([-wg,-we],[h(i),h(i)]);
    c5 = line1([wg,we],[h(i),h(i)]);
    c6 = line1([-wg,-wg],[h(i),ymax]);
    c7 = line1([wg,wg],[h(i),ymax]);
    c8 = line1([-wg,wg],[ymax,ymax]);
    g1 = geomcoerce('solid',{c1,c2,c3,c4,c5,c6,c7,c8});
    %geomplot(g1);
    draw.s.objs = {g1};
    draw.s.name = {'CO1'};
    draw.s.tags = {'g1'};
    fem.draw = draw;
    fem.geom = geomcsg(fem);

    % Initialize mesh
    mesh=5E-9;
    fem.mesh=meshinit(fem, ...
                      'hmaxedg',[2,mesh,4,mesh,6,mesh,7,mesh], ...
                      'hmaxsub',[1,0.5e-6]);

    % (Default values are not included)

    % Application mode 1
    clear appl
    appl.mode.class = 'Diffusion';
    appl.dim = {'Cox','Cre'};
    appl.module = 'CHEM';
    appl.gporder = 4;
    appl.cporder = 2;
    appl.assignsuffix = '_chdi';
    clear prop
    prop.analysis='static';
    appl.prop = prop;
    clear bnd
    bnd.type = {{'N0';'N0'},{'N';'N'},{'C';'C'},{'N';'N'}}; %sorted in groups
    bnd.c0 = {0,0,'c0',0}; %sorted in groups
    bnd.N = {0,{'-kfg*Cox+kbg*Cre';'kfg*Cox-kbg*Cre'},0,{'kbc*Cre-kfc*Cox';'-kbc*Cre+kfc*Cox'}}; %sorted in groups
    bnd.ind = [1,2,3,2,1,4,4,1]; %connect boundaries to groups
    appl.bnd = bnd;
    clear equ
    equ.D = {{'D';'D'}};
    equ.init = {{'c0';0}};
```

```
    equ.ind = [1];
    appl.equ = equ;
    fem.appl{1} = appl;
    fem.border = 1;
    fem.outform = 'general';
    fem.units = 'SI';

    % Subdomain expressions
    clear equ
    equ.ind = [1];
    equ.dim = {'Cox','Cre'};
    equ.expr = {'kfg','ks∗exp(−alpha∗F∗(Eg−Estd)/(R∗T))', ...
      'kbc','ks∗exp((1−alpha)∗F∗(Ec−Estd)/(R∗T))',...
      'kfc','ks∗exp(−alpha∗F∗(Ec−Estd)/(R∗T))',...
      'kbg','ks∗exp((1−alpha)∗F∗(Eg−Estd)/(R∗T))'};
    fem.equ = equ;

    % Descriptions
    clear descr
    descr.const= {'D','Diffusion␣constant','R','Gas␣constant␣J/K∗mol',...
        'F','Faradays␣constant␣C/mol','c0','Bulk␣concentration␣Oxidized␣species',...
        'T','Temp.␣K','Estd','Standard␣potenial␣from␣Yang'};
    fem.descr = descr;

    % Multiphysics
    fem=μltiphysics(fem);

    % Extend mesh
    fem.xmesh=meshextend(fem);

    % Solve problem
    fem.sol=femtime(fem, ...
                    'solcomp',{'Cox','Cre'}, ...
                    'outcomp',{'Cox','Cre'}, ...
                    'tlist',time, ...
                    'tout','tlist');

    % Save current fem structure for restart purposes
    fem0=fem;

    % Plot solution and save it to disk
    H=figure(i);
    set(H ,'Position',[50 50 scrsz(3)∗0.8 scrsz(4)∗0.8]);

  %  'contdata',{'Cox','cont','internal'}, ...
 %            'contlevels',20, ...
%              'contmap','cool(1024)', ...
%            'title',['t=',num2str(time(t)),'; Cox [mol/m^3];'], ...
    postplot(fem, ...
             'tridata',{'Cox','cont','internal'}, ...
             'trimap','jet(1024)', ...
             'arrowdata',{'Coxx','Coxy'}, ...
             'arrowxspacing',15, ...
             'arrowyspacing',15, ...
             'arrowtype','arrow', ...
             'arrowstyle','proportional', ...
             'arrowcolor',[1.0,0.0,0.0], ...
             'solnum',t, ...
             'refine',3, ...
             'axis',[−4E−6,4E−6,−1E−7,1E−5,−1,1]);
    xlabel('x␣(m)','FontSize',18);
    ylabel('y␣(m)','FontSize',18);
    grid on;
    set(gca,'linewidth',2,'FontSize',18);
    loc=['Z:\afstuderen\πcs\sim−ida−varied−h\h−',num2str(h(i)),'.jpg'];
    %loc=['/home/mathieu/odie/afstuderen/πcs/sim−ida−varied−h/h−',num2str(h(i)),'.jpg'];
    saveas(H,loc,'jpg');

    %integrate normal diffusive flux at electrode boundaries
    for j=1:size(time,2)
        % Integrate
        flux_coll(j,i)=F∗postint(fem,'ndflux_Cox_chdi', ...
                   'dl',[6,7], ...
                   'edim',1, ...
                   'solnum',j);

        % Integrate
        flux_gen(j,i)=F∗postint(fem,'ndflux_Cox_chdi', ...
                   'dl',[2,4], ...
                   'edim',1, ...
                   'solnum',j);
        %calculate collection efficiency and redox cycling
        coll_eff(j,i)=−flux_coll(j,i)/flux_gen(j,i)∗100;
        redox_cycles(j,i)=1/(1−(coll_eff(j,i)/100)^2);
    end
end

t=11;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% POST PROCESSING
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% PLOT OF COLLECTION EFFICIENCY VS. h
Ha=figure(size(h,2)+1);
set(Ha ,'Position',[50 50 scrsz(3)∗0.8 scrsz(4)∗0.8]);
%subplot(1,2,1);
plot(h,coll_eff(t,:),'Marker','o','color','b','LineWidth',2,'MarkerSize',8);
grid on;
xlabel('Electrode␣height␣in␣m','FontWeight','bold','FontSize',18);
ylabel('collection␣efficiency␣[o]␣(%)','FontWeight','bold','FontSize',18);
title(['He␣vs.␣collection␣efficiency␣at␣t=',num2str(time(t))],'FontWeight','bold','FontSize',18);
loc1=['Z:\afstuderen\πcs\sim−ida−varied−h\coll−eff−vs−h.jpg'];
```

```
set(gca,'linewidth',2,'fontsize',18);
%loc1=['/home/mathieu/odie/afstuderen/πcs/sim-ida-varied-h/coll-eff-vs-h.jpg'];
saveas(Ha,loc1,'jpg');

%% PLOT OF REDOX CYLES VS. h
Hb=figure(size(h,2)+2);
set(Hb ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
%subplot(1,2,1);
plot(h,redox_cycles(t,:),'Marker','+','color','g','LineWidth',2,'MarkerSize',8);
grid on;
xlabel('Electrode␣height␣in␣m','FontWeight','bold','FontSize',18);
ylabel('Redox␣cycles␣[+]','FontWeight','bold','FontSize',18);
title(['He␣vs.␣redox␣cycles␣at␣t=',num2str(time(t))],'FontWeight','bold','FontSize',18);
set(gca,'linewidth',2,'fontsize',18);

loc2=['Z:\afstuderen\πcs\sim-ida-varied-h\redox-cycles-vs-h.jpg'];
%loc2=['/home/mathieu/odie/afstuderen/πcs/sim-ida-varied-h/redox-cycles-vs-h.jpg'];
saveas(Hb,loc2,'jpg');

%% PLOT OF ELECTRODE FLUX VS. h
Hc=figure(size(h,2)+3);
set(Hc ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
plot(h,-flux_coll(t,:),'Marker','x','color','b','LineWidth',2,'MarkerSize',8);
hold on;
plot(h,flux_gen(t,:),'Marker','v','color','g','LineWidth',2,'MarkerSize',8);
hold off;
grid on;
xlabel('electrode␣height␣in␣m','FontWeight','bold','FontSize',18);
ylabel('Current␣at␣the␣electrodes␣(A/m)','FontWeight','bold','FontSize',18);
title(['He␣vs.␣electrode␣current␣at␣t=',num2str(time(t)),'␣s'],'FontWeight','bold','FontSize',18);
set(gca,'linewidth',2,'fontsize',18);
loc3=['Z:\afstuderen\πcs\sim-ida-varied-h\electrode-flux-vs-h.jpg'];
%loc3=['/home/mathieu/odie/afstuderen/πcs/sim-ida-varied-h/electrode-flux-vs-h.jpg'];
saveas(Hc,loc3,'jpg');

%% PLOT OF ELECTRODE FLUX VS. t
Hd=figure(size(h,2)+2);
set(Hd ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
hh=7;
[AX,H1,H2]=plotyy(time,-flux_coll(:,hh),time,flux_gen(:,hh));
ylimits1=get(AX(1),'YLim');
yinc1 = (ylimits1(2)-ylimits1(1))/5;
set(AX(1),'YTick',[ylimits1(1):yinc1:ylimits1(2)])
ylimits2=get(AX(2),'YLim');
yinc2 = (ylimits2(2)-ylimits2(1))/5;
set(AX(2),'YTick',[ylimits2(1):yinc2:ylimits2(2)]);
grid on;
set(H1,'Marker','x');
set(H2,'Marker','v');
xlabel('time␣(s)');
set(get(AX(1),'Ylabel'),'String','Current␣at␣collector␣[x]␣electrode␣(A/m)','FontWeight','bold','FontSize',14);
set(get(AX(2),'Ylabel'),'String','Current␣at␣generator␣[\nabla]␣electrode␣(A/m)','FontWeight','bold','FontSize',14);
title(['time␣vs.␣electrode␣current␣at␣electrode␣height␣',num2str(h(hh)),'␣m'],'FontWeight','bold','FontSize',14);
%loc3=['Z:\afstuderen\πcs\sim-ida-varied-h\electrode-flux-vs-t.jpg'];
loc4=['/home/mathieu/afstuderen/πcs/sim-ida-varied-h/electrode-flux-vs-t.jpg'];
saveas(Hd,loc4,'jpg');

%% save data to latex table
clear columnLabels;
columnLabels= {'He␣(m)','Current␣col.␣[x]␣(A/m)','Current␣gen.␣[∇] (A/m)','CE␣[o](\%)','RC␣[+]'};
clear matrix;
matrix(:,1)=h;
matrix(:,2)=-flux_coll(t,:);
matrix(:,3)=flux_gen(t,:);
matrix(:,4)=coll_eff(t,:);
matrix(:,5)=redox_cycles(t,:);

