DEGREE OF DOCTOR OF PHILOSOPHY IN
ELECTRONICS AND TELECOMMUNICATIONS

DOCTORATE SCHOOL IN
INFORMATION AND COMMUNICATION
TECHNOLOGIES

XXV CYCLE

UNIVERSITY OF MODENA AND REGGIO EMILIA

DEPARTMENT OF ENGINEERING "Enzo Ferrari"

Ph.D. DISSERTATION

SENSORS FOR BIOMEDICAL AND AGRI-FOOD INDUSTRY: THEORETICAL AND EXPERIMENTAL STUDIES

Candidate:
LUCA FERRARI

Advisor:
Prof. LUIGI ROVATI

The Director of the School:
Prof. GIORGIO MATTEO VITETTA

The Coordinator of the School:
Prof. GIORGIO MATTEO VITETTA

DOTTORATO DI RICERCA IN
ELECTRONICS AND TELECOMMUNICATIONS

SCUOLA DI DOTTORATO IN
INFORMATION AND COMMUNICATION
TECHNOLOGIES

XXV CICLO

UNIVERSITA' DI MODENA E REGGIO EMILIA

DIPARTIMENTO DI INGEGNERIA "Enzo Ferrari"

TESI PER IL CONSEGUIMENTO DEL TITOLO DI DOTTORE DI RICERCA

SENSORI PER L'INDUSTRIA BIOMEDICALE E AGROALIMENTARE: STUDI TEORICI E SPERIMENTALI

Candidato:
LUCA FERRARI

Relatore:
Prof. LUIGI ROVATI

Il Direttore della Scuola:
Prof. GIORGIO MATTEO VITETTA

Il Coordinatore della Scuola:
Prof. GIORGIO MATTEO VITETTA

Contents

Abstract

Nowadays, sensors are becoming more and more important in life sciences. Both in biomedical fields where the need of a continuous monitoring of body parameters is a desired characteristic and in food industry, where checking the quality of the rough materials is now a fundamental requirement. Nevertheless, different environments require specific sensing techniques. The present thesis focuses on the study, design and realization of different kinds of sensors for the measurement of important parameters in life sciences.

Firstly, this work deals with the realization of an optochemical sensor for biomedical applications, in particular to measure the pH and pCO_2 of blood during extracorporeal circulation. The developed sensor consists of a disposable low-cost polymer sensing element, the interrogation optical head and the front-end and control electronics. The pH-sensitive dye is a new kind of fluorescent monomer, the fluorescein O-methacrylate, which is able to covalently link to the polymer matrix, in order to minimize the dye leaching. Several tests have been performed; the sensor has been firstly tested on water, then in-vitro on cow blood and finally ex-vivo on a sheep and a pig. The sensor showed a linear behavior in the physiological range, making it suitable for biomedical applications.

Secondly, a combination of optics and electrochemistry will be presented, with the integration of the SPR (Surface Plasmon Resonance) and electrochemical techniques on the same sensing chip. The system is based on a four-channel SPR sensor system developed at the Institute of Photonics and Electronics AS CR, Prague. Since the SPR system provides four independent channels, an in-situ electrochemical cell has been realized in each channel, allowing four simultaneous electrochemical measurements. The gold substrate used for SPR measurement was used as the working electrode and two further planar gold electrodes (the reference and the counter electrodes) were inserted in the same flow cell. Together with the realization of the electrochemical cells, a custom electronics has been designed and realized. The electronics is able to set an arbitrary potential between the reference and working electrodes and to measure the current flowing between the counter and the working electrodes. The electrochemical SPR biosensor was used to study DNA desorption

and regeneration. It has been shown that desired probe density can be achieved by optimizing the potential and that potential can be used for regeneration of the SPR chip.

Finally, two sensors for the food industry will be presented, deploying both the electrochemical and optical techniques. An electrochemical sensor has been designed to estimate the concentration of glycerol in the grape juice during the grape harvest, in order to check the grapes quality. Together with the sensing element, an automatic fluidic system has been developed, using membrane micropumps. Finally, the sensor has been tested during the grape harvest, on different kinds of grapes. An optical sensor, instead, has been developed to estimate the concentration of antibiotics in milk. The sensor deploys the absorbance changes of an organic molecule in the presence of the antibiotics. Thus, the system consists of an optical head to interrogate the solution under test and a front-end and processing electronics to acquire the information of interest. The system has been tested both with buffer solutions and milk samples.

Introduction

Sensors in biomedical and agri-food industry play a fundamental role in modern life.

Biomedical sensors are routinely used in clinical medicine and biological research for measuring a wide range of physiological variables [1, 2, 3]. Moreover, system designs have been researched and developed to view real-time physiological data with remote access for clinical review in instances such as telemedicine, home-care, patient transport, as well as in hospital observation [4, 5, 6, 7, 8]. Among biomedical sensors, bioanalytic sensors incorporate biologic recognition reactions such as enzyme-substrate, antigen-antibody, or ligand-receptor to identify complex biochemical molecules. The use of biologic reactions gives bioanalytic sensors high sensitivity and specificity in identifying and quantifying biochemical substances [9].

At the same time, agri-food sensors are requested to control and certify food safety, quality and convenience. Thus, it has been necessary to characterize the biological and environmental processes in agri-food industry, relying on sensor technology. For examples, sensors are used to optimize the irrigation [10] and pest control programs [11], to measure the yield per unit area [12], to control the environmental parameters during storage and finally for on-line monitoring of processing parameters. Nevertheless, surrounding environment as well as processed agricultural materials are inherently complex. Presence of multiple microorganisms and other biological agents further makes the sensing of parameters in agri-food industry very challenging. As the agri-food processes are highly variable, the sensors require the capability to handle this variability.

Similar sensing techniques can be adopted to solve the sensing issue both in biomedical and agri-food industry. Both fields, in fact, are characterized by a great variety and a complex environment which usually makes the disposability of the sensing element a fundamental requirement. Risk of contamination between different samples and sterilization of the sensing element are typical characteristics of biomedical and agri-food sensors. For this reason, a non-contact measurement is usually a preferred approach in these application fields.

Moreover, in order to perform a real-time monitoring of important biological parameters, both in biomedical exams and in quality control tests, a fast response

sensor is a desired requirement.

Different environments can be faced with various sensing techniques; electrochemical and optical techniques are the most common ones.

Electrochemical methods are based on monitoring the changes of different parameters (e.g. potential or current), caused by oxidization or reduction of electroactive species on the working electrode surface as a function of the type or concentration of the target analyte. Electrochemical sensors usually belong to four main categories: potentiometric, amperometric, conductometric and impedimetric devices. With respect to optical sensors, electrochemical sensors do not require dark environments for measurements, there is no photobleaching of indicator molecules and they have a bigger dynamic range. Electrochemical sensors have been used in food analyses (e.g. measuring the freshness of food [13], the presence of genetically modified organisms [14] and analyzing food and beverages [15]) and in medical applications (e.g. drug response measurement [16], detection of pathogens [17, 18] and cancer diagnoses [19]).

Absorbance, reflectance, scattering and different types of luminescence are used in different kinds of optical sensors [20]. Optical sensors are a very powerful detection and analysis tool used in many different fields (e.g., healthcare [21, 22, 23, 24, 25, 26, 27], pharmaceuticals[28, 29] and environmental monitoring [30, 31]). With respect to electrochemical sensors, optical sensors may be much more easily miniaturized and used in in-vivo applications [32]. Moreover, they are not affected by electro-magnetic inteferences [33] and they are able to monitor more than one analyte simultaneously [34].

In addition, both electrochemical and optical methods allow real-time, sensitive and sometimes label-free detection of target analytes.

Since electrochemical and optical methods demonstrate advantages in different, yet sometimes overlapping applications, a combination of these two methodologies will permit a richer set of data in addition to the benefit of increased control over the sensing environment [35]. They both have different dynamic ranges that facilitate a more accurate measurement of the concentration of the target. Furthermore, these methods could provide a built-in control to each other.

The present thesis deals with the study and development of different sensors for biomedical and agri-food industry. The variety of the applications environment together with the specific requirements of each sensor, has led to the use of different sensing approaches (optical, electrochemical). In particular, in Chapter 1 a fluorescence based multi-analyte sensor platform is described, together with two experimental applications: the measurerement of pH and pCO_2 in extracorporeal circulation. In Chapter 2, a bioanalytical sensor is reported, combining the SPR and electrochemical techniques, to study DNA desorption and regeneration. Then,

in Chapter 3 and Chapter 4 two different sensors for agri-food industry will be presented, deploying an electrochemical and optical approach respectively. Finally conclusions are drawn.

Part I

Biomedical sensing

Chapter 1

Multi analyte fluorescence-based sensor platform

1.1 Introduction

Nowadays fluorescence phenomenon is widely exploited to realize optical sensors, due to the higher sensitivity and versatility with respect to other detection schemes. In particular, fluorescein is widely used for its high molar absorptivity at the wavelength of the argon laser ($\lambda = 488\ nm$), large fluorescence quantum yield and pH-dependent emission spectra [36].

Several optical fluorescence-based sensors consist of a solid matrix, including the sensitive dye, which can be reached by the analyte. Chemical bonding between the dye and the matrix is necessary to avoid undesirable leaching effects. Measurements are performed revealing reversible variations in the intensity or lifetime of the fluorescent indicator in the presence of the analyte; polymers are widely used as immobilization matrices for fluorescent dyes, thanks to the possibility of molecular tailoring to control and tune protons absorption [37, 38, 39, 40, 41]

Thanks to this approach, it is possible to design a multi-analyte sensor platform, by using the same electronics and optical interrogation head, changing the polymer sensing element including the sensitive dye.

pH (Latin: pondus hidrogenii) and CO_2 are commonly measured parameters of great interest in several application fields, such as environmental monitoring [42, 43, 44, 45], bioprocessing [46] and biomedical diagnostics [47, 48].

In particular, haematic pH and pCO_2 are ones of the most important blood parameters. In fact, they are strongly related to the functioning of important organs, such as lung, heart, liver and kidney. The haematic pH, in healthy human beings, is always between 7.30 and 7.45, while pCO_2 is always between 20 and 50 mmHg, thanks to the bicarbonate buffering system. Nevertheless, a little variation of pH or

pCO_2 can produce great damages for the body. Several efforts have been spent on the development of pH and pCO_2 sensors to be applied in Extracorporeal Circulation (ECC).

ECC is mainly applied in cardiopulmonary bypass, extracorporeal membrane oxygenation, and hemodialysis. Usually, during ECC, blood samples are drawn intermittently and analyzed by a blood gas analyzer. Major problems associated with this intermittent sampling are: (i) high cost of the blood gas analyzer and required reagents thus analyses are often performed only after adverse events, (ii) there is a risk of blood loss and infection, and (iii) therapeutic response can only be made after a delay. Considerable spontaneous variability in blood gases and other quickly-changing analytes that occurs during ECC require clinical decisions made on the basis of trends observed with continuous monitoring. Usually, the information of interest includes hemoglobin saturation, hematocrit, blood gases (O_2, CO_2), pH, electrolytes (Na^+, K^+, Ca^{2+}, Cl_2), certain metabolites (glucose, lactate, urea, creatinine) and physical measurands (temperature, pressure, flow). Several sensors capable of continuously monitoring these parameters have been so far proposed [49, 50]. Nevertheless, despite tremendous technological efforts, the proposed technologies are not yet sufficiently reliable and economic to be used routinely in ECC and frequent intermittent measurements made by a blood gas analyzer near the bedside are still the standard practice. Since ECC through flow cuvettes offers a perfect optical access to circulating blood, several optical approaches have been considered [21, 22, 23, 24, 25, 26, 27] and few commercial optical continuous intravascular blood gas monitoring devices are nowadays available: Paratrend 7+ (PT7+) for adults and Neotrend (NT) for newborns, both based on optical absorption and reflectance methods [51]. However, they have been rarely used due to their instability and inaccuracies [52, 53].

The measurement of pH is routinely performed using the glass electrode, nevertheless the electrochemical approach suffers from many drawbacks, such as electromagnetic interference, difficulty in miniaturisation and limitations when measuring aqueous suspensions of organic matter or low-ionic-strength solutions. Also continuous blood pH measurements can be made potentiometrically via ion-selective electrodes or transistors, but these devices are based on ionophores that may leach out of the ion-selective membranes, thus resulting lower stability and shorter lifetime [54]. This stability issue limits the use of ion-selective electrodes or transistors in ECC. Attempts to use electrochemical sensors were hampered by protein deposition on the active or reference electrodes that prevents their use unless regular recalibration [50]. While promising, none of these methods are suited to continuous monitoring in ECC. With respect to electrochemical sensors, optical pH sensors allow for a higher sensitivity and selectivity, due to the luminescence phenomena.

Moreover, they are insensitive to electromagnetic interference, they do not need any reference electrode and they are suitable for remote and disposable sensing, which is typical of in-situ applications [55, 56].

The most common devices for CO_2 determination are the Severinghaus electrode [57] and the infrared detector [58]. Although infrared absorption detectors are characterized by fast response times (0-90% in < 0.1 s), they are bulky, expensive and only applicable to gaseous CO_2. The Severinghaus type CO_2 electrode, instead, consists of a pH electrode in contact with a thin layer of bicarbonate buffer solution and the whole system is encapsulated by a thin, gas-permeable membrane. The working principle is based on the diffusion of the CO_2 in the sample under test through the gas permeable membrane, which determines a variation in the pH of the internal aqueous solution. Thus, the pH variation is monitored by the pH electrode. Nevertheless, the Severinghaus type electrode suffers from a long response time (5-15 minutes) and from the same drawbacks of the pH electrode [59].

In the last years, there has been considerable interest in optical methods of determining both pH and pCO_2. In literature, proposed pH sensors are mainly based on the immobilization of pH-sensitive absorbing dyes [60, 61, 62] or fluorophores [63, 64] on a solid substrate. One of the major limitation of such a sensor is still the dye leaching into blood [65], that, besides problems related to signal stability, can represent a risk for the patient, if the dye is toxic. A further disadvantage of the sensors based on fluorophores is their propensity for photobleaching [65]. For all these reasons, while many pH optical sensors have been proposed for continuous pH monitoring, in-vivo results have not been reported so far.

As far as pCO_2 sensors are concerned two types of optical sensors have been proposed to overcome the Severinghaus type electrode. The first type replaces the pH electrode with a pH-sensitive dye [66]; this kind of sensor is based on the reaction of CO_2 with a bicarbonate buffer solution placed behind a hydrophobic polymer. The pH change caused by the CO_2 is transduced in a variation of absorbance or fluorescence intensity of the dye. However, this kind of sensor has a long response time, due to the hydrophobic membrane. In addition, the membrane is impermeable to ions and liquid water, but not to water vapors. In this way, there could be a gradual drift due to the establishment of an osmotic equilibrium between the internal buffer and the sample solution. The second type of optical sensor uses a phase transfer agent (a quaternary ammonium hydroxide) [67] to replace the previously required sodium bicarbonate buffer solution. In fact, the quaternary ammonium hydroxide provides in the hydrophobic membrane the initial basic environment that is essential for a CO_2-sensing film to work. Since the hydrophobic membrane acts as a barrier for ions, no further gas-permeable membrane is required, thus this kind of sensors is called “naked sensors”.

Nevertheless, although lots of pH and pCO_2 optical sensors are commercially available at present, only few of them are suitable for the fabrication of disposable sensing elements [62]. In fact, the key aspect of a disposable sensor is a cheap and simple transduction mechanism, which converts the information of interest into a readable signal, so that it is possible to use the same optical and electronic instrumentation for different measurements changing only the cheap and disposable sensing element. Disposability is becoming a fundamental characteristic in life science sensors, where exams performed on different patients require always new and sterilized sensing elements in contact with biological tissues.

In this chapter a multi-analyte fluorescence-based sensor platform will be presented. In the next Section a description of the optical transducer, flow cuvette and the electronic circuitry is given. Then, two applications will be described. In particular, in Section 1.3 a polymer pH-sensitive membrane for measuring haematic pH will be presented, together with its characterization both in-vitro and ex-vivo. Then, a polymer pCO_2-sensitive membrane for measuring haematic pCO_2 will be described in Section 1.4.

1.2 Materials and Methods

In the next sections the developed multi-analyte fluorescence-based measurement system is presented. It basically consists of four parts: (i) polymer sensing element in contact with the solution (including the dye sensitive to the analyte of interest), (ii) optical reading head and (iii) front-end and signal elaboration electronics. As schematically shown in Figure 1.1, the optical transduction of the quantity of interest is based on the fluorescence intensity emitted by a polymer film directly in contact with the solution flowing inside the cuvette.

1.2.1 Polymer sensing element

The polymer sensing element, in contact with the solution, transduces the quantity of interest into an optical information; it consists of a polymer matrix and a fluorescent sensitive dye bonded to the polymer chains. The polymer matrix plays an important role because it is in direct contact with the solution and it contains the sensitive dye inside its molecular structure, so it has to be: (i) robust towards the flow, (ii) able to guarantee a fast penetration of the analyte of interest, (iii) fast in response time, (iv) well adherent to the substrate of the measuring cell, (v) characterized by absent or reduced dye leaching.

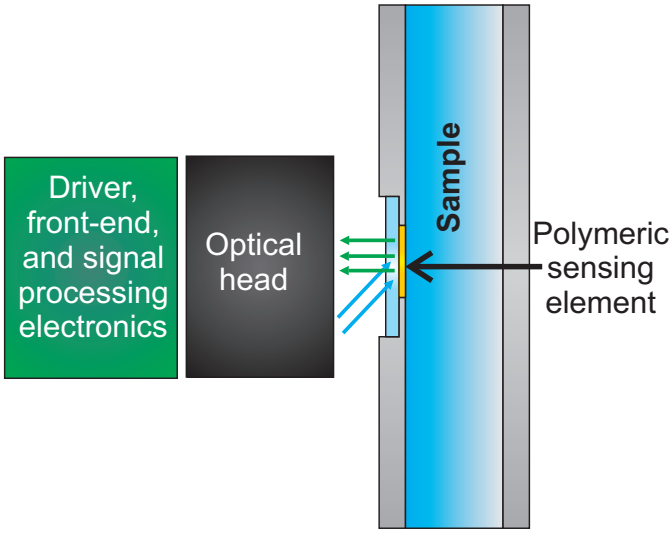


Figure 1.1: Schematic representation of the measurement principle. A polymer sensing matrix, placed in the flow cuvette directly in contact with the solution under test, is interrogated by the optical head properly driven by the electronic circuitry.

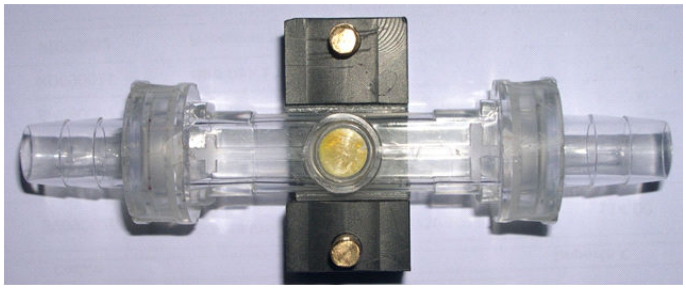
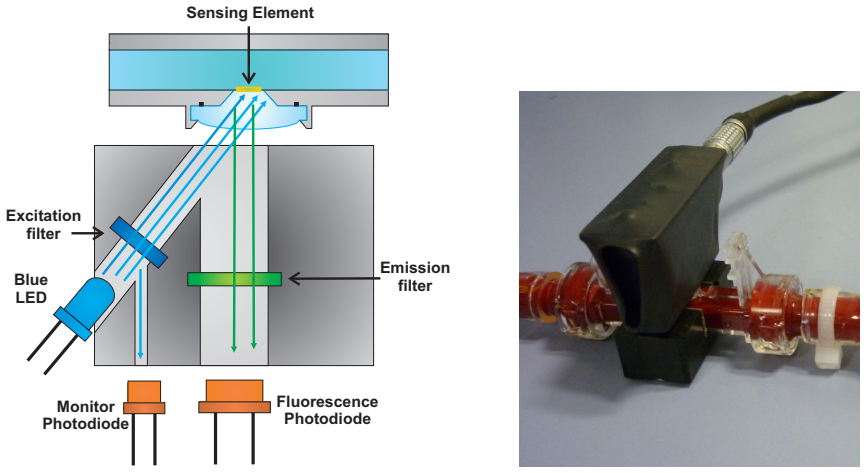


Figure 1.2: Picture of a polymer sensing element placed in a flow cuvette.

1.2.2 Optical reading head

A schematic representation of the realized optical setup is reported in Figure 1.3. To perform fluorescence intensity variation measurements, it is necessary to excite the fluorescent dye; in the present work, where the dye was fluorescein, excitation was induced by a blue LED ($\lambda_{peak} \cong 480\ nm$), and the emitted fluorescence signal was collected. Since the optical power emitted by the LED is much higher than the emitted fluorescence, two filters inside the optical head have been positioned; in particular an excitation filter on the light emitted by the LED ($\lambda_{cutoff} = 485\ nm$, ODL S.r.l., BG12) and an emission filter ($\Delta\lambda = 515 - 535\ nm$, Thorlabs, MF525-39) on the optical path of the fluorescence light towards the photodetector. The unwanted excitation light collected by the fluorescence channel produces an offset in the signal of interest comparable with the offsets introduced by the electronics. Thus, it can be compensated by a proper calibration of the sensor. Finally, two photodiodes have been used, one to collect the fluorescence signal and one to monitor the optical power emitted by the LED. The monitor photodiode is used to detect possible variations of the excitation optical power. The system is suitable to be used also with different fluorescent dyes; if their excitation and emission spectra are shifted compared to the fluorescein's ones, it will be sufficient to change the light source and the optical filters.

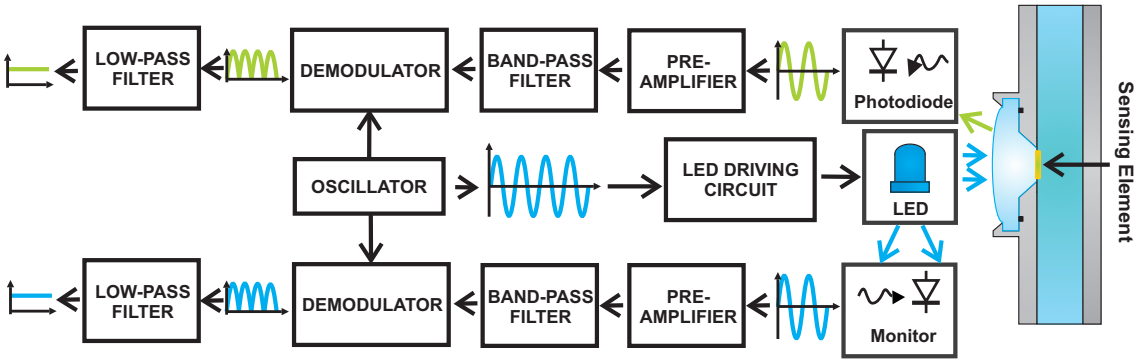


(a) (b)

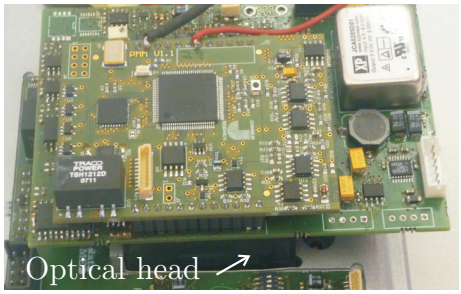
Figure 1.3: Schematic representation (a) and a picture (b) of the optical head. It consists of a blue LED, two photodiodes, to monitor the power emitted by the LED and to collect the fluorescence signal, and finally two optical filters: an excitation filter on the light path of the excitation beam and an emission filter on the light path of the fluorescence emission beam.

1.2.3 Front-end and signal elaboration electronics

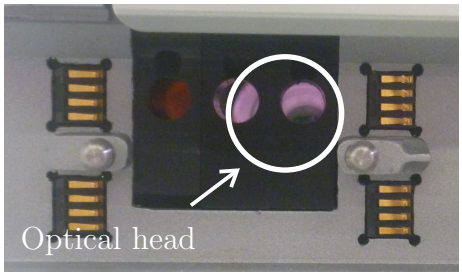
The operation of the measuring device is based on the electronic circuitry for exciting the blue LED and conditioning the signals provided at the photodiode detectors output. In order to reduce the effect of photobleaching, typical of fluorescent indicators, the blue LED was excited by current pulses (duration 800 ms) at a frequency of 0.125 Hz [68]. Moreover, to increase the robustness against the interferences and because of the low signal-to-noise ratio, the lock-in technique has been used, modulating sinusoidally the LED driving current during the pulse and demodulating the collected signal with the same signal (frequency=6.06 kHz), as shown in Figure 1.4.



(a)



(b)



(c)

Figure 1.4: Block diagram of the electronics (a) and pictures of the measuring system including the optical head, back view (b) and front view (c).

In this way, only the signal of interest presents a non-zero average value, which is representative of the excitation and fluorescence intensities. Thus, the front-end electronics consists of two channels devoted to the processing of the monitor and fluorescence photogenerated current signals. Each channel includes: (i) a low-noise transimpedance preamplifier (OPA380, Texas Instruments), (ii) a high-Q bandpass filter with center frequency at 6.06 kHz (OPA228, Texas Instruments) (iii) a lock-in amplifier (AD630, Anlog Devices), (iv) a lowpass filter with a 70 Hz passband cutoff (OPA2227, Texas Instruments). Finally, an analog-to-digital converter

(ADC128S102, Texas Instruments) and a microcontroller (C8051F305, Silicon Labs) are used to acquire and process the information of interest. Figure 1.4 shows the block diagram of the electronics and two pictures of the measuring system including the optical head. For the performed measurements, the fluorescence-to-monitor ratio was considered, thus avoiding that variations in the optical power emitted by the LED could influence the fluorescence measurements. In fact, the fluorescence intensity I_s is a fraction of the absorbed light [69], as reported in Eq. 1.1:

$$I_s = kI_0\phi\epsilon_\lambda lC, \tag{1.1}$$

where I_0 is the intensity of the exciting beam, ϕ is the quantum yield of the fluorophore, ϵ_λ is the molar absorptivity at λ_{ex}, l is the optical path length in the sample, C is the concentration of the fluorophore and k is an instrumental factor. Considering the fluorescence-to-monitor ratio, it yields:

$$I_{Ratio} = \frac{I_s}{I_0} = k\phi\epsilon_\lambda lC. \tag{1.2}$$

As shown in Eq. 1.2, the ratio signal is independent from the optical power emitted by the LED. The fluorescence-to-monitor ratio, I_{Ratio}, has been estimated by the voltage ratio, V_{Ratio}, between the signals generated by the front-end electronics, i.e. V_{Fluo} and V_{Mon}. In the next sections, it will be referred to the fluorescence-to-monitor ratio as I_{Ratio} and to the fluorescence to monitor ratio normalized to the greatest value as I_{Norm}.

1.3 Haematic fluorescence-based pH sensor

A fluorescence pH sensor for measuring blood pH during ECC has been developed using the multi-analyte fluorescence-based sensor platform previously described [70, 71]. In the next Section, a brief description of the theoretical background is given, while in Section 1.3.3 the characterization of the sensor is reported. In Section 1.3.4 and Section 1.3.5 the experimental protocols for in-vitro and ex-vivo tests are described and finally, in Section 1.3.6, the experimental results of both in-vitro and ex-vivo tests are reported and discussed.

1.3.1 Theoretical background

The development of the present sensor is based on the fluorimetric determination of pH [69]. Following the absorption of a photon, the excited molecules can lose energy through non radiative relaxation, emission of a photon or energy transfer to an acceptor. The reemitted photons usually possess less energy, so they are shifted

to the red part of the spectrum. This property represents a great advantage, compared to the absorption spectroscopy, since it decreases the level of the shot noise, which is proportional to the square root of the light intensity. In fact, when the emission light is only observed, the signal-to-noise ratio is greatly improved. The sensor presented in this work makes use of a polymerable fluorescein, namely fluorescein *O*-methacrylate to detect pH changes in the measured fluorescence intensity. Fluorescein *O*-methacrylate is a new kind of pH-sensitive fluorescent monomer characterized by an excitation spectrum centered at λ=490 nm and an emission spectrum centered at λ=520 nm. Fluorescein exhibits multiple, pH dependent ionic equilibria [36]. Both the phenol and carboxylic acid functional groups of fluorescein are almost completely ionized in aqueous solutions above pH 9.0 (Figure 1.5). Acidification of the fluorescein dianion first protonates the phenol ($pK_a \approx 6.4$) to yield the fluorescein monoanion, then the carboxylic acid ($pK_a < 5.0$) to produce the neutral species of fluorescein. Further acidification generates a fluorescein cation ($pK_a \approx 2.1$). Only the monoanion and dianion of fluorescein are fluorescent, with quantum yields of 0.37 and 0.93, respectively. A further equilibrium involves formation of a colorless, nonfluorescent lactone (Figure 1.5). Nevertheless, the fluorescence emission spectrum of fluorescein, even in acidic solution, is dominated by the dianion, with only small contributions from the monoanion. Consequently, the wavelength and shape of the emission spectra resulting from excitation close to the dianion absorption peak at 490 nm are relatively independent on pH, but the fluorescence intensity is dramatically reduced under acidic conditions. The mass-action law relationships

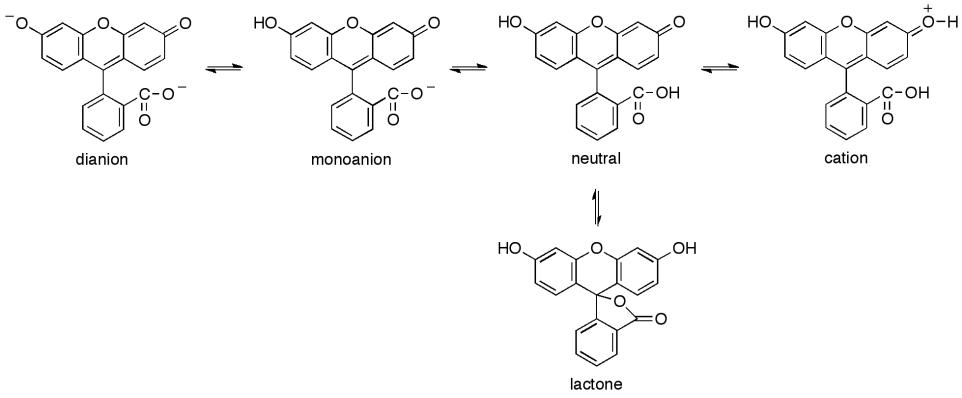


Figure 1.5: Ionization equilibria of fluorescein.

between pH and fluorescence intensity determines the response curve of this sensing approach [72]. Considering a dye covalently linked to a polymer matrix, defining I_{max} and I_{min} as the fluorescence contribution of the fully deprotonated and the fully protonated form of the fluorescent indicator, respectively, the relation between the fluorescence emission intensity I_s and pH, in the presence of the sample under

test, becomes [73, 74, 75]:

$$pH = pK_a - b \cdot log\left(\frac{I_{max} - I_s}{I_s - I_{min}}\right), \tag{1.3}$$

where pK_a is the acid-base constant of the indicator and b is a numerical coefficient, introduced to determine the slope of the function between I_{max} and I_{min}. In fact, the chemical and physical properties of the matrix including the dye (e.g. polarity and viscosity) could affect its sensitivity near the pK_a, thus having different slopes for the same indicator in different matrices. Rewriting Eq. 1.3 in terms of I_s, gives the well-known sigmoidal function [74]:

$$I_s = \frac{I_{max} + I_{min} \cdot 10^{-(\frac{pH-pKa}{b})}}{1 + 10^{-(\frac{pH-pKa}{b})}} = I_{min} + \frac{I_{max} - I_{min}}{1 + 10^{-(\frac{pH-pKa}{b})}}. \tag{1.4}$$

This equation results in a non-linear relationship between the fluorescence intensity versus pH, which has been used for the calculation of the pK_a of the dye included in the polymer matrix.

The pH of blood is related to the constituents of the bicarbonate buffer system by the modified Henderson-Hasselbalch equation [76]:

$$pH_{blood} = pK_{a_{H_2CO_3}} + log\left(\frac{[HCO_3^-]}{[H_2CO_3]}\right), \tag{1.5}$$

where $pK_{a_{H_2CO_3}}$ is the acid dissociation constant of the carbonic acid, which is equal to 6.1, $[HCO_3^-]$ is the concentration of bicarbonate in the blood and $[H_2CO_3]$ is the concentration of carbonic acid in the blood. Considering the partial pressure of carbon dioxide, pCO_2, rather than $[H_2CO_3]$, Eq. 1.5 can be written as:

$$pH_{blood} = 6.1 + log\left(\frac{[HCO_3^-]}{0.03 \times pCO_2}\right). \tag{1.6}$$

Equation 1.6 provides a pH value for healthy human beings between 7.30 and 7.45. This pH values are very close to the pK_a of our sensing membrane, i.e. $I_s = (I_{max} + I_{min})/2$, thus the argument of the logarithmic function in Eq. 1.3 is close to 1. According to the first-order Taylor polynomial of the logarithmic function, the approximation of Eq. 1.3 is:

$$pH = pK_a - 2 \cdot b \cdot \log(e) \cdot \left(\frac{\frac{I_{max}+I_{min}}{2} - I_s}{I_s - I_{min}}\right). \tag{1.7}$$

Since pH values are very close to the pK_a of our sensing membrane, i.e. $I_s = (I_{max} + I_{min})/2$, the term

$$\frac{\frac{I_{max}+I_{min}}{2} - I_s}{I_s - I_{min}}, \tag{1.8}$$

is close to zero. According to the first-order Taylor polynomial of the rational function, Eq. 1.7 can be rewritten as:

$$pH = pK_a + 2 \cdot b \cdot \log(e) \cdot \frac{2}{I_{max} - I_{min}} \cdot \left(I_s - \frac{I_{max} + I_{min}}{2} \right). \tag{1.9}$$

Rewriting Eq. 1.9 in terms of I_s, yields:

$$I_s = (pH - pK_a) \cdot \frac{I_{max} - I_{min}}{2} \cdot \frac{1}{2 \cdot b \cdot \log(e)} + \frac{I_{max} + I_{min}}{2} = A \cdot pH + B, \tag{1.10}$$

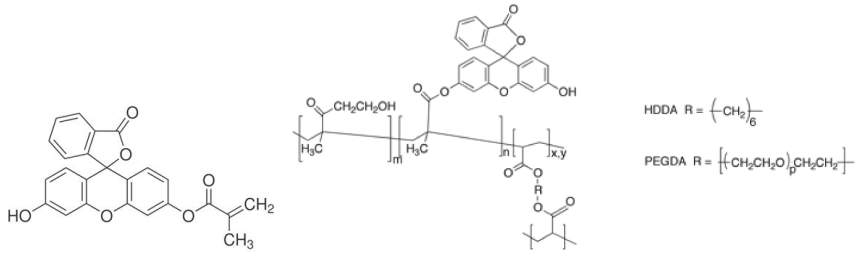
where

$$\begin{aligned} A &= \frac{I_{max} - I_{min}}{2} \cdot \frac{1}{2 \cdot b \cdot \log(e)}, \\ B &= -pK_a \cdot \frac{I_{max} - I_{min}}{2} \cdot \frac{1}{2 \cdot b \cdot \log(e)} + \frac{I_{max} + I_{min}}{2}. \end{aligned} \tag{1.11}$$

1.3.2 Fabrication of the polymer matrix

All chemicals needed for the realization of a hydrophilic, highly swellable, polymer matrix containing an optical sensing element were of analytical grade, purchased from Sigma-Aldrich and used without further purification. All aqueous solutions were prepared using distilled water. The matrix was prepared by the free-radical polymerization of 2-hydroxyethyl methacrylate (HEMA), whose thermodynamic affinity for water is well known [77], catalysed by the radical photoinitiator 4-(2-hydroxyethoxy)phenyl-(2-hydroxy-2-propyl)ketone (Irgacure® 2959, BASF), used at 1.0% wt. The polymerization was run in the presence of a tetrafunctional monomer, namely 1,6-hexanediol diacrylate (HDDA) added at 5.0% wt with respect to HEMA, acting as crosslinker to avoid an excessive swelling degree of the acrylate matrix in contact with water, that could result in a poor mechanically resistance. Fluorescein *O*-methacrylate 97% (Figure 1.6a) was added to the acrylates mixture at 1.0% wt with respect to HEMA monomer; its methacrylate moiety allows for a covalent inclusion of the fluoresceine comonomer within the swellable polymer matrix, thus suppressing the severe drawback of indicator leaching. The covalent bonding between the HEMA matrix and the fluorescent dye is depicted in Figure 1.6b. The overall swelling ratio of the polymer matrix in aqueous solutions was finely tuned by adding different amounts (up to 6.0% wt) of a flexible macromer, namely poly(ethylene glycol) diacrylate (PEGDA, Ebecryl®13, Cytec Industries), to further improve the polymer chain flexibility, and in turn ion mobility within the polymer matrix [78]. The mixture of chemicals was accurately mixed using a magnetic stirrer before being deposited by spin coating (500 rpm, 30 s) onto a transparent PVC substrate

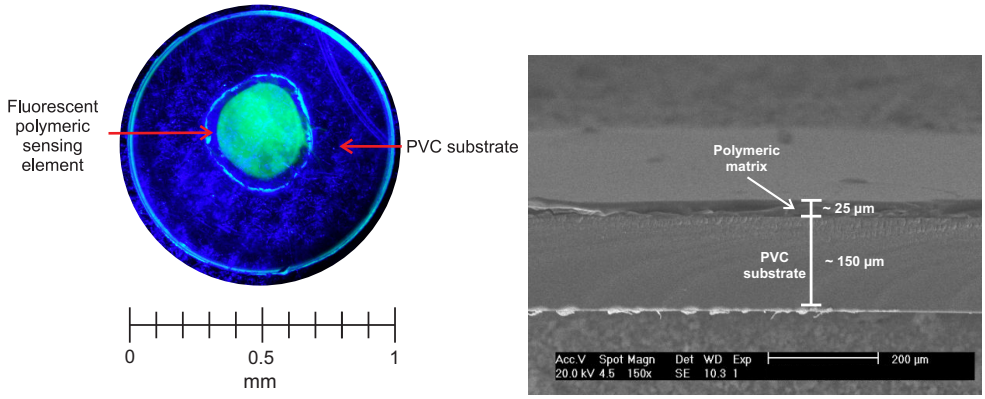
(Figure 1.7a), and cured by UV-irradiation with a medium pressure mercury lamp under nitrogen with a light intensity on the surface of the sample of 30 mW/cm^2 for different time up to 70 s.