loc4='Z:/afstuderen/eindverslag/data/he-study.tex';
%loc4='/home/mathieu/afstuderen/eindverslag/data/he-study.tex';
matrix2latex(matrix, loc4,...
    'columnLabels',columnLabels, 'alignment', 'c', 'format', '%10.2e',...
    'size', 'tiny','caption',...
    'Collection␣efficiency␣of␣different␣electrode␣heights',...
    'label','tbl:h-vs-coll-eff-IDA');
```

D.5 Parallel plate with varied fluid velocity

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%% Set some general variables
clear;
F=96485; %faraday constant
scrsz = get(0,'ScreenSize');
V_max=[1E-3 5E-3 1E-2 2E-2 5E-2 0.1];
w=9;
w=w*1E-6;
w_ins=1E-6;
h=1;
h=h*1E-6;
flux_coll=zeros(1,size(V_max,2));
flux_gen=zeros(1,size(V_max,2));
coll_eff=zeros(1,size(V_max,2));
redox_cycles=zeros(1,size(V_max,2));

%% clear fem structure
flclear fem

% COMSOL version
clear vrsn
vrsn.name = 'COMSOL␣3.2';
vrsn.ext = '';
vrsn.major = 0;
vrsn.build = 222;
vrsn.rcs = 'Name:';
vrsn.date = 'Date: 2005/09/0118:02:30';
fem.version = vrsn;

% set up Geometry
clear draw
c1 = line1([0 w_ins],[0 0]);
c2 = line1([w_ins w_ins+w],[0 0]);
c3 = line1([0 w_ins],[h h]);
c4 = line1([w_ins w_ins+w],[h h]);
c5 = line1([0 0],[0 h]);
c6 = line1([w_ins+w w_ins+w],[0 h]);
g1 = geomcoerce('solid',{c1,c2,c3,c4,c5,c6});
%geomplot(g1);
draw.s.objs = {g1};
draw.s.name = {'R1'};
draw.s.tags = {'g1'};
fem.draw = draw;
fem.geom = geomcsg(fem);

% Initialize mesh
mesh=5E-9;
fem.mesh=meshinit(fem, ...
                  'hmaxedg',[4,mesh,5,mesh]);

% (Default values are not included)

% Application mode 1
clear appl
appl.mode.class = 'FlConvDiff';
appl.dim = {'Cox'};
appl.assignsuffix = '_cd';
clear prop
prop.analysis='static';
appl.prop = prop;
clear bnd
bnd.type = {'C','N','Nc','N0','N0','N'};
bnd.c0 = {'c0',0,0,0,'c0','c0'};
bnd.N = {0,'-kfg*Cox+kbg*Cre',0,0,0,'kbc*Cre-kfc*Cox'};
bnd.ind = [1,4,5,2,6,3];
appl.bnd = bnd;
clear equ
equ.D = 'D';
equ.init = 'c0';
equ.u = 'Vmax*(1-((y-0.5*hc)/0.5*hc)^2)';
equ.ind = [1];
appl.equ = equ;
fem.appl{1} = appl;

% Application mode 2
clear appl
appl.mode.class = 'FlConvDiff';
appl.dim = {'Cre'};
appl.name = 'cd2';
appl.assignsuffix = '_cd2';
clear prop
prop.analysis='static';
clear weakconstr
weakconstr.value = 'off';
weakconstr.dim = {'lm2'};
prop.weakconstr = weakconstr;
appl.prop = prop;
clear bnd
bnd.type = {'C','N','Nc','N0','N0','N'};
bnd.c0 = {0,'c0',0,'c0',0,0};
bnd.N = {0,'kfg*Cox-kbg*Cre',0,0,0,'-kbc*Cre+kfc*Cox'};
bnd.ind = [1,4,5,2,6,3];
appl.bnd = bnd;
clear equ
equ.D = 'D';
```

```
equ.u = 'Vmax*(1-((y-0.5*hc)/0.5*hc)^2)';
equ.ind = [1];
appl.equ = equ;
fem.appl{2} = appl;
fem.border = 1;
fem.outform = 'general';
fem.units = 'SI';

% Subdomain expressions
clear equ
equ.ind = [1];
equ.dim = {'Cox','Cre'};
equ.expr = {'kfg','ks*exp(-alpha*F*(Eg-Estd)/(R*T))', ...
      'kbc','ks*exp((1-alpha)*F*(Ec-Estd)/(R*T))',...
      'kfc','ks*exp(-alpha*F*(Ec-Estd)/(R*T))',...
      'kbg','ks*exp((1-alpha)*F*(Eg-Estd)/(R*T))'};
fem.equ = equ;

for i=1:size(V_max,2)
    % define model Constants
    fem.const = {'c0','5', ...
      'D','7E-10', ...
      'Vmax',V_max(i), ...
      'hc','1e-6', ...
      'F','96485', ...
      'R','8.31', ...
      'T','290', ...
      'alpha','0.5', ...
      'Estd','0.265', ...
      'Eg','-0.2', ...
      'Ec','0.7', ...
      'ks','1.5E-4'};

    % Multiphysics
    fem=µltiphysics(fem);

    % Extend mesh
    fem.xmesh=meshextend(fem);

    % Solve problem
    fem.sol=femnlin(fem, ...
                    'solcomp',{'Cox','Cre'}, ...
                    'outcomp',{'Cox','Cre'});

    % Save current fem structure for restart purposes
    fem0=fem;

    % Plot solution
    %'title',['u_{max}=',num2str(V_max(i)),' [m/s]; Surface: Conc., Cox  [mol/m^3]'], ...
    H=figure(i);
    set(H ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
    postplot(fem, ...
            'tridata',{'Cre','cont','internal'}, ...
            'trimap','jet(1024)', ...
            'refine',2, ...
            'axis',[-0.1E-6,10.1E-6,-0.1E-6,1.1E-6,-1,1]);
    loc1=['Z:\afstuderen\πcs\sim-idee7b\Vmax-',num2str(V_max(i)),'.jpg'];
    loc1=['/home/mathieu/odie/afstuderen/πcs/sim-idee7b/Vmax-',num2str(V_max(i),'%10.0e'),'.jpg'];
    set(gca,'linewidth',2,'fontsize',18);
    saveas(H,loc1,'jpg');

    %integrate normal diffusive flux at electrode boundaries
    % Integrate
    flux_gen(i)=F*postint(fem,'ntflux_Cre_cd2', ...
               'dl',[4], ...
               'edim',1);

    % Integrate
    flux_coll(i)=F*postint(fem,'ntflux_Cre_cd2', ...
               'dl',[5], ...
               'edim',1);
    %calculate collection efficiency and redox cycling
    coll_eff(i)=-flux_coll(i)/flux_gen(i)*100;
    redox_cycles(i)=1/(1-(coll_eff(i)/100)^2);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% POST PROCESSING
%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% PLOT OF CE VS. V_max
Ha=figure(size(V_max,2)+1);
set(Ha ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
hold on;
plot(V_max,coll_eff,'Marker','o','color','b','LineWidth',2,'MarkerSize',8);
hold off;
grid on;
xlabel('Fluid␣velocity␣u_{max}␣(m/s)','FontWeight','bold','FontSize',18);
ylabel('collection␣efficiency␣[0]␣(%)','FontWeight','bold','FontSize',18);
set(gca,'linewidth',2,'fontsize',18);
title('Fluid␣velocity␣vs.␣collection␣efficiency','FontWeight','bold','FontSize',18);
%loc2=['Z:\afstuderen\πcs\sim-idee7b\Vmax-vs-CE.jpg'];
loc2=['/home/mathieu/odie/afstuderen/πcs/sim-idee7b/Vmax-vs-CE.jpg'];
saveas(Ha,loc2,'jpg');

%% plot of RC vs. V_max
Hb=figure(size(V_max,2)+2);
set(Hb ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
hold on;
plot(V_max,redox_cycles,'Marker','+','color','b','LineWidth',2,'MarkerSize',8);
hold off;
```

```
grid on;
xlabel('Fluid␣velocity␣u_{max}␣(m/s)','FontWeight','bold','FontSize',18);
ylabel('Redox␣cycles␣[+]','FontWeight','bold','FontSize',18);
set(gca,'linewidth',2,'fontsize',18);
title('Fluid␣velocity␣vs.␣collection␣efficiency','FontWeight','bold','FontSize',18);
loc3=['Z:\afstuderen\πcs\sim-idee7b\Vmax-vs-CE.jpg'];
loc3=['/home/mathieu/odie/afstuderen/πcs/sim-idee7b/Vmax-vs-RC.jpg'];
saveas(Hb,loc3,'jpg');

%% Electrode flux vs Vmax
Hc=figure(size(V_max,2)+3);
set(Hc ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
hold on;

plot(V_max,flux_coll,'Marker','x','color','b','LineWidth',2,'MarkerSize',8)
plot(V_max,-flux_gen,'Marker','v','color','g','LineWidth',2,'MarkerSize',8);
hold off;
grid on;
xlabel('Fluid␣velocity␣u_{max}␣(m/s)','FontWeight','bold','FontSize',18);
ylabel('Current␣(A/m)','FontWeight','bold','FontSize',18);
set(gca,'linewidth',2,'fontsize',18);
title('Current␣vs.␣fluid␣velocity','FontWeight','bold','FontSize',18);
loc4=['Z:\afstuderen\πcs\sim-idee7b\Vmax-vs-CE.jpg'];
loc4=['/home/mathieu/odie/afstuderen/πcs/sim-idee7b/Vmax-vs-flux.jpg'];
saveas(Hc,loc4,'jpg');

%% save data to latex table
columnLabels= {'$u_{max}$ (m/s)','Current␣col.␣[x](A/m)','Current␣gen.␣[∇](A/m)','CE␣[o](\%)','RC␣[+]'};
matrix(:,1)=V_max;
matrix(:,2)=flux_coll(1,:);
matrix(:,3)=- flux_gen(1,:);
matrix(:,4)=coll_eff(1,:);
matrix(:,5)=redox_cycles(1,:);

loc5='Z:/afstuderen/eindverslag/data/Vmax-study.tex';
loc5='/home/mathieu/odie/afstuderen/eindverslag/data/Vmax-study.tex';
matrix2latex(matrix,loc3 ,...
    'columnLabels',columnLabels, 'alignment', 'c', 'format', '%10.2e',...
    'size', 'tiny','caption',...
    'Collection␣efficiency␣and␣electrode␣current␣at␣different␣fluid␣velocities',...
    'label','tbl:v-vs-coll-eff');
```

D.6 Parallel plate with passive bulk supply

Parameter study (main file)

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%% Set some general variables
clear;
scrsz = get(0,'ScreenSize');
w_ee=[8 16 32];
%w_ee=[4 8 12 16];
w_ee=w_ee*1E-6;
w_gap_tot=[2E-6 4E-6 8E-6];
h_tot=[0.45E-6];
h_gap=5E-7;
ymax=5E-5;
time=[0:0.01:0.1];
t=size(time,2);
jb=90;

%% Place jobs
for k=1:size(h_tot,2)
    h=h_tot(k);
    for j=1:size(w_gap_tot,2)
        jobId=jb+10;
        jb=jb+1;
        w_gap=w_gap_tot(j);
        for i=1:size(w_ee,2)
            w=w_ee(i);
            job=['ideenr82DfunExp(',num2str(w),',',num2str(w_gap),',',num2str(h),',',num2str(h_gap),',',num2str(ymax),')'];
            setJob(job,i,jobId);
        end
    end
end

%% combine jobs
tot_res=struct([]);
jb=90;
for k=1:size(h_tot,2)
    h=h_tot(k);
    for j=1:size(w_gap_tot,2)
        jobId=jb+10;
        jb=jb+1;
        res=combineJob(jobId,size(w_ee,2),1,1);
        tot_res(j,k).result=res;
    end
end

%% plot results
H=figure(size(w_ee,2)+1);
set(H,'Position',[50 50 scrsz(3)*0.9 scrsz(4)*0.85]);
subplot(1,2,1);
hold on;
colors=get(gca,'ColorOrder');
colors(8,:)=[1 0.5 0.25];
colors(9,:)=[1 0 1];
set(gca,'ColorOrder',colors);
markers=['o','+','x','v','s','d','>','<','^','p','h'];
stringmatrix=java_array('java.lang.String', size(h_tot,2)*size(w_gap_tot,2));
s=0;
for k=1:size(h_tot,2)
    for j=1:size(w_gap_tot,2)
        s=s+1;
        string=['w_{gap}=',num2str(w_gap_tot(j)),'\μ␣m',';␣h=',num2str(h_tot(k)),'\μ␣m'];
        stringmatrix(s)=java.lang.String(string);
        %current_gen1=zeros(1,size(w_ee,2));
        %current_gen2=zeros(1,size(w_ee,2));
        %coll_eff=zeros(1,size(w_ee,2));
        redox_cycles=zeros(1,size(w_ee,2));

        for i=1:size(w_ee,2)
            %Order is:    current_gen1 current_gen2 redox_cycles coll_eff
            %current_gen1(i)=tot_res(j).result(i).result.jobres(1);
            %current_gen2(i)=tot_res(j).result(i).result.jobres(2);
            redox_cycles(i)=tot_res(j,k).result(i).result.jobres(3);
            %coll_eff(i)=tot_res(j).result(i).result.jobres(4);
        end
        plot(w_ee,redox_cycles,'color',colors(s,:),'Marker',markers(s));
        %[AX,H1,H2]=plotyy(w_ee,coll_eff,w_ee,redox_cycles);

    end
end
hold off;
grid on;
xlabel('w_{ee}␣(m)');
ylabel('Redox␣cycles');
title('Redox␣cycles␣with␣different␣sensor␣geometries');
stringm=char(stringmatrix);
L=legend(stringm,'location','NorthWest');
set(L,'FontSize',8);

subplot(1,2,2);
hold on;
stringmatrix=java_array('java.lang.String', size(h_tot,2)*size(w_gap_tot,2));
s=0;
for k=1:size(h_tot,2)
    for j=1:size(w_gap_tot,2)
        s=s+1;
```

```
        string=['w_{gap}=',num2str(w_gap_tot(j)),'\μ␣m',';␣h=',num2str(h_tot(k)),'\μ␣m'];
        stringmatrix(s)=java.lang.String(string);
       % current_gen1=zeros(1,size(w_ee,2));
        %current_gen2=zeros(1,size(w_ee,2));
        coll_eff=zeros(1,size(w_ee,2));
        %redox_cycles=zeros(1,size(w_ee,2));

        for i=1:size(w_ee,2)
            %Order is:    current_gen1 current_gen2 redox_cycles coll_eff
            %current_gen1(i)=tot_res(j).result(i).result.jobres(1);
            %current_gen2(i)=tot_res(j).result(i).result.jobres(2);
            %redox_cycles(i)=tot_res(j).result(i).result.jobres(3);
             coll_eff(i)=tot_res(j,k).result(i).result.jobres(4);
        end
        plot(w_ee,coll_eff,'color',colors(s,:),'Marker',markers(s));
        %[AX,H1,H2]=plotyy(w_ee,coll_eff,w_ee,redox_cycles);

    end
end
hold off;
grid on;
xlabel('w_{ee}␣(m)');
ylabel('collection␣efficiency␣(%)');
title('Averaged␣collection␣efficiency␣with␣different␣sensor␣geometries');
stringm=char(stringmatrix);
L1=legend(stringm,'location','southEast');
set(L1,'FontSize',8);
loc1=['Z:\afstuderen\πcs\sim-idee8\Wtotal-vs-CE.jpg'];
%loc1=['/home/mathieu/odie/afstuderen/πcs/sim-idee8/Wtotal-vs-CE.jpg'];
%saveas(H,loc1,'jpg');
%saveas(H,['/home/mathieu/afstuderen/πcs/sim-idee8/Wtotal-vs-CE.jpg'],'jpg');

H=figure(size(w_ee,2)+2);
set(H,'Position',[50 50 scrsz(3)*0.9 scrsz(4)*0.85]);
subplot(1,2,1);
hold on;
s=0;
for k=1:size(h_tot,2)
    for j=1:size(w_gap_tot,2)
        s=s+1;
        string=['w_{gap}=',num2str(w_gap_tot(j)),'\μ␣m',';␣h=',num2str(h_tot(k)),'\μ␣m'];
        stringmatrix(s)=java.lang.String(string);
        %string=['gen=on, w_{gap}=',num2str(w_gap_tot(j)),'\μ m'];
        %stringmatrix(s)=java.lang.String(string);
        %s=s+1;
        current_gen1=zeros(1,size(w_ee,2));
        %current_gen2=zeros(1,size(w_ee,2));
        %coll_eff=zeros(1,size(w_ee,2));
        %redox_cycles=zeros(1,size(w_ee,2));

        for i=1:size(w_ee,2)
            %Order is:    current_gen1 current_gen2 redox_cycles coll_eff
             current_gen1(i)=tot_res(j,k).result(i).result.jobres(1);
            %current_gen2(i)=tot_res(j).result(i).result.jobres(2);
            %redox_cycles(i)=tot_res(j).result(i).result.jobres(3);
            %coll_eff(i)=tot_res(j).result(i).result.jobres(4);
        end
        plot(w_ee,current_gen1,'color',colors(s,:),'LineStyle','-','Marker',markers(s));
        %plot(w_ee,current_gen2,'color',colors(j,:),'LineStyle','-');
        %[AX,H1,H2]=plotyy(w_ee,coll_eff,w_ee,redox_cycles);

    end
end
hold off;
grid on;
xlabel('w_{ee}␣(m)');
ylabel('Current␣(A/m)');
title('Generator␣electrode␣current␣(gen.=top)␣under␣different␣sensor␣geometries');
stringm=char(stringmatrix);
L2=legend(stringm,'location','NorthWest');
set(L2,'FontSize',8);

subplot(1,2,2);
hold on;
s=0;
for k=1:size(h_tot,2)
    for j=1:size(w_gap_tot,2)
        s=s+1;
        string=['w_{gap}=',num2str(w_gap_tot(j)),'\μ␣m',';␣h=',num2str(h_tot(k)),'\μ␣m'];
        stringmatrix(s)=java.lang.String(string);
        %string=['gen=on, w_{gap}=',num2str(w_gap_tot(j)),'\μ m'];
        %stringmatrix(s)=java.lang.String(string);
        %s=s+1;
        %current_gen1=zeros(1,size(w_ee,2));
        %current_gen2=zeros(1,size(w_ee,2));
        current_coll1=zeros(1,size(w_ee,2));
        %coll_eff=zeros(1,size(w_ee,2));
        %redox_cycles=zeros(1,size(w_ee,2));

        for i=1:size(w_ee,2)
            %Order is:    current_gen1 current_gen2 redox_cycles coll_eff
            %current_gen1(i)=tot_res(j,k).result(i).result.jobres(1);
            %current_gen2(i)=tot_res(j,k).result(i).result.jobres(2);
             current_coll1(i)=tot_res(j,k).result(i).result.jobres(5);
            %redox_cycles(i)=tot_res(j).result(i).result.jobres(3);
            %coll_eff(i)=tot_res(j).result(i).result.jobres(4);
        end
        plot(w_ee,current_coll1,'color',colors(s,:),'LineStyle','-','Marker',markers(s));
        %plot(w_ee,current_gen2,'color',colors(j,:),'LineStyle','-');
        %[AX,H1,H2]=plotyy(w_ee,coll_eff,w_ee,redox_cycles);
```

```
    end
end
hold off;
grid on;
xlabel('w_{ee}␣(m)');
ylabel('Current␣(A/m)');
title('Collector␣electrode␣current␣(col.=bottom)␣with␣different␣sensor␣geometries');
stringm=char(stringmatrix);
L3=legend(stringm,'location','NorthWest');
set(L3,'FontSize',8);
%saveas(H,['Z:\afstuderen\πcs\sim-idee8\W_total-vs-Flux.jpg'],'jpg');

%% save data to latex table
clear columnLabels;
columnLabels= {'w_ee (m)','Current␣gen.␣|␣coll.␣on␣[nabla](A/m)','Current␣gen.␣|␣coll.␣off␣[x](A/m))','CE␣[o](\%)','RC␣[+]'};
clear matrix;
matrix(:,1)=w_ee;
matrix(:,2)=-current_gen2(1,:);
matrix(:,3)=current_gen1(1,:);
matrix(:,4)=coll_eff(1,:);
matrix(:,5)=redox_cycles(1,:);

matrix2latex(matrix, 'Z:/afstuderen/eindverslag/data/w-total-study.tex',...
    'columnLabels',columnLabels, 'alignment', 'c', 'format', '%10.2e',...
    'size', 'tiny','caption',...
    'Collection␣efficiency␣with␣different␣edge-to-edge␣distances␣between␣the␣holes',...
    'label','tbl:ce-vs-w-total');
```

Parameter study (subfunction)

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk
function res =ideenr82Dfun(w_ee,w_gap,h,h_gap,ymax)
    w_tot=w_ee+w_gap;
    w=w_tot;
    time=[0:0.01:0.1];
    scrsz = get(0,'ScreenSize');
    res=[0 0 0 0];
    global jobResDir;
    t=size(time,2);
    F=96485; %faraday constant
    current_gen_on=0;
    current_gen_off=0;
    current_coll_on=0;
    current_coll_off=0;
    flclear fem

    % COMSOL version
    clear vrsn
    vrsn.name = 'COMSOL␣3.2';
    vrsn.ext = '';
    vrsn.major = 0;
    vrsn.build = 222;
    vrsn.rcs = 'Name:';
    vrsn.date = 'Date: 2005/09/0118:02:30';
    fem.version = vrsn;
    % Geometry
    %w=w_total(i);
    carr={curve2([-w/2,w/2],[0,0],[1,1]), ...
      curve2([w/2,w/2],[0,h],[1,1]), ...
      curve2([w/2,w_gap/2],[h,h],[1,1]), ...
      curve2([w_gap/2,w_gap/2],[h,h+h_gap],[1,1]), ...
      curve2([w_gap/2,w/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([w/2,w/2],[h+h_gap,ymax],[1,1]), ...
      curve2([w/2,-w/2],[ymax,ymax],[1,1]), ...
      curve2([-w/2,-w/2],[ymax,h+h_gap],[1,1]), ...
      curve2([-w/2,-w_gap/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([-w_gap/2,-w_gap/2],[h+h_gap,h],[1,1]), ...
      curve2([-w_gap/2,-w/2],[h,h],[1,1]), ...
      curve2([-w/2,-w/2],[h,0],[1,1])};
    g1=geomcoerce('solid',carr);
    clear s
    s.objs={g1};
    s.name={'CO1'};
    s.tags={'g1'};

    fem.draw=struct('s',s);
    fem.geom=geomcsg(fem);

    % Constants
    fem.const = {'c0','5', ...
      'D','7E-10', ...
      'Eg','-0.2', ...
      'Ec','0.7', ...
      'R','8.31', ...
      'T','290', ...
      'F','96000', ...
      'ks','1.5E-4', ...
      'Estd','0.265', ...
      'alpha','0.5'};

    for s=1:2
        % Initialize mesh
        dens_elec=1E-8; %use 1E-8 for "normal" structures.
        %if(s==1)
            fem.mesh=meshinit(fem, ...
                      'hmaxedg',[2,dens_elec,3,dens_elec,9,dens_elec]);
        %elseif(s==2)%collector off. Dont need dense grid at the switched off electrode.
        %    fem.mesh=meshinit(fem, ...
```

```
%                  'hmaxedg',[2,dens_elec,3,1E-7,9,1E-7]);
%else
%     error('wrong parameter for s!!');
%end

% (Default values are not included)

% Application mode 1
clear appl
appl.mode.class = 'FlDiffusion';
appl.dim = {'Cox'};
appl.assignsuffix = '_di';
clear prop
prop.analysis='static';
appl.prop = prop;
clear bnd
bnd.type = {'N0','N','C','N'};
bnd.c0 = {0,0,'c0',0};
bnd.N = {0,'kbc*Cre-kfc*Cox',0,'-kfg*Cox+kbg*Cre'};
if(s==1)
    bnd.ind = [1,4,2,1,1,3,1,1,2,1,1,1];
elseif(s==2)
    bnd.ind = [1,4,1,1,1,3,1,1,1,1,1,1];
else
    error('wrong␣parameter␣for␣s!!');
end
appl.bnd = bnd;
clear equ
equ.D = 'D';
equ.init = 'c0';
equ.ind = [1];
appl.equ = equ;
fem.appl{1} = appl;

% Application mode 2
clear appl
appl.mode.class = 'FlDiffusion';
appl.dim = {'Cre'};
appl.name = 'di2';
appl.assignsuffix = '_di2';
clear prop
prop.analysis='static';
clear weakconstr
weakconstr.value = 'off';
weakconstr.dim = {'lm2'};
prop.weakconstr = weakconstr;
appl.prop = prop;
clear bnd
bnd.type = {'N0','N','C','N'};
bnd.N = {0,'-kbc*Cre+kfc*Cox',0,'kfg*Cox-kbg*Cre'};
if(s==1)
    bnd.ind = [1,4,2,1,1,3,1,1,2,1,1,1];
elseif(s==2)
    bnd.ind = [1,4,1,1,1,3,1,1,1,1,1,1];
else
    error('wrong␣parameter␣for␣s!!');
end
appl.bnd = bnd;
clear equ
equ.D = 'D';
equ.ind = [1];
appl.equ = equ;
fem.appl{2} = appl;
fem.border = 1;
fem.outform = 'general';
fem.units = 'SI';

% Subdomain expressions
clear equ
equ.ind = [1];
equ.dim = {'Cox','Cre'};
equ.expr = {'kfg','ks*exp(-alpha*F*(Eg-Estd)/(R*T))', ...
       'kbc','ks*exp((1-alpha)*F*(Ec-Estd)/(R*T))',...
       'kfc','ks*exp(-alpha*F*(Ec-Estd)/(R*T))',...
       'kbg','ks*exp((1-alpha)*F*(Eg-Estd)/(R*T))'};

fem.equ = equ;

% Multiphysics
fem=µltiphysics(fem);

% Extend mesh
fem.xmesh=meshextend(fem);

% Solve problem
fem.sol=femtime(fem, ...
                'solcomp',{'Cox','Cre'}, ...
                'outcomp',{'Cox','Cre'}, ...
                'tlist',time, ...
                'tout','tlist');

% Save current fem structure for restart purposes
fem0=fem;

% Plot solution
H=figure(1);
set(H ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
subplot(1,2,1);
postplot(fem, ...
         'tridata',{'Cox','cont','internal'}, ...
         'trimap','jet(1024)', ...
```

```
                 'solnum',t, ...
                 'title',['t=',num2str(time(t)),'s;␣w=',num2str(w*1E6),'um;␣Conc.,␣Cox␣[mol/m^3]'], ...
                 'refine',3, ...
                 'axis',[-w/2*1.1,w/2*1.1,-1E-7,2E-5,-1,1]);

        % Plot in cross-section or along domain
        subplot(1,2,2);
        postcrossplot(fem,1,[-w/2*0.98 -w/2*0.98;0 h], ...
                      'lindata','dflux_Cox_di', ...
                      'solnum',t, ...
                      'title',['t=',num2str(time(t)),'s;␣w=',num2str(w*1E6),'um;␣Diff.␣flux,␣Cox␣[mol/(m^2*s)]'], ...
                      'axislabel',{'y','Diffusive␣flux','␣Cox␣␣[mol/(m^2*s)]'});
        if(s==1)
            coll='on';
            current_gen_on=F*postint(fem,'ndflux_Cox_di', ...
               'dl',[2], ...
               'edim',1,'solnum',t);
            current_coll_on=F*postint(fem,'ndflux_Cox_di', ...
                           'dl',[3,9], ...
                           'edim',1,'solnum',t);

        elseif(s==2)
            coll='off';
            current_gen_off=F*postint(fem,'ndflux_Cox_di', ...
               'dl',[2], ...
               'edim',1,'solnum',t);
            current_coll_off=F*postint(fem,'ndflux_Cox_di', ...
                           'dl',[3,9], ...
                           'edim',1,'solnum',t);

            redox_cycles=current_gen_on/current_gen_off;
            coll_eff=sqrt(-1/redox_cycles+1)*100;
        else
            error('wrong␣parameter␣for␣s!!');
        end
        saveas(H,[jobResDir,'Wee-',num2str(w_ee),'-coll-',coll,'-wg-',num2str(w_gap),'.jpg'],'jpg');
    end

    res=[current_gen_off current_gen_on redox_cycles coll_eff current_coll_off current_coll_on];
```

Bulk variation study

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
%% This file calculates the concentration profile in time of the parallel
%% plate sensor structure. It is possible to vary the total unit width,
%% which is the sum of the gapwidth and the distance between the gaps.
%%
%% If gen_movie is equal to 1, small timing is choosen and a movie which
%% shows the concentration profile development is generated. If it is equal
%% to zero a large time (60s) is choosen and during that time the fixed
%% concentration at the semi-infinite bulk boundary (at the top) is decreased in a number of steps.
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Set some general variables
clear;
scrsz = get(0,'ScreenSize');
%boolean which triggers the computing/time consuming movie generation
gen_movie=0;
w_ee=[16];
w_ee=w_ee*1E-6;
w_gap=1E-6;
h=1E-6;
h_gap=5E-6;
stepwidth=10;
nr_steps=5;
c0=5;
%cEnd=5;
time=[0:0.25:(nr_steps+1)*stepwidth];

if(gen_movie==1)
    time=[0:0.00001:0.001]; %ONLY use small timing when movie is generated
end

%generate a stepwise expression for the semi-infinite bulk boundary condition
expression=num2str((nr_steps+1)*0.999);
expression=['abs(floor(',expression,'-0.999*',num2str((nr_steps+1)/time(end)),'*t)/',num2str(nr_steps),'*',num2str(c0),')'];
%expression=num2str(c0);
%%% STUFF to test the validity of the expression
%%%test=zeros(1,size(time,2));
%%%for l=1:size(time,2)
%%%     t=time(l);
%%%     test(l)=eval(expression);
%%%end
%%%figure(4);
%%%plot(time,test);

D=7E-10;
F=96485;
%ymax=2*sqrt(2*D*time(end)); %some sort of expression to estimate the
                               %diffusion into a normal bulk
ymax=5E-5;



%t=size(time,2);
```

```
current_gen2=zeros(size(w_ee,2),size(time,2));
current_gen1=zeros(size(w_ee,2),size(time,2));
current_coll1=zeros(size(w_ee,2),size(time,2));
current_coll2=zeros(size(w_ee,2),size(time,2));
redox_cycles=zeros(size(w_ee,2),size(time,2));
coll_eff1=zeros(size(w_ee,2),size(time,2));
coll_eff2=zeros(size(w_ee,2),size(time,2));

flclear fem

% COMSOL version
clear vrsn
vrsn.name = 'COMSOL␣3.2';
vrsn.ext = '';
vrsn.major = 0;
vrsn.build = 222;
vrsn.rcs = 'Name:';
vrsn.date = 'Date:2005/09/0118:02:30';
fem.version = vrsn;

for i=1:size(w_ee,2)
    % Geometry
    w=w_ee(i)+w_gap;
    carr={curve2([-w/2,w/2],[0,0],[1,1]), ...
      curve2([w/2,w/2],[0,h],[1,1]), ...
      curve2([w/2,w_gap/2],[h,h],[1,1]), ...
      curve2([w_gap/2,w_gap/2],[h,h+h_gap],[1,1]), ...
      curve2([w_gap/2,w/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([w/2,w/2],[h+h_gap,ymax],[1,1]), ...
      curve2([w/2,-w/2],[ymax,ymax],[1,1]), ...
      curve2([-w/2,-w/2],[ymax,h+h_gap],[1,1]), ...
      curve2([-w/2,-w_gap/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([-w_gap/2,-w_gap/2],[h+h_gap,h],[1,1]), ...
      curve2([-w_gap/2,-w/2],[h,h],[1,1]), ...
      curve2([-w/2,-w/2],[h,0],[1,1])};
    g1=geomcoerce('solid',carr);
    clear s
    s.objs={g1};
    s.name={'CO1'};
    s.tags={'g1'};

    fem.draw=struct('s',s);
    fem.geom=geomcsg(fem);

    % Constants
    fem.const = {'D','7E-10', ...
      'Eg','-0.2', ...
      'Ec','0.7', ...
      'R','8.31', ...
      'T','290', ...
      'F','96000', ...
      'ks','1.5E-4', ...
      'Estd','0.265', ...
      'alpha','0.5'};

    for s=1:2
        % Initialize mesh
        %if(s==1)
            fem.mesh=meshinit(fem, ...
                         'hmaxedg',[2,2E-8,3,2E-8,9,2E-8]);
        %elseif(s==2)
        %    fem.mesh=meshinit(fem, ...
                      %  'hmaxedg',[2,1E-8,3,1E-7,9,1E-7]);
        %else
        %    error('wrong parameter for s!!');
        %end

        % (Default values are not included)