(a) (b)

Figure 1.6: Molecular structure of fluorescein *O*-methacrylate (a). Molecular structure of poly(HEMA) covalently bonded to fluorescein *O*-methacrylate units (b).

Figure 1.7b reports a SEM micrograph of a cross-section of the sensing polymer film (thickness about 25 μm) deposited onto the PVC substrate (thickness 150 μm). Because of the manual realization of the sensing elements, it was difficult to firmly control the thickness of the polymer sensing layer; this disuniformity between different sensing elements may cause some variability in terms of response time and sensitivity. The polymer film uniformity could be improved through the application of instrumental deposition techniques, like the one reported by Tian et al. [79, 80].

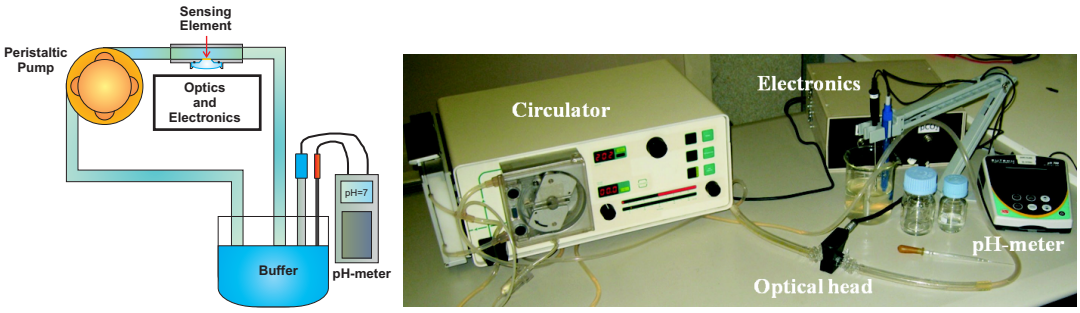


(a) (b)

Figure 1.7: Top-view picture of the hydrophilic polymer matrix including the fluorescent indicator, used as the sensing element, excited by a blue LED (a). SEM cross-section picture of the hydrophilic polymer matrix including the fluorescent indicator, used as the sensing element (b).

1.3.3 Characterization of the pH sensor

In order to test the sensor, a fluidic system has been realized. As shown in the schematic representation in Figure 1.8a and in the picture in Figure 1.8b, it consists of (i) a closed loop circulator, (ii) the optical head with the sensing element inside, (iii) a beaker with a reference glass electrode pH-meter (Eutech Instruments, XS pH 700, Netherlands) to measure the pH of the solution and a thermocouple to measure the temperature of the solution and correct the pH value.



(a) (b)

Figure 1.8: Schematic representation of the fluidic system used to test the pH sensor (a). It consists of (i) a closed loop circulator, (ii) the optical head with the sensing element inside, (iii) a beaker with a reference glass electrode pH-meter to measure the pH of the solution and a thermocouple to measure the temperature of the solution and correct the pH value. Picture of the fluidic system used to test the pH sensor (b).

All the measurements were performed at room temperature, with a flow rate of 200 ml/min. The pH of the solution was changed by adding acetic acid or ammonium hydroxide to the solution.

1.3.3.1 Characterization of the sensor pK_a

A fluorimetric titration has been performed to investigate the pK_a of the indicator inside the polymer matrix in aqueous solutions. In Figure 1.9, I_{Ratio} versus time acquired at different pH values is shown. The steady state pH values of the solution as measured by the reference instrument are reported close to the curve. In Figure 1.10, the average of the I_{Ratio} signal over a time interval of 60 seconds for every reference pH value is shown.

I_{Ratio} values have been fitted to a sigmoidal function, represented in Figure 1.10 and described by Eq. 1.4, where $I_{max} = 0.93$, $I_{min} = 0.48$ and $b = 1.10$. The pK_a of the indicator was thus estimated to be 7.9. This value is bigger than the

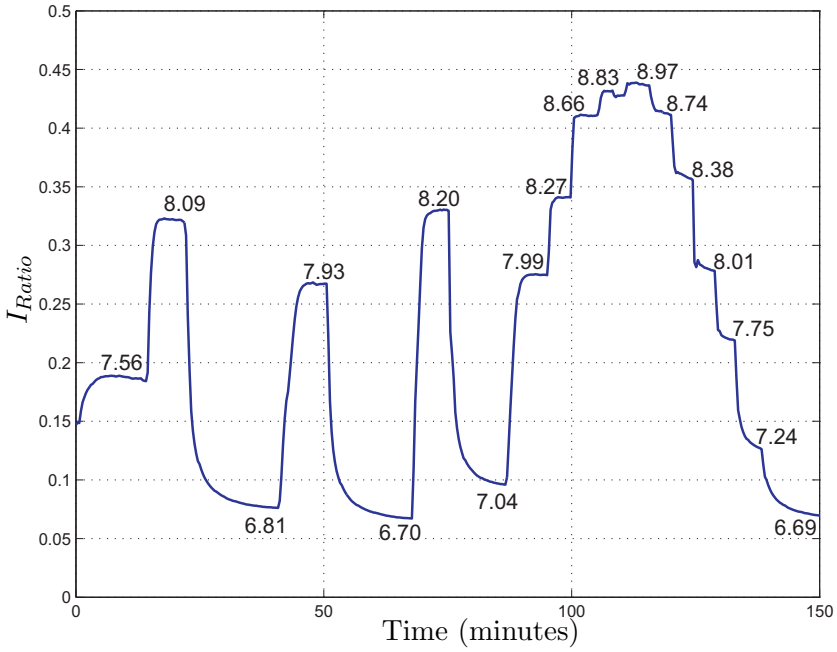


Figure 1.9: I_{Ratio} versus time of a measurement performed in an aqueous solution, changing the pH value of the solution. The steady state pH values of the solution as measured by the reference instrument are reported close to the curve.

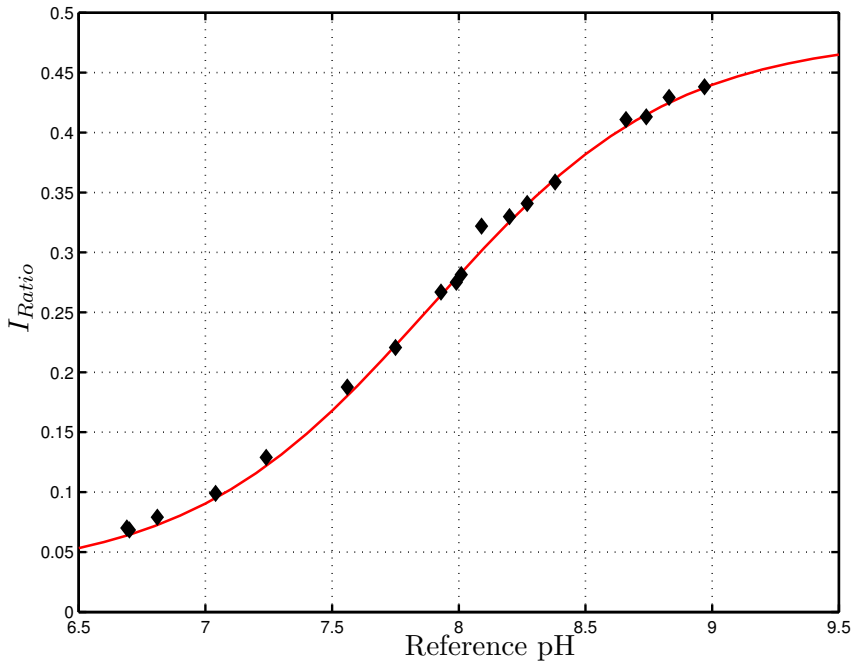


Figure 1.10: I_{Ratio} versus reference pH and sigmoidal interpolation curve.

one reported in literature for the fluorescein in aqueous solutions, i.e. 6.4 [81]. According to Vasylevska et al. [74], an increase in pK_a is observed upon covalent immobilization of the indicator and is attributed to the decrease in the polarity of the microenvironment.
The sigmoidal interpolation curve can be approximated with a linear curve for a range of values of pH close to the pK_a of the indicator. In particular, a linearity range between 7 and 8 has been considered. In the next section, the linearity characteristics of the sensor within this range are presented.

1.3.3.2 Evaluation of the non-linearity error

Since the proportionality coefficient between I_{Ratio} and parameters A and B in Eq. 1.10 are unknown, the relationship between pH and I_{Ratio} was determined using a linear regression model. Starting from Eq. 1.10, two regression coefficients $\tilde{A}$ and $\tilde{B}$ were introduced, calculated through a least square method relating the values of I_{Ratio} and the actual pH values measured by the blood gas analyzer according to the regression model:

$$I_{Ratio} = \tilde{A} \cdot pH + \tilde{B}. \tag{1.12}$$

This linear approximation is considered to be valid in the pH range of interest (RoI) (7 - 8). In Figure 1.11, I_{Ratio} versus time acquired at different pH values in the range 7.0-8.0 is shown. For every reference pH value, the average of I_{Ratio} has been calculated over an interval of 60 seconds. Afterward, these values were fitted to a linear function as:

$$\bar{I}_{Ratio} = 1.6834 \cdot pH_{Reference} - 10.1342, \tag{1.13}$$

where $\bar{I}_{Ratio}$ is the average value of I_{Ratio} and $pH_{Reference}$ is the pH measured by the reference pH electrode. Hence, the estimated pH values were calculated as:

$$pH_{Estimated} = \frac{\bar{I}_{Ratio} - q}{m} = \frac{\bar{I}_{Ratio} + 10.1342}{1.6834}. \tag{1.14}$$

In Figure 1.12a the estimated values versus reference pH are shown, together with the fitting line (R=0.9958), whereas Figure 1.12b shows the difference between the estimated and reference pH values, for every reference pH value. Thus, the Integral Non-linearity error, which is a parameter representing the maximum deviation between the sigmoidal interpolation curve of Eq. 1.4 and the linear interpolation curve of Eq. 1.13, was calculated to be 0.02 units of pH, which represents the 2% in the range 7.0-8.0. Finally, the Integral Non-linearity error was calculated also for the measurement reported in Figure 1.9 and was determined to be 0.05 units of pH,

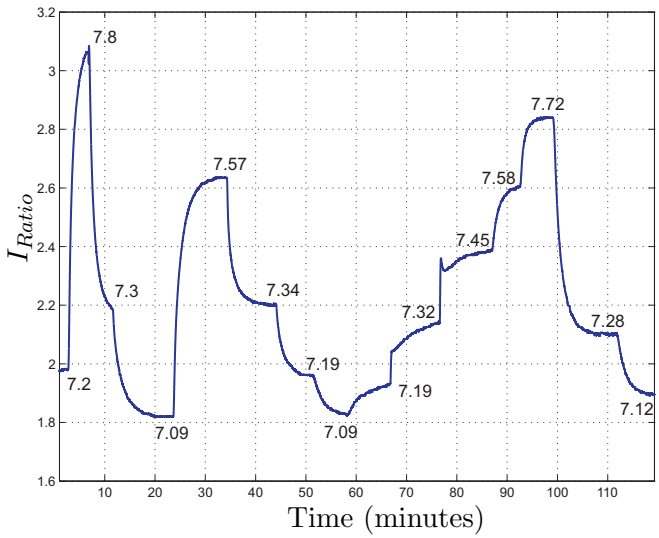
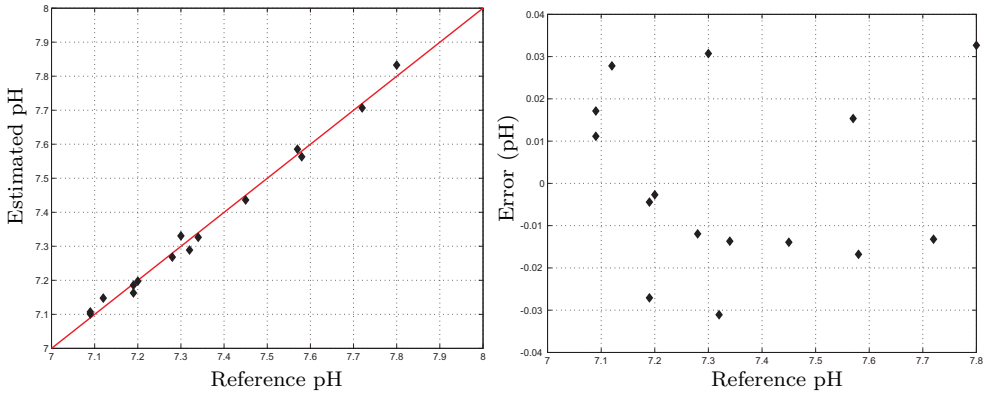


Figure 1.11: I_{Ratio} versus time acquired at different pH values in the range 7.0-8.0. The steady state pH values of the solution as measured by the reference instrument are reported close to the curve.

which represents the 5% in the range 7.0-8.0. These small variations between the Integral Non-linearity errors are probably due to different sensing elements used to perform these tests.



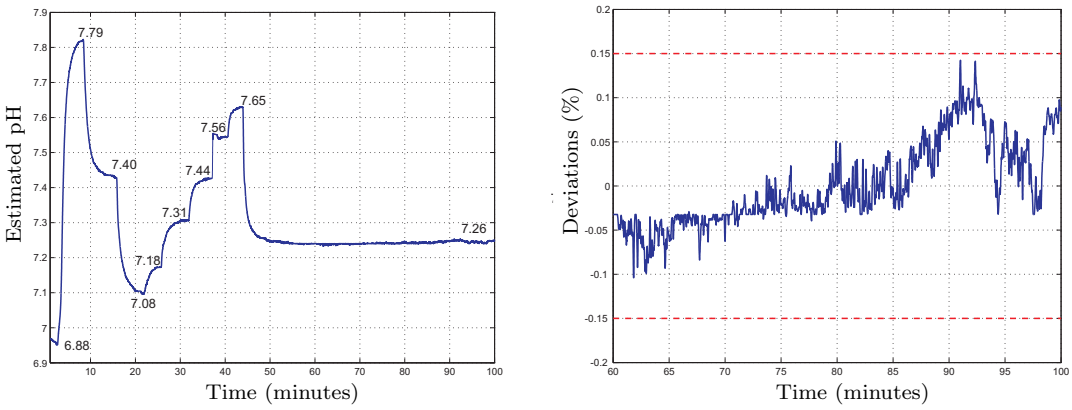
(a) (b)

Figure 1.12: Estimated versus reference pH and fitting line (a). Difference between estimated and reference pH versus reference pH (b).

1.3.3.3 Short term stability

Figure 1.13a shows the estimated pH values, determined through Equation 1.14, versus time acquired firstly changing the pH of the solution and then keeping it constant.

Assuming that the pH of the solution is completely stabilized at t = 60 min, the short term stability can be determined by evaluating the deviations of the estimated pH respect to its mean value in the time interval 60-100 min. As shown in Figure 1.13b, no significant drifts have been observed in this time interval; all deviations are included in a range of values of 0.15% of the mean value of the estimated pH.



(a) (b)

Figure 1.13: Estimated pH versus time of a measurement performed changing the pH value of the solution and then keeping it constant (a). The steady state pH values of the solution as measured by the reference instrument are reported close to the curve. Deviations of the estimated pH respect to its mean value in the time interval 60-100 min (b). All deviations are included in a range of values of 0.15% of the mean value of the estimated pH (dashed lines).

1.3.3.4 Response time

The sensor response time, the so called τ_{90}, is defined as the *time required for the sensor output to reach 90% of the change from its previous value to the final settled value*. To measure the response time of the developed sensor, the fluidic system reported in Figure 1.14 has been realized. The setup shown in Figure 1.8a has been

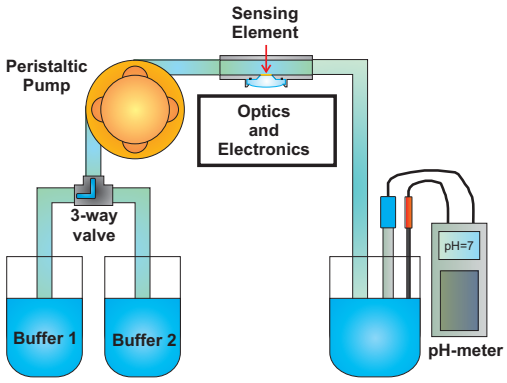


Figure 1.14: Schematic representation of the fluidic system used to measure the pH sensor response time. It consists of (i) a closed loop circulator, (ii) the optical head with the sensing element inside, (iii) a beaker with a glass electrode pH-meter to measure the pH of the solution and a thermocouple to measure the temperature of the solution and correct the pH value. A 3-way valve has been added to swith between two different buffer solutions.

integrated by a 3-way valve, thus allowing switching between two different buffer solutions (Fisher Scientific, Sigma-Aldrich). The measurement was performed by pumping the first buffer solution till the sensor stabilized, then switching to the second buffer solution and waiting for the stabilization of the sensor and finally switching back to the first one. The measurements were performed with the same sensing element (50 μm thick), at room temperature at a flow rate of 200 ml/min. The whole circuit volume was about 200 ml. In Figures 1.15a, 1.15b and 1.15c I_{Ratio} versus time is reported for three different test conditions.

In Table 1.1, the calculated rise and fall times (τ_{90}) of the performed measurements are reported.

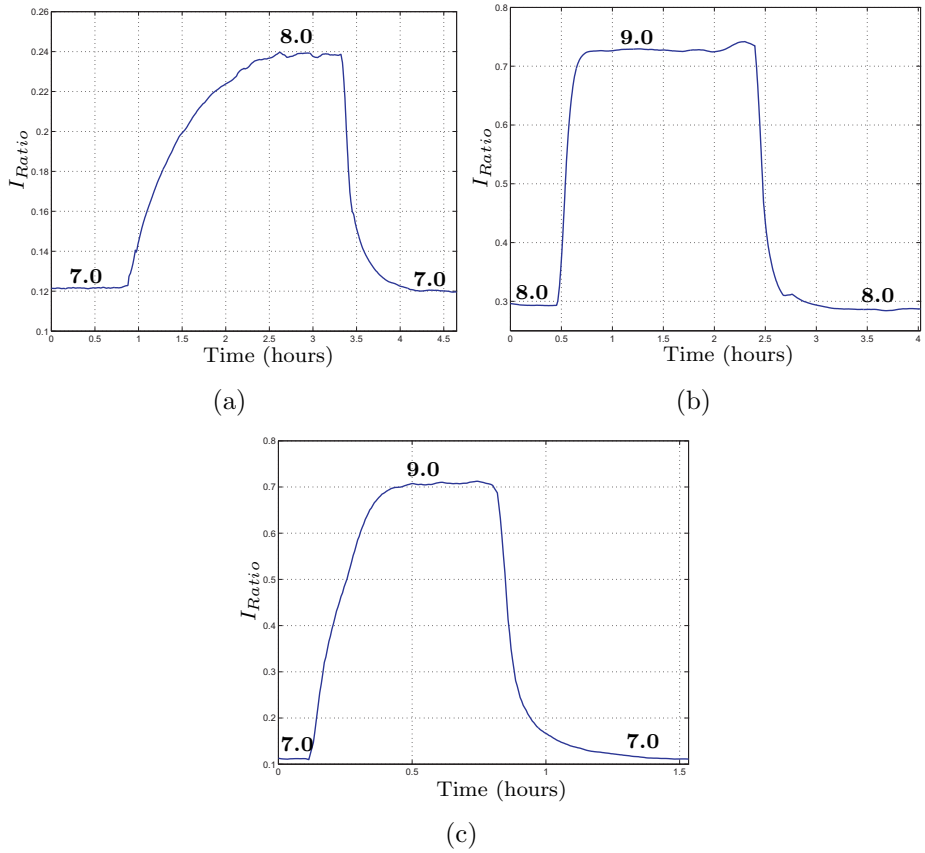


Figure 1.15: I_{Ratio} versus time acquired during the three tests performed to evaluate the response time of the sensor and its dependency on the pH value. Sensor response to a change in pH from 7 to 8 and back (a); sensor response to a change in pH from 8 to 9 and back (b); sensor response to a change in pH from 7 to 9 and back (c).

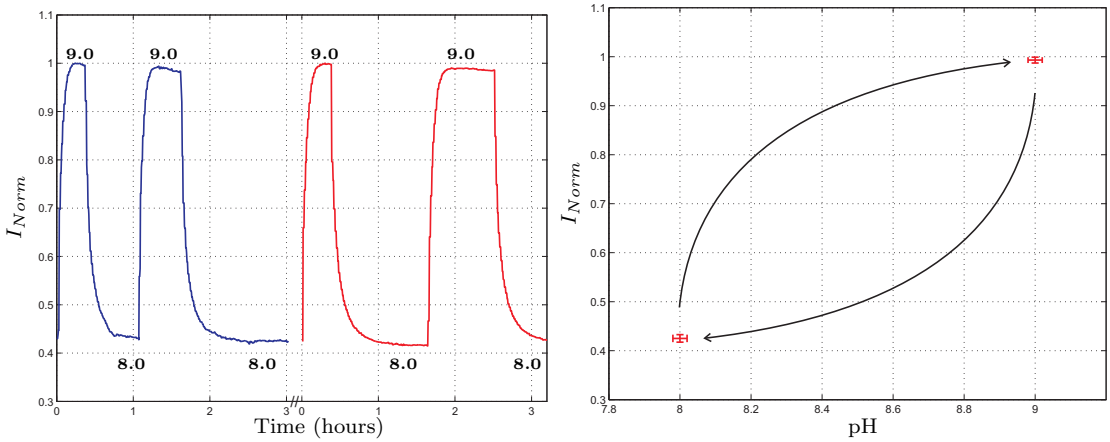
Table 1.1: Rise and fall times estimated during the three performed tests.

	Rise time τ_{90} (min)	Fall time τ_{90} (min)
$pH = 7.0 \rightarrow 8.0$	54.0	23.0
$pH = 8.0 \rightarrow 9.0$	9.5	17.0
$pH = 7.0 \rightarrow 9.0$	14.0	26.0

The response time of the sensor depends on several factors, such as the flow-rate and the thickness of the film. Response times reported in Figure 1.15 are substantially different compared to those of Figures 1.9, 1.11 and 1.13a. This large difference can be ascribed to the thickness of the sensing film. It has been observed that even small differences in thickness can result in large changes in the dynamic response of the sensor. Moreover, the response time depends on the initial and final pH value. In fact, from the data reported in Table 1.1, it can be noticed that the response time is highly dependent on the pH values with respect to the pK_a of the fluorescent dye. In fact, the response time is longer for the first variation of pH, ($pH = 7.0 \rightarrow 8.0$, Figure 1.15a), which has been done between pH values below or equal to the pK_a ($pK_a \approx 7.9$). If the final pH value is much higher than the pK_a, the response time is significantly reduced; in fact, considering the second variation ($pH = 8.0 \rightarrow 9.0$, Figure 1.15b), the response time is improved. Nevertheless, it can be noticed that, if the variation is done between two pH values chosen so that they are on either side of the pK_a value, ($pH = 7.0 \rightarrow 9.0$, Figure 1.15c), the rise and fall times are further improved (7 minutes per unit of pH and 13 minutes per unit of pH, respectively). Finally, it can be noticed that, if the final pH value is higher than the pK_a, the rise time is lower than the fall time, whereas, if the final pH value is lower than the pK_a, the fall time is lower than the rise time.
Nevertheless, this dynamic behavior can be improved by using thinner polymer sensing element. The thickness of the sensing element depends on a trade-off between the response time and the signal-to-noise ratio. As far as the signal-to-noise ratio of the developed sensor is high enough, it is possible to use thinner polymer films and thus achieve shorter response time.

1.3.3.5 Reproducibility

The reproducibility of the pH sensor has been evaluated using the fluidic system represented in Figure 1.14, by changing the pH between the same two buffer solutions, i.e. 8.0 and 9.0 (Fisher, Sigma-Aldrich). In Figure 1.16a, I_{Norm} versus time is reported. The first test (blue) was performed the day before with respect to the second one (red), using the same polymer sensing element. In Figure 1.16b the mean values of the I_{Norm} signal, measured at $pH = 8.0$ and $pH = 9.0$, are reported, together with the error bars.



(a) (b)

Figure 1.16: Acquired I_{Norm} versus time changing the pH value between 8.0 and 9.0 to evaluate the reproducibility of the pH sensor (a). Two tests were performed (blue line and red line) on two consecutive days. Mean values and error bars of the I_{Norm} signal, measured at $pH = 8.0$ and $pH = 9.0$ (b).

The mean values were calculated considering the average values of I_{Norm} in the steady-states at pH = 8.0 and pH = 9.0 respectively. The error bars represent the standard deviations; for the reference pH values they were determined through the uncertainty reported on the datasheet of the buffer solutions, i.e. $pH = 8.0 \pm 0.02$ and $pH = 9.0 \pm 0.02$ respectively, whereas for the I_{Norm} signal they were determined by the sample standard deviation (s). Finally, the mean values ($\bar{I}_{Norm}$) and the relative standard deviations ($\sigma = s/\bar{I}_{Norm}$) of I_{Norm} were determined to be $0.425 \pm 1.80\%$ at pH=8.0 and $0.993 \pm 0.60\%$ at pH=9.0.

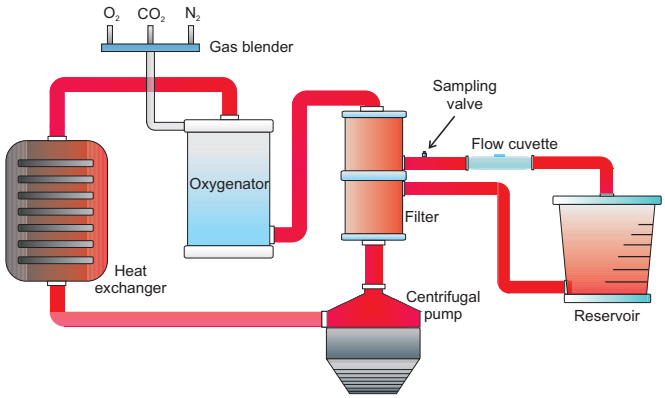
1.3.4 Experimental protocol for in-vitro test

Figure 1.17 shows the mock system assembled to simulate the blood condition during ECC. It consists of: (i) a centrifugal pump, capable of guaranteeing a flux up to 6 L/min; (ii) a heat exchanger based on the Peltier effect (temperature power ranging 15-39°C); (iii) an oxygenator and (iv) an arterial/venous bubble filter. The whole circuit volume, including the centrifugal pump, the heat exchanger, the oxygenator, the arterial-venous bubble filter and the tubing set is 760 ml. The flow cuvette was positioned in the tubing connecting the filter and the reservoir, on the arterial line. The optical probe was fixed to the flow cuvette according to the setup shown in Figure 1.3 and the fuorescence and monitor signals were collected and analyzed by the front-end electronics that generates V_{ratio}. A constant blood flow (2 L/min) was maintained throughout the experiments. Blood samples were obtained from the extracorporeal line close to the flow cuvette (sampling valve). The protocol described above was repeated for two different blood temperatures, 37°C and 32°C, since frequently in cardiac surgery a moderate ipothermia condition is induced. The dependence of the sensor on the blood saturation was not investigate since it is correlated with the variation of pH.

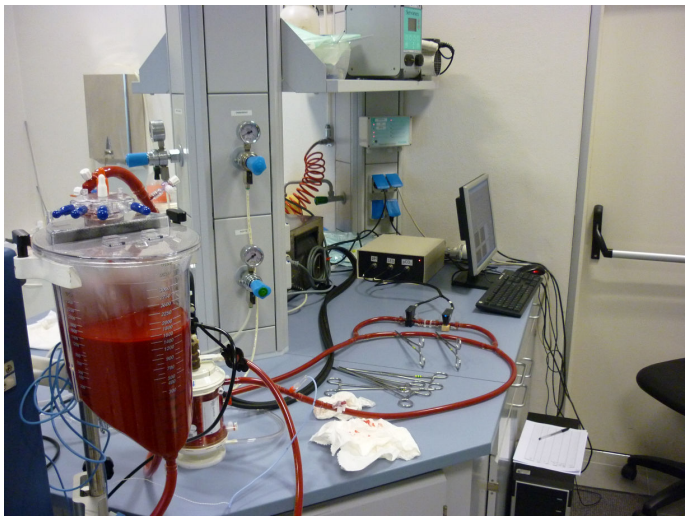
The cow was exsanguinated at local slaughterhouse one hour before the experiment. The collected blood was stored in heparinized container at 4°C during the transportation. Whole blood is an intrinsically complex fluid and difficult to manipulate, thus great care was devoted to not alterate natural condition of the sample. Before the experiment, the blood was diluted with saline solution and used to fill the reservoir with 3 L of fluid. The hemodilution was adjusted to provide sample with a hematocrit of 33-34%. The values of partial pressure of oxygen (pO_2), carbon dioxide (pCO_2) and pH were adjusted to normal physiological range previous the measurement session. All blood gas analyses were performed with a commercial blood gas analyzer (ABL, Radiometer A/S, Copenhagen, Denmark).

1.3.5 Experimental protocol for ex-vivo test

The study was approved by the the Animal Care Ethical Committee of the University of Modena and Reggio Emilia. The experiments followed the recommendations described in the Italian law (D.L. 27-1-1992 n. 116). A veterinary doctor was constantly involved in the activities to control animals wellness and health. First experiment was performed on an adult female pig (see note Table 1.2). To confirm the results obtained during this experiment a second sensor has been tested on an adult female sheep (see note Table 1.3).



(a)



(b)

Figure 1.17: Block diagram (a) and picture (b) of the mock system realized to simulate the blood condition during ECC. It consists of: a centrifugal pump, an oxygenator, a heat exchanger, an arterial/venous bubble filter, a sampling valve, a flow cuvette and tubings.

Table 1.2: Details of the pig involved in the ex-vivo experiment.

Race	Large White (Female)	**Weight**	80 kg
Sedation	Zolazepam (Zoletil®) 5 mg/kg + Atropine 1 ml		
Induction	Propofol		
Maintenance	Isoflurane + O_2		
Analgesia	Morphine + 0.3 mg/kg		

Table 1.3: Details of the sheep involved in the ex-vivo experiment.

Sheep (Female)	**Weight**	55 kg
Sedation	Tiletamine and Zolazepam (Zoletil®) + Atropine 0.66 mg/kg	
Induction	Propofol	
Maintenance	Fentanyl 0.1 mg/kg + Propofol 22.5 mg/kg	
	+ 2% isofluorane in 50% N_2O/ 50% O_2	

During each surgery the sensor (different sensing elements have been used in the two experiments) was attached to the arterial line of the ECC circuit. A total of 20 samples were analyzed by a a blood gas analyzer (ABL, Radiometer A/S, Copenhagen, Denmark) during the 2 surgeries with pH varying in the range 7.2 and 7.5. The experimental setup for ex-vivo evaluation of the measuring system is shown in Figure 1.18. It consists of the standard ECC setup including the flow cuvette placed between the animal and the arterial filter.

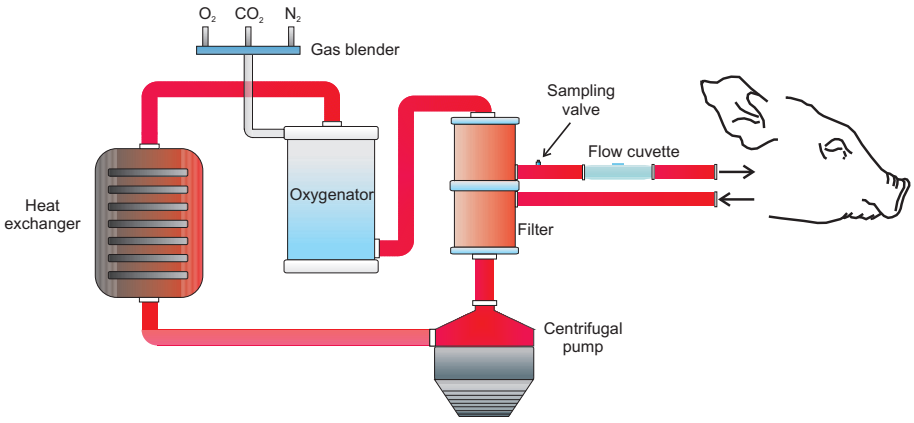
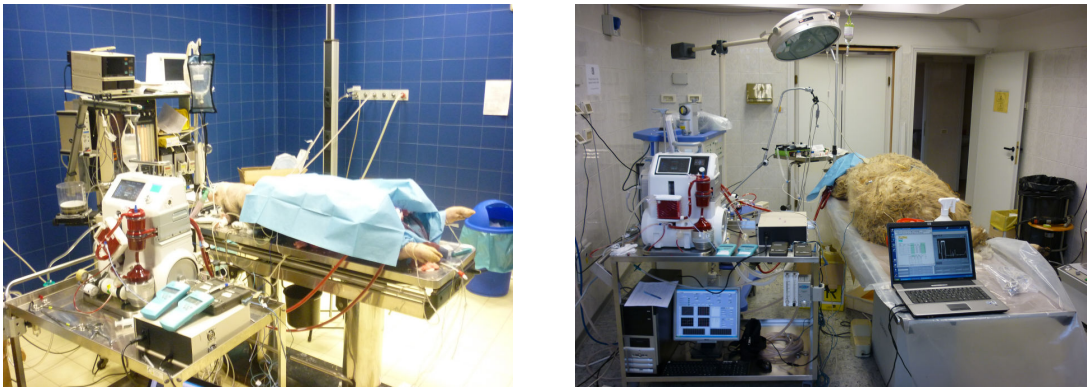


Figure 1.18: Block diagram of the experimental setup for ex-vivo evaluation of the measuring system.