        % Application mode 1
        clear appl
        appl.mode.class = 'FlDiffusion';
        appl.dim = {'Cox'};
        appl.assignsuffix = '_di';
        clear prop
        prop.analysis='static';
        appl.prop = prop;
        clear bnd
        bnd.type = {'N0','N','C','N'};
        bnd.c0 = {0,0,'c0',0};
        bnd.N = {0,'kbc*Cre-kfc*Cox',0,'-kfg*Cox+kbg*Cre'};
        if(s==1)
            bnd.ind = [1,4,2,1,1,3,1,1,2,1,1,1];
        elseif(s==2)
            bnd.ind = [1,2,4,1,1,3,1,1,4,1,1,1];
        else
            error('wrong␣parameter␣for␣s!!');
        end
        appl.bnd = bnd;
        clear equ
        equ.D = 'D';
        equ.init = 'c0'; %TEMP!!
        equ.ind = [1];
        appl.equ = equ;
        fem.appl{1} = appl;

        % Application mode 2
        clear appl
        appl.mode.class = 'FlDiffusion';
        appl.dim = {'Cre'};
        appl.name = 'di2';
```

```
appl.assignsuffix = '_di2';
clear prop
prop.analysis='static';
clear weakconstr
weakconstr.value = 'off';
weakconstr.dim = {'lm2'};
prop.weakconstr = weakconstr;
appl.prop = prop;
clear bnd
bnd.type = {'N0','N','C','N'};
bnd.N = {0,'-kbc*Cre+kfc*Cox',0,'kfg*Cox-kbg*Cre'};
if(s==1)
    bnd.ind = [1,4,2,1,1,3,1,1,2,1,1,1];
elseif(s==2)
    bnd.ind = [1,2,4,1,1,3,1,1,4,1,1,1];
else
    error('wrong␣parameter␣for␣s!!');
end
appl.bnd = bnd;
clear equ
equ.D = 'D';
equ.init = 0;
equ.ind = [1];
appl.equ = equ;
fem.appl{2} = appl;
fem.border = 1;
fem.outform = 'general';
fem.units = 'SI';

% Subdomain expressions
clear equ
equ.ind = [1];
equ.dim = {'Cox','Cre'};
equ.expr = {'kfg','ks*exp(-alpha*F*(Eg-Estd)/(R*T))', ...
      'kbc','ks*exp((1-alpha)*F*(Ec-Estd)/(R*T))',...
      'kfc','ks*exp(-alpha*F*(Ec-Estd)/(R*T))',...
      'kbg','ks*exp((1-alpha)*F*(Eg-Estd)/(R*T))',...
      'c0',expression};

fem.equ = equ;

% Multiphysics
fem=μltiphysics(fem);

% Extend mesh
fem.xmesh=meshextend(fem);

% Solve problem
fem.sol=femtime(fem, ...
                'solcomp',{'Cox','Cre'}, ...
                'outcomp',{'Cox','Cre'}, ...
                'tlist',time, ...
                'tout','tlist');

% Save current fem structure for restart purposes
fem0=fem;
%integrate normal diffusive flux at electrode boundaries

for j=1:size(time,2)
    if(gen_movie==0)
        % Integrate
        if(s==1)
            % Integrate
            current_coll1(i,j)=F*postint(fem,'ndflux_Cox_di', ...
                   'dl',[3,9], ...
                   'edim',1, ...
                   'solnum',j);
            current_gen1(i,j)=F*postint(fem,'ndflux_Cox_di', ...
                       'dl',[2], ...
                       'edim',1, ...
                       'solnum',j);
            coll_eff1(i,j)=current_coll1(i,j)/current_gen1(i,j);
        else
            current_coll2(i,j)=F*postint(fem,'ndflux_Cox_di', ...
                   'dl',[2], ...
                   'edim',1, ...
                   'solnum',j);
            current_gen2(i,j)=F*postint(fem,'ndflux_Cox_di', ...
                       'dl',[3,9], ...
                       'edim',1, ...
                       'solnum',j);
            coll_eff2(i,j)=current_coll2(i,j)/current_gen2(i,j);
        end
        %%calculate collection efficiency and redox cycling
        %coll_eff(i,j)=-current_coll(i,j)/current_gen(i,j)*100;
        redox_cycles(i,j)=1/(1-(abs(coll_eff1(i,j)*coll_eff2(i,j))));

    else
        %plot concentration profile and save it to a frame in the movie
        H=figure(1);
        set(H ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);
        postplot(fem, ...
                 'tridata',{'Cox','cont','internal'}, ...
                 'trimap','jet(1024)', ...
                 'arrowdata',{'Coxx','Coxy'}, ...
                 'arrowxspacing',15, ...
                 'arrowyspacing',15, ...
                 'arrowtype','arrow', ...
                 'arrowstyle','proportional', ...
                 'arrowcolor',[1.0,0.0,0.0], ...
                 'solnum',j, ...
```

```
                        'title',['Time=',num2str(time(j)),';␣Surface+contour:␣Conc.,␣Cox␣[mol/m^3];␣Arrow:␣Conc.␣grad.,␣Cox␣[mol/m^4]'], ...
                        'refine',3, ...
                        'axis',[-w/2,w/2,-1E-7,ymax/5,-1,1]);
                mov(j) = getframe(H);
            end
        end
        %'contdata',{'Cox','cont','internal'}, ...
    end             %'contlevels',20, ...
    if(gen_movie==1)             %'contmap','cool(1024)', ...
        movie2avi(mov,['Z:\afstuderen\πcs\sim-idee8\concentration-profile-development-Wunit-',num2str(w_ee(i)),'.avi'],'compression','Cinepak');
    else

    end
end
%PLOT OF ELECTRODE FLUX VS. t
        H1=figure(i);
         set(H1 ,'Position',[50 50 scrsz(3)*0.8 scrsz(4)*0.8]);

        %subplot(1,2,1);
        %plot(time,-current_coll1(i,:),'color','g','marker','+');
        %hold on;
        %plot(time,-current_coll2(i,:),'color','r','marker','o');
        plot(time,current_gen1(i,:),'color','b','marker','x');
        %plot(time,current_gen2(i,:),'color',[0 0 0],'marker','v');
        ylim([0 6E-3]);
        %hold off;
        ylabel('current␣at␣the␣generator␣electrode␣(A/m)');
        grid on;
        xlabel('time␣(s)');
        %set(H1,'Marker','x');
        %set(H2,'Marker','v');
        %title(['time vs. electrode current with w_{ee}=',num2str(w_ee(i)),' m']);
        %legend('Collector current, top','Collector current, bottom','Generator current, bottom','Generator current, top','location','Southwest');
        %h = axes('Position',[0.55 0.45 0.45 0.46],'Visible','on');
        %set(gcf,'CurrentAxes',h);
        %start=38;
        %stop=65;
        %plot(time(start:stop),-current_coll1(i,start:stop),'color','b','marker','+');
        %hold on;
        %plot(time(start:stop),-current_coll2(i,start:stop),'color','r','marker','o');
        %plot(time(start:stop),current_gen1(i,start:stop),'color','g','marker','x');
        %plot(time(start:stop),current_gen2(i,start:stop),'color',[0 0 0],'marker','v');
        %xlabel('time (s)');
        %ylabel('i (A/m)');
        %xlim([time(start) time(stop)]);
        %grid on;
%         subplot(1,2,2);
%         [AX,H1,H2]=plotyy(time,coll_eff1(i,:),time,redox_cycles(i,:));
%         set(get(AX(1),'Ylabel'),'String','Collection efficiency');
%         set(get(AX(2),'Ylabel'),'String','Redox cycles');
%         ylimits1=get(AX(1),'YLim');
%         yinc1 = (ylimits1(2)-ylimits1(1))/5;
%         set(AX(1),'YTick',[ylimits1(1):yinc1:ylimits1(2)])
%         ylimits2=get(AX(2),'YLim');
%         yinc2 = (ylimits2(2)-ylimits2(1))/5;
%         grid on;
%         set(AX(2),'YTick',[ylimits2(1):yinc2:ylimits2(2)]);
%         xlabel('time (s)');
%        %set(H1,'Marker','x');
%        %set(H2,'Marker','v');
%         title(['time vs. collection efficiency and RC with w_{ee}=',num2str(w_ee(i)),' m']);
        loc=['Z:\afstuderen\πcs\sim-idee8\electrode-current-vs-t-bulkvar-Wunit-',num2str(w_ee(i)),'.jpg'];
        loc=['/home/mathieu/odie/afstuderen/πcs/sim-idee8/electrode-current-vs-t-bulkvar-Wunit-',num2str(1E6*w_ee(i)),'um.jpg'];
        saveas(H1,loc,'jpg');
    %end
```

D.7 Single cyclic voltammogram

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%Set some general variables
clear;
%dirnme='Z:\afstuderen\πcs\sim-idee8\CV-calculations\';
dirnme='/home/mathieu/afstuderen/πcs/sim-idee8/CV-calculations/';

scrsz = get(0,'ScreenSize');
F=96485;
w_total=[3 5 7];
%w_total=3;
w_total=w_total*1E-6;
w_gap=1E-6;
h=0.5E-6;
h_gap=0.5E-6;
ymax=1E-3;
%time=[0:0.01:0.1];
%t=size(time,2);
Emax=0.8;
Emin=-0.3;
Eg=0.3;
Estd=0.265;
sweeprate=0.02;
%t_end=(Emax-Emin)/sweeprate;
%time=[0:0.2:2*t_end];
t_start=2; %s
t_end=(Emax-Emin)/sweeprate;
```

```
t1=(Emax-Estd)/sweeprate;
time=[0:1:2*t_end+t_start-1];
sweep_expr=['(',num2str(Estd-0.02),'+',num2str(sweeprate),'*t+(2*(',num2str(Emax),'-',num2str(Estd),')-2*',num2str(sweeprate),'*t)*(heaviside(t-',num2
test=zeros(size(time,2),1);
for l=1:size(time,2)
    t=time(l);
    test(l)=eval(sweep_expr);
end
plot(time,test);

flux_gen_off=zeros(size(w_total,2),size(time,2));
flux_gen_on=zeros(size(w_total,2),size(time,2));
coll_eff=zeros(size(w_total,2),size(time,2));
redox_cycles=zeros(size(w_total,2),size(time,2));

C_re=zeros(size(w_total,2),size(time,2));
C_ox=zeros(size(w_total,2),size(time,2));
E_nernst=zeros(size(w_total,2),size(time,2));

flclear fem

% COMSOL version
clear vrsn
vrsn.name = 'COMSOL␣3.2';
vrsn.ext = '';
vrsn.major = 0;
vrsn.build = 222;
vrsn.rcs = 'Name:';
vrsn.date = 'Date:2005/09/0118:02:30';
fem.version = vrsn;

for i=1:size(w_total,2)
    % Geometry
    w=w_total(i);
    carr={curve2([-w/2,w/2],[0,0],[1,1]), ...
      curve2([w/2,w/2],[0,h],[1,1]), ...
      curve2([w/2,w_gap/2],[h,h],[1,1]), ...
      curve2([w_gap/2,w_gap/2],[h,h+h_gap],[1,1]), ...
      curve2([w_gap/2,w/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([w/2,w/2],[h+h_gap,ymax],[1,1]), ...
      curve2([w/2,-w/2],[ymax,ymax],[1,1]), ...
      curve2([-w/2,-w/2],[ymax,h+h_gap],[1,1]), ...
      curve2([-w/2,-w_gap/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([-w_gap/2,-w_gap/2],[h+h_gap,h],[1,1]), ...
      curve2([-w_gap/2,-w/2],[h,h],[1,1]), ...
      curve2([-w/2,-w/2],[h,0],[1,1])};
    g1=geomcoerce('solid',carr);
    clear s
    s.objs={g1};
    s.name={'CO1'};
    s.tags={'g1'};

    fem.draw=struct('s',s);
    fem.geom=geomcsg(fem);

    % Constants
    fem.const = {'c0','5', ...
      'D','7E-10', ...
      'R','8.31', ...
      'T','290', ...
      'F',F, ...
      'ks','1.5E-4', ...
      'Estd','0.265', ...
      'alpha','0.5'};

    %for s=1:2
        % Initialize mesh
        %if(s==1)
            fem.mesh=meshinit(fem, ...
                      'hmaxedg',[2,5E-9,3,5E-9,9,5E-9]);
        %elseif(s==2)
        %    fem.mesh=meshinit(fem, ...
        %              'hmaxedg',[2,1E-8,3,1E-7,9,1E-7]);
        %else
        %    error('wrong parameter for s!!');
        %end

        % (Default values are not included)

        % Application mode 1
        clear appl
        appl.mode.class = 'FlDiffusion';
        appl.dim = {'Cox'};
        appl.assignsuffix = '_di';
        clear prop
        prop.analysis='static';
        appl.prop = prop;
        clear bnd
        bnd.type = {'N0','N','C','N'};
        bnd.c0 = {0,0,'c0',0};
        bnd.N = {0,'kbc*Cre-kfc*Cox',0,'-kfg*Cox+kbg*Cre'};
        %if(s==1)
            bnd.ind = [1,1,2,1,1,3,1,1,2,1,1,1];
        %elseif(s==2)
        %    bnd.ind = [1,4,1,1,1,3,1,1,1,1,1,1];
        %else
        %    error('wrong parameter for s!!');
        %end
        appl.bnd = bnd;
        clear equ
        equ.D = 'D';
```

```
equ.init = 'c0';
equ.ind = [1];
appl.equ = equ;
fem.appl{1} = appl;

% Application mode 2
clear appl
appl.mode.class = 'FlDiffusion';
appl.dim = {'Cre'};
appl.name = 'di2';
appl.assignsuffix = '_di2';
clear prop
prop.analysis='static';
clear weakconstr
weakconstr.value = 'off';
weakconstr.dim = {'lm2'};
prop.weakconstr = weakconstr;
appl.prop = prop;
clear bnd
bnd.type = {'N0','N','C','N'};
bnd.N = {0,'-kbc*Cre+kfc*Cox',0,'kfg*Cox-kbg*Cre'};
%if(s==1)
    bnd.ind = [1,1,2,1,1,3,1,1,2,1,1,1]; %entry 2 was 4 is now 1 (insulation)
%elseif(s==2)
%    bnd.ind = [1,4,1,1,1,3,1,1,1,1,1,1];
%else
%    error('wrong parameter for s!!');
%end
appl.bnd = bnd;
clear equ
equ.D = 'D';
equ.init = 'c0';
equ.ind = [1];
appl.equ = equ;
fem.appl{2} = appl;
fem.border = 1;
fem.outform = 'general';
fem.units = 'SI';

% Subdomain expressions
clear equ
equ.ind = [1];
equ.dim = {'Cox','Cre'};
equ.expr = {'kfg','ks*exp(-alpha*F*(Eg-Estd)/(R*T))', ...
  'kfc','ks*exp(-alpha*F*(Ec-Estd)/(R*T))', ...
  'kbc','ks*exp((1-alpha)*F*(Ec-Estd)/(R*T))', ...
  'kbg','ks*exp((1-alpha)*F*(Eg-Estd)/(R*T))', ...
  'Ec',sweep_expr, ...
  'Eg',Eg};

fem.equ = equ;

% Multiphysics
fem=µltiphysics(fem);

% Extend mesh
fem.xmesh=meshextend(fem);

% Solve problem
fem.sol=femtime(fem, ...
                'solcomp',{'Cox','Cre'}, ...
                'outcomp',{'Cox','Cre'}, ...
                'tlist',time, ...
                'tout','tlist');

% Save current fem structure for restart purposes
fem0=fem;

%integrate normal diffusive flux at electrode boundaries
for j=1:size(time,2)
    t=time(j);
    E_sweep(j)=eval(sweep_expr);
    % Integrate
    flux_coll(i,j)=F*postint(fem,'ndflux_Cox_di', ...
              'dl',[3,9], ...
              'edim',1, ...
              'solnum',j);

    % Integrate
    flux_gen(i,j)=F*postint(fem,'ndflux_Cox_di', ...
              'dl',[2], ...
              'edim',1, ...
              'solnum',j);
    %calculate open circuit potential generator electrode via nernst
    C_ox(i,j)=F*postinterp(fem,'Cox',[0; 0],'Dom',[1],'solnum',j);
    C_re(i,j)=F*postinterp(fem,'Cre',[0; 0],'Dom',[1],'solnum',j);
    E_nernst(i,j)=Estd+2.303*8.31*290/F*log10(C_ox(i,j)/C_re(i,j));
    %calculate collection efficiency and redox cycling
    coll_eff(i,j)=-flux_coll(i,j)/flux_gen(i,j)*100;
    redox_cycles(i,j)=1/(1-(coll_eff(i,j)/100)^2);
end
% Plot solution
H=figure(i);
set(H ,'Position',[50 50 scrsz(3)*0.9 scrsz(4)*0.8]);
%PLOT OF ELECTRODE FLUX VS. E
%subplot(size(w_total,2),1,i);
plot(E_sweep(t_start:end),-flux_coll(i,t_start:end),'LineWidth',2);
%hold on;
%plot(E_sweep(t_start:end),-flux_gen(i,t_start:end),'color','r','LineWidth',2);
%hold off;
grid on;
```

```
        set(gca,'linewidth',2,'fontsize',14);
        xlabel('E␣(V)','FontWeight','bold','FontSize',18);
        ylabel('Current␣(A/m)','FontWeight','bold','FontSize',18);
       %text(0.1,-flux_gen(i,find(E_sweep<0.1,1)),' \leftarrow generator','FontSize',18,'color','red');
        text(0.1,-flux_coll(i,find(E_sweep<0.1,1)),'␣\leftarrow␣collector','FontSize',18,'color','blue');
        title(['CV-diagram␣top␣electrode;␣w_{ee}=',num2str((w_total(i)-w_gap)),'␣m'],'FontWeight','bold','FontSize',18);
        saveas(H,[dirnme,'electrode-current-vs-Ec-EgFixedToOCV-Wunit-',num2str(w_total(i)),'.png'],'png');
        saveas(H,[dirnme,'electrode-current-vs-Ec-EgFixedToOCV-Wunit-',num2str(w_total(i)),'.fig'],'fig');

       H=figure(i+3);
        set(H ,'Position',[50 50 scrsz(3)*0.9 scrsz(4)*0.8]);
        plot(time(t_start:end),E_nernst(i,t_start:end),'LineWidth',2);
        grid on;
        set(gca,'linewidth',2,'fontsize',14);
        xlabel('time␣(s)','FontWeight','bold','FontSize',18);
        ylabel('E_{nernst}␣(V)','FontWeight','bold','FontSize',18);
        title(['Nernst␣potential␣at␣(x=0,y=0);␣w_{ee}=',num2str((w_total(i)-w_gap)),'␣m'],'FontWeight','bold','FontSize',18);
        saveas(H,[dirnme,'OCV-generator-Wunit-',num2str(w_total(i)),'.png'],'png');
        saveas(H,[dirnme,'OCV-generator-Wunit-',num2str(w_total(i)),'.fig'],'fig');
    %end
end
save([dirnme,'electrode-current-vs-Ec-EgFixedToOCV-data.mat']);

% %save data to latex table
% clear columnLabels;
% columnLabels= {'W_total p/unit','Flux gen. | coll. on [v](mol/(m*s))','Flux gen. | coll. off [x](mol/(m*s))','CE [o](\%)','RC [+]'};
% clear matrix;
% matrix(:,1)=w_total;
% matrix(:,2)=-flux_gen_on(1,:);
% matrix(:,3)=flux_gen_off(1,:);
% matrix(:,4)=coll_eff(1,:);
% matrix(:,5)=redox_cycles(1,:);
%
% matrix2latex(matrix, 'Z:/afstuderen/eindverslag/data/w-total-study.tex', ...
%     'columnLabels',columnLabels, 'alignment', 'c', 'format', '%10.2e', ...
%     'size', 'tiny','caption', ...
%     'Collection efficiency with different edge-to-edge distances between the holes', ...
%     'label','tbl:ce-vs-w-total');
```

D.8 Double cyclic voltammogram with two work electrodes

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%Set some general variables
clear;
%dirnme='Z:\afstuderen\πcs\sim-idee8\CV-calculations\';
dirnme='/home/mathieu/afstuderen/πcs/sim-idee8/CV-calculations/';

scrsz = get(0,'ScreenSize');
F=96485;
w_total=[3 5 7];
%w_total=3;
w_total=w_total*1E-6;
w_gap=1E-6;
h=0.5E-6;
h_gap=0.5E-6;
ymax=1E-3;
%time=[0:0.01:0.1];
%t=size(time,2);
Emax=0.8;
Emin=-0.3;
Eg=0.3;
Estd=0.265;
sweeprate=0.02;
%t_end=(Emax-Emin)/sweeprate;
%time=[0:0.2:2*t_end];
t_start=5; %s
t_end=(Emax-Emin)/sweeprate;
t1=(Emax-Estd)/sweeprate;
time=[0:0.25:2*t_end+1];
sweep_expr=['(',num2str(Estd-0.02),'+',num2str(sweeprate),'*t+(2*(',num2str(Emax),'-',num2str(Estd),')-2*',num2str(sweeprate),'*t)*(heaviside(t-',num2
test=zeros(size(time,2),1);
for l=1:size(time,2)
    t=time(l);
    test(l)=eval(sweep_expr);
end
plot(time,test);

flux_gen_off=zeros(size(w_total,2),size(time,2));
flux_gen_on=zeros(size(w_total,2),size(time,2));
coll_eff=zeros(size(w_total,2),size(time,2));
redox_cycles=zeros(size(w_total,2),size(time,2));

flclear fem

% COMSOL version
clear vrsn
vrsn.name = 'COMSOL␣3.2';
vrsn.ext = '';
vrsn.major = 0;
vrsn.build = 222;
vrsn.rcs = 'Name:';
vrsn.date = 'Date: 2005/09/0118:02:30';
fem.version = vrsn;
```

```
for i=1:size(w_total,2)
    % Geometry
    w=w_total(i);
    carr={curve2([-w/2,w/2],[0,0],[1,1]), ...
      curve2([w/2,w/2],[0,h],[1,1]), ...
      curve2([w/2,w_gap/2],[h,h],[1,1]), ...
      curve2([w_gap/2,w_gap/2],[h,h+h_gap],[1,1]), ...
      curve2([w_gap/2,w/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([w/2,w/2],[h+h_gap,ymax],[1,1]), ...
      curve2([w/2,-w/2],[ymax,ymax],[1,1]), ...
      curve2([-w/2,-w/2],[ymax,h+h_gap],[1,1]), ...
      curve2([-w/2,-w_gap/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([-w_gap/2,-w_gap/2],[h+h_gap,h],[1,1]), ...
      curve2([-w_gap/2,-w/2],[h,h],[1,1]), ...
      curve2([-w/2,-w/2],[h,0],[1,1])};
    g1=geomcoerce('solid',carr);
    clear s
    s.objs={g1};
    s.name={'CO1'};
    s.tags={'g1'};

    fem.draw=struct('s',s);
    fem.geom=geomcsg(fem);

    % Constants
    fem.const = {'c0','5', ...
      'D','7E-10', ...
      'R','8.31', ...
      'T','290', ...
      'F',F, ...
      'ks','1.5E-4', ...
      'Estd','0.265', ...
      'alpha','0.5'};

    %for s=1:2
        % Initialize mesh
        %if(s==1)
            fem.mesh=meshinit(fem, ...
                         'hmaxedg',[2,5E-9,3,5E-9,9,5E-9]);
        %elseif(s==2)
        %    fem.mesh=meshinit(fem, ...
        %                 'hmaxedg',[2,1E-8,3,1E-7,9,1E-7]);
        %else
        %    error('wrong parameter for s!!');
        %end

        % (Default values are not included)

        % Application mode 1
        clear appl
        appl.mode.class = 'FlDiffusion';
        appl.dim = {'Cox'};
        appl.assignsuffix = '_di';
        clear prop
        prop.analysis='static';
        appl.prop = prop;
        clear bnd
        bnd.type = {'N0','N','C','N'};
        bnd.c0 = {0,0,'c0',0};
        bnd.N = {0,'kbc*Cre-kfc*Cox',0,'-kfg*Cox+kbg*Cre'};
        %if(s==1)
            bnd.ind = [1,4,2,1,1,3,1,1,2,1,1,1];
        %elseif(s==2)
        %    bnd.ind = [1,4,1,1,1,3,1,1,1,1,1,1];
        %else
        %    error('wrong parameter for s!!');
        %end
        appl.bnd = bnd;
        clear equ
        equ.D = 'D';
        equ.init = 'c0';
        equ.ind = [1];
        appl.equ = equ;
        fem.appl{1} = appl;

        % Application mode 2
        clear appl
        appl.mode.class = 'FlDiffusion';
        appl.dim = {'Cre'};
        appl.name = 'di2';
        appl.assignsuffix = '_di2';
        clear prop
        prop.analysis='static';
        clear weakconstr
        weakconstr.value = 'off';
        weakconstr.dim = {'lm2'};
        prop.weakconstr = weakconstr;
        appl.prop = prop;
        clear bnd
        bnd.type = {'N0','N','C','N'};
        bnd.N = {0,'-kbc*Cre+kfc*Cox',0,'kfg*Cox-kbg*Cre'};
        %if(s==1)
            bnd.ind = [1,4,2,1,1,3,1,1,2,1,1,1];
        %elseif(s==2)
        %    bnd.ind = [1,4,1,1,1,3,1,1,1,1,1,1];
        %else
        %    error('wrong parameter for s!!');
        %end
        appl.bnd = bnd;
        clear equ
        equ.D = 'D';
```

```
        equ.init = 'c0';
        equ.ind = [1];
        appl.equ = equ;
        fem.appl{2} = appl;
        fem.border = 1;
        fem.outform = 'general';
        fem.units = 'SI';

        % Subdomain expressions
        clear equ
        equ.ind = [1];
        equ.dim = {'Cox','Cre'};
        equ.expr = {'kfg','ks*exp(-alpha*F*(Eg-Estd)/(R*T))', ...
          'kfc','ks*exp(-alpha*F*(Ec-Estd)/(R*T))', ...
          'kbc','ks*exp((1-alpha)*F*(Ec-Estd)/(R*T))', ...
          'kbg','ks*exp((1-alpha)*F*(Eg-Estd)/(R*T))', ...
          'Ec',sweep_expr, ...
          'Eg',sweep_expr};

        fem.equ = equ;

        % Multiphysics
        fem=μltiphysics(fem);

        % Extend mesh
        fem.xmesh=meshextend(fem);

        % Solve problem
        fem.sol=femtime(fem, ...
                        'solcomp',{'Cox','Cre'}, ...
                        'outcomp',{'Cox','Cre'}, ...
                        'tlist',time, ...
                        'tout','tlist');

        % Save current fem structure for restart purposes
        fem0=fem;

        %integrate normal diffusive flux at electrode boundaries
        for j=1:size(time,2)
             t=time(j);
             E_sweep(j)=eval(sweep_expr);
             % Integrate
             flux_coll(i,j)=F*postint(fem,'ndflux_Cox_di', ...
                           'dl',[3,9], ...
                           'edim',1, ...
                           'solnum',j);

             % Integrate
             flux_gen(i,j)=F*postint(fem,'ndflux_Cox_di', ...
                           'dl',[2], ...
                           'edim',1, ...
                           'solnum',j);
             %calculate collection efficiency and redox cycling
             coll_eff(i,j)=-flux_coll(i,j)/flux_gen(i,j)*100;
             redox_cycles(i,j)=1/(1-(coll_eff(i,j)/100)^2);
        end
        % Plot solution
        H=figure(i);
        set(H ,'Position',[50 50 scrsz(3)*0.9 scrsz(4)*0.8]);
        %PLOT OF ELECTRODE FLUX VS. E
        %subplot(size(w_total,2),1,i);
        plot(E_sweep(t_start:end),-flux_coll(i,t_start:end),'LineWidth',2);
        hold on;
        plot(E_sweep(t_start:end),-flux_gen(i,t_start:end),'color','r','LineWidth',2);
        hold off;
        grid on;
        set(gca,'linewidth',2,'fontsize',14);
        xlabel('E␣(V)','FontWeight','bold','FontSize',18);
        ylabel('Current␣(A/m)','FontWeight','bold','FontSize',18);
        text(0.2,-flux_gen(i,find(E_sweep<0.2,1)),'␣\leftarrow␣generator','FontSize',18,'color','red');
        text(0.2,-flux_coll(i,find(E_sweep<0.2,1)),'␣\leftarrow␣collector','FontSize',18,'color','blue');
        title(['Siμltaneous␣sweep␣CV;␣w_{ee}=',num2str((w_total(i)-w_gap)),'␣m'],'FontWeight','bold','FontSize',18);
        saveas(H,[dirnme,'electrode-current-vs-Ec-Eg-Wunit-',num2str(w_total(i)),'.png'],'png');
        saveas(H,[dirnme,'electrode-current-vs-Ec-Eg-Wunit-',num2str(w_total(i)),'.fig'],'fig');

    %end
end
save([dirnme,'electrode-current-vs-Ec-Eg-data.mat']);