The following blood parameters were monitored during the experiment: blood flow rate by flow probes (H9XL Flow Probe, Transonic System Inc., Artisan Sci. Co., Champaign, IL, USA), blood pressures by digital sensors (DPI 705, Druck Ltd, Leichester, England), blood temperatures by thermistor probes (Exacon® Medical Temperature Probe, Exacon Scientific A/S, Roskilde, Denmark) and digital thermometer (Omega HH41, Engineering Inc., Stamford, CT, USA), blood gases (p_aO_2: oxygen partial pressure in the arterial blood; p_vO_2: oxygen partial pressure in the venous blood; p_aCO_2: carbon dioxide partial pressure in the arterial

blood; S_aO_2: hemoglobin oxygen saturation in the arterial blood) by a blood gas analyzer (ABL, Radiometer A/S, Copenhagen, Denmark); hematocrit by a HCT minicentrifuge (Hematocrit Centrifugette 4203, ALC International S.r.L., Cologno Monzese, MI, Italy), blood electrolytes and activated clotting time (ACT) (ACT Plus™, Medtronic Perfusion Systems, Minneapolis, MN, USA). Heparin was continuously infused during the experiment in order to keep the ACT between 300 s and 450 s. The pig was perfused for 9 hours in venous-arterial mode (iliac artery and vein were cannulated by two bypass cannulas), in normothermic condition (37.5-38° C). The sheep was perfused for 11 hours in venous-arterial mode (right carotid artery and jugular vein were cannulated by two bypass cannulas), under controlled hypothermia (33.2°C and 34.5°C). The pH of blood was sampled with a time interval of one hour in each ex-vivo experiment, obtaining 9 samples in the ex-vivo experiment on the pig and 11 on the sheep. Pictures of the environment test during the two ex-vivo experiments are shown in Figure 1.19.



(a) (b)

Figure 1.19: Pictures of the environment test during the experiment performed on the pig (a) and sheep (b).

1.3.6 Results

1.3.6.1 In-vitro experimental results

In Figure 1.20, I_{Ratio} versus time is shown, for a measurement performed on cow blood, changing the pH of the blood. The red dots are the average values of the I_{Ratio} signal recorded at each reference pH (green dots), measured by the blood gas analyzer. During this test, both the flow rate and the temperature of blood were kept constant, respectively at a value of 2 L/min and 37°C. The pH of the blood was changed using gases, in particular carbon dioxide and nitrogen. The oxygen was

kept quite constant, thus the saturation was always above 90%. The hematocrit of the blood was always kept at a value of 33-34%.

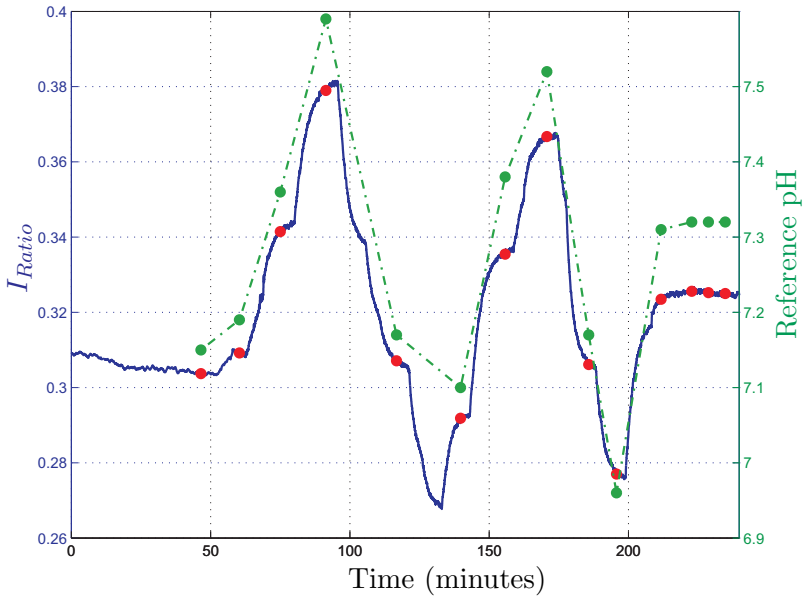


Figure 1.20: I_{Ratio} versus time measured in-vitro, changing the pH of the blood. The red dots are the average values of the I_{Ratio} signal recorded at each reference pH (green dots) measured by the blood gas analyzer.

The data interpolation was obtained by applying the regression model on the mean values of I_{Ratio}, calculated over a time interval of 60 s, at each reference pH value. The coefficients $\tilde{A}$, $\tilde{B}$ and the correlation index determined by the regression model are reported in Table 1.4.

Table 1.4: Coefficients $\tilde{A}$ and $\tilde{B}$ of the regression model and correlation index, determined considering the mean values of I_{ratio} (averaged over a time interval of 60 s).

$\tilde{A}$	0.1628
$\tilde{B}$	-0.8615
R	0.9887

The linearity and the difference between the pH measured by the developed system and the pH determined by the blood gas analyzer are shown in Figure 1.21.

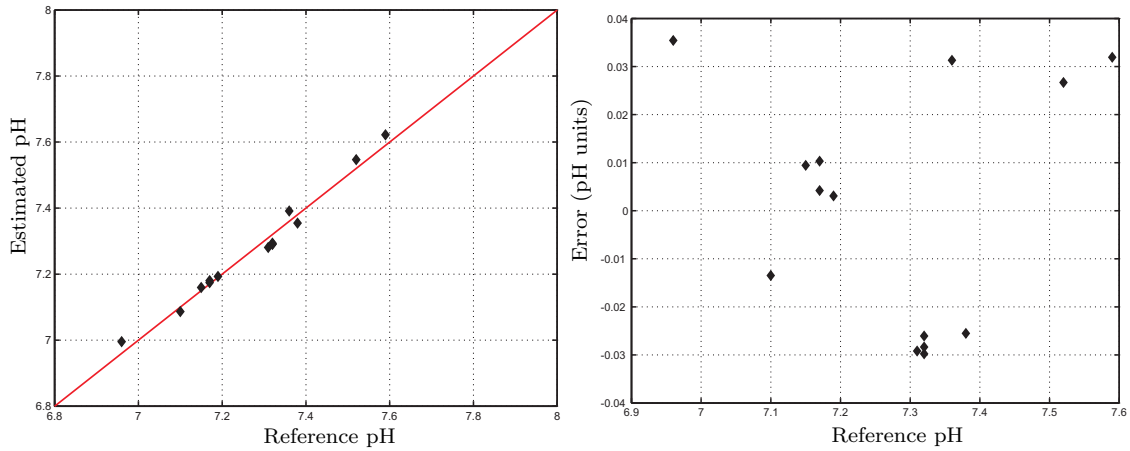


Figure 1.21: Estimated pH versus reference pH and the interpolation line (a). The interpolation line was obtained by applying the regression model on the mean values of I_{Ratio}, calculated over an interval of 60 s, at every reference pH value. Difference between the estimated pH and the reference pH, versus the reference pH (b).

The mean square error value, calculated as:

$$\epsilon = \sqrt{\frac{\sum_i (pH_{Reference_i} - pH_{Estimated_i})^2}{N}}, \tag{1.15}$$

was determined to be 0.0243 units of pH (2.2% of the RoI).

Frequently in cardiac surgery a moderate ipothermia condition is induced. To investigate the potential influence of this procedure on the recorded signal, measurements were performed varying the temperature from 37°C to 32°C. During this test, the temperature was initially kept at a value of 37°C, then was brought to 32 °C, and finally changed back to 37°C. In Figure 1.22, I_{Ratio} versus time is shown; during the time period included between the two black vertical dashed lines the temperature was brought to 32°C. The red dots are the average values of the I_{Ratio} signal measured at each reference pH (green dots).

Results of the regression method, calculated considering only the pH measured at 37°C, applied to the mean values of I_{Ratio} (averaged over a time interval of 60 s) are shown in Figure 1.23. The diamonds are the reference pH measured at 37°C, whereas the triangles were measured at 32°C.

The coefficients $\tilde{A}$, $\tilde{B}$ and the correlation index determined by the regression model are reported in Table 1.5.

The mean square errors, calculated according to (1.15), were 0.0150 units of pH for the acquisition at 37°C (1.3% of the RoI) and 0.0280 for the acquisition at 32°C (2.5% of the RoI).

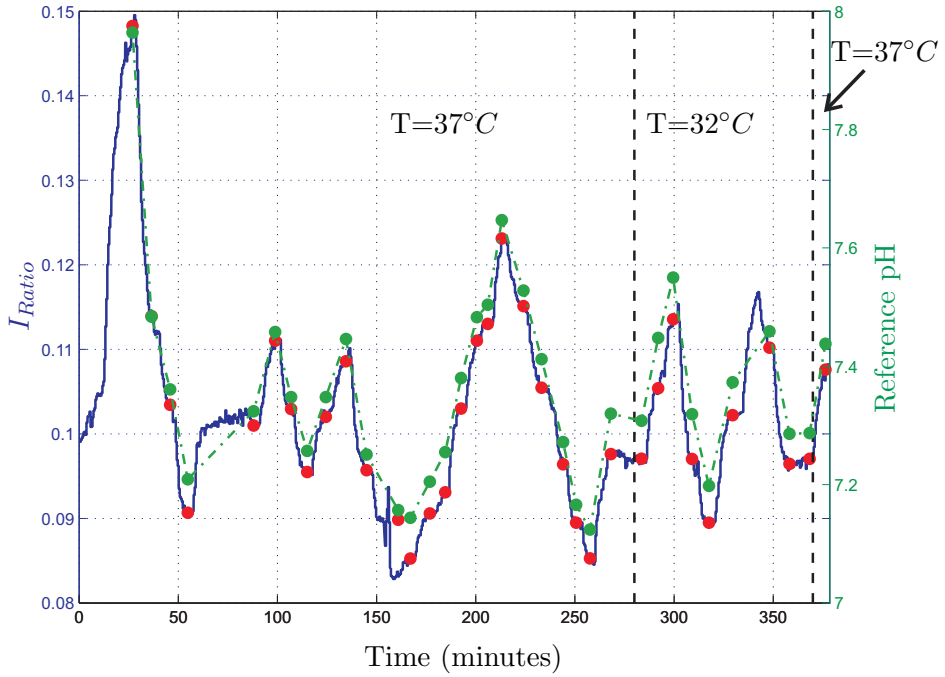


Figure 1.22: I_{Ratio} versus time measured during an in-vitro experiment. The red dots are the average values of the I_{Ratio} signal measured at each reference pH (green dots). To investigate the potential influence of a moderate ipothermia condition on the recorded signal, the temperature was initially kept at a value of 37°C, then was brought to 32°C (between the two black vertical dashed lines), and finally changed back to 37°C.

Table 1.5: Coefficients $\tilde{A}$ and $\tilde{B}$ of the regression model and correlation index, determined considering the mean values of I_{Ratio} (averaged over a time interval of 60 s) of the pH recorded at 37°C.

$\tilde{A}$	0.0743
$\tilde{B}$	-0.4440
R	0.9936

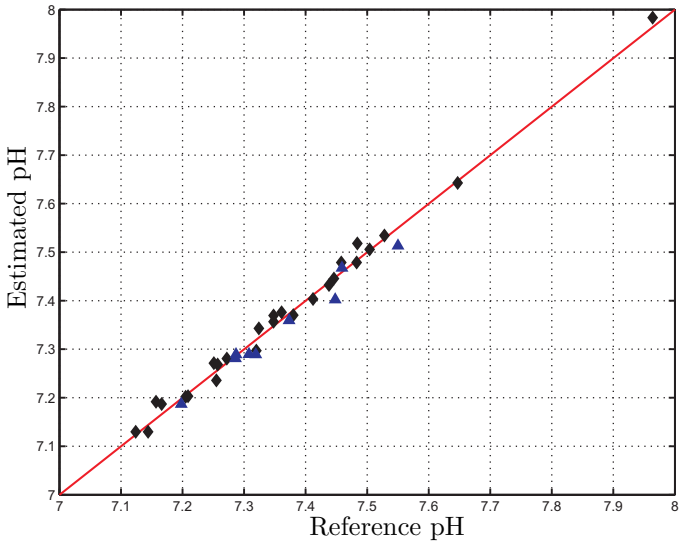


Figure 1.23: Estimated pH versus reference pH and the interpolation line, obtained by applying the regression model on the mean values of the I_{Ratio} recorded at 37°C. The diamonds are the data acquired at 37°C, whereas the triangles were measured at 32°C.

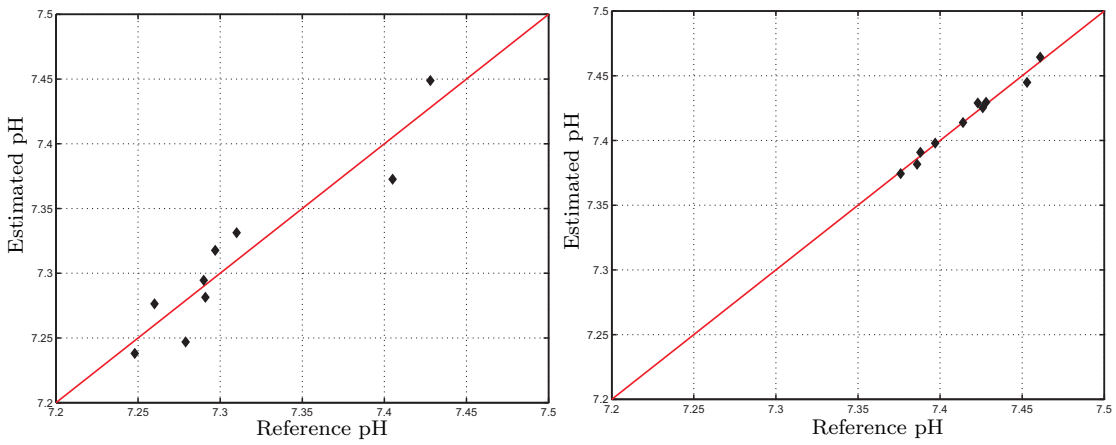
1.3.6.2 Ex-vivo experimental results

In ex-vivo experiments, the pH was changed only with little variations, thus avoiding to compromise the physiological functions of the animals. The linearity between the pH measured by the developed system and the pH determined by the commercial blood gas analyzer is shown in Figure 1.24. The coefficients $\tilde{A}$, $\tilde{B}$ and the correlation index determined by the regression model are reported in Table 1.6.

Table 1.6: Coefficients $\tilde{A}$ and $\tilde{B}$ of the regression model and correlation index, determined considering the mean values of the I_{Ratio} (averaged over a time interval of 60 s) in the ex-vivo experiments on a pig and on a sheep.

	Pig	Sheep
$\tilde{A}$	0.1213	0.4889
$\tilde{B}$	-0.6541	-3.516
R	0.9431	0.9901

The mean error was determined to be 0.0038 units of pH (0.3% of the RoI) for the experiment performed on the sheep and 0.0208 units of pH (1.9% of the RoI) for the one performed on the pig.



(a) (b)

Figure 1.24: Estimated pH measured by the developed system versus reference pH determined by the commercial blood gas analyzer and the interpolation line, measured in the ex-vivo tests performed on the pig (a) and on the sheep (b).

1.3.7 Conclusions

A fluorescence pH sensor for measuring blood pH during ECC has been developed using the multi-analyte fluorescence-based sensor platform previously described. The sensor is based on the pH-dependent fluorescence of a purposely developed polymer matrix including a fluorescent monomer (fluorescein *O*-methacrylate 97%). Since the pK_a of the sensing element has been calculated to be 7.9, the sensor is suitable for measurement of near neutral solutions. Good performance in terms of linearity (in the range 7.0-8.0), stability and reproducibility were observed. It has to be noticed that the response time strongly depends on the measuring conditions, i.e. amplitude of the pH range and its limit values. Obviously, this parameter also depends on the physico-chemical characteristics of the sensing element. In fact, it has been observed that the response time could change considerably for different sensing elements. This variation is mainly due to different thicknesses and cross-linking degrees of the sensing elements, caused by the manual realization. For a thickness of the sensing film of 50 μm, a response time between ten minutes and one hour was measured. For many applications, this dynamic response may be sufficient; nevertheless it is always possible to reduce the thickness of the polymer film to improve the response time.

In the tests performed on blood, the sensor has shown a very high correlation (R of 0.9887 and 0.9936 in-vitro and R of 0.9431 and 0.9901 ex-vivo) between pH value determined by the developed device and pH measured by a commercial blood gas analyzer during in-vitro and ex-vivo experiments. After calibration, a mean

square error lower than the 3% of the RoI was determined both for in-vivo and ex-vivo measurements. Considering an ipothermic treatment at 32°C, the mean square error, calculated with the calibration curve obtained considering only the pH values recorded at 37°C, was lower than 3% of the measuring range. This means that this sensor is suitable for iphotermic treatment without the need of a new calibration curve.

However a different set of calibration coefficients was obtained for each sensing element. This variation is mainly associated to the reproducibility of the deposition of the fluorescence film on the substrate. Therefore a calibration procedure before each EEC treatment is required. This might be problematic, since at least two points of calibration are required. This procedure leads to two main disadvantages: (i) a commercial blood gas analyzer is always required in order to calibrate the sensor (limiting its field of application); (ii) calibrating the sensor on a human being may be difficult since, in normal conditions, pH has little variations. This means that the sensor is likely to be calibrated with two near points, with a high probability of committing a great error for pH values out of the range of calibration. These problems could be overcome calibrating the sensor with two buffer physiological solutions with an already-known pH. Nevertheless, this approach could be problematic for perfusionists who are accustomed to use only one priming physiological solution.

1.4 Haematic fluorescence-based pCO_2 sensor

A fluorescence-based sensor for measuring blood pCO_2 during ECC has been developed using the multi-analyte fluorescence-based sensor platform previously described. In particular, the developed sensor platform allows to use the same light source, photodetectors, optics and electronics of the pH sensor, by simply changing the polymer sensing element. In the next Section, a brief description of the theoretical background is given, while in Section 1.4.2 the polymer sensing matrix fabrication will be reported. Finally, in Section 1.4.3 and Section 1.4.4 the experimental protocol is described and the results are reported and discussed.

1.4.1 Theoretical background

Carbon dioxide is usually sensed via an acid-base chemistry occurring inside a sensor membrane, e.g using a bicarbonate buffer of pH 8-9 contained in a gas-permeable polymer, impermeable to protons in order to prevent interference by pH. The following reversible reaction occurs in the buffer [82]:

$$CO_2 + H_2O + HCO_3^- \rightleftharpoons 2HCO_3^- + H^+. \tag{1.16}$$

The change in pH can be detected by a pH indicator contained in the sensor layer. Nevertheless, this detection scheme suffers from several drawbacks, which include the control of the osmolarity and ionic strength of the water-buffer system. To overcome this problem, Mills et al. [83] proposed the plastic type CO_2 sensors, also called "naked sensors", where the dye and a lipophilic base (rather than an aqueous solution) were contained in a plastic material. In this way, there is no more need of a buffer solution in the hydrophilic polymer. On the contrary, this sensing material consists of an indicator dye along with a quaternary ammonium hydroxide (phase transfer agent) in a polymer solution. This kind of sensor is faster and more reliable even though it suffers from high cross-sensitivity to foreign acid gases. Considering an acid-base indicator (HD), and a phase transfer agent (Q^+OH^-), the preparation of the polymer sensing matrix is based on the extraction, as the ion pair Q^+D^-, of the deprotonated form of the indicator, D^-, from a high polar protic medium (i.e. methanol/ethanol) into the less polar medium of the polymer. Nevertheless, when a phase transfer agent is used to extract an anion from an aqueous solution into an organic medium, in several cases a small number of molecules of water of solvation are found to be associated with the ion pair, thus it is more correct to represent the ion pair of D^- as $Q^+D^- \cdot xH_2O$ rather than Q^+D^- [83]. The working principle of the CO_2-sensing film can be summarized by the following equilibrium reactions:

$$\begin{aligned} Q^+D^- \cdot xH_2O + CO_2(g) &\overset{K_1}{\rightleftharpoons} Q^+D^- \cdot xH_2O \cdot CO_2(aq), \\ Q^+D^- \cdot xH_2O \cdot CO_2(aq) &\overset{K_2}{\rightleftharpoons} Q^+D^- \cdot (x-1)H_2O \cdot H_2CO_3, \\ Q^+D^- \cdot (x-1)H_2O \cdot H_2CO_(3) &\overset{K_3}{\rightleftharpoons} Q^+HCO_3^- \cdot (x-1)H_2O \cdot HD. \end{aligned} \tag{1.17}$$

If it is assumed that K_1 and K_2 are very small and K_3 is very large, the relationship between pCO_2 and the concentrations of the two main species present, $Q^+D^- \cdot xH_2O$ and $Q^+HCO_3^- \cdot (x-1)H_2O \cdot HD$, which are likely to correspond to different fluorescence intensities, will be given by the expression:

$$\alpha^{'} pCO_2 = R = \frac{Q^+HCO_3^- \cdot (x-1)H_2O \cdot HD}{Q^+D^- \cdot xH_2O}, \tag{1.18}$$

where $\alpha^{'} = K_1 \cdot K_2 \cdot K_3$. Thus, the overall process can be summarized as follow:

$$\underbrace{Q^+D^- \cdot xH_2O}_{Fluorescent} + CO_2(g) \overset{\alpha^{'}}{\rightleftharpoons} \underbrace{Q^+HCO_3^- \cdot (x-1)H_2O \cdot HD}_{Nonfluorescent}. \tag{1.19}$$

Calibrating the sensor by plotting R versus pCO_2 yields a linear graph. The variable R in Eq. 1.18 can be calculated from experimental fluorescence measurements using the following expression [84]:

$$R = \frac{I_{max} - I_s}{I_s - I_{min}}, \tag{1.20}$$

where I_{max} and I_{min} are the fluorescence intensities in the absence and 100% of CO_2, respectively, and I_s is the observed fluorescence intensity. Combining Eq. 1.18 and 1.20, yields:

$$\alpha^{'} pCO_2 = \frac{I_{max} - I_s}{I_s - I_{min}}, \tag{1.21}$$

and rewriting Eq. 1.21 in function of pCO_2, I_s can be expressed as:

$$I_s = \frac{I_{max} + I_{min} \cdot \alpha^{'} pCO_2}{1 + \alpha^{'} pCO_2} = I_{min} + \frac{I_{max} - I_{min}}{1 + \alpha^{'} pCO_2}. \tag{1.22}$$

This expression can be used to make a calibration curve from which the pCO_2 concentration can be inversely extracted based on experimental measurements of fluorescence intensity [85, 86]. Finally, combining Eq. 1.22 and Eq. 1.4, the following equation is obtained:

$$pH = pK_a + b \log\left(\frac{1}{\alpha^{'} pCO_2}\right), \tag{1.23}$$

which represents the modified Henderson-Hasselbalch equation (Eq. 1.6), where $\alpha^{'}$ is equal to $0.03/[HCO_3^-]$.

1.4.2 Fabrication of the polymer matrix

Organic-inorganic nanocomposites represent a class of innovative materials which can be defined as heterogeneous materials in which organic and inorganic components are intimately mixed, and at least one of the phase domains has a dimension lower than 100 nm. These nanocomposites are increasingly important materials due to their peculiar properties resulting from the synergism between the characteristics of each individual phase. It is important to point out that the combination of inorganic and organic phases at the nanometer scale into a single material opens a vast new area of materials science that has extraordinary implications in the development of multifunctional materials. Organic-inorganic hybrids can be classified according to the nature of their interface: the material can be defined as a class I hybrid if the organic and inorganic phases are simply mixed and only weak bonds (van der Waals, ionic and/or hydrogen bonds) are present at their interface; on the opposite, class II hybrids are materials in which the organic and inorganic components are bonded together by strong covalent or ionic-covalent linkages. Organic-inorganic hybrid materials have gained a lot of interest due to the remarkable change in mechanical, thermal, barrier, electrical and magnetic properties compared to pure organic polymers. These hybrid materials already found commercial applications for different uses in particular in the field of protective and functional coatings for plastic, glass, ceramic and metal substrates. Different strategies can be formulated to prepare these materials referring to the following two main approaches: (a) assembling or self-assembling or dispersion of well-defined nanobuilding blocks consisting of perfectly calibrated pre-formed objects that keep their integrity in the final material; (b) synthesis based on very convenient soft chemistry including conventional sol-gel chemistry, the use of specific bridged and polyfunctional precursors and hydrothermal synthesis. Thanks to its very mild operative conditions, the most commonly applied preparation methods are based on the sol-gel process. The classical sol-gel process consists in a two-step hydrolysis-condensation reaction occurring in a solution of metal alkoxides $M(OR)_x$ (most commonly a silicon alkoxide $Si(OR)_4$) under acidic or basic conditions, according to the following reaction scheme:

$$\begin{aligned} M(OR)_4 + nH_2O &\rightarrow M(OR)_{4-n}(OH)_n + nROH \\ \equiv M-OH + HO-M \equiv &\rightarrow\equiv\ M-O-M \equiv +H_2O \\ \equiv M-OH + RO-M \equiv &\rightarrow\equiv\ M-O-M \equiv +ROH \end{aligned} \tag{1.24}$$

The presence in the reactive system of an organic oligomer or polymer bearing suitable functional groups which are reactive towards the other reactants of the sol-gel process (typically alkoxysilane groups) leads to the formation of covalent

bonds between organic and inorganic moieties, which remain phase separated on a nanometric scale (class II hybrids), but give rise to a resulting material which is macroscopically uniform. It is well known that the final morphology of these hybrid materials, as their physical properties, is strictly dependent on the characteristics of the organic polymer, such as its molecular weight and its solubility in the sol-gel solution, as well as the presence of reactive functionalities and associated chemical reaction rate.

For the preparation of the pCO_2 sensor, an organic-inorganic hybrid matrix was prepared by using an organic component based on commercial perfluoropolyethers (PFPE). PFPE are polymer chains mainly composed by -(CF_2O)- and -(C_2F_4O)- repeating units alternating along the chain; PFPE are characterized by an extremely high chain mobility, hence very low glass transition temperatures, chemical inertia, solvent and high temperature resistance, barrier properties towards water and water vapor, and a very high permeability towards CO_2 gas. These features certainly make PFPE suitable candidates for the preparation of functional materials for CO_2 sensing because their properties are suitable for the preparation of high-reponse membranes. Commercial triethoxysilane-terminated PFPE [PFPE-$Si(OEt)_3$] were reacted with tetraethoxysilane [TEOS, $Si(OEt)_4$] for the preparation of a sol-gel hybrid containing 90% wt of organic fluorinated phase and 10% wt of inorganic silica phase. Diluted HCl (37 %) was used as catalyst to induce condensation reactions between the -Si(OEt) units, for getting a crosslinked structure able to rapidly sense CO_2 without risk of physical unstability. HPTS (8-hydroxypyrene 1,3,6, trisulfonic acid trisodium salt) was dissolved in small amounts in the solution of organic and inorganic precursors and embedded inside the hybrid matrix to act as optical indicator of CO_2 level. This fluorescent dye is characterized by excitation and emission spectra similar to those of fluorescein *O*-methacrylate, which has been used for the pH sensor. Thus, the same light source and optical filters were used for the pCO_2 sensor.

1.4.3 Experimental protocol

The sensor has been firstly tested in air, then on cow blood. For the test in air, the measurement setup shown in Figure 1.25 has been realized. It consists of the multi-analyte fluorescence-based sensor platform previously described with the flow cuvette filled with carbon dioxide and nitrogen respectively, in order to simulate the condition of absence and 100% of CO_2. In order to generate CO_2 and N_2, a CO_2 cylinder and a nitrogen generator have been used. Finally, a three-way valve has been used to commute from one gas to the other one. All the measurements were performed at room temperature.

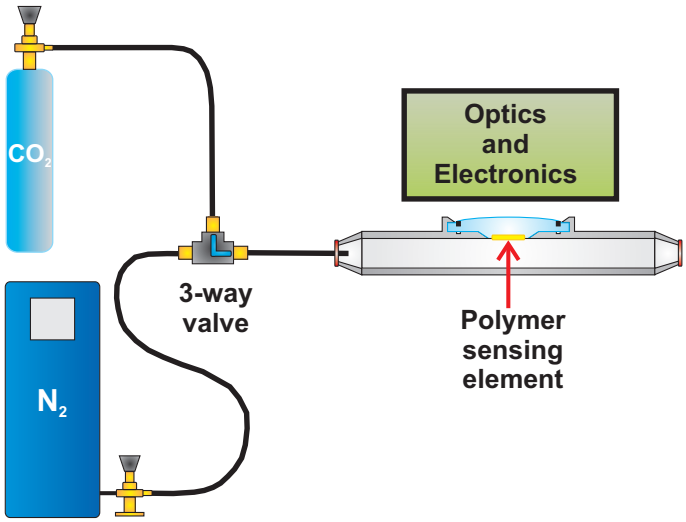


Figure 1.25: Schematic representation of the measurement setup used to test the pCO_2 sensor. It consists of the multi-analyte fluorescence-based sensor platform previously described, a CO_2 cylinder, a nitrogen generator and a three-way valve to commute between the two gases.

In the in-vitro test, the experimental protocol and the measurement setup were the same of the pH sensor, previously described in Section 1.3.4.

1.4.4 Experimental results

1.4.4.1 Test in air

In Figure 1.26, I_{Ratio} (defined in Eq. 1.2) versus time is shown, for a measurement performed in air, changing the concentration of CO_2 between 0% and 100%, using the measurement setup described in Figure 1.25. The valleys correspond to the maximum concentration of CO_2, whereas the peaks correspond to the minimum concentration of CO_2.

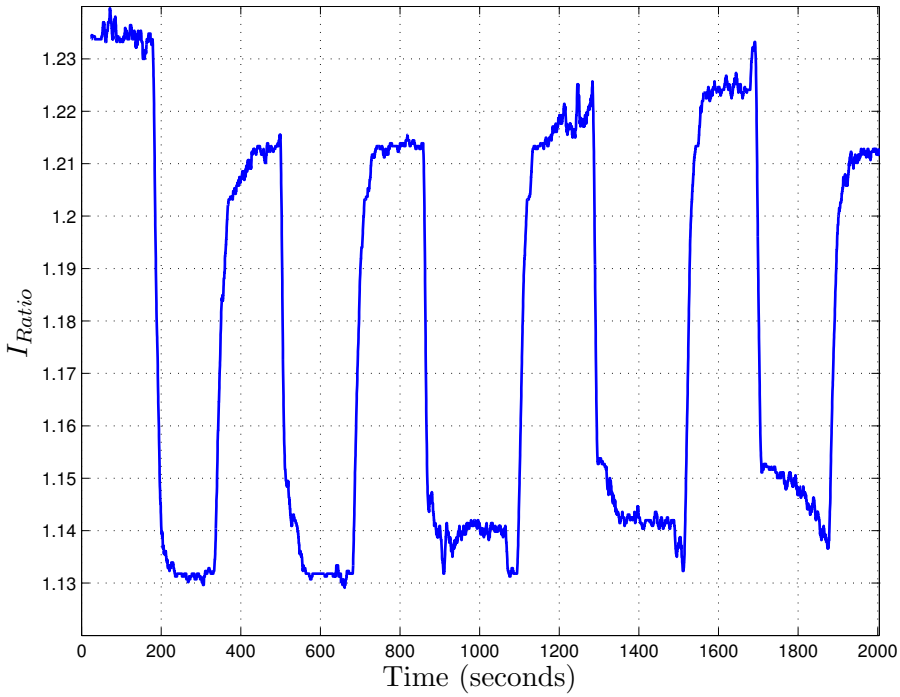


Figure 1.26: I_{Ratio} versus time acquired changing the concentration of CO_2 between 0% and 100%. The valleys correspond to the maximum concentration of CO_2, whereas the peaks correspond to the minimum concentration of CO_2.

1.4.4.2 Test in-vitro on cow blood

In Figure 1.27, I_{Ratio} versus time is shown, for a measurement performed on cow blood, changing the pCO_2 of the blood. The red dots are the average values of the I_{Ratio} signal, calculated over a time interval of 60 s, recorded at each reference pCO_2, measured by the blood gas analyzer (green dots). During this test, both the flow rate and the temperature of blood were kept constant, at a value of 2 L/min and 37°C respectively. The pCO_2 of the blood was changed using gases, in particular carbon dioxide and nitrogen. The oxygen was kept quite constant, thus the saturation was always above 90%. The hematocrit of the blood was always kept at a value of 33-34%.

The mean values of the I_{Ratio} signal versus the pCO_2 values determined by the blood gas analyzer are shown in Figure 1.28, together with the interpolation curve expressed by Eq. 1.22. I_{max}, I_{min} and α' were determined to be 0.493, 0.299 and 0.012 respectively (R=0.989).

As previously described in Section 1.4.1, there exist a linear relation between parameter R (Eq. 1.20) and measured pCO_2 values. Thus, in Figure 1.29, R values versus measured pCO_2 values are shown, together with the interpolation line (slope=80.78, intercept=2.31, R=0.9666).

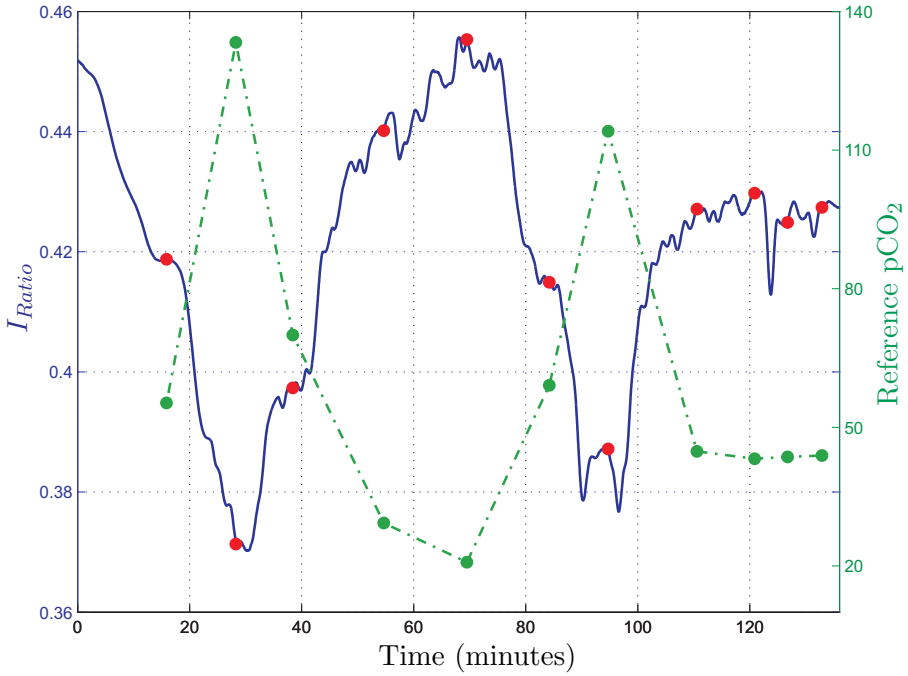


Figure 1.27: I_{Ratio} versus time acquired at different pCO_2 values during a test performed on cow blood. The red dots are the mean values of the I_{Ratio} signal, calculated over a time interval of 60 s, recorded at each reference pCO_2, measured by the blood gas analyzer (green dots)

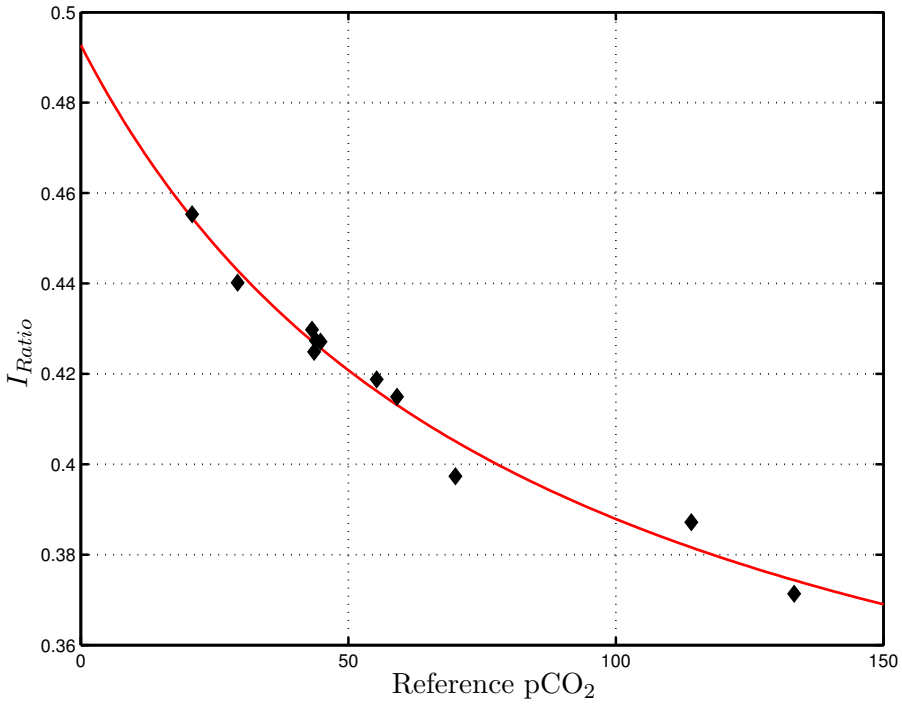


Figure 1.28: Mean values of the I_{Ratio} signal, calculated over a time interval of 60 s, versus reference pCO_2, measured by the blood gas analyzer and interpolation curve.

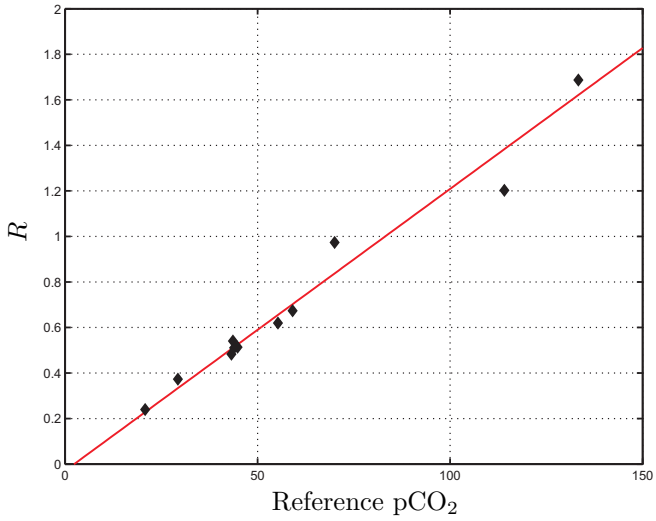
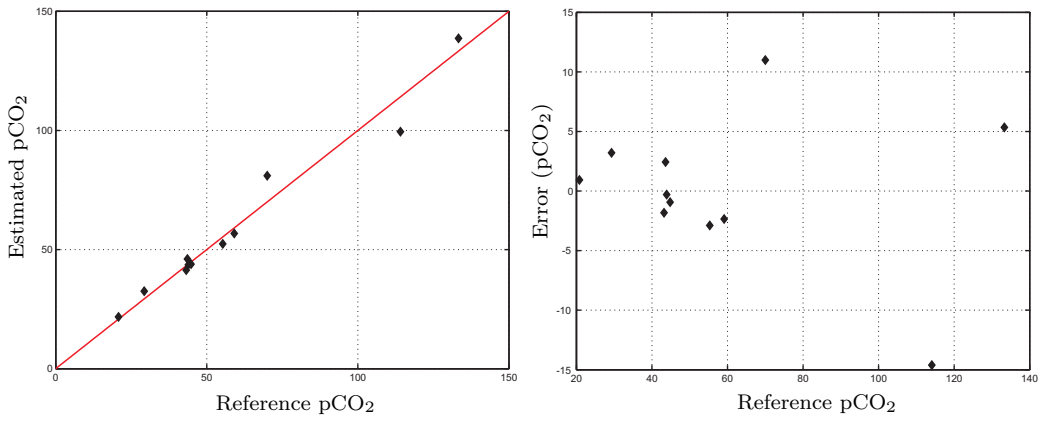


Figure 1.29: R values versus reference pCO_2, measured by the blood gas analyzer and interpolation line.

Finally, in Figure 1.30a, the estimated pCO_2 values versus the reference ones, are shown. They have been determined trough the linear interpolation of the R values. The difference between the estimated and the reference pCO_2, is shown in Figure 1.30a. The mean error value, calculated through Eq. 1.15, was determined to be 6 mmHg.



(a) (b)

Figure 1.30: Estimated pCO_2 versus reference pCO_2 and interpolation line (a). Difference between the estimated pCO_2 and the reference pCO_2, versus the reference pCO_2 (b).

1.4.5 Conclusions

A fluorescence pCO_2 sensor for measuring blood pCO_2 during ECC has been developed using the multi-analyte fluorescence-based sensor platform previously described. The sensor is based on the pH-dependent fluorescence of a purposely developed polymer matrix including a fluorescent dye (HPTS).

Preliminary tests have been performed both in air and on cow blood. The sensor response has been coherent with the pCO_2 variations, showing a linear behavior for pCO_2 concentrations between 0 and ca. 70 mmHg.

The linear range depends both on the pK_a of the fluorescent indicator and on the concentration of the phase transfer agent. Thus, by properly tuning the phase transfer agent concentration, the linear range can be increased.

1.5 Comparison with the state of the art

Table 1.7: Characteristics comparison between the developed system and the main commercial systems.

pH				
Name	**Description**	**Type of measurement**	**Measuring range**[a]	**Test range**[b]
Developed system	Optical fluorescence sensor for measuring haematic pH	Continuous non-contact measurement	6.50-9.00	7.00-8.00
ABL 800 Flex	Blood gas analyzer	Non continuous measurement	6.30-8.00	6.85-7.55
pCO_2				
Name	**Description**	**Type of measurement**	**Measuring range**	**Test range**
Developed system	Optical fluorescence sensor for measuring haematic pCO_2	Continuous non-contact measurement	10-70 mmHg	10-130 mmHg
ABL 800 Flex	Blood gas analyzer	Non continuous measurement	5-250 mmHg	17-160 mmHg

[a] The Measuring range for a parameter is the range within which the analyzer is physically capable of measuring. The measuring range corresponds to the 'range of indication' as defined in the 'International vocabulary of basic and general terms in the metrology' (VIM).

[b] The Test range for a parameter is the range within which accuracy and precision of a measured parameter has been specified and intended to lie within specified limits. The test range corresponds to the 'measuring range' as defined in the 'International vocabulary of basic and general terms in the metrology' (VIM).

Chapter 2

Electrochemical SPR sensor

2.1 Introduction

With the advent of the "genomic era" nucleic acid testing technologies have become more important than ever. Nowadays, "DNA chips" are used in several research laboratories. A variety of detection schemes is currently under investigation, ranging from surface plasmon resonance sensors [87, 88], electrochemical methods [89, 90], mass sensors [91] and DNA conductivity [92], to electronic field effect sensors [93, 94].

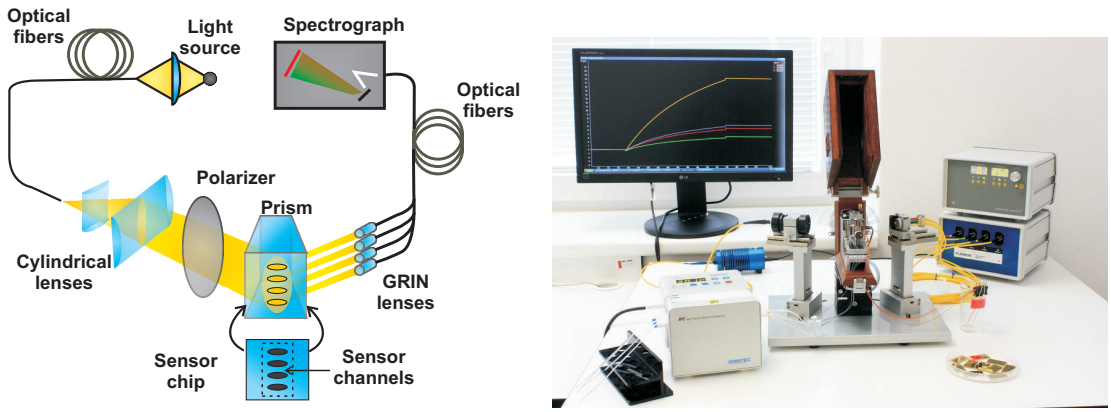
Recently, it has been recognized that the molecular packing density within the DNA layer crucially determines the functionality of the nucleic acids. For instance, the ability of surface immobilized probe strands to capture complementary target sequences from solution is largely suppressed if the layer density is too high [95, 96]. Electrically induced desorption of DNA [97] is a common technique to control the DNA surface density. Starting from relatively densely packed monolayers, the surface density is gradually reduced by applying negative potentials to the substrate which leads to a release of nucleic acids from the surface. The basic principle relies on electrostatic interactions and electrochemical reduction. Since DNA is highly negative in solutions of pH>1, it is repelled from the surface when a negative potential is applied to the substrate [98, 99]. In addition, also the sulfur-gold bond, which tethers the DNA to the surface, can be broken by electrochemical reduction through the application of negative potentials [100, 101, 102].

Surface Plasmon Resonance (SPR) is a powerful tool to monitor the changes in the refractive index at a solid-liquid interface. Thus, it has been widely used to characterize interactions between biomolecules [103, 104] and, in particular, to study DNA hybridization [105, 106, 107]. In addition, since SPR needs a metal film, it can be inherently combined with electrochemical (EC) methods, where a metal film on a glass slide is usually used as both the surface plasmon medium and the working electrode [108, 109, 110, 111].

In this chapter the design and the realization of a system which combines electrochemical and SPR techniques on the same sensing chip is presented. A comparison between the acquired electrochemical and SPR signals has been performed. Furthermore, the electrochemical SPR biosensor was used to study DNA desorption and regeneration. In the following Section, the system will be described. In particular, after a brief theoretical introduction, the design and realization of the electrochemical electrodes will be presented in Section 2.2.2, while in Section 2.2.3 the realized custom electronics will be described, together with the management software. Finally, in Section 2.3, the experimental results are reported and discussed.

2.2 Materials and Methods

The developed system is based on a four-channel SPR sensor system developed at the Institute of Photonics and Electronics AS CR (Prague) [112], shown in Figure 2.1.



(a) (b)

Figure 2.1: Schematic representation (a) and picture (b) of a four-channel SPR sensor system developed at the Institute of Photonics and Electronics AS CR, Prague.

The sensor consists of a halogen lamp, a SPR sensor platform and a spectrometer. White light from the halogen lamp is brought to the SPR sensor platform via a multimode optical fiber. The sensor platform comprises an input collimator producing a large diameter parallel beam, a glass prism with an attached SPR chip (coated with a 50 nm thick gold layer), polarizer and multichannel output collimator. The output collimator couples the light into optical fibers, which are connected to inputs of the spectrograph. Since the SPR system provides four independent channels, an in-situ electrochemical cell has been realized in each channel, allowing four simultaneous electrochemical measurements. In particular, the gold substrate

used for SPR measurement was used as the working electrode and two further planar gold electrodes (the reference and the counter electrodes) were inserted in each channel of the flow cell. Together with the realization of the electrochemical cells, a custom electronics has been designed and realized. The electronics is able to set an arbitrary potential between the reference and working electrodes and to measure the current flowing between the counter and the working electrodes.

2.2.1 Theoretical background

The developed system aims to combine SPR and electrochemical techniques on the same sensing chip. In the following paragraphs, a brief theoretical introduction on SPR sensors and on electrochemical techniques is given.

2.2.1.1 Surface Plasmon Resonance sensors

In principle, Surface Plasmon Resonance (SPR) sensors are thin-film refractometers that measure changes in the refractive index occurring at the surface of a metal film supporting a surface plasmon [113]. A surface plasmon excited by a light wave propagates along the metal film, and its evanescent field probes the medium (sample) in contact with the metal film. A change in the refractive index of the dielectric gives rise to a change in the propagation constant of the surface plasmon, which alters the characteristics of the light wave coupled to the surface plasmon (e.g., coupling angle, coupling wavelength, intensity, phase). Four different kinds of modulation can be exploited in SPR sensors, on the basis of which characteristic of the light wave modulated by a surface plasmon is measured:

angular modulation a monochromatic light wave is used to excite a surface plasmon. The strength of coupling between the incident wave and the surface plasmon is observed at multiple angles of incidence, typically by employing a convergent light beam. The excitation of surface plasmons is observed as a dip in the angular spectrum of reflected light. The angle of incidence yielding the strongest coupling is measured and used as a sensor output;

wavelength modulation a surface plasmon is excited by a collimated light wave containing multiple wavelengths, typically a beam of polychromatic light. The excitation of surface plasmons is observed as a dip in the wavelength spectrum of reflected light. The wavelength yielding the strongest coupling is measured and used as a sensor output;

intensity modulation a surface plasmon is excited by a light wave at a single angle of incidence and wavelength, and the intensity of light wave serves as a

sensor output;

phase modulation a surface plasmon is excited by a light wave at a single angle of incidence and wavelength, and the shift in phase of the light wave coupled to the surface plasmon is measured as the sensor output.

A light wave can couple to a surface plasmon at a metal-dielectric interface if the component of light's wavevector that is parallel to the interface matches the propagation constant of the surface plasmon. As the propagation constant of a surface plasmon at a metal-dielectric interface is larger than the wavenumber of the light wave in the dielectric, surface plasmons cannot be excited directly by light incident onto a smooth metal surface. The wavevector of light can be increased to match that of the surface plasmon by the attenuated total reflection or diffraction. This enhancement and subsequently the coupling between light and a surface plasmon are performed in a coupling device, such as a prism coupler. In the Kretschmann configuration of the attenuated total reflection (ATR) method (Figure 2.2)[114], a light wave passes through a high refractive index prism and is totally reflected at the base of the prism, generating an evanescent wave penetrating a thin metal film.

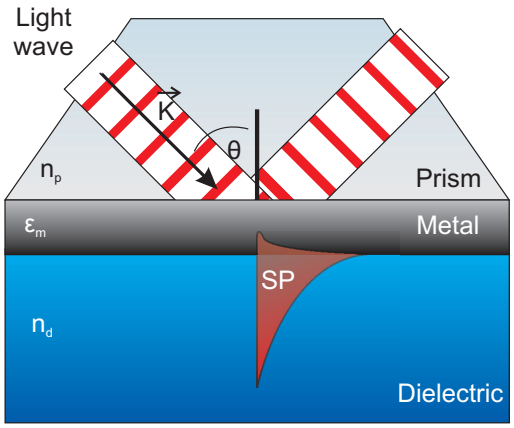


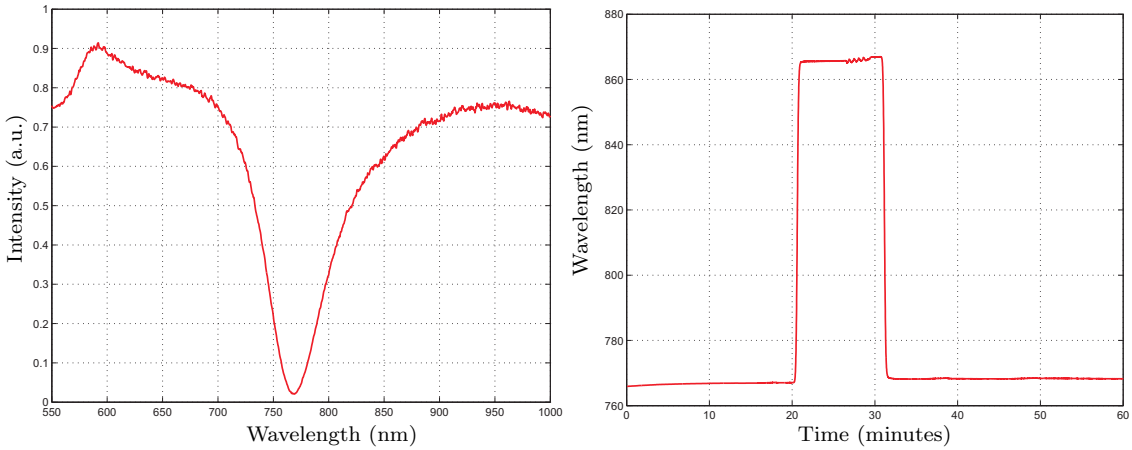
Figure 2.2: Coupling of light to a surface plasmon via a prism coupler.

The evanescent wave propagates along the interface with the propagation constant, which can be adjusted to match that of the surface plasmon by controlling the angle of incidence. Thus, the matching condition

$$\frac{2\pi}{\lambda} n_p sin(\theta) = Re\beta_{SP} \tag{2.1}$$

can be fulfilled, allowing the evanescent wave to be coupled to the surface plasmon. θ denotes the angle of incidence, n_p denotes the refractive index of the prism ($n_p > n_d$), and β_{SP} denotes the propagation constant of the surface plasmon. Figure 2.3a shows a typical SPR spectrum, with the dip in correspondence to the

resonance wavelength. Through an ad-hoc algorithm, the dip is tracked and plotted versus time, obtaining the curve shown in Figure 2.3b.



(a) (b)

Figure 2.3: SPR spectrum, with a dip in corrispondence to the resonance wavelength (a) and the tracked dip versus time (b).

2.2.1.2 Electrochemical techniques

The area of analytical chemistry which studies the movement of electrons in an oxidation-reduction reaction is called electrochemistry. In particular, electrochemical techniques deploy the measurement of potential, charge, or current to determine an analyte concentration or to characterize an analyte chemical reactivity [115, 116].

Although there are only three basic electrochemical signals, there are several possible experimental designs. The simplest division of electrochemical techniques is between bulk techniques, where a property of the solution in the electrochemical cell is measured, and interfacial techniques, where the potential, charge, or current depend on the species present at the interface between an electrode and the solution. The measurement of a solution conductivity, which is proportional to the total concentration of dissolved ions, is an example of a bulk electrochemical technique. A determination of pH using a pH electrode is an example of an interfacial electrochemical technique.

Electrochemical measurements are made in an electrochemical cell consisting of two or more electrodes and the electronic circuitry for controlling and measuring the current and the potential.

The simplest electrochemical cell uses two electrodes. The potential of one electrode is sensitive to the analyte concentration, and is called *working electrode*. The

second electrode, called *auxiliary electrode*, completes the electrical circuit and provides a reference potential to measure the working electrode potential.

Ideally the counter electrode potential remains constant. Nevertheless, since in several applications it is not constant, the auxiliary electrode is replaced with two electrodes: a *reference electrode*, whose potential remains constant, and a *counter electrode* that completes the electrical circuit.

Basically, there are three experimental designs: (i) measurement of potential with zero current, (ii) measurement of potential while controlling the current, and (iii) measurement of current while controlling the potential.

Each of these experimental designs uses a different type of instrument, in particular: potentiometer to measure the potential under zero current condition, galvanostat, to control the current flowing in the electrochemical cell and finally potentiostat, to control the potential of the working electrode. In particular, potentiostats are amplifiers used to control the potential between the working and the reference electrode, to a constant value, as schematically shown in Figure 2.4.

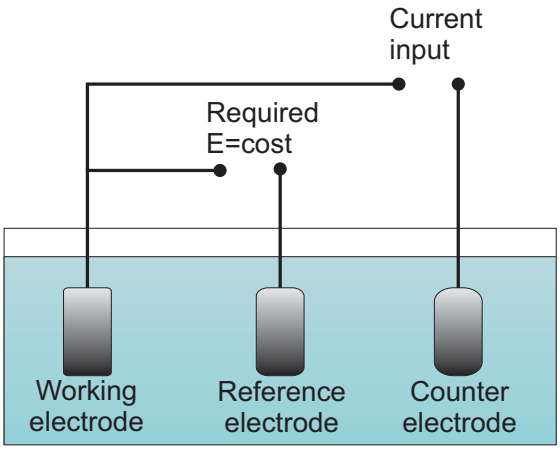


Figure 2.4: Schematic representation of the working principle of a potentiostat.

A current is forced to flow between the counter and the working electrodes in order to keep the potential of the working electrode to a fixed value with respect to the reference electrode. Thus, the potentiostat has to measure the potential difference between the working and the reference electrode, compare this value with a preset voltage and set a current between counter and working electrodes to counteract the difference between the preset voltage and the actual working electrode potential.

Interfacial electrochemical techniques are divided in two main categories: static and dynamic techniques. In the former a current is not allowed to pass through the analyte solution (e.g. potentiometry), whereas in the latter a current can flow in the analyte solution. In addition, dynamic techniques are divided in controlled-current

techniques (e.g. controlled-current coulometry) and controlled-potential techniques. Finally, the last ones are divided in variable and fixed potential techniques, such as cyclic voltammetry and chronoamperometry respectively. In the next Sections, a brief description of these two techniques is provided.

2.2.1.2.1 Cyclic voltammetry In a cyclic voltammetry, the potential applied to the working electrode is changed in order to complete a scan in both direction. In particular, a typical potential-excitation signal is shown in Figure 2.5a. In this example, the potential is initially scanned to more positive values, resulting in the following oxidation reaction for the species R:

$$R \rightleftharpoons O + ne^{-}. \tag{2.2}$$

When the potential reaches a predetermined switching potential, the direction of the scan is reversed towards more negative potentials. The species O, generated during the forward scan, is reduced back to R during the reverse scan.

$$O + ne^{-} \rightleftharpoons R. \tag{2.3}$$

The graph obtained by plotting the measured current versus the applied potential is called voltammogram. A typical voltammogram is shown in Figure 2.5b. It has separate peaks for the oxidation and the reduction reactions, each characterized by a peak potential and a peak current.

The peak current in cyclic voltammetry is given by the Randles-Sevcik equation:

$$i_P = (2.69 \cdot 10^5)\, n^{\frac{3}{2}}\, A D^{\frac{1}{2}}\, \nu^{\frac{1}{2}}\, C = KC, \tag{2.4}$$

where n is the number of electrons in the redox reaction, A is the area of the working electrode, D is the diffusion coefficient for the electroactive species, ν is the scan rate, and C is the concentration of the electroactive species at the electrode. For a well-behaved system, the anodic and cathodic peak currents are equal, and the ratio $i_{p,a}/i_{p,c}$ is 1.00. The half-wave potential, $E_{1/2}$, is midway between the anodic and cathodic peak potentials:

$$E_{1/2} = \frac{E_{p,a} + E_{p,c}}{2}. \tag{2.5}$$

Scanning the potential in both directions allows to explore the electrochemical behavior of species generated at the electrode.

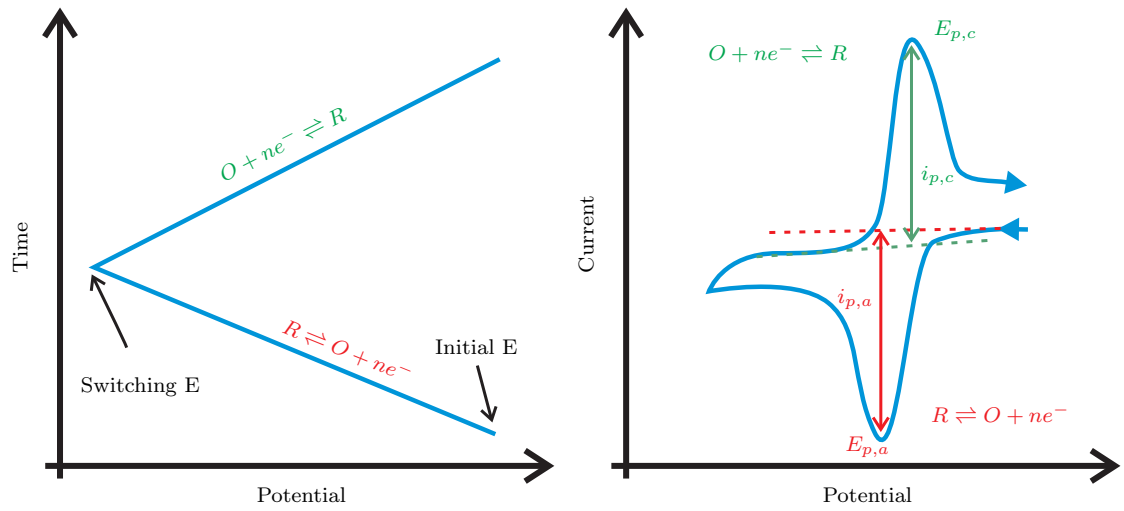


Figure 2.5: One cycle of the triangular potential-excitation signal showing the initial potential and the switching potential (a) and resulting cyclic voltammogram showing the measurement of the peak currents and peak potentials (b).

2.2.1.2.2 Chronoamperometry In chronoamperometry technique, a constant potential is applied to the working electrode and the current is measured as a function of time.

2.2.2 Electrodes realization

Three electrodes had to be placed inside each channel of the flow cell, in order to combine the SPR and electrochemical techniques. The gold substrate used for SPR measurement was used as the working electrode and two further planar gold electrodes (the reference and the counter electrodes) were deposited on a PCB and inserted in the same channel, as schematically shown in Figure 2.6.

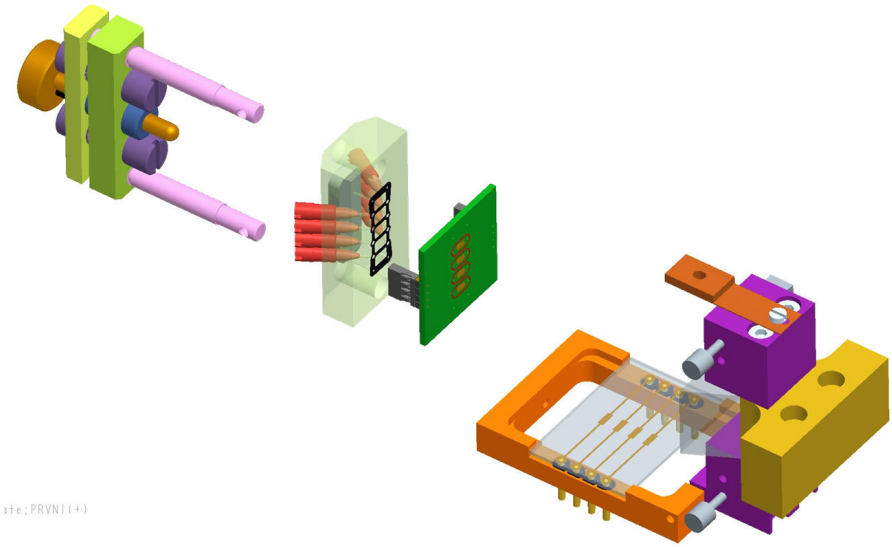
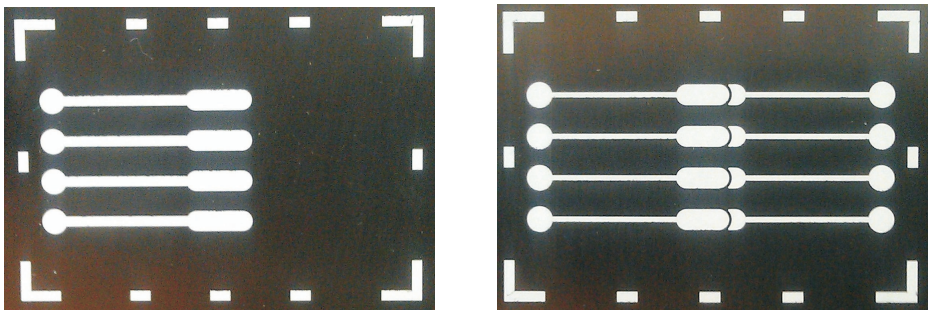


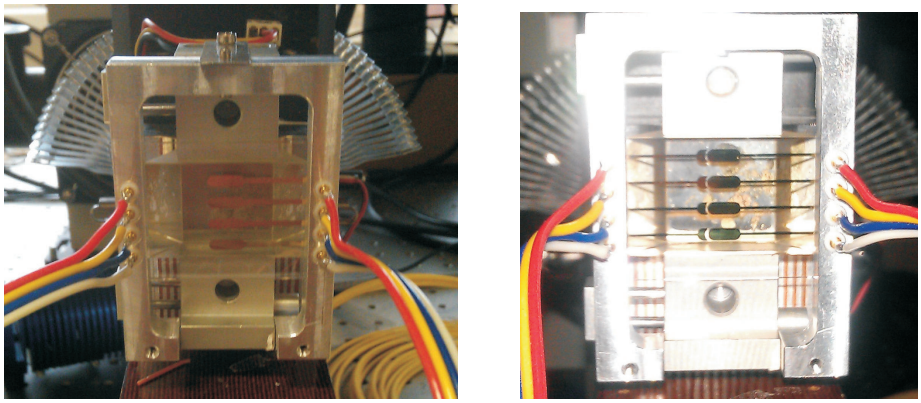
Figure 2.6: Schematic representation of an open flow cell. The gold substrate of the SPR chip was used as the working electrode and two further planar gold electrodes were deposited on a PCB and inserted in the same flow cell.

Two different geometries have been proposed; in the first one the working and the reference electrodes are realized on the gold substrate used for the SPR measurement while the counter electrode is realized on the upper part of the flow cell (PCB); in the second one only the working electrode is realized on the gold substrate and the reference and counter electrodes are realized in the upper part of the flow cell (PCB). The electrodes on the glass substrate used for SPR measurement were realized through gold evaporation. Two different masks were designed and realized for the two different geometries. The gold layer was 50 nm thick and before depositing the gold, a 5 nm thick Ti layer has been deposited. In Figure 2.7, the masks and two SPR chips (with different geometries) are shown.

The electrodes on PCB, shown in Figure 2.8, were realized in chemical gold.

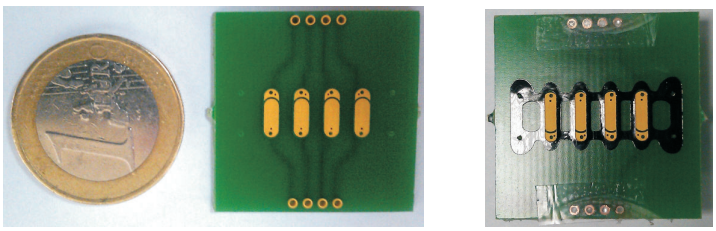


(a) (b)



(c) (d)

Figure 2.7: Pictures of the realized masks and SPR chips, with only the working electrode on the SPR chip (a),(c), and with both the working and the reference electrodes on the SPR chip (b),(d).



(a) (b)

Figure 2.8: Pictures of the electrodes realized in chemical gold on a PCB. It can be noticed the presence of two electrodes, the reference and the counter ones. In addition, around the electrodes, a vynil gasket mask has been placed to form the boundaries of the flow cell. (b)

The flow cell was built using the SPR glass chip and the electrodes PCB as the lower and upper surface respectively, while using a vynil gasket (60 um thick) to form its boundaries. Finally, the electrical contacts were realized through wire connectors for PCB electrodes whereas using spring loaded contacts, directly inserted in the mechanical cage, for the electrodes realized on the SPR glass chip. A scheme of a section of the whole flow cell is shown in Figure 2.9.

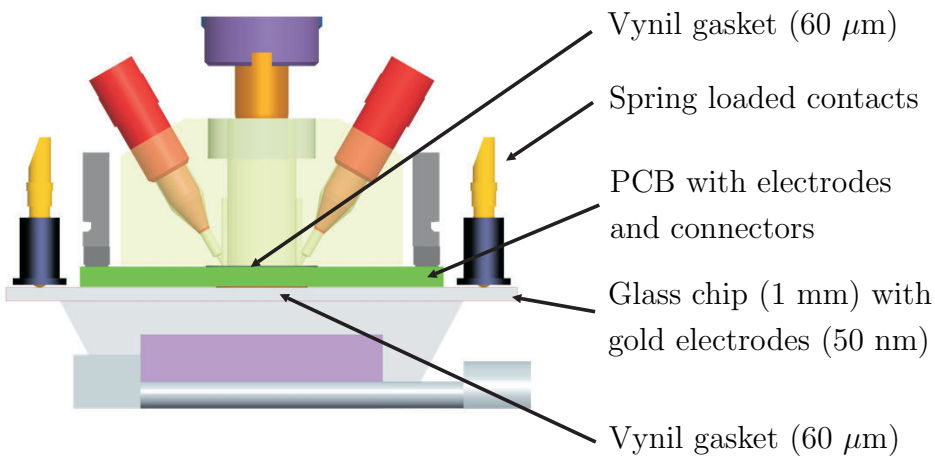
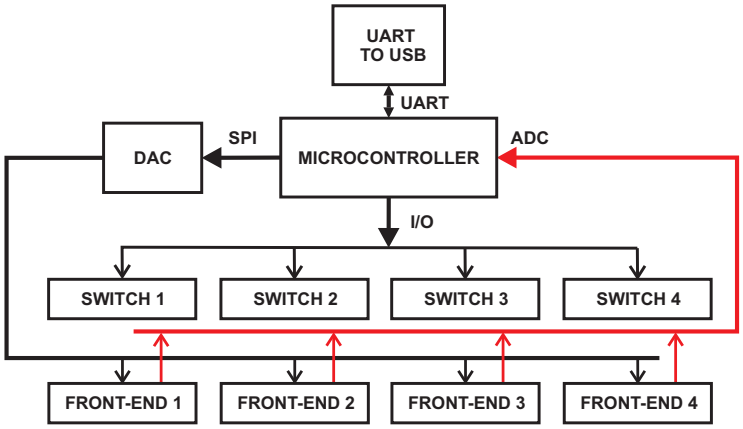


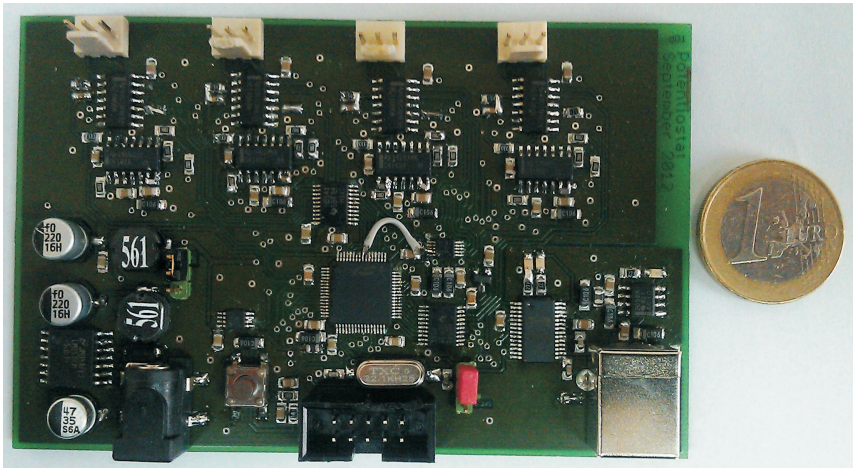
Figure 2.9: Schematic representation of a section of the flow cell. It can be noticed: the glass chip with gold electrodes, the vynil gasket and the PCB with the other gold electrodes and connectors.

2.2.3 Electronics

The realized electronics is able to set an arbitrary potential between the working and reference electrodes, to force and measure a current flowing between the counter and the working electrodes and to communicate with a PC. A scheme and a picture of the developed electronics is shown in Figure 2.10.



(a)



(b)

Figure 2.10: Schematic representation (a) and picture (b) of the developed electronics. It basically consists of two parts, a digital and an analog one. The former controls the board and communicates with the PC, while the latter represents the front-end electronics, which has to measure a potential between the working and the reference electrodes and force a current between the working and the counter electrodes.

Basically, it consists of a digital and an analog part. The former controls the board and communicate with the PC, the latter represents the front-end part, which measures the potential between the reference and the working electrodes and force

a current between the working and the counter electrodes. A brief description of both parts will be provided in the next paragraphs.