% %save data to latex table
% clear columnLabels;
% columnLabels= {'$W_{total}$ p/unit','Flux gen. | coll. on [v](mol/(m*s))','Flux gen. | coll. off [x](mol/(m*s))','CE [o](\%)','RC [+]'};
% clear matrix;
% matrix(:,1)=w_total;
% matrix(:,2)=-flux_gen_on(1,:);
% matrix(:,3)=flux_gen_off(1,:);
% matrix(:,4)=coll_eff(1,:);
% matrix(:,5)=redox_cycles(1,:);
%
% matrix2latex(matrix, 'Z:/afstuderen/eindverslag/data/w-total-study.tex',...
%     'columnLabels',columnLabels, 'alignment', 'c', 'format', '%10.2e',...
%     'size', 'tiny','caption',...
%     'Collection efficiency with different edge-to-edge distances between the holes',...
%     'label','tbl:ce-vs-w-total');
```

D.9 Single cyclic voltammogram with second fixed-potential work electrode

```
% COMSOL Multiphysics Model M-file
% Generated by COMSOL 3.2 and heavily modified by M. Odijk

%Set some general variables
clear;
%dirnme='Z:\afstuderen\πcs\sim-idee8\CV-calculations\';
dirnme='/home/mathieu/afstuderen/πcs/sim-idee8/CV-calculations/';

scrsz = get(0,'ScreenSize');
F=96485;
w_total=[3 5 7];
%w_total=3;
w_total=w_total*1E-6;
w_gap=1E-6;
h=0.5E-6;
h_gap=0.5E-6;
ymax=1E-3;
%time=[0:0.01:0.1];
%t=size(time,2);
Emax=0.8;
Emin=-0.3;
Eg=-0.2;
Estd=0.265;
sweeprate=0.02;
%t_end=(Emax-Emin)/sweeprate;
%time=[0:0.2:2*t_end];
t_start=2; %s
t_end=(Emax-Emin)/sweeprate;
t1=(Emax-Estd)/sweeprate;
time=[0:1:2*t_end+t_start-1];
sweep_expr=['(',num2str(Estd-0.02),'+',num2str(sweeprate),'*t+(2*(',num2str(Emax),'-',num2str(Estd),')-2*',num2str(sweeprate),'*t)*(heaviside(t-',num2
test=zeros(size(time,2),1);
for l=1:size(time,2)
    t=time(l);
    test(l)=eval(sweep_expr);
end
plot(time,test);

flux_gen_off=zeros(size(w_total,2),size(time,2));
flux_gen_on=zeros(size(w_total,2),size(time,2));
coll_eff=zeros(size(w_total,2),size(time,2));
redox_cycles=zeros(size(w_total,2),size(time,2));
E_sweep=zeros(1,size(time,2));

flclear fem

% COMSOL version
clear vrsn
vrsn.name = 'COMSOL␣3.2';
vrsn.ext = '';
vrsn.major = 0;
vrsn.build = 222;
vrsn.rcs = 'Name:';
vrsn.date = 'Date:2005/09/0118:02:30';
fem.version = vrsn;

for i=1:size(w_total,2)
    % Geometry
    w=w_total(i);
    carr={curve2([-w/2,w/2],[0,0],[1,1]), ...
      curve2([w/2,w/2],[0,h],[1,1]), ...
      curve2([w/2,w_gap/2],[h,h],[1,1]), ...
      curve2([w_gap/2,w_gap/2],[h,h+h_gap],[1,1]), ...
      curve2([w_gap/2,w/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([w/2,w/2],[h+h_gap,ymax],[1,1]), ...
      curve2([w/2,-w/2],[ymax,ymax],[1,1]), ...
      curve2([-w/2,-w/2],[ymax,h+h_gap],[1,1]), ...
      curve2([-w/2,-w_gap/2],[h+h_gap,h+h_gap],[1,1]), ...
      curve2([-w_gap/2,-w_gap/2],[h+h_gap,h],[1,1]), ...
      curve2([-w_gap/2,-w/2],[h,h],[1,1]), ...
      curve2([-w/2,-w/2],[h,0],[1,1])};
    g1=geomcoerce('solid',carr);
    clear s
    s.objs={g1};
    s.name={'CO1'};
    s.tags={'g1'};

    fem.draw=struct('s',s);
    fem.geom=geomcsg(fem);

    % Constants
    fem.const = {'c0','5', ...
      'D','7E-10', ...
      'R','8.31', ...
      'T','290', ...
      'F',F, ...
      'ks','1.5E-4', ...
      'Estd','0.265', ...
      'alpha','0.5'};

    %for s=1:2
        % Initialize mesh
        %if(s==1)
            fem.mesh=meshinit(fem, ...
                      'hmaxedg',[2,5E-9,3,5E-9,9,5E-9]);
```

```
%elseif(s==2)
%    fem.mesh=meshinit(fem, ...
%               'hmaxedg',[2,1E-8,3,1E-7,9,1E-7]);
%else
%    error('wrong parameter for s!!');
%end

% (Default values are not included)

% Application mode 1
clear appl
appl.mode.class = 'FlDiffusion';
appl.dim = {'Cox'};
appl.assignsuffix = '_di';
clear prop
prop.analysis='static';
appl.prop = prop;
clear bnd
bnd.type = {'N0','N','C','N'};
bnd.c0 = {0,0,'c0',0};
bnd.N = {0,'kbc*Cre-kfc*Cox',0,'-kfg*Cox+kbg*Cre'};
%if(s==1)
    bnd.ind = [1,4,2,1,1,3,1,1,2,1,1,1];
%elseif(s==2)
%    bnd.ind = [1,4,1,1,1,3,1,1,1,1,1,1];
%else
%    error('wrong parameter for s!!');
%end
appl.bnd = bnd;
clear equ
equ.D = 'D';
equ.init = 'c0';
equ.ind = [1];
appl.equ = equ;
fem.appl{1} = appl;

% Application mode 2
clear appl
appl.mode.class = 'FlDiffusion';
appl.dim = {'Cre'};
appl.name = 'di2';
appl.assignsuffix = '_di2';
clear prop
prop.analysis='static';
clear weakconstr
weakconstr.value = 'off';
weakconstr.dim = {'lm2'};
prop.weakconstr = weakconstr;
appl.prop = prop;
clear bnd
bnd.type = {'N0','N','C','N'};
bnd.N = {0,'-kbc*Cre+kfc*Cox',0,'kfg*Cox-kbg*Cre'};
%if(s==1)
    bnd.ind = [1,4,2,1,1,3,1,1,2,1,1,1];
%elseif(s==2)
%    bnd.ind = [1,4,1,1,1,3,1,1,1,1,1,1];
%else
%    error('wrong parameter for s!!');
%end
appl.bnd = bnd;
clear equ
equ.D = 'D';
equ.init = 'c0';
equ.ind = [1];
appl.equ = equ;
fem.appl{2} = appl;
fem.border = 1;
fem.outform = 'general';
fem.units = 'SI';

% Subdomain expressions
clear equ
equ.ind = [1];
equ.dim = {'Cox','Cre'};
equ.expr = {'kfg','ks*exp(-alpha*F*(Eg-Estd)/(R*T))', ...
  'kfc','ks*exp(-alpha*F*(Ec-Estd)/(R*T))', ...
  'kbc','ks*exp((1-alpha)*F*(Ec-Estd)/(R*T))', ...
  'kbg','ks*exp((1-alpha)*F*(Eg-Estd)/(R*T))', ...
  'Ec',sweep_expr, ...
  'Eg',Eg};

fem.equ = equ;

% Multiphysics
fem=µltiphysics(fem);

% Extend mesh
fem.xmesh=meshextend(fem);

% Solve problem
fem.sol=femtime(fem, ...
                'solcomp',{'Cox','Cre'}, ...
                'outcomp',{'Cox','Cre'}, ...
                'tlist',time, ...
                'tout','tlist');

% Save current fem structure for restart purposes
fem0=fem;

%integrate normal diffusive flux at electrode boundaries
for j=1:size(time,2)
```

```
            t=time(j);
            E_sweep(j)=eval(sweep_expr);
            % Integrate
            flux_coll(i,j)=F*postint(fem,'ndflux_Cox_di', ...
                        'dl',[3,9], ...
                        'edim',1, ...
                        'solnum',j);

            % Integrate
            flux_gen(i,j)=F*postint(fem,'ndflux_Cox_di', ...
                        'dl',[2], ...
                        'edim',1, ...
                        'solnum',j);
            %

            %calculate collection efficiency and redox cycling
            coll_eff(i,j)=-flux_coll(i,j)/flux_gen(i,j)*100;
            redox_cycles(i,j)=1/(1-(coll_eff(i,j)/100)^2);
        end
        % Plot solution
        H=figure(i);
        set(H ,'Position',[50 50 scrsz(3)*0.9 scrsz(4)*0.8]);
        %PLOT OF ELECTRODE FLUX VS. E
        %subplot(size(w_total,2),1,i);
        plot(E_sweep(t_start:end),-flux_coll(i,t_start:end),'LineWidth',2);
        hold on;
        plot(E_sweep(t_start:end),-flux_gen(i,t_start:end),'color','r','LineWidth',2);
        hold off;
        grid on;
        set(gca,'linewidth',2,'fontsize',14);
        xlabel('E␣(V)','FontWeight','bold','FontSize',18);
        ylabel('Current␣(A/m)','FontWeight','bold','FontSize',18);
        text(0.2,-flux_gen(i,find(E_sweep<0.2,1)),'␣\leftarrow␣generator','FontSize',18,'color','red');
        text(0.2,-flux_coll(i,find(E_sweep<0.2,1)),'␣\leftarrow␣collector','FontSize',18,'color','blue');
        title(['CV␣top␣electrode␣with␣E_{bottom}=',num2str(Eg),'V;␣w_{ee}=',num2str((w_total(i)-w_gap)),'␣m'],'FontWeight','bold','FontSize',18);
        saveas(H,[dirnme,'electrode-current-vs-Ec-EgFixedTo',num2str(Eg),'-Wunit-',num2str(w_total(i)),'.png'],'png');
        saveas(H,[dirnme,'electrode-current-vs-Ec-EgFixedTo',num2str(Eg),'-Wunit-',num2str(w_total(i)),'.fig'],'fig');

    %end
end
save([dirnme,'electrode-current-vs-Ec-EgFixedTo',num2str(Eg),'-data.mat']);

% %save data to latex table
% clear columnLabels;
% columnLabels= {'$W_{total}$ p/unit','Flux gen. | coll. on [v](mol/(m*s))','Flux gen. | coll. off [x](mol/(m*s))','CE [o](\%)','RC [+]'};
% clear matrix;
% matrix(:,1)=w_total;
% matrix(:,2)=-flux_gen_on(1,:);
% matrix(:,3)=flux_gen_off(1,:);
% matrix(:,4)=coll_eff(1,:);
% matrix(:,5)=redox_cycles(1,:);
%
% matrix2latex(matrix, 'Z:/afstuderen/eindverslag/data/w-total-study.tex',...
%     'columnLabels',columnLabels, 'alignment', 'c', 'format', '%10.2e',...
%     'size', 'tiny','caption',...
%     'Collection efficiency with different edge-to-edge distances between the holes',...
%     'label','tbl:ce-vs-w-total');
```

D.10 General functions

Matrix to latex table converter

```
function matrix2latex(matrix, filename, varargin)

% function: matrix2latex(...)
% Author:   M. Koehler
% Contact:  koehler@in.tum.de
% Version:  1.1
% Date:     May 09, 2004

% This software is published under the GNU GPL, by the free software
% foundation. For further reading see: http://www.gnu.org/licenses/licenses.html#GPL

% Usage:
% matrix2late(matrix, filename, varargs)
% where
%   - matrix is a 2 dimensional numerical or cell array
%   - filename is a valid filename, in which the resulting latex code will
%   be stored
%   - varargs is one ore more of the following (denominator, value) combinations
%      + 'rowLabels', array -> Can be used to label the rows of the
%      resulting latex table
%      + 'columnLabels', array -> Can be used to label the columns of the
%      resulting latex table
%      + 'alignment', 'value' -> Can be used to specify the alignment of
%      the table within the latex document. Valid arguments are: 'l', 'c',
%      and 'r' for left, center, and right, respectively
%      + 'format', 'value' -> Can be used to format the input data. 'value'
%      has to be a valid format string, similar to the ones used in
%      fprintf('format', value);
%      + 'size', 'value' -> One of latex' recognized font-sizes, e.g. tiny,
%      HUGE, Large, large, LARGE, etc.
%
% Example input:
%   matrix = [1.5 1.764; 3.523 0.2];
%   rowLabels = {'row 1', 'row 2'};
%   columnLabels = {'col 1', 'col 2'};
%   matrix2latex(matrix, 'out.tex', 'rowLabels', rowLabels, 'columnLabels', columnLabels, 'alignment', 'c', 'format', '%-6.2f', 'size', 'tiny');
%
% The resulting latex file can be included into any latex document by:
% /input{out.tex}
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% NOTES BY M. Odijk:
% I've altered this code a bit. It now adds \table, \topcaption, \label and
% \centering tags. The topcaption tag is dependent on topcapt.sty, which
% can be downloaded from: ftp://cam.ctan.org/tex-archive/macros/latex/contrib/misc/topcapt.sty
%
%
% Enjoy life!!!

    rowLabels = [];
    colLabels = [];
    alignment = 'l';
    format = [];
    textsize = [];
    caption = [];
    label = [];
    if (rem(nargin,2) == 1 || nargin < 2)
        error('matrix2latex:␣', 'Incorrect␣number␣of␣arguments␣to␣%s.', mfilename);
    end

    okargs = {'rowlabels','columnlabels', 'alignment', 'format', 'size','caption','label'};
    for j=1:2:(nargin-2)
        pname = varargin{j};
        pval = varargin{j+1};
        k = strmatch(lower(pname), okargs);
        if isempty(k)
            error('matrix2latex:␣', 'Unknown␣parameter␣name:␣%s.', pname);
        elseif length(k)>1
            error('matrix2latex:␣', 'Ambiguous␣parameter␣name:␣%s.', pname);
        else
            switch(k)
                case 1  % rowlabels
                    rowLabels = pval;
                    if isnumeric(rowLabels)
                        rowLabels = cellstr(num2str(rowLabels(:)));
                    end
                case 2  % column labels
                    colLabels = pval;
                    if isnumeric(colLabels)
                        colLabels = cellstr(num2str(colLabels(:)));
                    end
                case 3  % alignment
                    alignment = lower(pval);
                    if alignment == 'right'
                        alignment = 'r';
                    end
                    if alignment == 'left'
                        alignment = 'l';
                    end
                    if alignment == 'center'
                        alignment = 'c';
                    end
                    if alignment ≠ 'l' && alignment ≠ 'c' && alignment ≠ 'r'
                        alignment = 'l';
```