2.2.3.1 Front-ent electronics

The front-end electronics is based on the Wenking potentiostat scheme [117]. This architecture has been integrated with analog switches, thus allowing to activate each channel separately and to change the gain of the transimpendace amplifier (Figure 2.11).

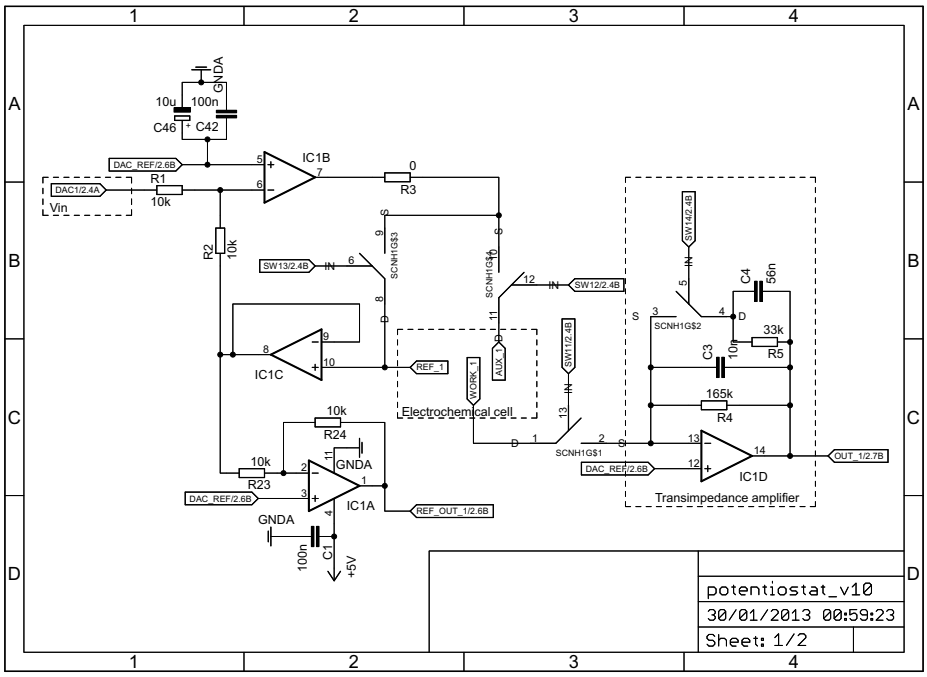


Figure 2.11: Schematic representation of the front-end electronics. It consists of a Wenking potentiostat circuit and a transimpedance amplifier to measure the current flowing between the working and the counter electrodes. Analog switches are used to select the active channels and to change the gain of the transimpedance amplifier.

2.2.3.2 Digital electronics

The digital part consists of a microcontroller (C8051F005, Silicon Labs), an UART-to-USB converter (FT232R, FTDI Chip) and a four channel DAC (DAC7554, Texas Instrument). The DAC generates the input voltages for each channel independently.

2.2.4 Software interface

A software interface has been realized in LabVIEW to manage all the system. In particular, it allows to set the active channels and to use already implemented electrochemical techniques, such as cyclic voltammetry and chronoamperometry. In addition, it allows to set a square wave with tunable amplitude and frequency (range -2 V to 2 V, frequency: 0.2-2000 Hz). In Figure 2.12, the main interface is shown, whereas the VI hierarchy is reported in Figure 2.13.

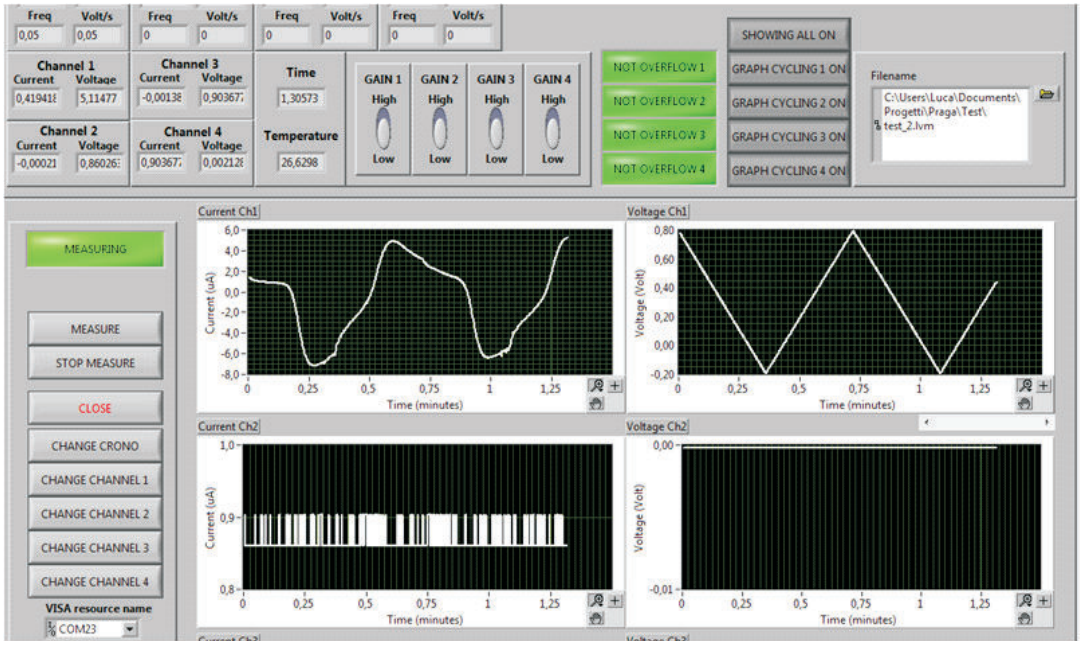


Figure 2.12: Developed software interface to manage all the system. In particular, it allows to select the active channels and the electrochemical techniques.

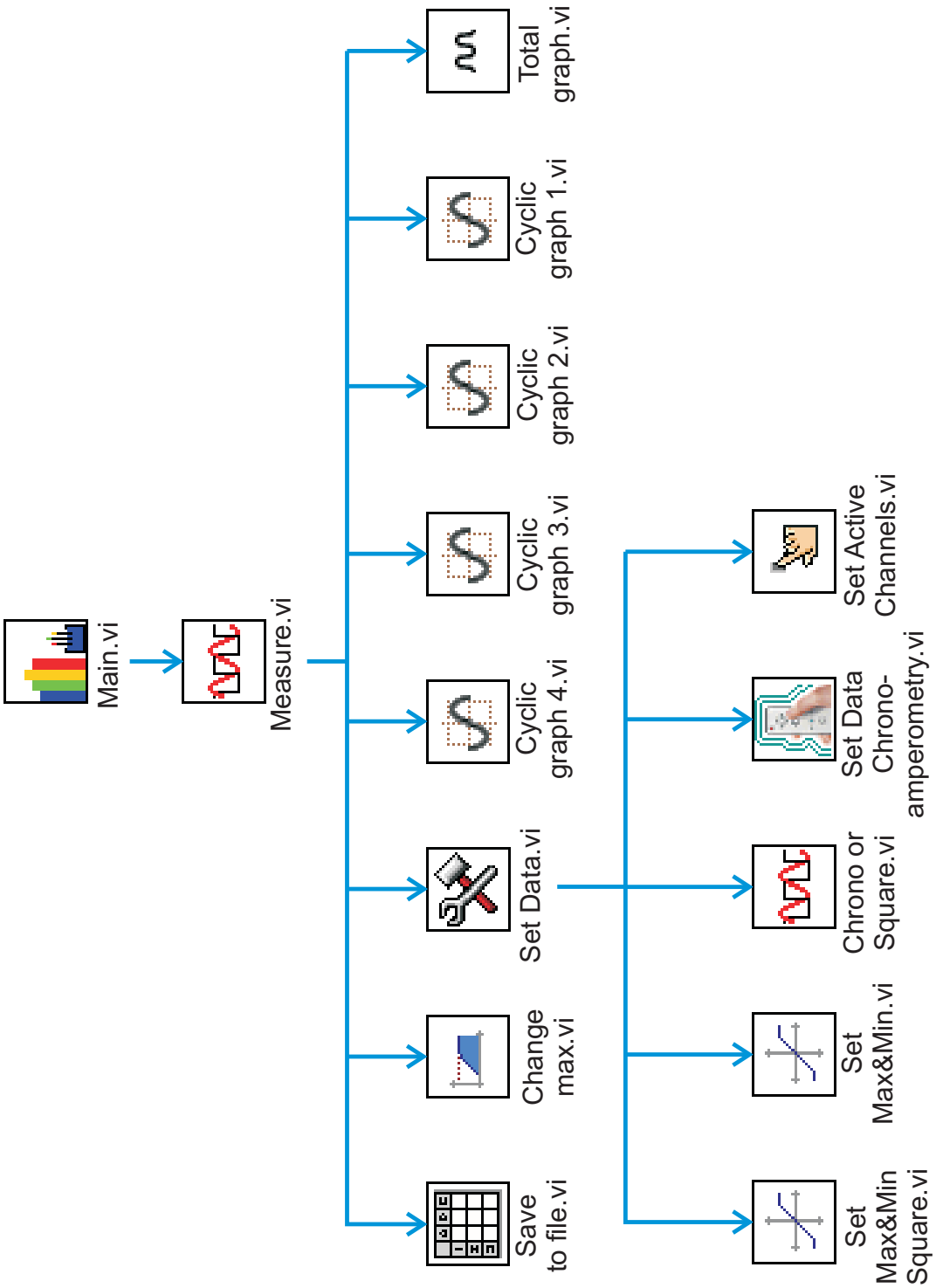
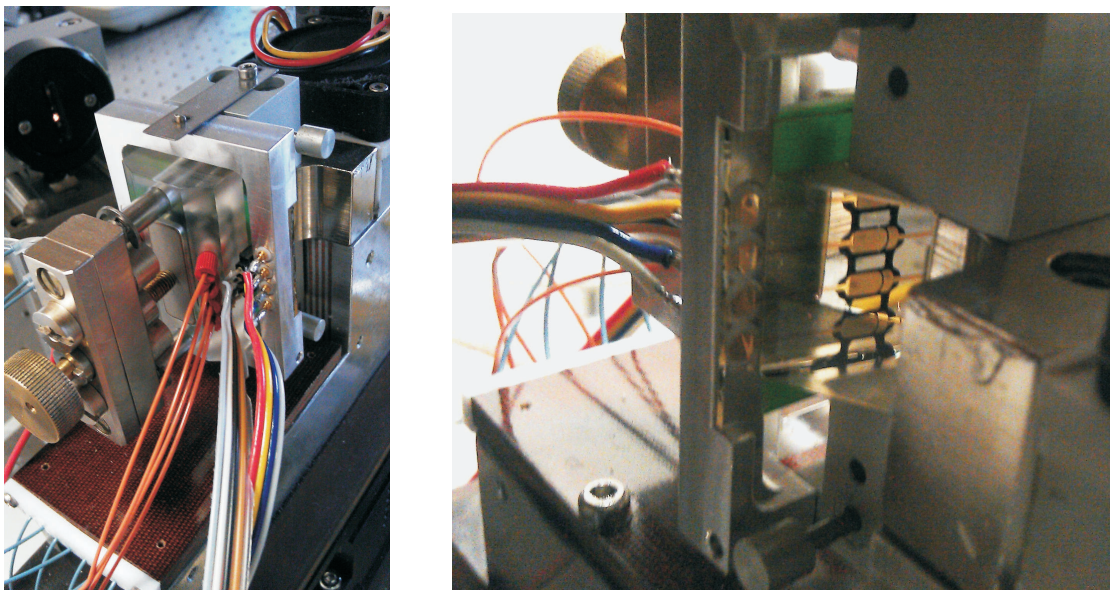


Figure 2.13: VI hierarchy of the software interface, developed in LabVIEW, to manage all the system.

2.3 Experimental results

The system has been assembled following the scheme shown in Figure 2.6. Pictures of the flow cell are shown in Figure 2.14 representing the front and the back view respectively.



(a) (b)

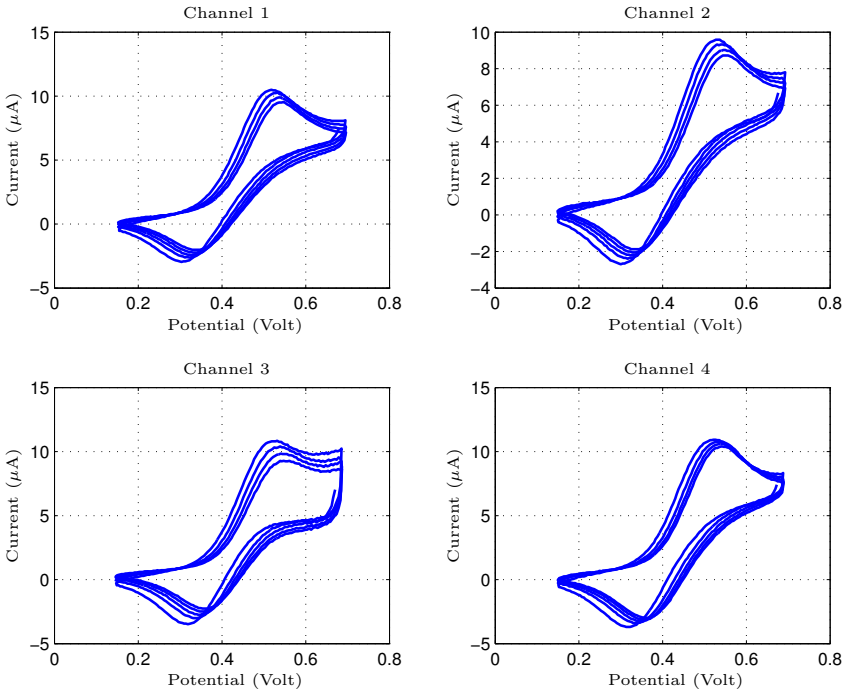
Figure 2.14: Pictures of the flow cell. Front (a) and back view (b). It can be noticed the gold electrodes from the back view (seen through the prism coupler).

During the experimental tests, the signals from the potentiostat and from the SPR system were recorded simultaneously, in order to compare the acquired data. Firstly, a cyclic voltammetry in buffer solution (PBS 100 mM, pH 7.15) with ferricyanide (1 mM) has been performed, in order to characterize the electrochemical cell, using both the geometries. Then, several tests have been done to study the influence of the electrical potential on DNA desorption and regeneration.

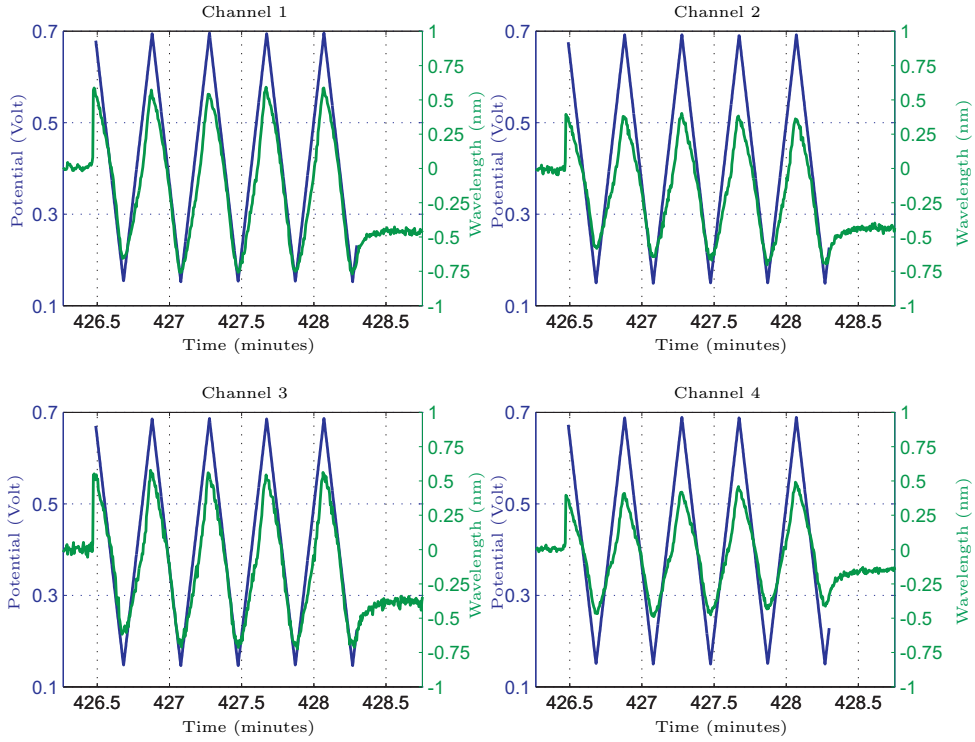
2.3.1 Cyclic voltammetry

In Figure 2.15 and Figure 2.16, the cyclic voltammetries performed in PBS with ferricyanide ($Fe(CN)_6^{3-}$) with the two electrochemical cells are shown.

The cyclic voltammetry is useful to determine the pseudo-reference potential. In particular, this corresponds to the middle potential between the reduction and oxidation peaks of $[FeCN_6]^{3-}$ and $[FeCN_6]^{4-}$ respectively. Thus, it was calculated to be 0.45 V and 0.16 V for the first and the second kind of geometry respectively.

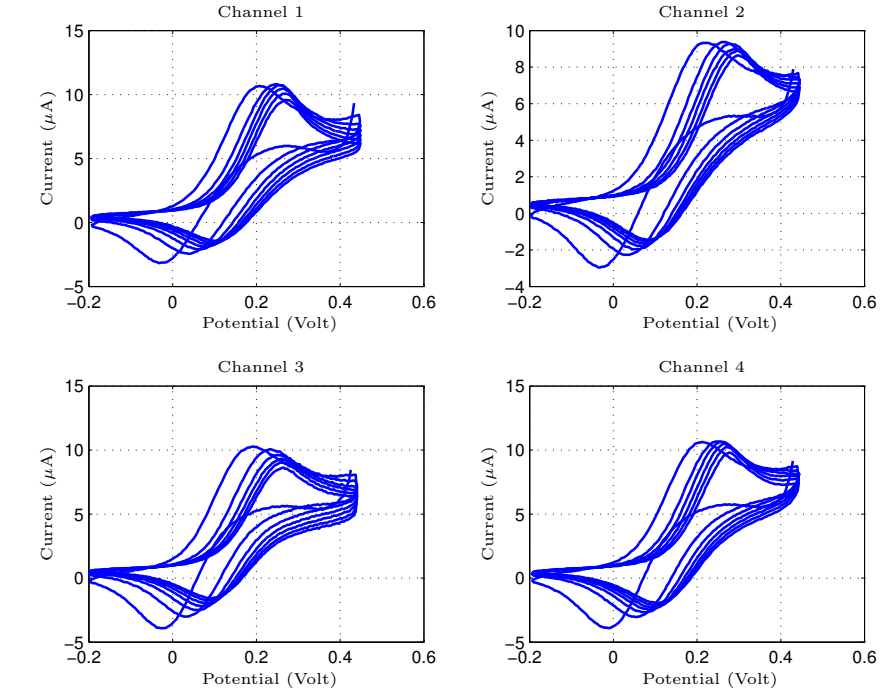


(a)

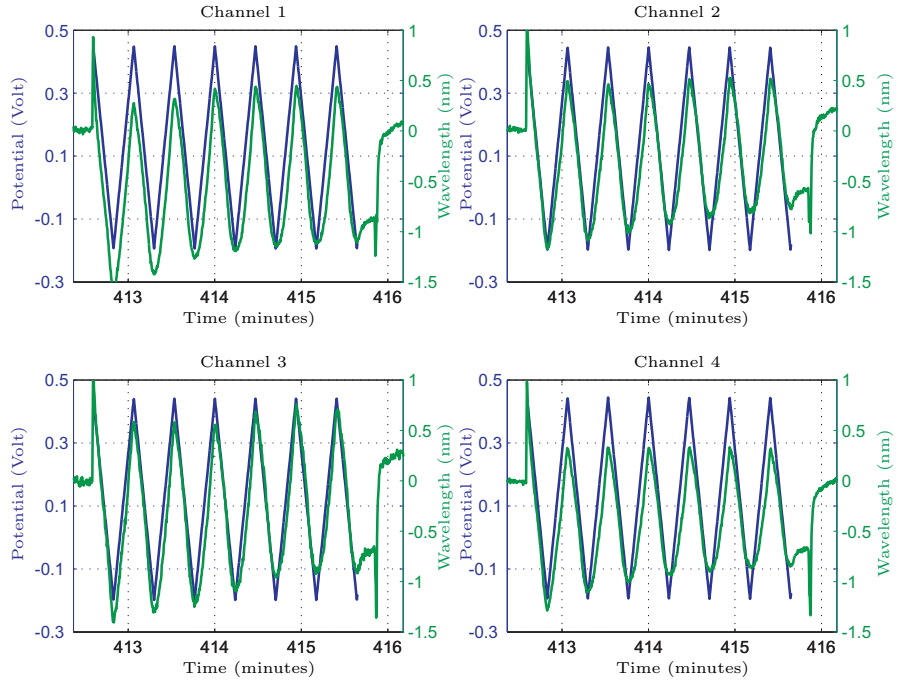


(b)

Figure 2.15: Cyclic voltammetry performed in PBS (100 mM) with ferricyanide (1 mM) in all the four channels using the first geometry (only working electrode on the SPR chip) (a). Applied potential (blue) and recorded SPR signal (green) versus time (b).



(a)



(b)

Figure 2.16: Cyclic voltammetry performed in PBS (100 mM) with ferricyanide (1 mM) in all the four channels using the second geometry (both working and reference electrodes on the SPR chip) (a). Applied potential (blue) and recorded SPR signal (green) versus time (b).

2.3.2 DNA regeneration

The following experimental tests were performed to study the influence of the applied potential on DNA desorption and regeneration. The chip was functionalized with single stranded DNA probes (ssDNA) (Sd273BATT - 16-mer) by exposing the surface to buffered aqueous DNA solution. Then, the chip was exposed to mercaptohexanol (MCH), which has been used as the blocking layer. The complementary DNA (cDNA) was used for the preparation of double stranded DNA. When the complementary DNA tethers with the DNA probes, a shift in the SPR wavelength is recorded. In the first test, the experimental protocol reported in Table 2.1 has been followed.

Table 2.1: Experimental protocol for the preparation of the SPR chip.

Component	Time (minutes)	Flow ($\mu l/min$)
DNA probes (ssDNA)	15	6.69
Blocking Layer (MCH)	20	6.69
PBS (100 mM pH=7.15)	3	30
NaOH	5	30
PBS (100 mM pH=7.15)	5	30
Complementary DNA (cDNA)	10	30
PBS (100 mM pH=7.15)	5	30
NaOH	3	30
PBS (100 mM pH=7.15)	10	30

At the end of each protocol, a negative potential was applied to the SPR chip, in order to completely remove both the DNA probes and the blocking layer, i.e. regenerate the chip. In Figure 2.21a, the SPR signals of the first response to the cDNA is shown, without having applied any potential. After this test, different negative potentials were applied to the four channels, for 30 seconds, to study their effect on DNA regeneration. All the tests were performed at a temperature of $15° C$.

2.3.2.1 1° test

In the first test, the following negative potentials were applied to the SPR chip:

- Channel 1: -0.4 V;
- Channel 2: -0.5 V;

- Channel 3: -0.6 V;
- Channel 4: -0.7 V.

In Figure 2.17a, SPR signals and applied potentials are shown, whereas Figure 2.17b shows SPR signals and measured currents.

After the application of the negative potentials, the protocol described in Table 2.1 was applied. Finally, in Figure 2.21b the response to cDNA is shown.

2.3.2.2 2° test

In the second test, the following negative potentials were applied to the SPR chip:

- Channel 1: -0.7 V;
- Channel 2: -0.6 V;
- Channel 3: -0.5 V;
- Channel 4: -0.4 V.

In Figure 2.18a, SPR signals and applied potentials are shown, whereas Figure 2.18b shows SPR signals and measured currents.

After the application of the negative potentials, the protocol described in Table 2.1 was applied. Finally, in Figure 2.21c the response to cDNA is shown.

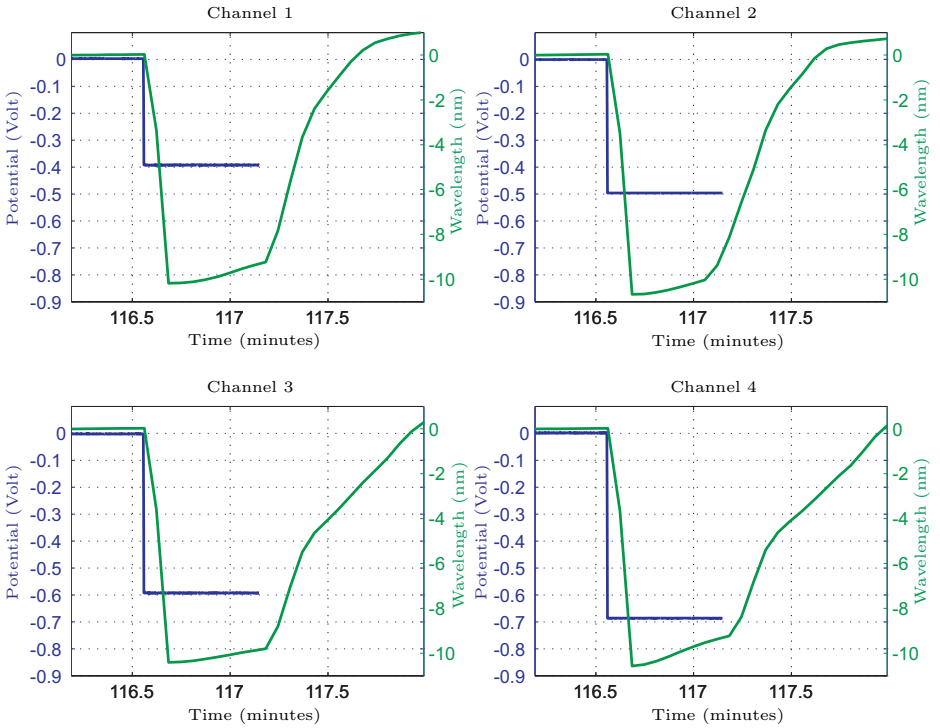
2.3.2.3 3° test

In the third test, the following negative potentials were applied to the SPR chip:

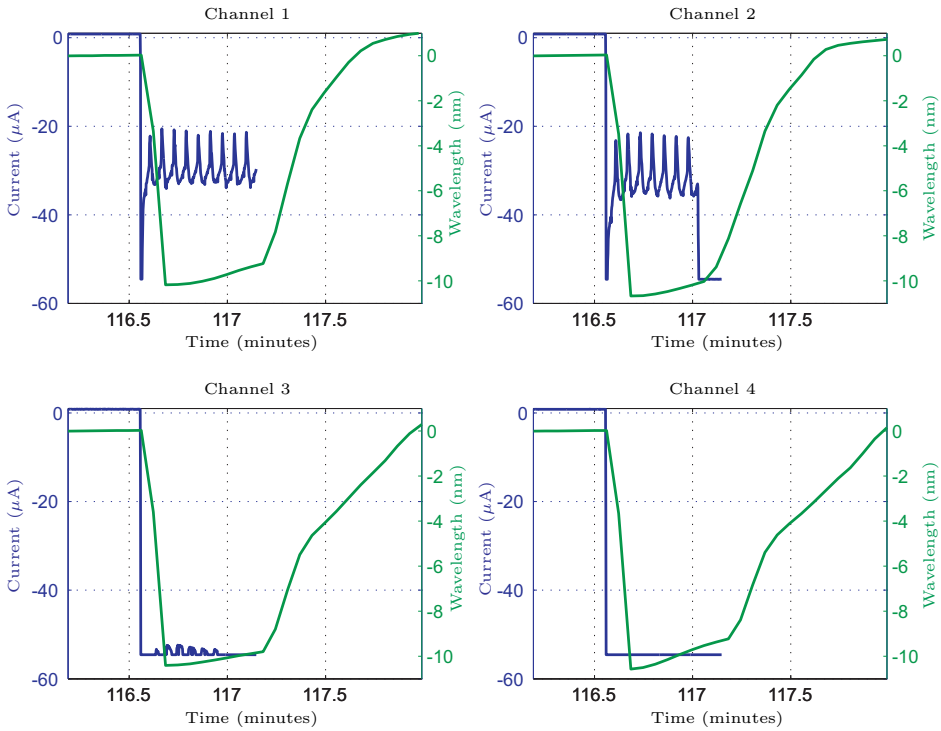
- Channel 1: -0.5 V;
- Channel 2: -0.5 V;
- Channel 3: -0.5 V;
- Channel 4: -0.5 V.

In Figure 2.19a, SPR signals and applied potentials are shown, whereas Figure 2.19b shows SPR signals and measured currents.

After the application of the negative potentials, the protocol described in Table 2.1 was applied. Finally, in Figure 2.21d the response to cDNA is shown.

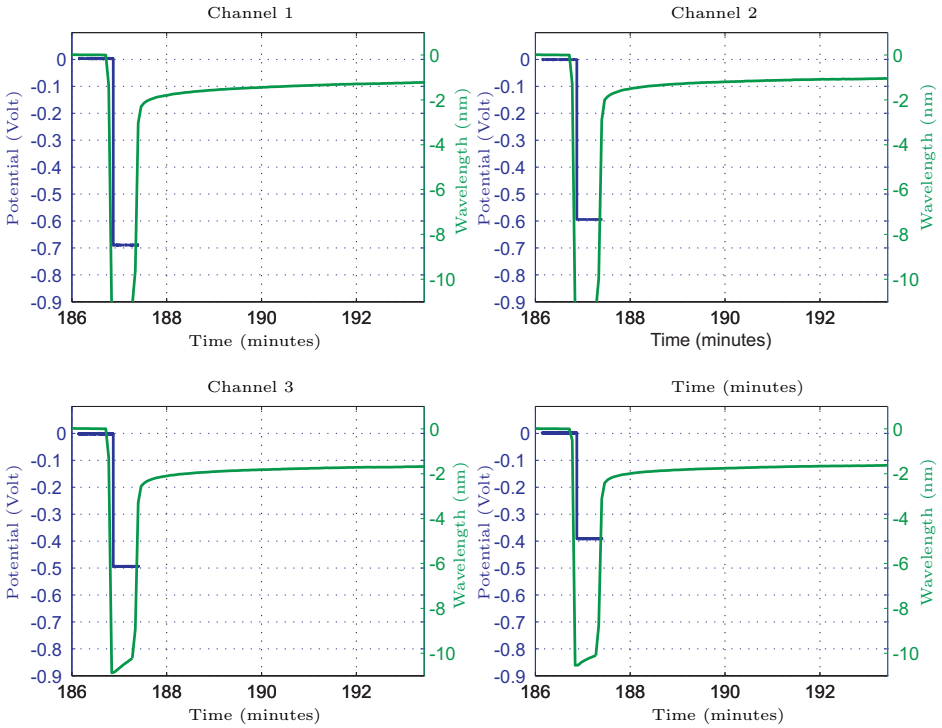


(a)

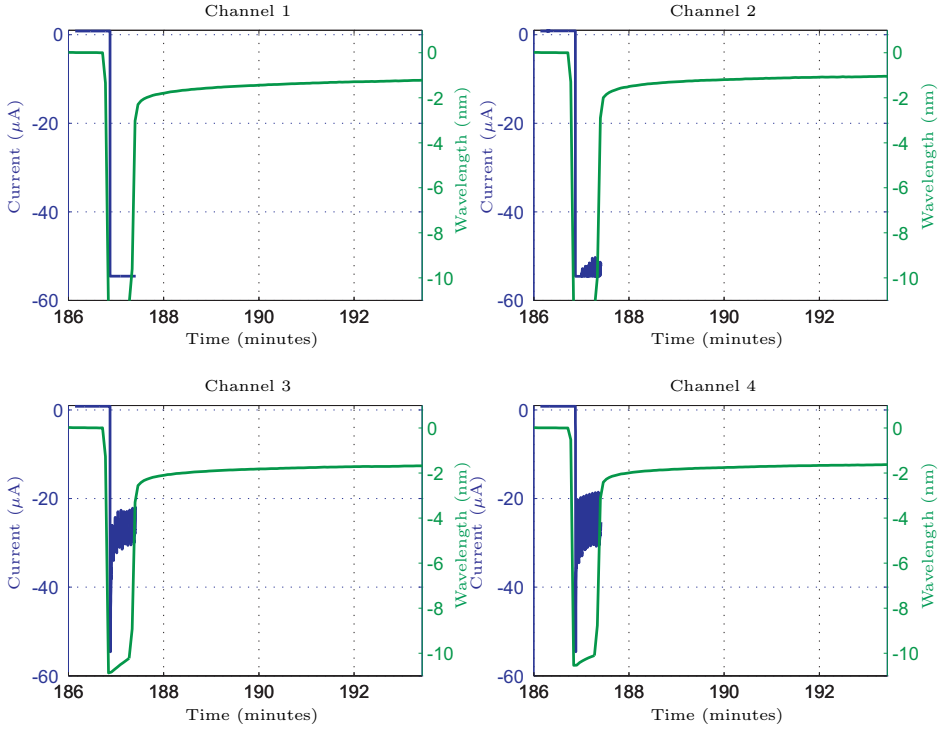


(b)

Figure 2.17: SPR signals (green) and applied potentials (blue) (a), SPR signals (green) and measured currents (blue) (b), 1° test.

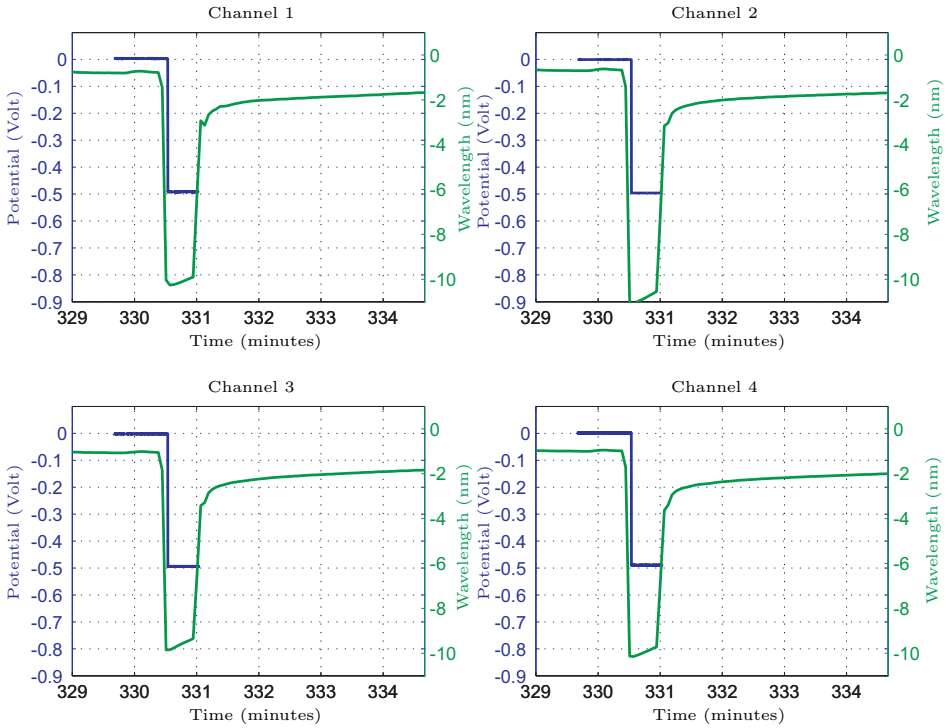


(a)

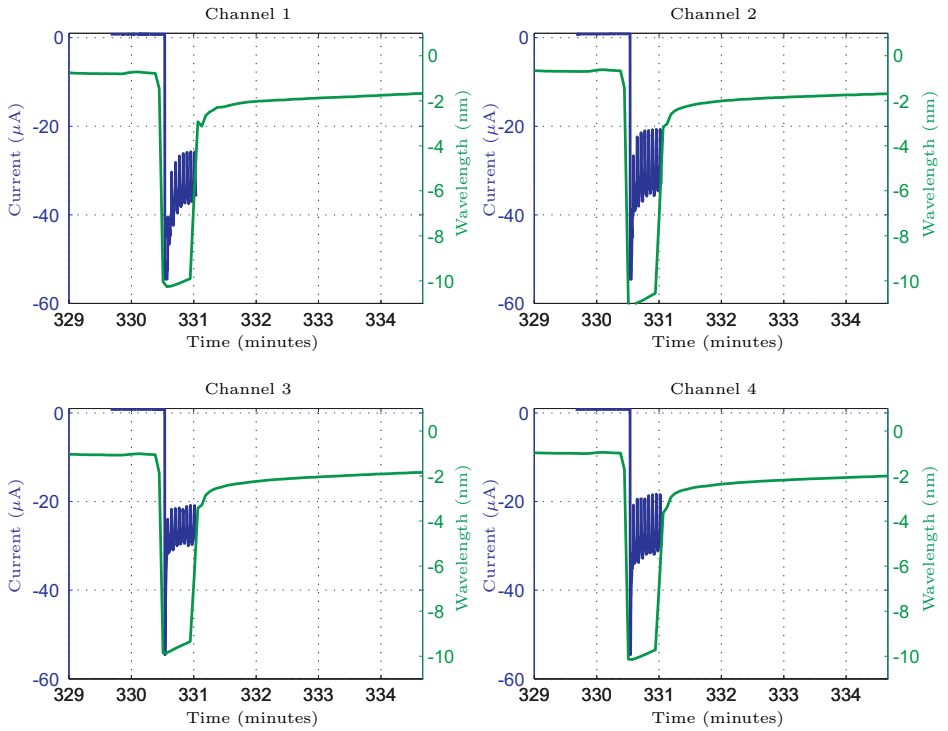


(b)

Figure 2.18: SPR signals (green) and applied potentials (blue) (a), SPR signals (green) and measured currents (blue) (b), 2° test.



(a)



(b)

Figure 2.19: SPR signals (green) and applied potentials (blue) (a), SPR signals (green) and measured currents (blue) (b), 3° test.

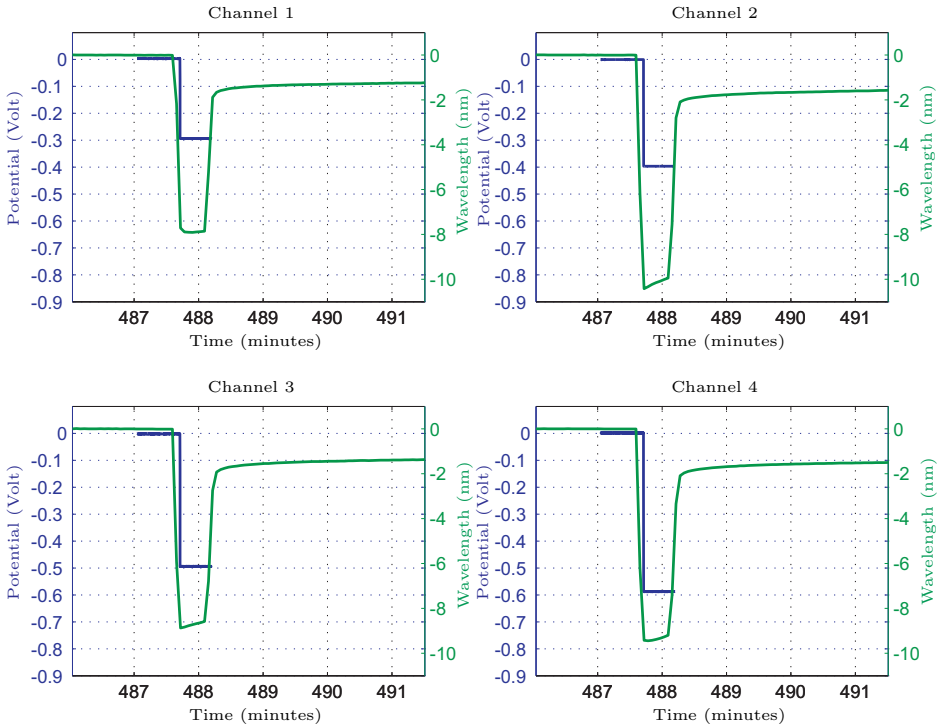
2.3.2.4 4° test

In the fourth test, the following negative potentials were applied to the SPR chip:

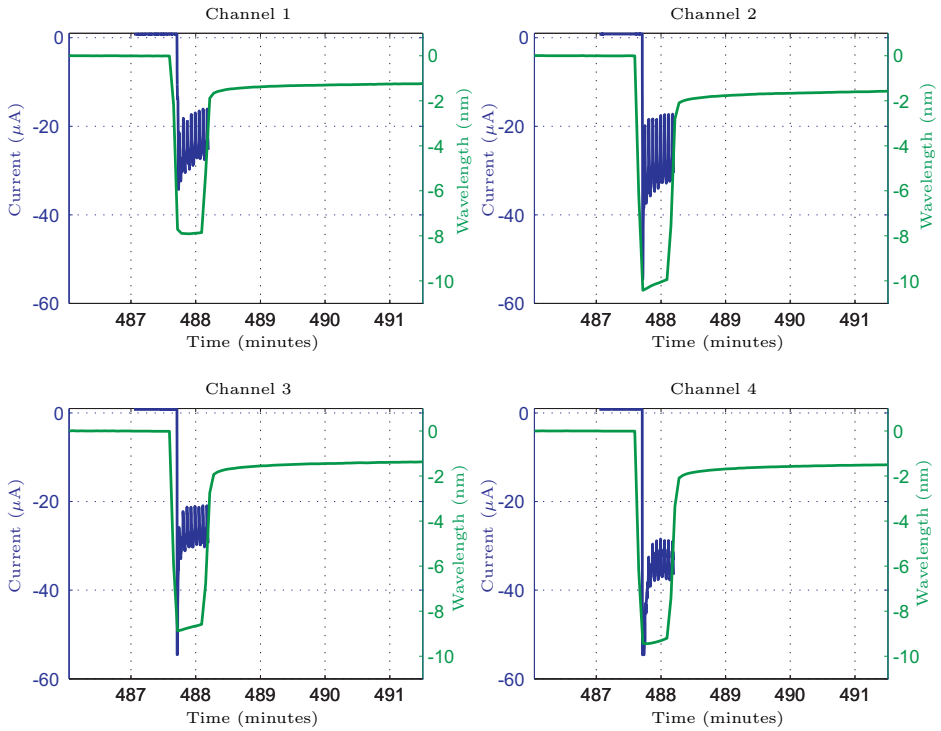
- Channel 1: -0.3 V;
- Channel 2: -0.4 V;
- Channel 3: -0.5 V;
- Channel 4: -0.6 V.

In Figure 2.20a, SPR signals and applied potentials are shown, whereas Figure 2.20b shows SPR signals and measured currents.

After the application of the negative potentials, the protocol described in Table 2.1 was applied. Finally, in Figure 2.21e the response to cDNA is shown.



(a)



(b)

Figure 2.20: SPR signals (green) and applied potentials (blue) (a), SPR signals (green) and measured currents (blue) (b), 4° test.

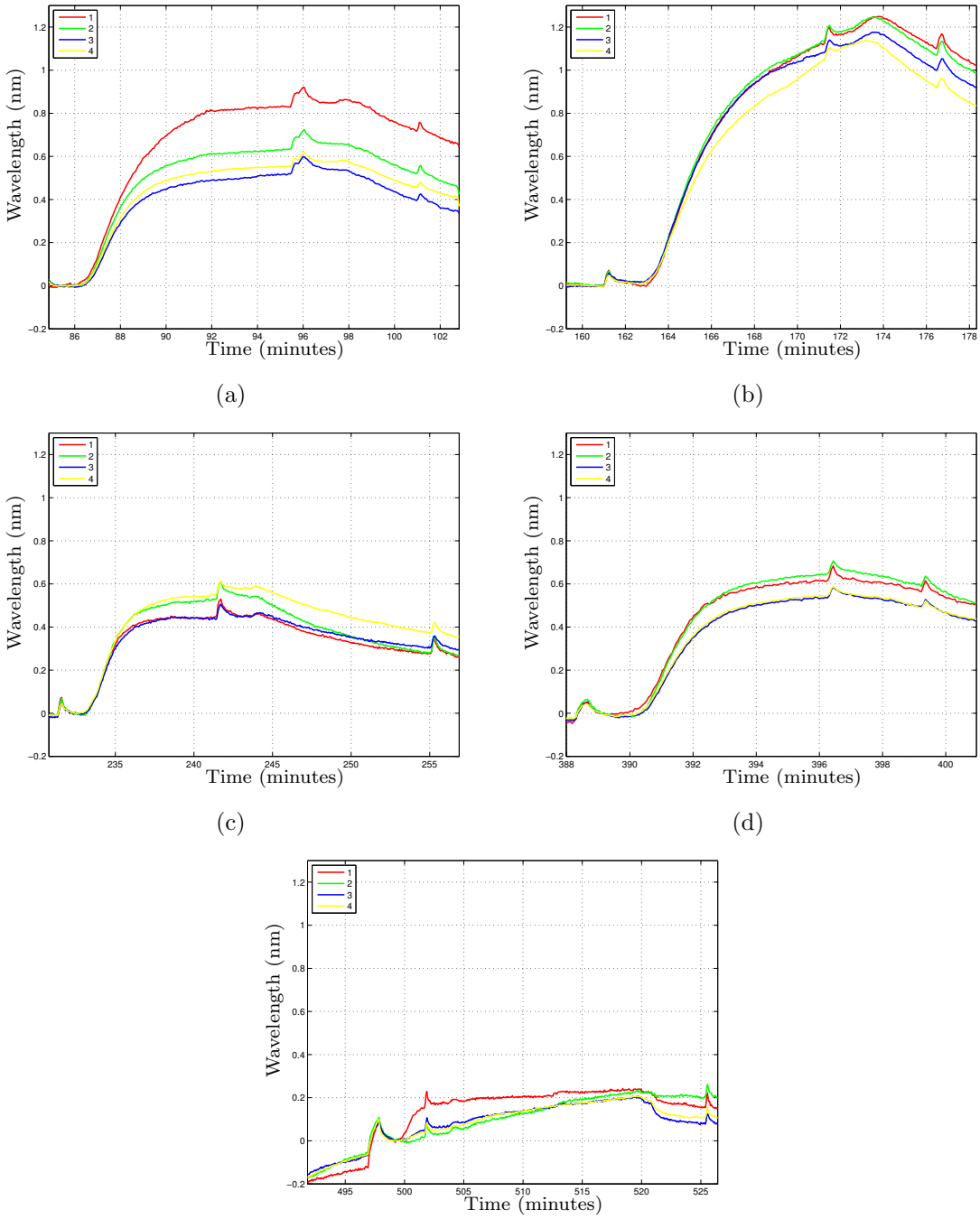


Figure 2.21: Response of the SPR system to the cDNA, after the application of negative potentials and the regeneration of the chip. Initial response (a), after 1° test (b), after 2° test (c), after 3° test (d), after 4° test (e).

2.3.3 Study on the electrical induced desorption threshold

Further experimental tests have been performed in order to study the dependence of the electrical induced desorption threshold on the lengths of the DNA probe chains. The test has been performed with DNA probes with different lengths, i.e. they have the same final sequence but with different spacers. In particular:

- channel 1: Sd273B (10-mer);
- channel 2: Sd273BATT (16-mer);
- channel 3: Sd273BATT2 (24-mer);
- channel 4: Sd273BATT3 (34-mer).

Increasing negative potentials were applied to the SPR chip to find the desorption potential threshold. The beginning of the desorption process is identified by a sudden decrease of the SPR signal. In Figure 2.22, the applied potentials (blue) and the SPR signals (green) are shown.

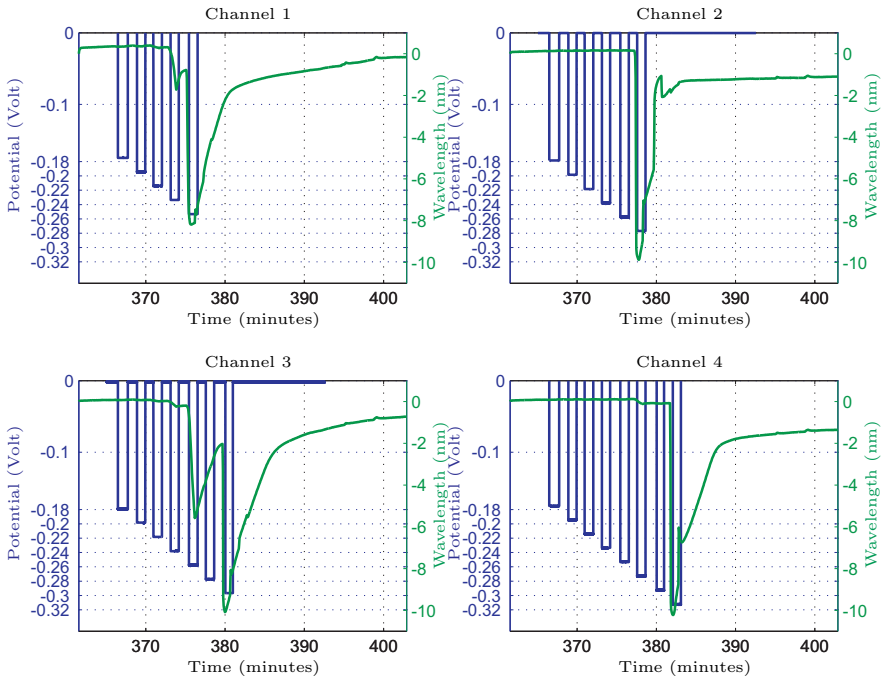


Figure 2.22: Increasing negative potentials (blue) applied to the SPR chip and SPR signals (green). The desorption potential depends on the length of the DNA probes.

Finally, in Figure 2.23, the four SPR signals versus time are shown. It can be noticed that the desorption potential depends on the length of the DNA probes. In particular, the longer the probes the more negative potential is required.
In Table 2.2 the desorption threshold potentials for each channel are reported.

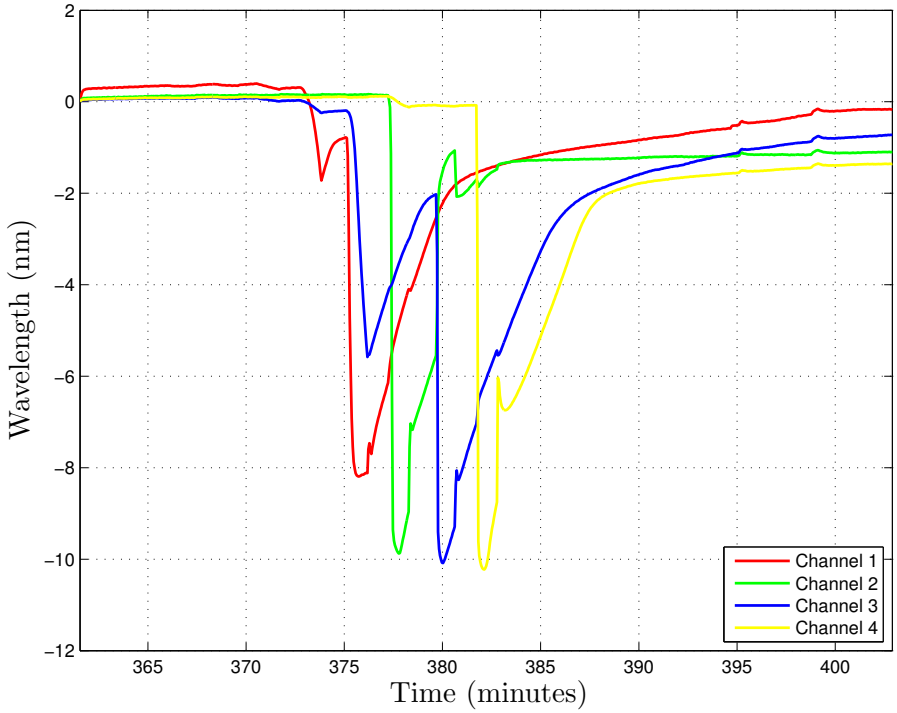


Figure 2.23: Increasing negative potentials applied to the SPR chip and SPR signals. The desorption threshold potential depends on the length of the DNA probes.

Table 2.2: Desorption threshold potentials of the four different DNA probes.

Channel	DNA probe	Desorption threshold potential (Volt)
1	Sd273B (10-mer)	-0.25
2	Sd273BATT (16-mer)	-0.28
3	Sd273BATT2 (24-mer)	-0.30
4	Sd273BATT3 (34-mer)	-0.31

2.4 Conclusions

A system combining electrochemical and SPR techniques has been developed. The system is based on a four-channel SPR system developed at the Institute of Photonics and Electronics AS CR, Prague. An in-situ electrochemical cell has been realized in each channel of the flow cell. Two different geometries have been proposed for the electrodes realization. Experimental tests have been performed with the whole system, comparing the electrochemical and SPR signals. Cyclic voltammetries performed in PBS and ferricyanide have been used to determine the pseudo-reference potential of the electrochemical cells. Preliminary tests have been performed to study the DNA probes desorption and regeneration. In particular, different negative potentials and different DNA probes have been used. In general, it has been noticed that it is possible to regenerate the DNA sensing chip through an electrical induced desorption. The desorption threshold potential and the desorption efficiency depend both on the applied negative potentials and on the DNA probes. It has been noticed that the longer DNA probes, the more negative potential is required to induce the desorption. Nevertheless, further experimental tests are required in order to better characterize the system and study the electrical induced desorption process.

Part II

Agri-food sensing

Chapter 3

Sensor to determine the glycerol concentration in grape juice

3.1 Introduction

Every year, consumption of unsafe or contaminated food causes serious health problems to millions of people [118]. In addition, billions of dollars are lost annually due to inspection processes, inefficient production and insect damage.

Traditional food-inspection techniques rely on sample collection and subsequent off-line analysis in a laboratory through periodic chemical and microbiological tests. These procedures conventionally use techniques as chromatography, spectrophotometry, electrophoresis, titration and others. These methods do not allow an easily continuous monitoring, because they are expensive, slow, need well trained operators and in some cases require steps of extraction or sample pre-treatment, increasing the time of analysis. Thus there is an increasing interest in newer non-destructive approaches to measure the analytes of interest [119, 120, 121]. These new methods aim to realize new sensors which should be fast, selective, multi-analyte in order to test several samples in real time.

Among these, biosensors are now playing a fundamental role. Biosensors are a sub group of chemical sensors; they consist of a biological recognition element (such as enzyme, antibody, receptor or microorganisms) coupled to a chemical or physical transducer (electrochemical, mass, optical and thermal). These devices represent a promising tool for food analysis due to the possibility to fulfil some demand that the classic methods of analysis are not able to satisfy. In fact, they are characterized by a high selectivity, relative low cost of construction and storage, potential for miniaturization, facility of automation and simple and portable equipment construction for a fast analysis and monitoring in platforms of raw material reception, quality control laboratories or some stage during the food processing [122, 123].

There are several methods for the immobilization of enzymes onto the electrodes surfaces and, among these ones, electrochemical methods are very effective and suitable [124]. In fact, immobilization of enzymes by using conducting polymer is an effective method for the preparation of the enzyme-modified electrodes [125, 126].

The quality control of wine, since the grape harvest, is becoming a problem of great importance. Nevertheless, defining a quality index for wine is not a simple procedure, thus both global and specific tests are needed, in order to give a general judgment of the sample characteristics and, on the other side, to determine the concentration of single analytes, considered to be representative of the future quality of wine, such as ethanol, lactic and malic acid, sulphites, polyphenol and sugars [127, 128, 129, 130, 131, 132].

Nowadays, due to the lack of portable instruments capable of making any quantitative estimation directly on the field and to the rather restrict times available during the reception of the grapes in the wine cellar, the evaluation is generally made by visual criteria that suffer from individual bias: the possibility of using more objective and even quantitative criteria appears definitely preferable.

Glycerol is a secondary fermentation product of alcoholic fermentation, with a favourable effect on the taste. The amount of glycerol formed during the fermentation process is about 1:10 of the alcohol formed, with final concentrations varying from 1 to 10 g/L. Deviations from this value might indicate technological alterations during the process or deterioration of the harvested grapes [133]. For example, *Botrytis* is a fungal disease that affects the integrity of the berries leading to undesired fermentation processes.

Glycerol concentration is routinely analysed either by a liquid chromatographic method or by spectrophotometric assessment of the effect of an enzymatic reaction (enzymatic kit). However, both these methods require the presence of qualified personnel carrying out the analysis in a proper laboratory, and they are not compatible with the times required for the analysis (not longer than 3 min).

In the present chapter, a fast portable sensor for the estimation of the glycerol concentration in grape juice is presented. It deploys an electrochemical reaction, using two enzymes, glycerol dehydrogenase (GDH) and diaphorase (DP). In particular, in the following Section a brief theoretical introduction is given together with the description of the realized system. In Section 3.3 the experimental protocol is described, whereas in Section 3.4 the experimental results are reported. Finally conclusion are drawn in Section 3.5.

3.2 Materials and Methods

The developed system consists of: (i) an electrode functionalized to reveal glycerol concentration, (ii) a fluidic system to dose the reagents, (iii) control and processing electronics and (iv) software interface to manage the whole system.

In the next Section, the electrochemical principle of measurement is described, whereas in Section 3.2.2, 3.2.3, 3.2.4 and 3.2.5 the electrode functionalization, the fluidic system, the electronics and the software interface are presented respectively.

3.2.1 Theoretical background

The developed system is based on a disposable three electrodes cell, where the enzymes glycerol dehydrogenase (GDH) and diaphorase (DP) have been immobilized. The whole process requires the presence of an oxidizer, NAD^+ (β-Nicotinamide adenine dinucleotide), called cofactor, and a redox mediator (M^{red}/M^{ox}), ferricyanide ($Fe(CN)_6^{3-}$), which represents the electroactive species. The following reactions take place at the working electrode:

$$\begin{aligned} Glycerol + NAD^+ &\rightarrow dihydroxyacetone + NADH \ (catalyzed\ by\ GDH), \\ NADH + M^{ox} &\rightarrow NAD^+ + M^{red} \ (catalyzed\ by\ diaphorase), \\ M^{red} &\rightarrow M^{ox} \ (electrochemical\ process). \end{aligned} \tag{3.1}$$

The information of interest is represented by a current variation recorded at a fixed potential ($\approx$ 0.4 $Volt$), which determines the oxidation of the ferrocyanide. In the presence of the glycerol, the ferricyanide (M^{ox}) is reduced to ferrocyanide (M^{red}), thus determining an increase in the recorded current. In this way, there is a proportional relation between the concentration of glycerol and the measured current. A schematic representation of the electrochemical reactions that take place at the working electrode is shown in Figure 3.1.

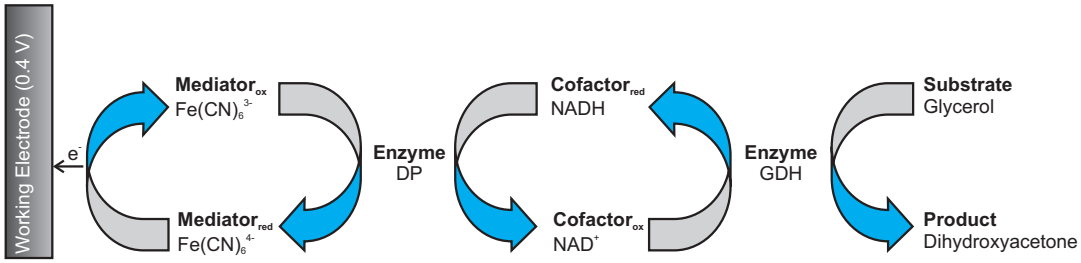


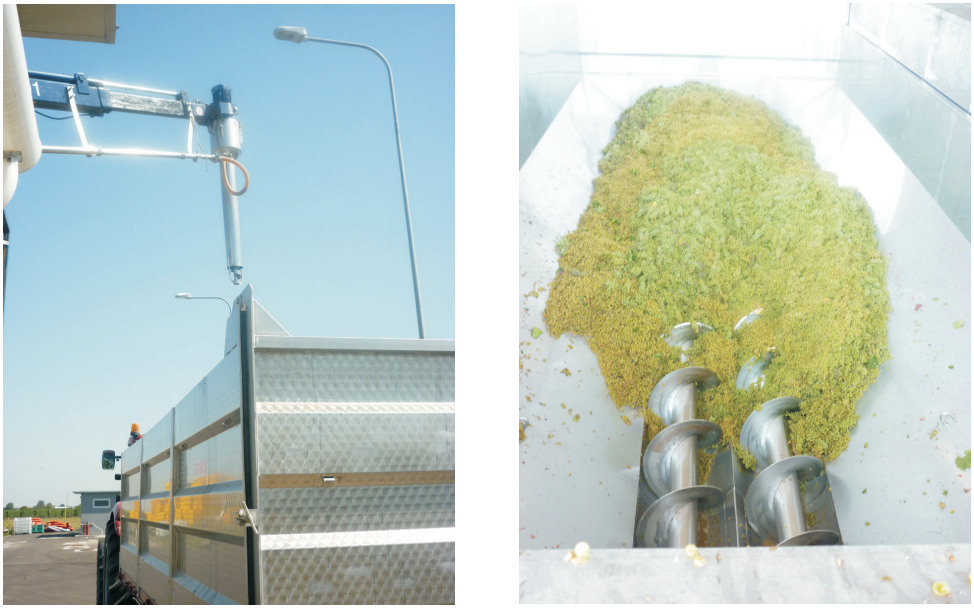
Figure 3.1: Schematic representation of the electrochemical reactions at the working electrode ($V \approx$ 0.4 $Volt$).

3.2.2 Electrode realization

A disposable electrochemical cell (Screen Printed Electrode - SPE) (EcoBioServices, Florence) has been used. It consists of a 3 mm diameter graphite working electrode, an Ag pseudo-reference electrode and a graphite auxiliary electrode. The working electrode was modified by deposition of a solution of glutaraldehyde, PBS, GDH, DP and Nafion. Enzymes *glycerol dehydrogenase* (GDH, E.C. 1.1.1.6, 90 U/mg) from *Cellulomonas* sp. and *diaphorase* (DP, E.C. 1.8.1.4, 59 U/mg) from *Clostridium* sp. were acquired from Sigma Aldrich and used without further purification. Enzymatic kits for the spectrophotometric quantification of glycerol in grapes were acquired from Boehringer Mannheim. All the other reactants were acquired from Sigma Aldrich and used as received.

3.2.3 Fluidic system

Nowadays, when the trucks with the grape arrive at the winery during the grape harvest, a core probe takes some grape samples, as shown in Figure 3.2a, and crushes them to prepare a must sample for preliminary analysis, such as pH, total acidity and alcoholic degree from sugar concentrations. Then, the grapes are dumped into a receiving hopper (Figure 3.2b) that conveys them to the destemmer/crusher. During this procedure, a visual judgment is given by an expert, from AA (best quality) to C (lowest quality), and the cost of the grape depends on this judgment.



(a) (b)

Figure 3.2: Pictures of the corer (a) and the receiving hopper (b).

Thus, an automated system has been developed in order to collect the sample directly after the corer, using suitable filters. The system is also capable of diluting the sample under test and adding the reactants. Basically, it consists of: (i) three MEMS membrane micropumps (mp6, Bartels Mikrotechnik), (ii) a thin layer flow cell (EcoBioServices, Florence) for flow injection analysis (FIA), where the electrode is placed, (iii) a one-way valve in the NAD channel and finally (iv) four reservoirs for grape juice, buffer, NAD and dumping respectively. A schematic representation of the developed fluidic system is shown in Figure 3.3, whereas pictures of the flow cell, micropumps and SPE electrode are shown in Figure 3.4.

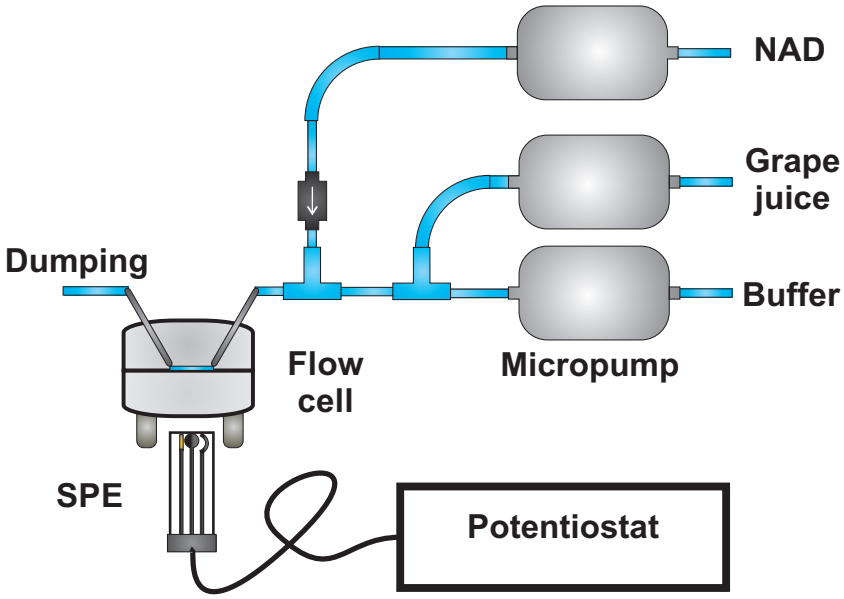


Figure 3.3: Schematic representation of the developed fluidic system. It consists of: (i) three micropumps, (ii) a flow cell, (iii) a one-way valve on NAD channel, (iv) four reservoirs for grape juice, buffer, NAD and dumping.

(a) (b) (c)

Figure 3.4: Pictures of: Screen printed electrode (SPE) (a), flow cell (b), membrane micropump (c).

3.2.4 Electronics

In order to collect the information of interest, i.e. the current, a commercial handheld potentiostat/galvanostat has been used (Palmsens, PalmSens BV, The Netherlands). In addition, a control electronics has been realized, as schematically shown in Figure 3.5.

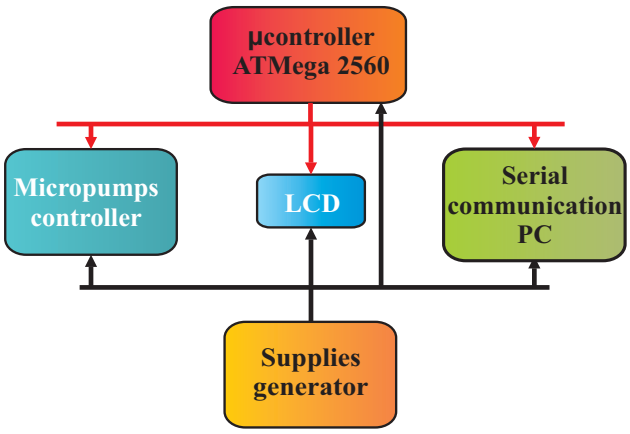
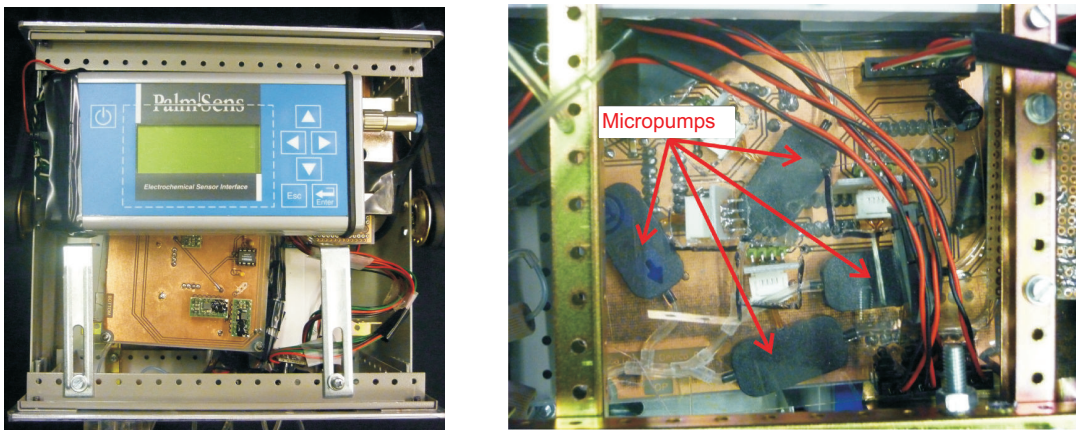


Figure 3.5: Schematic representation of the realized electronics. It consists of a power supplies generator, a micropumps controller, a LCD and a commercial microcontroller (ATMega 2560)

In particular, it generates all the power supplies, it drives the micropumps, it controls a LCD and finally it communicates with a PC. The micropumps control has been realized through the use of ad-hoc controller chips (mp6-oem, Bartels Mikrotechnik), whereas the communication with the PC has been performed using a commercial microcontroller (ATMega 2560).

Finally, a portable prototype instrument has been realized placing together the potentiostat, the developed electronics and the fluidic system. Pictures of the realized electronics and the developed portable prototype instrument are shown in Figure 3.6 and 3.7 respectively.



(a) (b)

Figure 3.6: Pictures of the electronics: commercial potentiostat (a) and micropumps control circuitry (b).

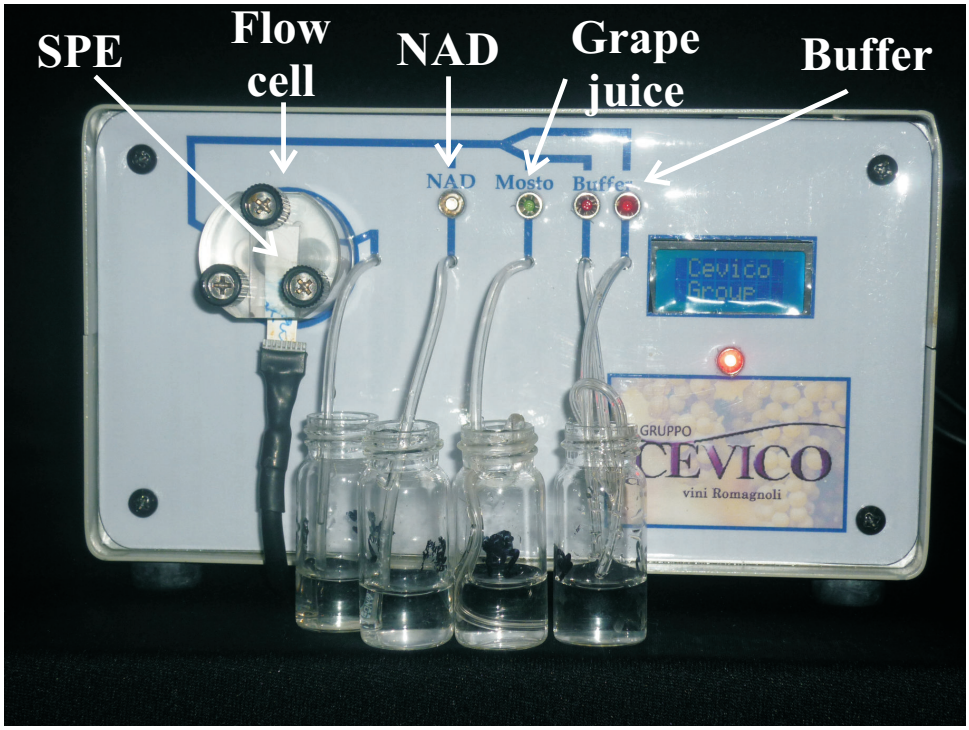
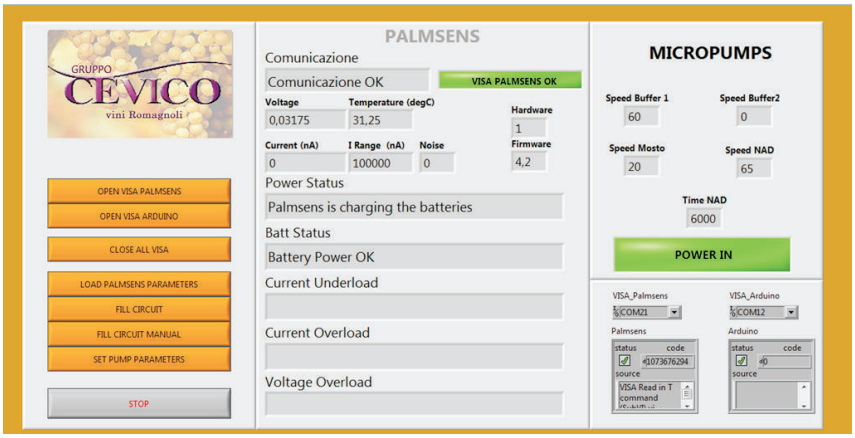


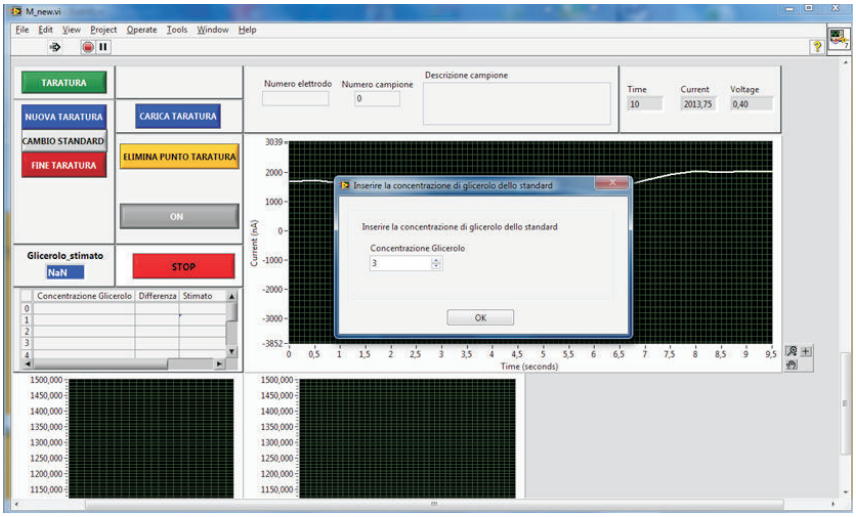
Figure 3.7: Picture of the developed prototype instrument, including the fluidic system and the electronics. It can be noticed the flow cell and the inlets for the sample and the reactants.