```
                        warning('matrix2latex:␣', 'Unkown␣alignment.␣(Set␣it␣to␣\''left\''.)');
                    end
                case 4  % format
                    format = lower(pval);
                case 5  % textsize
                    textsize = pval;
                case 6  % caption
                    %display('yes!');
                    caption = pval;
                case 7  % label
                    %display('no!');
                    label = pval;
            end
        end
    end

    fid = fopen(filename, 'w');

    width = size(matrix, 2);
    height = size(matrix, 1);

    if isnumeric(matrix)
        matrix = num2cell(matrix);
        for h=1:height
            for w=1:width
                if(¬isempty(format))
                    matrix{h, w} = num2str(matrix{h, w}, format);
                else
                    matrix{h, w} = num2str(matrix{h, w});
                end
            end
        end
    end
    fprintf(fid,'\\begin{table∗}[h]\r\n');
    fprintf(fid,'\\centering\r\n');
    if(¬isempty(caption))
       fprintf(fid, '\\topcaption{%s}\r\n', caption);
    end
    if(¬isempty(textsize))
        fprintf(fid, '\\begin{%s}', textsize);
    end

    fprintf(fid, '\\begin{tabular}{|');

    if(¬isempty(rowLabels))
        fprintf(fid, 'l|');
    end
    for i=1:width
        fprintf(fid, '%c|', alignment);
    end
    fprintf(fid, '}\r\n');

    fprintf(fid, '\\hline\r\n');

    if(¬isempty(colLabels))
        if(¬isempty(rowLabels))
            fprintf(fid, '&');
        end
        for w=1:width−1
            fprintf(fid, '\\textbf{%s}&', colLabels{w});
        end
        fprintf(fid, '\\textbf{%s}\\\\\\hline\r\n', colLabels{width});
    end

    for h=1:height
        if(¬isempty(rowLabels))
            fprintf(fid, '\\textbf{%s}&', rowLabels{h});
        end
        for w=1:width−1
            fprintf(fid, '%s&', matrix{h, w});
        end
        fprintf(fid, '%s\\\\\\hline\r\n', matrix{h, width});
    end
    fprintf(fid, '\\end{tabular}\r\n');

    if(¬isempty(textsize))
        fprintf(fid, '\\end{%s}\r\n', textsize);
    end
    if(¬isempty(label))
        fprintf(fid, '\\label{%s}\r\n', label);%
    end
    fprintf(fid, '\\end{table∗}\r\n');
    fclose(fid);
```

Simple distributed computing script (set job)

```
%set Job
function setJob(functionExpression,jobNr,jobId)
    jobdir='Z:\afstuderen\scripts\jobs'; %always leave \ or / at the end
    jobname=['\job−',num2str(jobId),'−',num2str(jobNr),'−'];
    command=functionExpression;
    save([jobdir,jobname],'command');
```

Simple distributed computing script (combine job)

```
function res=combineJob(jobId,maxJobNr,help,old)
    display('Start␣job␣combiner..');
    jobdir='Z:\afstuderen\scripts\jobs'; %always leave \ or / at the end
    jobResDir='Z:\afstuderen\scripts\jobresults';%always leave \ or / at the end
    finished=0;
    res=struct([]);
    interval=10;
    %put some polling here to check if all jobs are πcked up by clients...
    while(finished==0)
        keep_looπng=0;
        jobs=dir(jobdir);
        for i=1:size(jobs,1)
            work_done=0;
            name=jobs(i).name;
            fullname1=[jobdir,'\',name];
            fullname2=[jobResDir,'\',name];
            if(isdir(fullname1)==0)%check if it is a file
                [id,nr]=getIdNr(name);
                if(id==jobId)
                    keep_looπng=1;
                    if(help==1 & work_done==0) %help, but only if polling just took place!
                        clear command;
                        display('start␣helπng␣with␣executing␣job..');
                        load(fullname1,'-mat');
                        delete(fullname1);
                        jobres=eval(command);
                        save(fullname2,'jobres');
                        work_done=1;
                    end
                end
            end
        end
        if(keep_looπng==0)
            finished=1;
            %break;
        else
            pause(interval);
        end
    end
    if(finished==1) %no more scheduled jobs
        display('All␣jobs␣in␣the␣que␣are␣taken,␣continuing␣with␣combining␣jobs');
        %check if all results are already present
        check=ones(1,maxJobNr);
        all_results_back=0;
        max_wait=20*60; %equal to x minutes
        wait_time=0;
        while(all_results_back==0)
            res_jobs=dir(jobResDir);
            for i=1:size(res_jobs,1)
                res_name=res_jobs(i).name;
                res_fullname=[jobResDir,'\',res_name];
                if(isdir(res_fullname)==0)%check if it is a file
                    [id,nr]=getIdNr(res_name);
                    if(id==jobId)
                        check(nr)=0;
                    end
                end
            end
            if(size(find(check),2)==0)
                all_results_back=1;%stop looking for results
            else
                display('Not␣all␣jobs␣finished␣by␣clients.␣Going␣on␣wait␣now..');
                pause(interval);
                wait_time=wait_time+interval;
                if(wait_time>max_wait)
                    display('Max␣wait␣time␣exceeded!␣Maybe␣a␣client␣failed␣a␣job!');
                    all_results_back=1;%stop looking for results
                end
            end
        end
        %combine results and put it in a struct
        display('Now␣combining␣results..please␣wait␣just␣a␣little␣longer');
        for i=1:maxJobNr
            jobres=0;
            if(old==0)
                Fullname=[jobResDir,'\job-',num2str(jobId),'-',num2str(i),'-.mat'];
            else
                Fullname=[jobResDir,'\job-',num2str(jobId),'-',num2str(i),'-old.mat'];
            end
            display('Fullname');
            jobres=load(Fullname);
            res(i).result=jobres;
            if(old==0)
                copyfile([jobResDir,'\job-',num2str(jobId),'-',num2str(i),'-.mat'],[jobResDir,'\job-',num2str(jobId),'-',num2str(i),'-old.mat']);
                delete([jobResDir,'\job-',num2str(jobId),'-',num2str(i),'-.mat']);
            end
        end
    end
end
```

Simple distributed computing script (poll job)

```
%poll job
clear;

global jobdir;
jobdir='Z:\afstuderen\scripts\jobs'; %always leave \ or / at the end
global jobResDir;
jobResDir='Z:\afstuderen\scripts\jobresults';%always leave \ or / at the end
interval=30; %pollinterval in seconds
counter=0;

while(1);
    display(['start␣polling␣again..(',num2str(counter),')']);
    counter=counter+1;
    jobs=dir(jobdir);
    work_done=0;
    for i=1:size(jobs,1)
        expression='';
        name=jobs(i).name;
        fullname1=[jobdir,'\',name];
        fullname2=[jobResDir,'\',name];
        if(isdir(fullname1)==0 & work_done==0)
            clear command;
            display('start␣executing␣job..');
            load(fullname1,'-mat');
            delete(fullname1);
            jobres=eval(command);
            work_done=1;
            save(fullname2,'jobres');
        end
    end
    drawnow; %draw pending grafic processes
    pause(interval);
end
%save([jobdir,'\job00101'],'command');
%command='3+3';
```

D.11 Measurement process scripts

Main measurement process file

```
%mainMeasurements This is a script to process measurements
%
% Author:
%   v.2.1. Mathieu Odijk − 24−11−2006
%
%
%   See also GETDATA
%
%   type doc('mainmeasurements') to display this help in the helpbrowser


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
%% The adjustable settings are displayed here below:
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear;

%dirnme='Z:\afstuderen\measurements\SINsensors\electrode1\';
%dirnme='Z:\afstuderen\measurements\electrode2—G8—W8−900s\';
%dirnme='Z:\afstuderen\measurements\electrode3—G4—W8−600s\';
%dirnme='/home/mathieu/afstuderen/measurements/SiNsensors/electrode1/';
%dirnme='/home/mathieu/afstuderen/measurements/SiNsensors/electrode4—G8—W32−600s/';
%dirnme='/home/mathieu/afstuderen/measurements/SiNsensors/electrode5—G4—W32−300s/';
%dirnme='Z:\afstuderen\measurements\SiNsensors\electrode4—G8—W32−600s\';
%dirnme='Z:\afstuderen\measurements\tests\test\';
%dirnme='Z:\afstuderen\measurements\PIsensors\G2—W32—ronde2\';

%dirnme='/home/mathieu/afstuderen/measurements/PIsensors/G2—W32−047/';
%dirnme='/home/mathieu/afstuderen/measurements/PIsensors/G8—w32−047/';

dirnme='Z:\afstuderen\measurements\PIsensors\G8—W32—047−1uur\';
%dirnme='/home/mathieu/afstuderen/measurements/PIsensors/G8—W32—047−1uur/';
%plot_cycles=[2 12];
%show_log=1;

filterSize=1;
%offset=685;
offset=570;
%blocks=[660 500 100 600 700];
blocks=[700 500 100 600 600];
windowSize=1;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
%% In principle no editing below this line is necessary.
%% MAIN PROGRAM STARTS HERE
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
scrsz = get(0,'ScreenSize');


%content=getData(dirnme); %retrieve all data from HD
%save([dirnme,'matlab−data.mat']);
%display('Loading of data finished!');

% optional load data
old_dirnme=dirnme;
load([dirnme,'matlab−data.mat']);
dirnme=old_dirnme;

%% process data
fig=1;
for i=1:size(content,1) %TEMP CHANGED!!
    for j=1:size(content,2)
        if(isempty(content(i,j,1).name))
            display(['NOT␣working␣on␣',num2str(i),',',num2str(j)]);
        else
            display(['Working␣on␣',num2str(i),',',num2str(j)]);
           H=figure(fig);
            set(H ,'Position',[50 50 scrsz(3)∗0.9 scrsz(4)∗0.85]);

            if(strcmp(content(i,j,1).type,'CV') || strcmp(content(i,j,3).type,'CV'))
                plotCV(content(i,j,:),filterSize);
            end
            if(strcmp(content(i,j,1).type,'CA') & strcmp(content(i,j,3).type,'CA'))
                plotCA(content(i,j,:),filterSize,blocks,offset,0);
            end
           AX=axes('Position',[0 0 1 1],'Visible','off');
            set(gcf,'CurrentAxes',AX,'paperpos',[0 0 32 32∗scrsz(4)/scrsz(3)]);
            text(0.2,0.98,[dirnme,content(i,j,1).name],'Fontsize',16,'FontWeight','bold');
           %print(['−f',num2str(H)], '−dpng', [dirname,content(i,j,1).name,'_',num2str(j)]);
            display(['Saving:␣',dirnme,content(i,j,1).name,'_0',num2str(j)]);
            saveas(H,[dirnme,content(i,j,1).name,'_0',num2str(j)],'png');
            saveas(H,[dirnme,content(i,j,1).name,'_0',num2str(j)],'fig');
            fig=fig+1;
        end
    end
end

%% Figure window closer
for i=1:4
   H=figure(i);
    close(H);
end
% CUSTOM graph generator for the final report
```

```
filename='Z:\afstuderen\πcs\measurements\test';
%filename='/home/mathieu/afstuderen/πcs/measurements/meas-sens1-1500s';
%grtitle='Measurements on the bottom electrode of a W_g=8\μ m, W_{ee}=32\μ m sensor before etching.';
%define colors
color(1,:)=[0 0 1];
color(2,:)=[1 0 0];
color(3,:)=[0 0.8 0.8];
color(4,:)=[1 0 1];

windowSize=5;
%start graph

%start first subplot
%subplot(1,2,1);
H=figure(1);
set(H ,'Position',[50 50 scrsz(3)*0.9 scrsz(4)*0.8]);
clear meas;
clear data_ch;
l=1;
k=1;
meas=content(1,1,:); %(1,1,:) %(2,3,:)
data_ch=meas(l).data;
A=plot(data_ch(:,8),data_ch(:,9),'color',color(k,:),'linewidth',2);
title(['CV:␣Ch',num2str(l)],'FontWeight','bold','FontSize',24);
xlabel(char(meas(l).fields(8)),'FontWeight','bold','FontSize',24);
ylabel(char(meas(l).fields(9)),'FontWeight','bold','FontSize',24);
set(gca,'linewidth',2,'FontSize',24);
grid('on');
AX=axes('Position',[0 0 1 1],'Visible','off');
set(gcf,'CurrentAxes',AX,'paperpos',[0 0 32 32*scrsz(4)/scrsz(3)]);
saveas(H,[filename,'-CV-ch',num2str(l)],'png');
saveas(H,[filename,'-CV-ch',num2str(l)],'fig');

%second subplot
%subplot(1,2,2);
H=figure(2);
set(H ,'Position',[50 50 scrsz(3)*0.9 scrsz(4)*0.8]);
clear meas;
clear data_ch;
l=3;
k=2;
meas=content(1,2,:); %(1,2,:)
data_ch=meas(l).data;
A=plot(data_ch(:,8),data_ch(:,9),'color',color(k,:),'linewidth',2);
title(['CV:␣Ch',num2str(l)],'FontWeight','bold','FontSize',24);
xlabel(char(meas(l).fields(8)),'FontWeight','bold','FontSize',24);
ylabel(char(meas(l).fields(9)),'FontWeight','bold','FontSize',24);
set(gca,'linewidth',2,'FontSize',24);
grid('on');
AX=axes('Position',[0 0 1 1],'Visible','off');
set(gcf,'CurrentAxes',AX,'paperpos',[0 0 32 32*scrsz(4)/scrsz(3)]);
saveas(H,[filename,'-CV-ch',num2str(l)],'png');
saveas(H,[filename,'-CV-ch',num2str(l)],'fig');

%third subplot
%subplot(3,2,3:4);
clear meas;
clear data_ch1;
clear data_ch2;
meas=content(1,3,:);
data_ch1=meas(1).data;
data_ch2=meas(3).data;%(943:end,:);
%for i=1:size(data_ch2,1)
%    if(data_ch1(i,10)<-0.02)
%        data_ch1(i,10)=-0.02;
%    end
%end
for i=0:1
    H=figure(3+i);
    set(H ,'Position',[50 50 scrsz(3)*0.9 scrsz(4)*0.8]);

    %plot data
    AX=axes('Position',[0.1 0.1 0.75 0.8],'Visible','off');
    set(gca,'linewidth',2,'FontSize',24);
    plot(data_ch1(:,7),data_ch1(:,9+i),'color',color(1+i*2,:),'linewidth',2);
    hold on;
    plot(data_ch2(:,7),data_ch2(:,9+i),'color',color(2+i*2,:),'linewidth',2);
    hold off;
    xlabel(char(meas(1).fields(7)),'FontWeight','bold','FontSize',24);
    ylabel(char(meas(1).fields(9+i)),'FontWeight','bold','FontSize',24);
    if(i==0)
        title(['RC␣measurement:␣Potential␣Ch1␣+␣Ch3'],'FontWeight','bold','FontSize',24);
    else
        title(['RC␣measurement:␣Current␣Ch1␣+␣Ch3'],'FontWeight','bold','FontSize',24);
    end
    set(gca,'linewidth',2,'FontSize',24);
    grid('on');
    AX=axes('Position',[0 0 1 1],'Visible','off');
    set(gcf,'CurrentAxes',AX,'paperpos',[0 0 32 32*scrsz(4)/scrsz(3)]);
    if(i==0)
        saveas(H,[filename,'-RC-E'],'png');
        saveas(H,[filename,'-RC-E'],'fig');
    else
        saveas(H,[filename,'-RC-I'],'png');
        saveas(H,[filename,'-RC-I'],'fig');
    end
end
```

Get data files from harddrive