3.2.5 Software interface

A software interface has been developed in LabVIEW to control all the system. Several functions have been developed to initialize the instrument, allowing to make an automatic filling of tubings and to set the potentiostat parameters. In addition, functions to determine the calibration curve and to measure the glycerol concentration in unknown grape juice samples have been implemented, together with all the required calculations. The software interface is simple and user-friendly, allowing also an inexpert user to easily manage the instrument. In Figure 3.8 the main and the measurement software interfaces are shown, whereas the VI hierarchy is reported in Figure 3.9.



(a)



(b)

Figure 3.8: Main interface (a) and measurement interface (b) of the management software developed in LabVIEW.

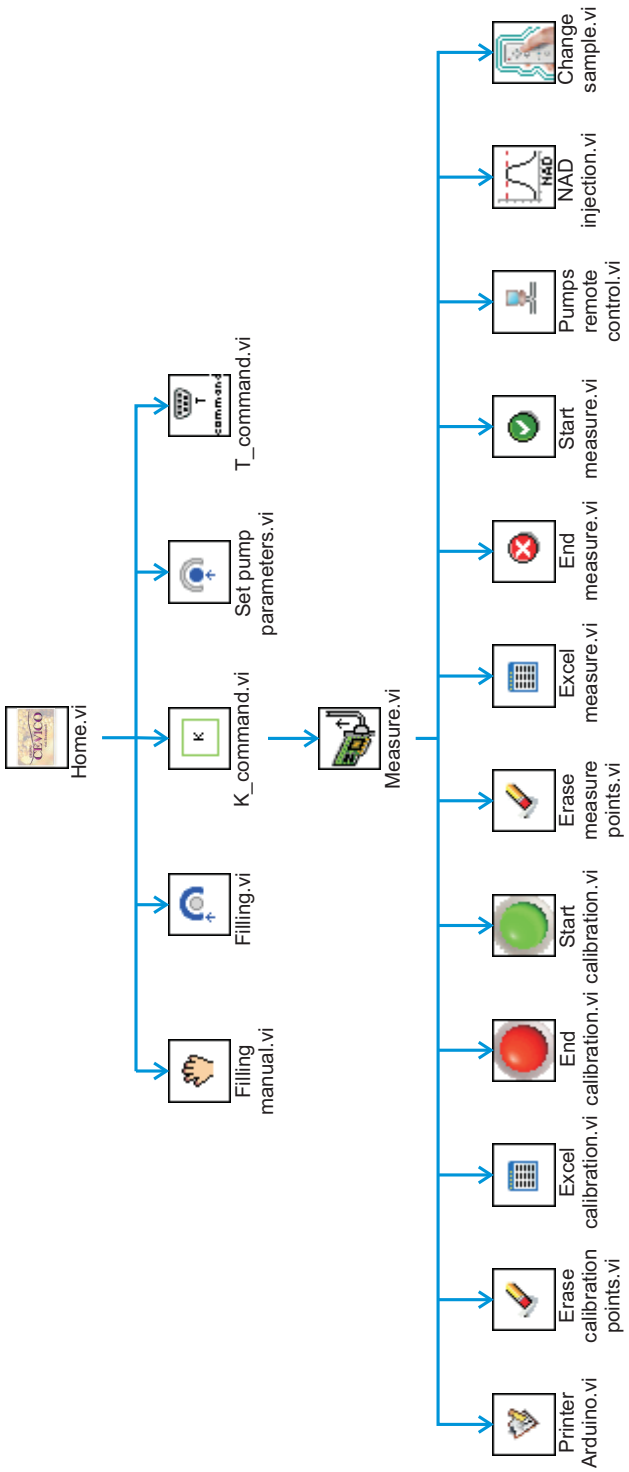


Figure 3.9: VI hierarchy of the software interface, developed in LabVIEW, to manage all the system.

3.3 Experimental protocol

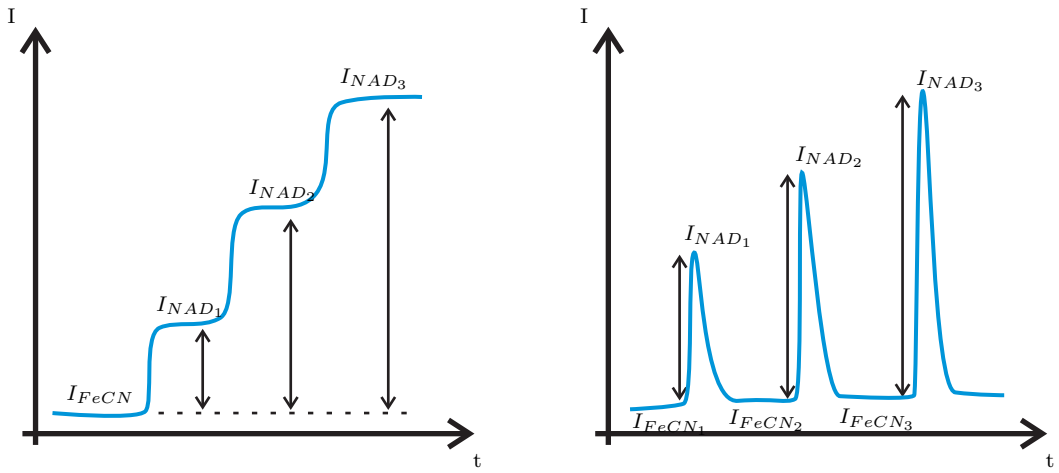
As described in Section 3.2.1, the information of interest is represented by a current variation measured at a fixed potential of 0.4 Volt. In particular, the glycerol concentration is proportional to the measured current variation. The measurement can be performed in two different ways: (i) using an external calibration curve or (ii) applying the standard additions method in each sample.

In the first method, the calibration curve is previously determined in a healthy grape juice, with standard additions of glycerol, thus the same calibration curve is used for all the grape juice samples. In the second method, standard additions of glycerol are done in each sample and the measured current variations are interpolated through a line. The first method is faster, but it assumes that the same calibration curve can be used for different grapes (e.g red and white), while the second method is slower, but at the same time does not suffer from the matrix effect.

In the standard additions method, the point of interception of the abscissa is the endogenous concentration of the glycerol in the sample. Thus, it can be determined as:

$$\begin{aligned} I_{diff} &= m \cdot [Glycerol] + q, \\ [Glycerol] &= \frac{q}{m}, \end{aligned} \tag{3.2}$$

where $[Glycerol]$ represents the glycerol concentration, m and q are the slope and the intercept of the interpolation line and I_{diff} represents the current variation. In particular, considering a closed-loop fluidic circuit, I_{diff} represents the current variation between the current measured after the glycerol addition (I_{NAD}) and before the first glycerol addition (I_{FeCN}), as schematically shown in Figure 3.10a. In a closed-loop fluidic circuit, the NAD^+ injection is performed only at the beginning before the first glycerol addition. On the other side, considering an analysis performed under continuous flow, I_{diff} represents the current variation between the current measured just after (I_{NAD}) and before (I_{FeCN}) the injection of the cofactor NAD^+, as schematically shown in Figure 3.10b. Under continuous flow, in fact, the cofactor is injected whenever the sample is changed, whereas in the closed-loop fluidic circuit it is injected only at the beginning.



(a) (b)

Figure 3.10: Schematic representation of the current variation I_{diff} in a closed loop circuit (a) and under continuous flow (b).

3.4 Experimental results

Several tests have been performed to study the feasibility of the sensing approach and to characterize the developed system. In particular, in the next Section, the correlation between the glycerol concentration and the healthy level of grape juices is investigated. In Section 3.4.2 the matrix effect is discussed, whereas in Section 3.4.3 a preliminary test with a closed-loop system is reported. Finally, in Section 3.4.4, experimental tests on grape juices with the developed system are presented.

3.4.1 Feasibility study

Firstly, a feasibility study has been performed to verify the efficiency of the glycerol concentration as a parameter to quantify the presence of *Botrytis* in the grape samples. Thus, seven different samples were prepared using the grape of the same vine, with different amounts of moldy bunches. In particular, the estimated glycerol concentrations, determined through the enzymatic kit, versus the percentage in weight of moldy grapes are shown in Figure 3.11.

We observed that the estimated glycerol concentrations are correlated with the amount of moldy grapes in the sample (interpolation line: R=0.9760), thus demonstrating that the glycerol concentration can be considered a suitable parameter for the sanitary level of the collected grapes.

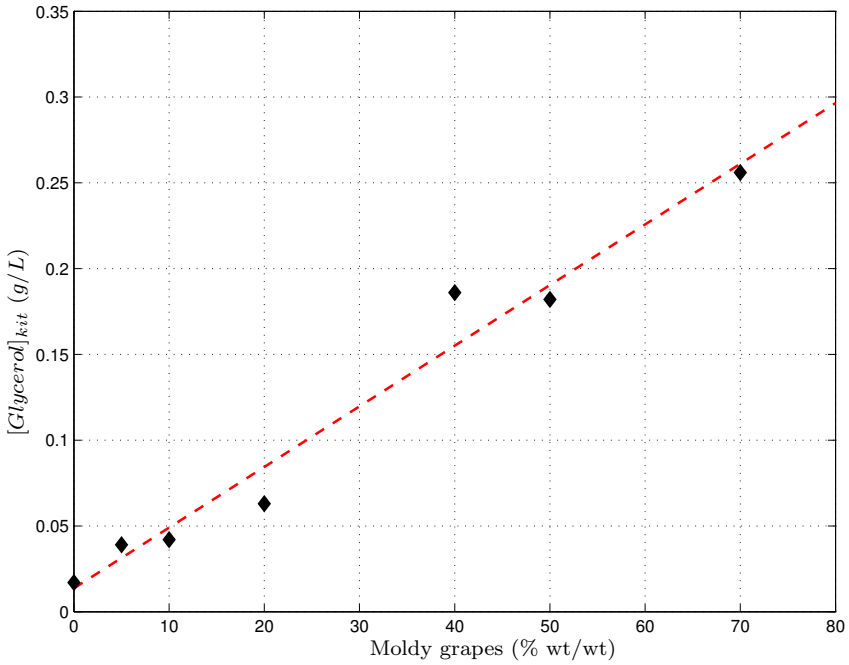


Figure 3.11: Glycerol concentration estimated with the enzymatic kit versus moldy grapes percentage.

3.4.2 Matrix effect study

A set of measurements have been carried out with grapes of good quality, i.e. not affected by *Botrytis*. Initially, a set of tests have been performed in solutions containing different amounts of grape juice. Nevertheless, by using the standard additions method, different interpolation lines have been determined. In particular, both the slope and the intercept of the interpolation line depend on the amount of grape juice in the test solution. In particular, the higher the grape juice concentration, the lower the slope and the higher the intercept. The decrease of the slope is due to the matrix effect, since the measured current depends on the diffusion speed of the species at the electrode surface which, in this experiment, depends on the amount of grape juice in the sample. For this reason, standard additions method is usually performed in each sample unless a high dilution ratio is used. The increase of the intercept, instead, is due to the presence of chemical species directly oxidised by the redox mediator.

For these reasons, an external calibration line, determined on a buffer solution, cannot be used and the difference between the current measured after (I_{NAD}) and just before the glycerol addition (I_{FeCN}) had to be considered representative of the glycerol concentration.

3.4.3 Preliminary test on grape juice with standard additions method

A preliminary test has been performed on grape juice (Trebbiano) affected by *Botritis*, using the standard additions method, under reflux regime with a closed loop system. For this reason, only an initial NAD^+ injection has been performed. Then, after every glycerol addition, as soon as the current signal reached the stability, the mean current value was determined over a time period of 20 seconds. Thus, the mean current values represent the I_{NAD} signals defined in Section 3.3, whereas, for this measurement, the I_{FeCN} is represented by the mean current value before the first glycerol addition. In Figure 3.12 the amperometric signal obtained at 0.35 V, by subsequent addition of glycerol standard solution, versus time is reported. For this measurement, the grape juice was diluted eight times, with a 3 mM $Fe(CN)_6^{3-}$, 0.05 M phosphate buffer (pH = 8.5), 0.1 M KCl solution.

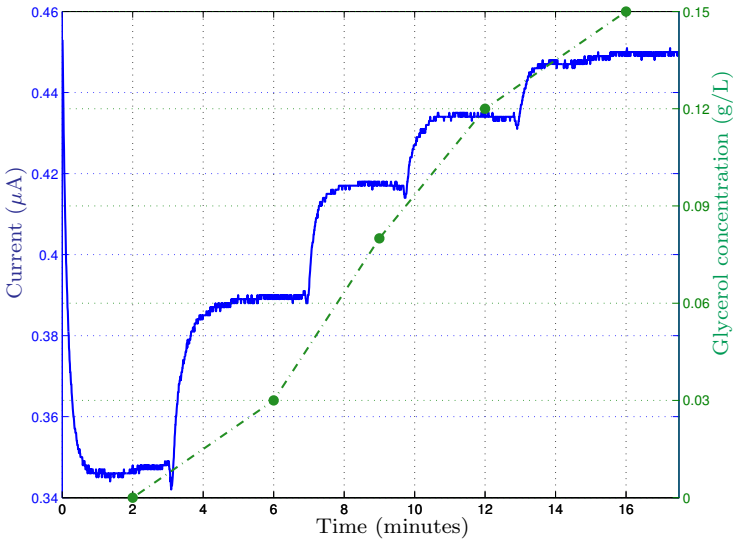


Figure 3.12: Amperometric signal obtained at +0.35 V, by subsequent addition of glycerol standard solution in a grape juice diluted eight times with a 3 mM $Fe(CN)_6^{3-}$, 0.05 M phosphate buffer (pH = 8.5), 0.1 M KCl solution.

Finally, in Figure 3.13, I_{diff} values versus glycerol concentration are shown, together with the interpolation lines (R=0.9905). The glycerol concentration of the sample under test was determined by applying Eq. 3.2 and calculated to be 0.1227 g/L.

Other several tests have been performed with the standard additions methods, using different grape juices. In general, we observed that calibration curves are reproducible on the same grape juice and that, considering three different grape juices, the calibration curves determined in each sample are not significantly differ-

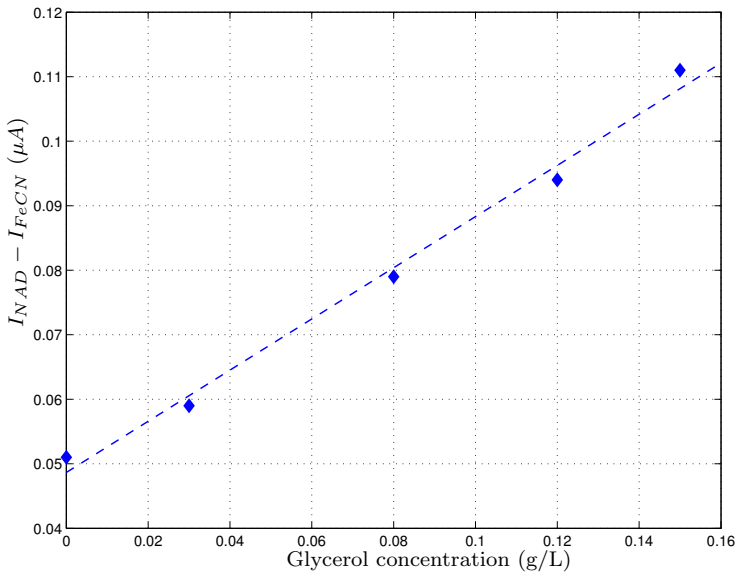


Figure 3.13: Mean values of the I_{diff} signal (diamonds) and interpolation line. Amperometric signal obtained at +0.35 V, by subsequent addition of glycerol standard solution in a grape juice diluted eight times with a 3 mM $Fe(CN)_6^{3-}$, 0.05 M phosphate buffer (pH = 8.5), 0.1 M KCl solution.

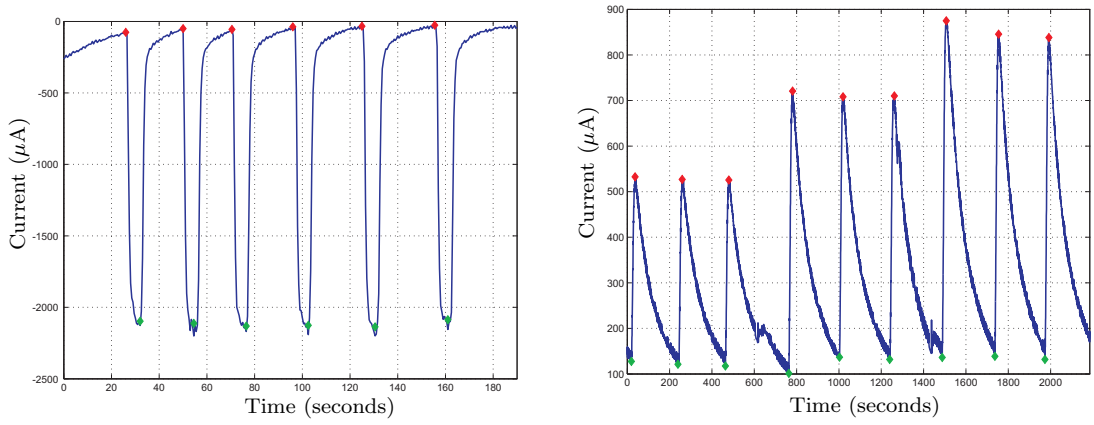
ent. Thus, an external calibration curve, previously determined on a healthy grape juice, can be used to estimate the glycerol concentration in different grape juices.

3.4.4 Experimental tests with the developed system

Finally, several tests have been performed to characterize the developed system and to make it suitable for the measurements during the grape harvest. In particular, due to the long time required for closed loop analysis (Figure 3.12), the tests have been performed under continuous flow, injecting the cofactor NAD^+ in the buffer/ grape juice solution. Thus, the developed system collects the sample under test and dilutes it with buffer solution (different micropumps speeds allow for different dilution ratios); then, an injection of 3 seconds of NAD^+ is performed and the current variation between the measured current value after (I_{NAD}) and just before (I_{FeCN}) the NAD^+ injection is considered to be representative of the glycerol concentration. In particular, in the next Section a characterization of the reproducibility of the system is reported whereas in Section 3.4.4.2 tests performed during the grape harvest are described.

3.4.4.1 Characterization of the reproducibility of the sensor response

In order to test the reproducibility of the sensor response, two different tests have been performed. Firstly, the NAD^+ cofactor has been replaced by a redox mediator ($Fe(CN)_6^{3-}$) and the current variation, due to its reduction, after its injection into a solution containing only the supporting electrolyte, has been measured and shown in Figure 3.14a.



(a) (b)

Figure 3.14: Amperometric measurements of six subsequent injection of $Fe(CN)_6^{3-}$ through the NAD^+ micropump in a flux of 0.05 M PBS and 0.1 M KCl (E=-0.2 V) (a). NAD^+ injections in flux containing glycerol (0.11, 0.19, 0.25 g/L - 3 replicates for each sample) mixed with the "buffer solution" accordingly with the dilution factor due to the micropumps (b). Red diamonds represent the I_{NAD} values whereas green diamonds represent the I_{FeCN} values.

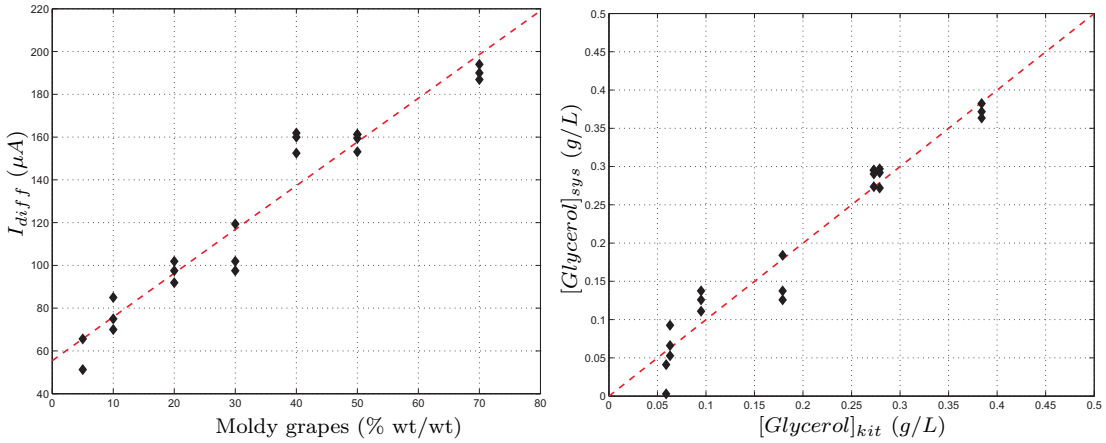
Then, the system has been tested by adding the NAD^+ cofactor to three standard solutions of glycerol. The information of interest, as previously described (Section 3.3), is represented by the difference between the current peak after the injection of NAD^+ (I_{NAD}) and the steady state regime just before the injection of the cofactor (I_{FeCN}), averaged over a time period of 5 seconds. In Figure 3.14b the measured current versus time is shown, whereas in Table 3.1 the mean values ($\bar{I}_{diff}$), the standard deviations (σ) and the relative standard deviations (σ_r) are reported for the performed tests.

Table 3.1: Mean values, standard deviations and relative standard deviations for the tests performed to characterize the sensor reproducibility.

	Test with $Fe(CN)_6^{4-}$	Test with NAD^+ with different $[Glycerol]$ concentrations		
		0.11 g/L	0.19 g/L	0.25 g/L
$\bar{I}_{diff}$ (μA)	2127	406.25	589.58	717.30
σ (μA)	42.13	1.65	26.03	18.59
σ_r (%)	1.98	0.41	4.42	2.59

3.4.4.2 Tests on grape juice samples

A first test on grape juice was performed on the same samples prepared for the test described in Section 3.4.1. In particular, healthy grapes were mixed with moldy grapes in different weight percentages. For each sample, three measurements were performed. In Figure 3.15a, the I_{diff} signal versus the moldy grapes percentage is shown, together with the interpolation line (R=0.9354), whereas in Figure 3.15b glycerol concentrations estimated with the developed system ($[Glycerol]_{sys}$) versus the ones estimated with the enzymatic kit ($[Glycerol]_{kit}$) are shown, together with the interpolation line (slope=1, intercept=-3.68e-17, R=0.9496). Preliminary estimation has determined an uncertainty of 27 mg/L.



(a) (b)

Figure 3.15: I_{diff} versus moldy grapes percentage (a). Glycerol concentration estimated with the developed system ($[Glycerol]_{sys}$) versus glycerol concentration estimated with the enzymatic kit ($[Glycerol]_{kit}$) (b).

Finally, the instrument was tested during the grape harvest. In Figure 3.16, $[Glycerol]_{sys}$ versus $[Glycerol]_{kit}$ is shown, together with the interpolation line (slope:

0.9245, intercept: 0.2096, R: 0.8514), for grape juice samples coming from different vines, considering both white and red grapes. Glycerol concentrations were estimated using the same calibration curve for all the samples, previously determined on a healthy grape juice.

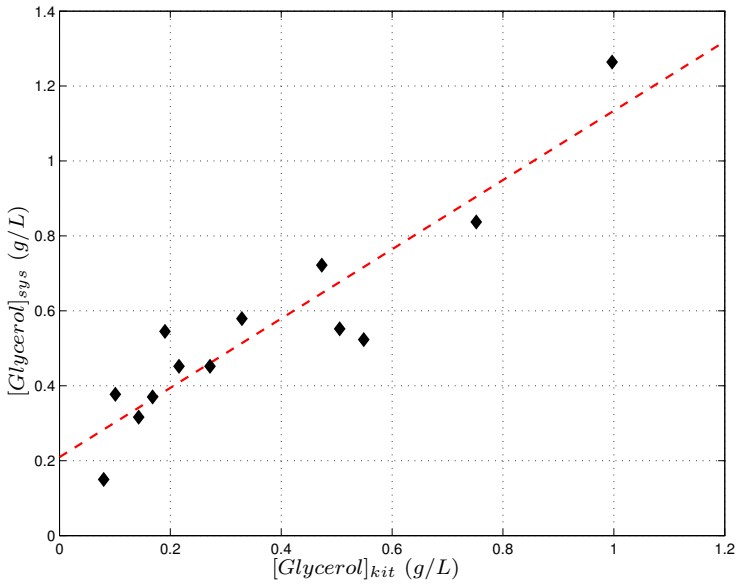


Figure 3.16: Glycerol concentration estimated with the developed system ($[Glycerol]_{sys}$) versus glycerol concentration estimated with the enzymatic kit ($[Glycerol]_{kit}$), during the grape harvest, considering both white and red grapes.

3.5 Conclusions

A portable system for the fast estimation of glycerol concentration in grape juice has been developed. Since the glycerol concentration has been proven to be correlated to the healthy level of the grapes, the system can be used during the grape harvest to control the grapes quality. Experimental tests have been performed both in laboratory and during the grape harvest. In order to perform the measurement in less than 3 minutes (time constraint defined by the winery), an external calibration curve has been used for all the grape kinds (white and red), previously determined in a mute grape juice, and only one replicate has been considered. Although this approach worsens the measurement accuracy, the instrument can correctly classifies grapes based on the content of glycerol, providing an important tool for an objective classification of the grapes, not biased by the visual judgment.

3.6 Comparison with the state of the art

Name	Description	Principle	Method	Reaction time	Detection limit	Price
Developed system	Electrochemical sensor for the determination of Glycerol in grape juices	$Glycerol + NAD^+ \overset{GDH}{\rightarrow} dihydroxyacetone + NADH$ $NADH + Fe(CN)_6^{3-} \overset{DP}{\rightarrow} NAD^+ + Fe(CN)_6^{4-}$ $Fe(CN)_6^{4-} \overset{0.4\,V}{\rightarrow} Fe(CN)_6^{3-}$	Current variation measured at ≈0.4 V	<3 min	27 mg/L	2 €
Megazyme K-GCROL	UV-method for the determination of Glycerol in foodstuffs, beverages, and other materials	$Glycerol + ATP \overset{glycerokinase}{\rightarrow} L-glycerol-3-phospate + ADP$ $ADP + PEP \overset{pyruvatekinase}{\rightarrow} ATP + pyruvate$ $Pyruvate + NADH + H^+ \overset{L-lactatedehydrogenase}{\rightarrow} L-lacticacid + NAD^+$	Spectro-photometric at 340 nm	≈5 min	0.34 mg/L	3.50 €
Megazyme K-GCROLGK	UV-method for the determination of Glycerol in foodstuffs, beverages, and other materials	$Glycerol + ATP \overset{glycerokinase}{\rightarrow} L-glycerol-3-phospate + ADP$ $ADP + D-glucose \overset{ADP\rightarrow GK}{\rightarrow} G-6-P + AMP$ $G-6-P + NAD^+ \overset{glucose-6-phospatedehydrogenase}{\rightarrow} gluconate-6-phospate + NADH + H^+$	Spectro-photometric at 340 nm	≈7 min	0.37 mg/L	4 €

Table 3.2: Characteristics comparison between the developed system and the main commercial systems.

Chapter 4

Sensor to estimate the concentration of antibiotics in milk

4.1 Introduction

Nowadays, almost all food-production animals receive medication during their lives, representing a threat for the public health, with regard to the presence of drug residues in the food chain [134]. Lots of animals, in fact, are treated with antibacterial drugs to attend infectious diseases [135, 136]. These could lead to adverse health effects, such as toxicity and potential allergic response. In addition, they could determine a transfer of antibiotic resistance genes to human pathogens [135, 137]. Although it is hard to determine the consumer health risks of drug residuals, the presence of antibiotic residues in food has been regulated with maximal residue limits (MRLs) set by the European Commission (EC Regulation 2377/90) (Table 4.1).

Milk is one of the most important food, both as a beverage and to produce milk derivatives, such as cheese and butter. For this reason, milk is one of the most regulated product in food industry. As a consequence, there is an increasing request of a method capable of a fast and multi-antibiotic sensing.

Traditional methods include enzymatic colorimetric assays, receptor binding assays, chromatographic methods, microbial growth inhibition assays, microbial receptor assays and immunoassays [138, 139]. The milk samples are usually firstly analyzed with qualitative fast methods and only the suspected samples are subsequently quantified and confirmed with high pressure liquid chromatography (HPLC) and mass spectroscopy (MS).

Nowadays, several immunoassays have been developed which are faster and more specific, in order to create an alternative to unspecific and time-consuming microbial growth inhibition assays. These immunoassays are in ELISA format, which allows quantitative and highly specific measurements [140]. However, since milk

samples have to be screened for several antibiotics, they become laborious and time-consuming and they require modifications of the antibody used in the assay.

For this reason, there has been an increasing interest in the development of sensors for antibiotics that do not require extensive chemical modification or pre-treatment steps and that are rapid, quantitative and selective [141]. Basically, an immunosensor consists of three parts:

- a receptor species, capable of recognizing and binding the analyte of interest;
- a transducer, on which the receptor is immobilized and which is able to convert the binding event into a measurable physical change;
- a method to convert the change detected at the transducer into an useful information. The most common ones are electrochemical [142, 143] and optical methods [144, 145, 146].

Finally, there are two basic type of immunoassays: direct assays and competitive assays. In the former, the binding of the analyte of interest with a particular enzyme is measured, whereas, in the latter, a known amount of an another analyte, competitive with the analyte of interest, is mixed with the sample under test, containing the enzyme. If there is no analyte of interest in the sample, the second analyte just binds to the enzyme, whereas if there is the analyte of interest in the sample, there is a competition between the two analytes. This phenomenon leads to an inhibition of enzyme binding.

The developed sensor deploys an optical competitive method. As described in the following Section, it deploys the competition between the antibiotic of interest and a particular substrate. This competition leads to a variation in absorbance at a specific wavelength. In the next Section, together with the theoretical background, a brief description of the whole system is given. Finally, preliminary experimental tests are reported and discussed in Section 4.3 and 4.4.

Table 4.1: List of pharmacologically active substances for which maximum residue limits have been fixed. Council Regulation (EEC) no. 2377/90.

Pharmacologically active substance(s)	**Marker residue**	**MRLs in milk μg/kg**	**MRLs in food producing species μg/kg**
Penicillins			
Amoxicyllin	Amoxicyllin	4	50
Ampicillin	Ampicillin	4	50
Benzylpenicillin	Benzylpenicillin	4	50
Cloxacillin	Cloxacillin	30	300
Dicloxacillin	Dicloxacillin	30	300
Nafcillin	Nafcillin	30	300
Oxacillin	Oxacillin	30	300
Penethamate	Benzylpenicillin	4	50
Cephalosporins			
Cefacetrile	Cefacetrile	125	-
Cefalexin	Cefalexin	100	200 (muscle, fat, liver) 1000 (kidney)
Cefalonium	Cefalonium	20	-
Cefapirin	Sum of cephapirin and desacetylcephapirin	60	50 (muscle, fat) 100 (kidney)
Cefazolin	Cefazolin	50	-
Cefoperazone	Cefoperazone	50	-
Cefquinome	Cefquinome	20	50 (muscle, fat) 100 (liver) 200 (kidney)
Ceftiofur	Sum of all residues retaining the betalactam structure expressed as desfuroylceftiofur	100	1000 (muscle) 2000 (fat, liver) 6000 (kidney)

4.2 Materials and Methods

The developed system deploys the absorbance variation of a particular substrate in the presence of the antibiotic. Thus it consists of an optical system, a front-end and control electronics and a software interface. In the next sections the theoretical background is presented, and a brief description of the optics, electronics and software interface is given.

4.2.1 Theoretical background

The system aims to the detection of beta-lactam antibiotics in milk through the use of the beta-lactamase enzyme (AmpC). The enzyme is able to link to the antibiotic determing its inactivity. The idea is to synthesize a substrate, CHROMOTIN (CENTA), with different chemo-physical properties capable of linking to the same enzyme active site. The substrate changes its optical properties with respect to the interaction with the enzyme. In particular, this interaction determines a change in absorbance of the substrate from about 345 nm to 405 nm. Thus, in the presence of the antibiotic, the antibiotic and the substrate compete for the same enzyme active site (Figure 4.1). The substrate, which links to the enzyme, is inversely proportional to the concentration of antibiotic, thus the variation in absorbance at 405 nm is inversely proportional to the concentration of antibiotic.

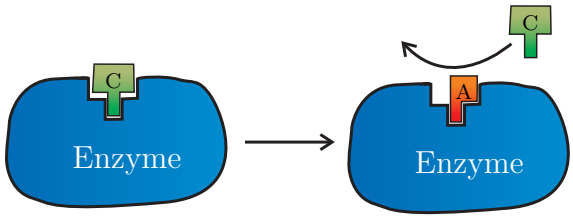


Figure 4.1: Schematic representation of the competition between CENTA (C) and antibiotic (A) for the enzyme active site.

In order to describe the enzyme kinetics, the Michaelis-Menten equation (Eq. 4.16) plays a fundamental role [147]. It is characterized by two constants: the Michaelis constant, K_m, and the indirectly obtained catalytic constant k_{cat}.