```
function [returncont]=getData(dirname)
%getData script to read files from a specified directory given by dirname.
%  It returns a structure containing each file, it's data and filename along
% some other variables.
%
%   function [returncont]=getData(dirname)
%
%   Input arguments:
%        dirname − the directory the data files are in. (Full name is recommended)
%
%   Output arguments:
%        returncont − structure with the contents of the directory.
%
%   Author:
%        M. Odijk − 24−11−2006
%
%   See also MAINMEASUREMENTS
%
%   Type doc('getData') to display this help in the helpbrowser

%initialize some vars
clear returncont;
returncont=struct([]);
item=1;
%first check if the directory exist
if exist(dirname)==7,
    content=dir(dirname);%retrieve contents of the dir.
    for z=1:size(content,1) %cycle through the files
        filename=getfield(content(z),'name'); %retrieve filename
        if getfield(content(z),'isdir')==0, %if it is not a directory, do...
            %find decimal point to check filetype letter positions
            finding=find(filename=='.');
            type='.abc'; %set dummy type string of equal length
            %get real filetype with a for loop.
            j=1;
            for i=finding:size(filename,2)
                type(j)=filename(i);
                j=j+1;
            end
            %get filename without type and find _ positions.
            if(strcmp(type,'.mpt'))
                splitpos=find(filename=='_');
                if(size(splitpos,2)≠3)
                    warning('A␣file␣with␣wrong␣name␣found.␣Not␣enough␣splitchars␣found!');
                else
                    %fetch experiment name
                    realname=filename(1:splitpos(1)−1);
                    %fetch experiment nr
                    expnr=str2num(filename(splitpos(1)+1:splitpos(2)−1));
                    %fetch experiment type
                    exptype=filename(splitpos(2)+1:splitpos(3)−1);
                    %fetch experiment channel
                    channel=str2num(filename(splitpos(3)+1:finding−1));
                    %open file
                    display([dirname,filename]);
                    fid = fopen([dirname,filename]); %get a filepointer
                    %if(exist(dirname)
                    %open data file, read first line for number of rows in
                    %the file.
                    fgetl(fid);
                    headerlines=fgetl(fid);

                    num=[headerlines(19),headerlines(20),headerlines(21),headerlines(22)];

                    rows=str2num(num)−2;
                    %get file description (part of log file)
                    description = java_array('java.lang.String', rows);
                    if(rows>0)
                        for i=1:rows
                            tline = fgetl(fid);
                            description(i)=java.lang.String(tline);
                            if ¬ischar(tline),   break,   end
                            %disp(tline)
                        end
                    end
                    last=size(description,1);
                    remain=char(description(last));
                    %read the actual data fields (columns)
                    f=1;
                    clear fields;
                    fields(1)=java.lang.String('');
                    while true
                       [str, remain] = strtok(remain,char(9));
                       if isempty(str),  break;  end
                       fields(f)=java.lang.String(str);
                       f=f+1;
                       %disp(sprintf('%s', str));
                    end
                    %prepare a scanstring to read the data from the lines,
                    %%f used as a format. (floating point?)
                    scanstring='';
                    for j=1:f−1;
                        scanstring=[scanstring,'%f'];
                    end


                    %close data file
                    %fclose(fid);%read data with textscan function
```

```
                    tline = fgetl(fid);
                    data=cell2mat(textscan(strrep(tline,',','.'),scanstring,'delimiter',char(9)));
                    while 1
                        tline = fgetl(fid);
                        if ¬ischar(tline),   break,   end
                        data=cat(1,data,cell2mat(textscan(strrep(tline,',','.'),scanstring,'delimiter',char(9))));
                        %disp(tline)
                    end

                    fclose(fid);
                    %open logfile
                    filename(end)='l';
                    fid = fopen([dirname,filename]); %get a filepointer to logfile
                    j=1;
                    clear log;
                    log{1}='log';
                    while true
                        tline = fgetl(fid);
                        log{j,1}=tline;
                        j=j+1;
                        if ¬ischar(tline),   break,   end
                        %disp(tline)
                    end
                    %close logfile
                    fclose(fid);

                    found=0;
                    for n=1:item−1
                        if(size(returncont,1)>0)
                            %display(['n=',num2str(n)]);
                            %display(returncont(1,1,1).name);
                            if(strcmp(returncont(n,1,1).name,realname))
                                found=n;
                            end
                        end
                    end
                    if found>0

                        %save all data in the right place.
                        returncont(found,expnr,channel).name = realname;
                        returncont(found,expnr,1).name = realname;
                        returncont(found,expnr,channel).fields=fields;
                        returncont(found,expnr,channel).channel=channel;
                        returncont(found,expnr,channel).data = data;
                        returncont(found,expnr,channel).description = cell(description);
                        returncont(found,expnr,channel).log = log;
                        returncont(found,expnr,channel).type =exptype;
                    else
                        returncont(item,1,1).name = realname; %this is needed for the found search
                        returncont(item,expnr,channel).name = realname;
                        returncont(item,expnr,channel).fields=fields;
                        returncont(item,expnr,channel).channel=channel;
                        returncont(item,expnr,channel).data = data;
                        returncont(item,expnr,channel).description = cell(description);
                        returncont(item,expnr,channel).log = log;
                        returncont(item,expnr,channel).type =exptype;
                        item=item+1;
                    end
                end
            end
        end
    end
else
    display('This␣is␣not␣a␣valid␣directory!');
end
```

Plot CV diagrams

```
function plotCV(meas,windowSize)
color(1,:)=[0 0 1];
color(2,:)=[1 0 0];
color(3,:)=[0 0.8 0.8];
color(4,:)=[1 0 1];


for k=1:2
    if(k==2)
        l=3;
    else
        l=1;
    end
    if(strcmp(meas(l).type,'CV'))
        data_ch=meas(l).data;
        if(strcmp(meas(1).type,'CV') & strcmp(meas(3).type,'CV'))
            subplot(2,2,k);
        else
            subplot(2,2,1:2);
        end
        A=plot(data_ch(:,8),filter2(ones(1,windowSize)/windowSize,data_ch(:,9)),'color',color(k,:),'linewidth',2);
        title(['CV:␣Ch',num2str(l)],'FontWeight','bold','FontSize',14);
        xlabel(char(meas(l).fields(8)),'FontWeight','bold','FontSize',14);
        ylabel(char(meas(l).fields(9)),'FontWeight','bold','FontSize',14);
        set(gca,'linewidth',2,'FontSize',14);
        grid('on');

        subplot(2,2,k+2);
        [AX,H1,H2]=plotyy(data_ch(:,6),filter2(ones(1,windowSize)/windowSize,data_ch(:,8)),data_ch(:,6),data_ch(:,9));
        title(['CV:␣E␣and␣I␣Ch',num2str(l)],'FontWeight','bold','FontSize',14);
        xlabel(char(meas(l).fields(6)));
```

```
        set(get(AX(1),'Ylabel'),'String',char(meas(1).fields(8)),'FontWeight','bold','FontSize',14);
        set(get(AX(2),'Ylabel'),'String',char(meas(1).fields(9)),'FontWeight','bold','FontSize',14);
        set(gca,'linewidth',2);
        set(get(AX(1),'Xlabel'),'FontWeight','bold','FontSize',14);
        set(get(AX(2),'Xlabel'),'FontWeight','bold','FontSize',14);
        set(AX(1),'FontSize',14,'linewidth',2,'Ycolor',color(k,:));
        set(AX(2),'FontSize',14,'linewidth',2,'Ycolor',color(k+2,:));
        set(H1,'color',color(k,:),'linewidth',2);
        set(H2,'color',color(k+2,:),'linewidth',2);
        grid('on');
    end
    if(strcmp(meas(1).type,'MP') || strcmp(meas(1).type,'CA'))
        data_ch=meas(1).data;
        %subplot(2,2,k);
        %plot(data_ch(:,9),data_ch(:,10));
        %xlabel(char(meas(1).fields(9)));
        %ylabel(char(meas(1).fields(10)));
        %title(['CV channel ',num2str(1)]);
        subplot(2,2,k+2);
        set(gca,'linewidth',2,'fontsize',14);
        [AX,H1,H2]=plotyy(data_ch(:,7),filter2(ones(1,windowSize)/windowSize,data_ch(:,9)),data_ch(:,7),filter2(ones(1,windowSize)/windowSize,data_ch(:
        xlabel(char(meas(1).fields(7)),'FontWeight','bold','FontSize',14);
        set(get(AX(1),'Ylabel'),'String',char(meas(1).fields(9)),'FontWeight','bold','FontSize',14);
        set(get(AX(2),'Ylabel'),'String',char(meas(1).fields(10)),'FontWeight','bold','FontSize',14);
        set(get(AX(1),'Xlabel'),'FontWeight','bold','FontSize',14);
        set(get(AX(2),'Xlabel'),'FontWeight','bold','FontSize',14);
        set(AX(1),'FontSize',14,'Ycolor',color(k,:));
        set(AX(2),'FontSize',14,'Ycolor',color(k+2,:));
        set(H1,'color',color(k,:),'linewidth',2);
        set(H2,'color',color(k+2,:),'linewidth',2);
        title([meas(1).type,':␣E␣and␣I␣Ch',num2str(1)],'FontWeight','bold','FontSize',14);
        set(gca,'linewidth',2,'fontsize',14);
        grid('on');
    end
end
```

Plot RC diagrams and calculate RC values

```
function plotCA(meas,windowSize,blocks,offset,calcRC)
color(1,:)=[0 0 1];
color(2,:)=[1 0 0];
color(3,:)=[0 0.8 0.8];
color(4,:)=[1 0 1];

filtSize=15;

%% Plot results
for k=1:2
    if(k==2)
        l=3;
    else
        l=1;
    end
    subplot(2,1,k);
    %RC-calculation script:
    if(calcRC==1)
        data_ch=meas(1).data;%get data
        data_ch(:,10)=filter(ones(1,filtSize)/filtSize,1,data_ch(:,10)); %filter the I data



        t=offset;
        %plot block markings and RC values
        hold on;
        for i=1:size(blocks,2)
            RC=0;
            %find t range to calculate RC for. the first and last 100
            %datapoints are rejected.
            range=[find(data_ch(:,7)>t+50,1) find(data_ch(:,7)>t+blocks(i)-50,1)];
            timeperiod=[data_ch(range(1),7) data_ch(range(2),7)];
            %determine on which channel the RC can be found. If the
            %difference is small, it is assumed this channel is kept at
            %constant value.
            MinMax=max(data_ch(range(1):range(2),9)) - min(data_ch(range(1):range(2),9));
            if(MinMax<0.2) %this is the one!
                %calculate redox amplification on the filtered channel
                RC=max(abs(data_ch(range(1):range(2),10)))/min(abs(data_ch(range(1):range(2),10)));
                %draw RC value as text on chart.
                text(t,abs(max(data_ch(range(1):range(2),9))*1.1),['RC=',num2str(RC,'%6.2f'),'x'],'color',[0 0.8 0],'FontWeight','bold','FontSize',14);
                %calc maxiµm and miniµm value to plot on chart later.
%                   maxvalue=max(data_ch(range(1):range(2),10));
%                   minvalue=min(data_ch(range(1):range(2),10));
%                   plot(timeperiod, [minvalue minvalue],'color',[0 0.9 0],'linewidth',2,'linestyle','--');
%                   plot(timeperiod, [maxvalue maxvalue],'color',[0 0.9 0],'linewidth',2,'linestyle','--');
%
            end
            t=t+blocks(i);
            %plot the green bars to indicate the blocks
            plot([t t],[min(data_ch(:,9))*1.1 max(data_ch(:,9))*1.1],'color',[0 0.8 0],'linewidth',3);
            % calculate RC-factors and display them in the figure
        end
    end

    data_ch=meas(1).data;
    %plot data
    set(gca,'linewidth',2,'fontsize',14);
    [AX,H1,H2]=plotyy(data_ch(:,7),filter2(ones(1,windowSize)/windowSize,data_ch(:,9)),data_ch(:,7),filter2(ones(1,windowSize)/windowSize,data_ch(:,10)
    xlabel(char(meas(1).fields(7)),'FontWeight','bold','FontSize',14);
    set(get(AX(1),'Ylabel'),'String',char(meas(1).fields(9)));
```

```
    set(get(AX(2),'Ylabel'),'String',char(meas(1).fields(10)));
    set(get(AX(1),'Xlabel'),'FontWeight','bold','FontSize',14);
    set(get(AX(2),'Xlabel'),'FontWeight','bold','FontSize',14);
    set(AX(1),'FontSize',14,'YColor',color(k,:));
    set(AX(2),'FontSize',14,'YColor',color(k+2,:));
    set(H1,'color',color(k,:),'linewidth',2);
    set(H2,'color',color(k+2,:),'linewidth',2);
    title([meas(1).type,':␣E␣and␣I␣Ch',num2str(1)],'FontWeight','bold','FontSize',14);
    set(gca,'linewidth',2,'fontsize',14);
    grid('on');


    if(calcRC==1)
        t=offset;
        data_ch=meas(1).data;%get data
        data_ch(:,10)=filter(ones(1,filtSize)/filtSize,1,data_ch(:,10)); %filter the I data

        for i=1:size(blocks,2)
          %find t range to calculate RC for. the first and last 90
          %s are rejected.
          range=[find(data_ch(:,7)>t+50,1) find(data_ch(:,7)>t+blocks(i)-50,1)];
          timeperiod=[data_ch(range(1),7) data_ch(range(2),7)];
          MinMax=max(data_ch(range(1):range(2),9)) - min(data_ch(range(1):range(2),9));
          if(MinMax<0.2) %this is the one!
              %calc maxiμm and miniμm value to plot on chart later.
              maxvalue=max(data_ch(range(1):range(2),10));
              minvalue=min(data_ch(range(1):range(2),10));
              line(timeperiod, [minvalue minvalue],'color',[0 0.9 0],'linewidth',2,'linestyle','--','Parent',AX(2));
              line(timeperiod, [maxvalue maxvalue],'color',[0 0.9 0],'linewidth',2,'linestyle','--','Parent',AX(2));
          end
          t=t+blocks(i);
        end
     end
    hold off;

end
```