The simplest possible reaction is the irreversible conversion of substance S to product P.

$$S \xrightarrow{k_1} P. \tag{4.1}$$

The arrow is drawn from S to P to signify that the equilibrium lies far to the right, and the reverse reaction is infinitesimally small. The reaction rate or velocity

of the reaction (v) can be defined in terms of the time (t)-dependent production of product P. Since formation of P involves the loss of S, v can be defined in terms of the time-dependent consumption of substance S. Finally the operator [.] represents the concentration of the substance inside the square brackets.

$$v = \frac{\delta[P]}{\delta t} = -\frac{\delta[S]}{\delta t} = k_1\,[S]. \tag{4.2}$$

Considering a simple conversion of substrate (S) into product (P) catalyzed by an enzyme (E), Eq. 4.1 can be rewritten as:

$$E + S \underset{k_{-1}}{\overset{k_1}{\rightleftarrows}} ES \overset{k_{cat}}{\rightarrow} E + P. \tag{4.3}$$

Assuming that the dissociation rate k_{cat} of the enzyme-substrate complex (ES) is slow compared to association k_1 and redissociation k_{-1} reactions and that the reverse reaction ($P \rightarrow S$) is negligible, following Eq. 4.2, the formation of the product can be expressed as:

$$v = k_{cat}\,[ES]. \tag{4.4}$$

At steady-state, the enzyme-substrate concentration is stable, thus

$$\frac{\delta[ES]}{\delta t} = 0, \tag{4.5}$$

which yields to:

$$k_1[E][S] = k_{-1}[ES] + k_{cat}[ES], \tag{4.6}$$

since the formation of the ES complex and the breakdown of the ES complex are equal. Equation 4.6 can be written as:

$$\frac{k_1[E][S]}{k_{-1} + k_{cat}} = [ES]. \tag{4.7}$$

The three rate constants can be combined in one term, K_m, known as the Michaelison-Menten constant:

$$K_m = \frac{k_{-1} + k_{cat}}{k_1}, \tag{4.8}$$

and Eq. 4.7 can be rewritten as:

$$\frac{[E][S]}{K_m} = [ES]. \tag{4.9}$$

The amount of free enzyme (E) and enzyme that is bound to the substrate (ES) varies over the course of a reaction, but the total amount of enzyme (E_t) is constant such that:

$$E = E_t - ES. \tag{4.10}$$

Substituting into Eq. 4.9 yields:

$$\frac{([E_t] - [ES])[S]}{K_m} = [ES], \tag{4.11}$$

which can be rearranged as:

$$\frac{[E_t][S]}{K_m + [S]} = [ES]. \tag{4.12}$$

Substituting into Eq. 4.4 yields:

$$v = \frac{k_{cat}\,[E_t]\,[S]}{K_m + [S]}. \tag{4.13}$$

The maximum possible reaction rate (V_{max}) would be achieved when all the available enzyme is bound to the substrate and involved in catalysis, i.e.,

$$[ES] = [E_t]. \tag{4.14}$$

Substituting Eq. 4.14 into Eq. 4.4 under conditions of saturating substrate concentration, yields:

$$V_{max} = k_{cat}\,[E_t], \tag{4.15}$$

and substituting Eq. 4.15 into Eq. 4.13 yields the Michaelis-Menten equation:

$$v = \frac{V_{max}\,[S]}{K_m + [S]}. \tag{4.16}$$

In a competitive inhibition, the competitive inhibitor binds at the catalytic site, without undergoing catalysis. In this way, the presence of an inhibitor decreases the ability of the enzyme to bind with its substrate. The reaction scheme below details the mechanism. Here k_1 and k_{-1} are the rates for the association and redissociation reactions for the enzyme-substrate complex (Eq. 4.3), and k_2 and k_{-2} are the rates for the association and redissociation reactions between the enzyme and inhibitor (I).

$$\begin{array}{l} E + S \underset{k_{-1}}{\overset{k_1}{\rightleftarrows}} ES \overset{k_{cat}}{\rightarrow} E + P \\ + \\ I \\ {}^{k_2}\downarrow\uparrow_{k_{-2}} \\ EI \end{array} \tag{4.17}$$

At steady state, the enzyme-inhibitor concentration is stable; thus following Eqs 4.5-4.8, the association and redissociation rate constants for the enzyme-inhibitor complex can be combined in one term K_i, the dissociation constant for inhibitor, and following Eq. 4.9 can be expressed as:

$$K_i = \frac{[E][I]}{[EI]}. \tag{4.18}$$

Since some enzyme is bound to the inhibitor, the equation describing the total amount of enzyme has an extra term and Eq. 4.10 becomes:

$$E = E_t - ES - EI. \tag{4.19}$$

The resulting rate equation becomes:

$$v = \frac{k_{cat}\,[E_t]\,[S]}{K_m(1 + \frac{[I]}{K_i}) + [S]}, \tag{4.20}$$

and following Eq. 4.15:

$$v = \frac{V_{max}\,[S]}{K_m(1 + \frac{[I]}{K_i}) + [S]} = \frac{V_{max}\,[S]}{K_m^{app} + [S]}, \tag{4.21}$$

where:

$$K_m^{app} = K_m(1 + \frac{[I]}{K_i}). \tag{4.22}$$

As can be seen from Eq. 4.22, an increase in the concentration of a competitive inhibitor will increase the apparent K_m of the enzyme. However, since an infinite substrate concentration will exclude the competitive inhibitor, there is no effect on V_{max}.

Moreover, Eq. 4.21 can be expressed also as [148]:

$$\frac{\delta y}{\delta t} = \frac{V_{max}\,(s_0 - y)}{K_m^{app} + (s_0 - y)}, \tag{4.23}$$

where s_0 is the initial substrate concentration and the concentration at time t is $s_0 - y$.

Integrating Eq. 4.23 yields:

$$V_{max}\,t = \int_0^y \frac{s_0 - y + K_m}{s_0 - y}\,dy = y + K_m \log\left(\frac{s_0}{s_0 - y}\right). \tag{4.24}$$

Rearranging, it yields:

$$\frac{1}{t}\log(\frac{s_0}{s_0 - y}) = \frac{1}{K_m}\left(V_{max} - \frac{y}{t}\right). \tag{4.25}$$

According to Beer-Lambert law, in a non-scattering medium, the substrate concentration is related to the absorbance of an incident light by the following equation [149]:

$$A = \log\left(\frac{I_0}{I}\right) = \epsilon\,[S]\,l, \tag{4.26}$$

where A is light absorbance, I_0 is the incident light, I is the emergent light, ϵ is the specific extinction coefficient, $[S]$ is the substrate concentration and l is the light pathlength (distance between points where light enters and leaves the medium). Nevertheless, in a scattering medium, the Beer-Lambert law is modified and becomes [149]:

$$A = \log\left(\frac{I_0}{I}\right) = \epsilon\,[S]\,l \cdot DPF + G, \tag{4.27}$$

where the differential pathlength factor (DPF) is included to account for the extended pathlength due to scattering, and G represents the losses due to scattering. Consequently, quantitative measurements of $[S]$ cannot be made without knowledge of DPF and G.

Nevertheless, assuming that DPF and G are always constant, changes in $[S]$ from an arbitrary baseline are then calculated from changes in light attenuation. Thus, considering Eq. 4.25 and Eq. 4.27, s_0 and $s_0 - y$ can be expressed as:

$$\begin{aligned} s_0 &= k\,A_\infty - G_\infty - (k\,A_0 - G_0) \\ y &= k\,A_t - G_t - (k\,A_0 - G_0), \end{aligned} \tag{4.28}$$

where A_0, A_∞ and A_t represent the initial, final and at time t absorbance values respectively, G_0, G_∞ and G_t represent the initial, final and at time t scattering losses respectively and finally k represents a numerical factor, defined as:

$$k = \frac{1}{\epsilon\,l\,DPF}. \tag{4.29}$$

Assuming that DPF and G are always constant ($G_0 = G_t = G_\infty$), s_0 and $s_0 - y$ can be expressed as:

$$\begin{aligned} s_0 &= k\,(A_\infty - A_0) \\ y &= k\,(A_t - A_0), \end{aligned} \tag{4.30}$$

and Eq. 4.25 can be rewritten as:

$$\frac{1}{t}\log\left(\frac{A_\infty - A_0}{A_\infty - A_t}\right) = \frac{1}{K'_m}\left(V'_{max} - \frac{A_\infty - A_t}{t}\right), \tag{4.31}$$

where $K_m^{'}$ and $V_{max}^{'}$ are K_m and V_{max} expressed in absorbance and absorbance variation per unit time respectively. Eq. 4.31 represents the integrated Michaelis-Menten equation.

4.2.1.1 Graphical determination of K_m and V_{max}

In Figure 4.2, a schematic representation of the rate of product formation is shown.

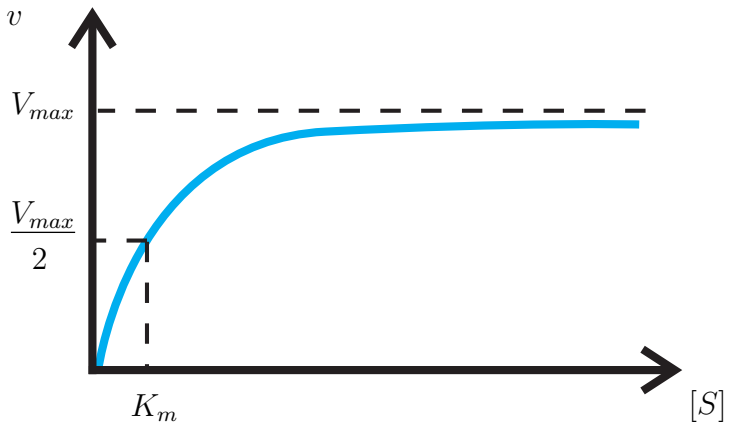


Figure 4.2: Initial rate (v) of an enzyme-catalyzed reaction versus substrate concentration ($[S]$) and graphical determination of V_{max} and K_m.

In principle, V_{max} and K_m can be determined from a graph as the one reported in Figure 4.2. Nevertheless, the plot of v versus $[S]$ is not very useful in determining V_{max} when high substrate concentrations are used. A better approach, suggested by the American chemists Lineweaver and Burk, is to use the double-reciprocal plot of $1/v$ and $1/[S]$. Thus, Eq. 4.16 can be rewritten as:

$$\frac{1}{v} = \frac{K_m}{V_{max}\,[S]} + \frac{1}{V_{max}}. \tag{4.32}$$

As shown in Figure 4.3a, V_{max} and K_m can be obtained from the slope and the intercept of the straight line. The disadvantage of the Lineweaver-Burk is that it compresses the points at high substrate concentrations whereas it emphasizes the less concentrations points which are the least accurate.

Another way of plotting the kinetic data is the Eadie-Hofstee method, whose describing equation can be obtained by Eq. 4.32 multiplying both sides for $v\,V_{max}$, thus obtaining:

$$V_{max} = v + \frac{v\,K_m}{[S]}. \tag{4.33}$$

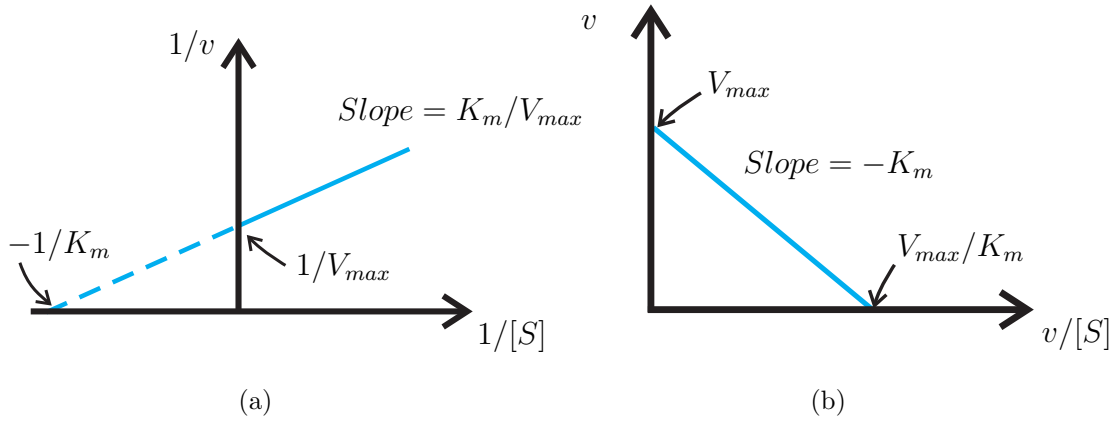


Figure 4.3: Lineweaver-Burk (a) and Eadie-Hofstee (b) plot for an enzyme-catalyzed reaction obeying Michaelis-Menten kinetics.

Rewriting Eq. 4.33 in terms of v, it gives:

$$v = V_{max} - \frac{v_0 \, K_m}{[S]}. \tag{4.34}$$

The Eadie-Hofstee plot, described by Eq. 4.34, gives a straight line with slope equal to K_m and intercepts V_{max} on the v_0 axis and V_{max}/K_m on the $v_0/[S]$ (Figure 4.3b).

Finally, considering Eq. 4.31 and applying the Eadie-Hofstee method, it gives a straight line with slope equal to $1/K_m^{'}$ and intercepts $V_{max}^{'}/K_m^{'}$ on the $1/t \, \log(A_\infty - A_0)/(A_\infty - A_t)$ axis and $V_{max}^{'}$ on the $|A_t - A_0|/t$ axis (Figure 4.4).

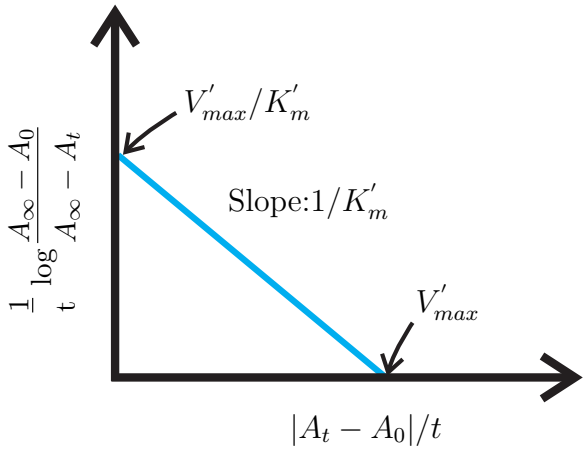


Figure 4.4: Eadie-Hofstee plot applied to the integrated Michaelis-Menten equation.

4.2.2 Instrument optical design

The optics consists of four light sources: two LEDs centered at a wavelength of $\lambda_{peak} = 365\ nm$ (XSL-365-3E, Roithner LaserTechnik) and two LEDs centered at a wavelength of $\lambda_{peak} = 405\ nm$ (LED-405-33V, Roithner LaserTechnik) respectively, and a photodiode (HFD 1100, EG&G), placed directly in contact with the walls of a cuvette holder. In order to realize a low cost device, no other optics were used. In addition, since only one photodiode has been used, the measurement at the two different wavelengths has been performed with a time division multiplexing (TDM) technique. In this way, the two LEDs are switched on at different times, for a time interval of 3 seconds each. After switching off each LED, no LED is switched on, for a time interval of 3 seconds, in order to measure the blank signal. Thus, the signal of interest is determined considering the difference between the mean value, determined with the LED on, and the baseline value, measured when all LEDs are off. Different cuvettes where used, in particular a glass cuvette with an optical path of 1 mm and disposable plastic cuvettes with an optical path of 10 mm. Finally, a schematic design of the optical setup is shown in Figure 4.5.

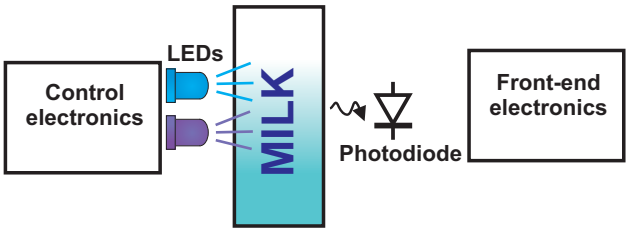
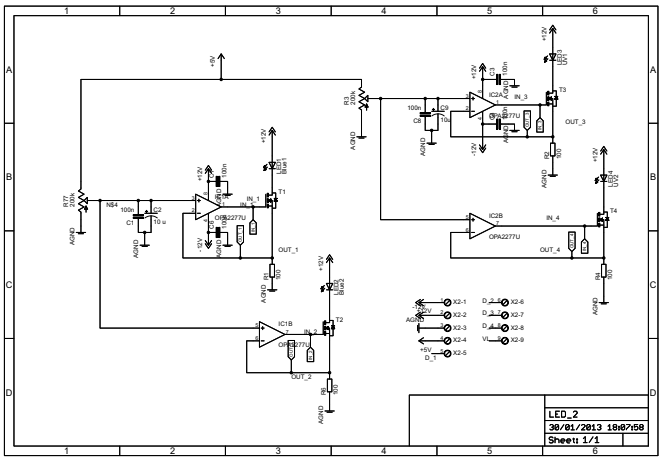


Figure 4.5: Schematic representation of the optical setup for the absorbance variation measurement. It consists of four LEDs (two centered at $\lambda_{peak} = 365\ nm$ and two centered at $\lambda_{peak} = 405\ nm$ respectively) and a photodiode.

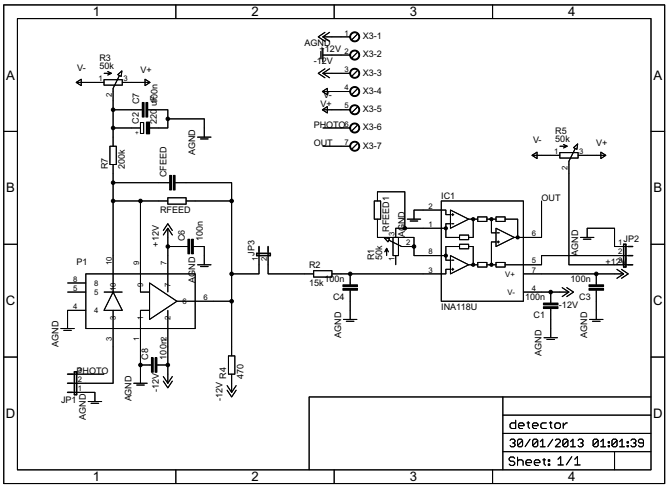
4.2.3 Electronics

A driver, front-end and control electronics has been designed and realized. The driver electronics (Figure 4.6a) consists of four current generators to drive the LEDs. Each current generator can be switched off through an analog switch (short-circuit between the gate and the source of the MOSFET of the current generator). The front-end electronics consists of a transimpedance amplifier to convert the current photogenerated by the incident light on the photodiode, together with an instrumentation amplifier to amplify the signal (Figure 4.6b). Finally, a commercial Arduino Uno board has been used to acquire the signals of interest, to control the system and to communicate with the PC through a USB port, as schematically shown in

Figure 4.7.



(a)



(b)

Figure 4.6: Electrical schematics of the driver (a) and front-end electronics (b). The first one consists of four current generators to drive the LEDs, whereas the second one consists of a transimpedance amplifier and an instrumentation amplifier.

Pictures of the realized system are shown in Figure 4.8; it can be noticed the cuvette holder, optics and electronics (4.8(a-b)) and the whole system (4.8(c-d)).

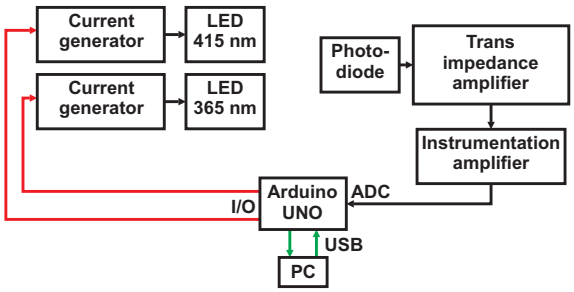
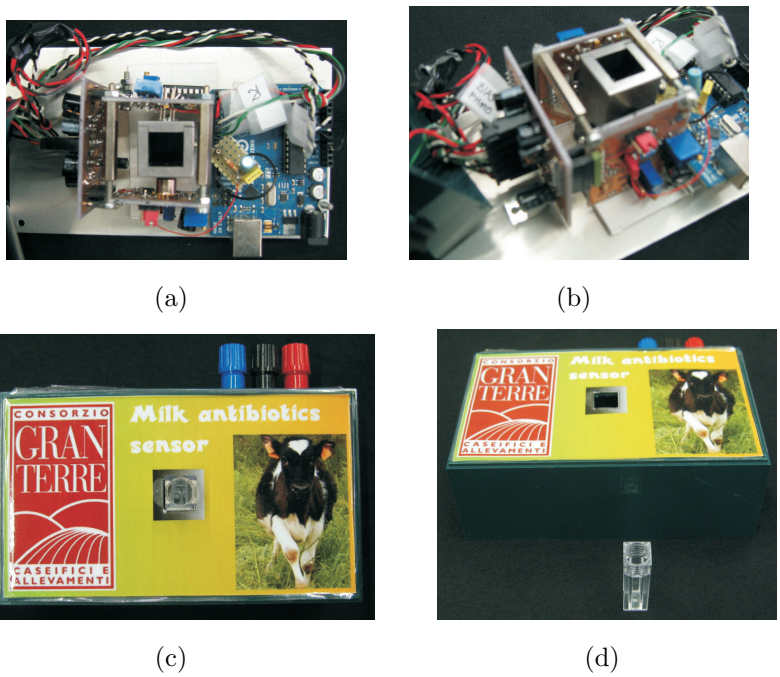


Figure 4.7: Schematic representation of the electronics. It consists of four current generators to drive the LEDs, a transimpendance amplifier and an instrumentation amplifier to convert and amplify the photogenerated current, and an Arduino Uno board to communicate with the PC and to control all the electronics.



(a) (b)

(c) (d)

Figure 4.8: Pictures of the realized system: cuvette holder, optics and electronics (a-b) and the whole system (c-d).

4.2.4 Software interface

A software interface has been developed in LabVIEW to manage all the system. In particular, it allows to record the signals of interest and calibrate the system. Moreover, it automatically determines the quantities of interest, representative of the antibiotic concentrations. A picture of the main interface is shown in Figure 4.9, whereas the VI hierarchy is reported in Figure 4.10.

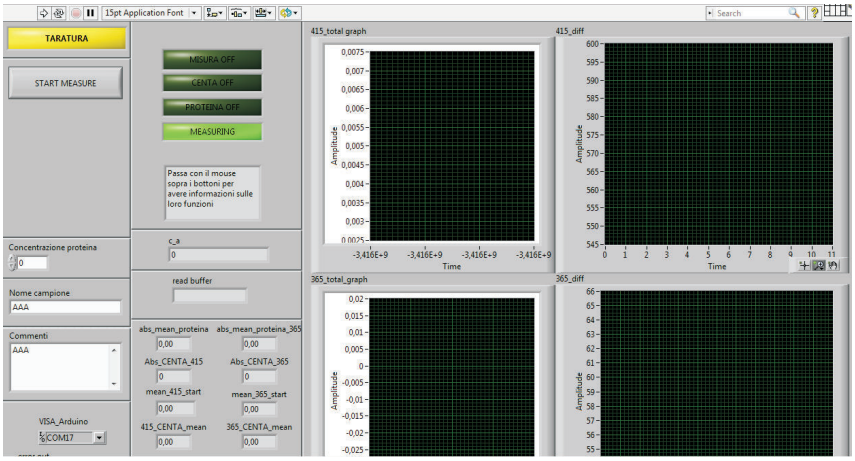


Figure 4.9: Software interface realized in LabVIEW.

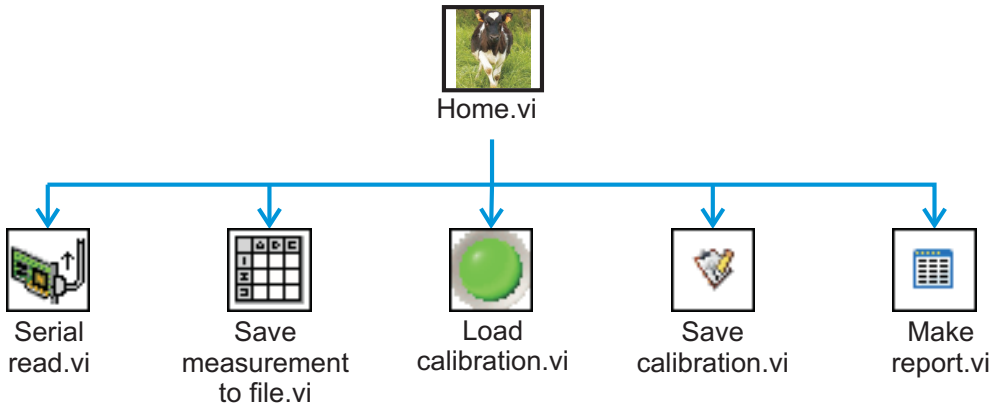


Figure 4.10: VI hierarchy of the software interface, developed in LabVIEW, to manage all the system.

4.3 Experimental protocol

The information of interest is represented by the absorbance variation at 405 nm and 365 nm in the presence of the antibiotic. In particular, the presence of the antibiotic determines a change both in the enzyme kinetic (K_m^{app}) and in the final settled absorbance value (A_∞) [150], as described in Section 4.2.1. Thus, combining these two information, it is possible to have a better estimation of the antibiotic concentration.

Two different experimental tests have been performed. The first one has been done using a spectrometer whereas the second one using the developed system. In the first one, since the spectrometer is able to determine the absorbance comparing the signal of interest with a reference channel, no further calculations have been performed. On the other side, in the developed system there is no reference path. For this reason, it is necessary to measure the output voltage of the developed system in the following conditions:

- Milk;
- Milk + CENTA;
- Milk + CENTA + Protein (Enzyme AmpC).

In this way, it is possible to define the following quantities:

$$A_{CENTA} = \frac{V_{CENTA}}{V_{Milk}}, \tag{4.35}$$

$$A_{Protein} = \frac{V_{Protein}}{V_{Milk}}, \tag{4.36}$$

where V_{Milk} is the output signal with only milk in the sample at 405 nm, V_{CENTA} the output signal with milk and CENTA at 405 nm and finally $V_{Protein}$ is the output signal with milk and protein after having added CENTA at 405 nm.
In general, the output voltage signal is proportional to the light incident on the photodiode. Thus, considering the modified Beer-Lambert law (Eq. 4.27), V_{CENTA} and $V_{Protein}$ can be expressed as:

$$\begin{aligned} V_{CENTA} &= I_0 10^{-G}, \\ V_{Protein} &= I_0 10^{-(\epsilon_B\,[C_B]\,l\cdot DPF+G)}, \end{aligned} \tag{4.37}$$

where ϵ_B is the specific extinction coefficient of the binded substrate at $\lambda = 405\ nm$ and $[C_B]$ is the concentration of substrate (CENTA) binded with the protein.

When V_{CENTA} is measured, $[C_B]$ is equal to zero because no protein is present in the cuvette. Thus, the following quantity $Abs(t)$ can be defined as:

$$Abs(t) = \log\left(\frac{A_{CENTA}}{A_{Protein}}\right) = \log\left(\frac{V_{Milk}}{V_{Protein}} \cdot \frac{V_{CENTA}}{V_{Milk}}\right) = \log\left(\frac{V_{CENTA}}{V_{Protein}}\right). \quad (4.38)$$

Substituting Eq. 4.37 into Eq. 4.38 yields:

$$\begin{aligned} Abs(t) &= \log\left(\frac{I_0 10^{-G}}{I_0 10^{-(\epsilon_B [C_B] l \cdot DPF + G)}}\right), \\ &= \epsilon_B [C_B] l \cdot DPF. \end{aligned} \quad (4.39)$$

Thus, $[C_B]$, is proportional to:

$$[C_B] = k_B \, Abs(t), \quad (4.40)$$

$$(4.41)$$

where k_B is defined as:

$$k_B = \frac{1}{\epsilon_B \, l \cdot DPF}. \quad (4.42)$$

Since the antibiotic concentration is inversely proportional to $[C_B]$, the final settled absorbance value (A_∞) is inversely proportional to the antibiotic concentration (Eq. 4.40). The higher the antibiotic concetration, the lower the A_∞ value.

Finally, in order to determine the A_∞ values, the absorbance values ($Abs(t)$) have been interpolated with the following function:

$$Abs(t) = A_\infty - b \, e^{-a \, t}, \quad (4.43)$$

where a and b are numerical values and A_∞ represents the final settled absorbance value.

Thus, for the measurements performed with the developed system, the following quantities have been determined: $K_m^{'}$ and A_∞, as defined in Eq. 4.31 and Eq. 4.43 respectively.

4.4 Experimental results

The experimental tests have been performed firstly using a spectrometer to study the feasibility of the sensing approach and then with the developed system, in order to verify its functionality. In the next Sections, the obtained experimental results are presented.

4.4.1 Preliminary spectroscopic analysis

The first experimental tests have been performed in buffer solution with a glass cuvette with an optical path of 1 mm, using a spectrometer (Perkin Elmer). The volume of buffer solution, in each experiment, was 300 μl. The components concentrations are reported in Table 4.2.

Table 4.2: Component concentrations for the tests performed with the spectrometer.

Component	**Stock concentration**	**Added volume**	**Cuvette concentration**
Test on Buffer			
Buffer		300 μl	
CENTA	5 mM	15 μl	240 μM
Protein	13.85 μM	3 μl	130 nM
Test on Diluted Milk			
Buffer		150 μl	
Milk		150 μl	
CENTA	5 mM	15 μl	240 μM
Protein	13.85 μM	3 μl	130 nM
Test on Milk			
Milk		300 μl	
CENTA	5 mM	15 μl	240 μM
Protein	13.85 μM	3 μl	130 nM
Cloxacillina	0.5 μM	3 μl	5 nM ($\approx$ 2.2 $\mu g/kg$)[a]
		6 μl	10 nM ($\approx$ 4.4 $\mu g/kg$)

[a] Cloxacillina molecular weight: 435.88 g/mol.

The scan-rate was 60 nm/min. In Figure 4.11 the measurements performed in buffer solution are shown. They have been performed following the protocol described in Section 4.3, firstly measuring the absorbance of buffer, then the absorbance of buffer and CENTA and finally measuring the absorbance of buffer, CENTA and protein, at consecutive time intervals (3 minutes).

It can be noticed that CENTA (green line) presents an absorption peak at 345 nm. In the presence of the protein, the CENTA reacts with the protein active site and the absorption peak is shifted to 405 nm.

These measurements have been performed also in milk and diluted milk, using the same protocol. In Figure 4.12, the measurements performed in milk (volume 300 μl), are shown, whereas, Figure 4.13 shows the measurements performed in diluted milk (milk 150 μl and buffer 150 μl).

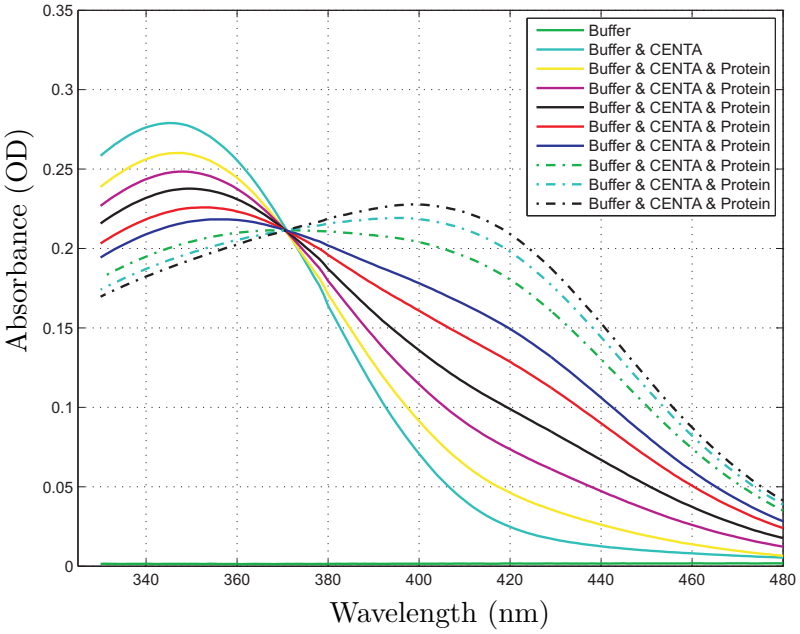


Figure 4.11: Absorbance curves of the buffer solution, buffer solution and CENTA, buffer solution and CENTA and protein, at consecutive time intervals (3 minutes). The measurements have been performed with a spectrometer (cuvette: 1 mm, scan-rate: 60 nm/min).

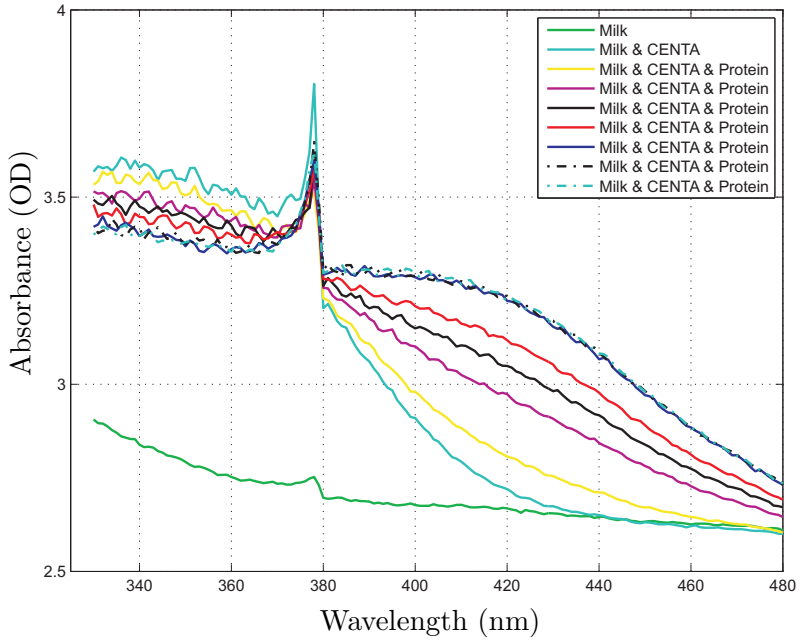


Figure 4.12: Absorbance curves of milk, milk and CENTA, milk and CENTA and protein, at consecutive time intervals (3 minutes). The measurements have been performed with a spectrometer (cuvette: 1 mm, scan-rate: 60 nm/min).

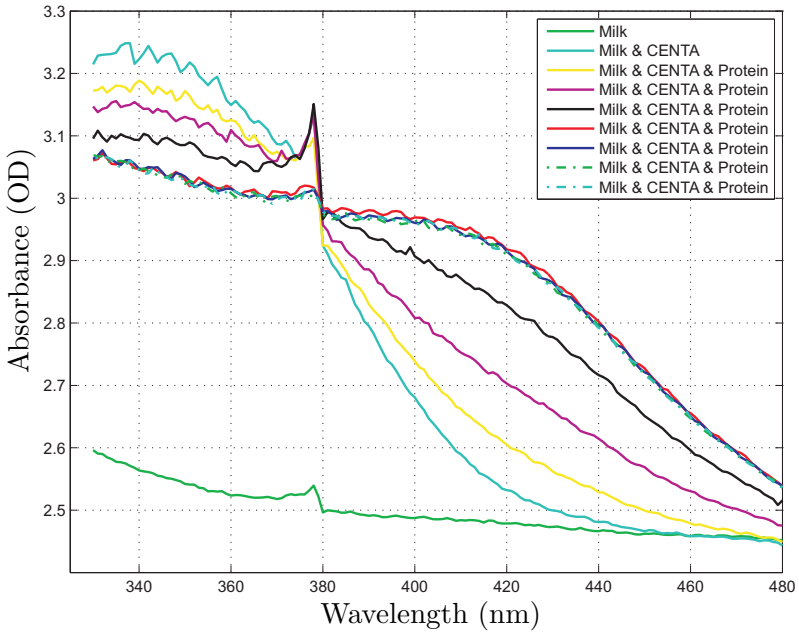


Figure 4.13: Absorbance curves of diluted milk, diluted milk and CENTA, diluted milk and CENTA and protein, at consecutive time intervals (3 minutes). The measurements have been performed with a spectrometer (cuvette: 1 mm, scan-rate: 60 nm/min).

Finally, these measurements were repeated in milk with a non zero concentration of antibiotic. In particular, the first test has been performed with no antibiotic, the second test with antibiotic (cloxacillina), concentration 5 nM, and finally the third test with antibiotic (cloxacillina), concentration 10 nM. In Figure 4.14, the final settled absorbance curves for the three different tests are shown, whereas, in Figure 4.15, the absorbance versus time, measured at 405 nm, is shown, for the three different samples.

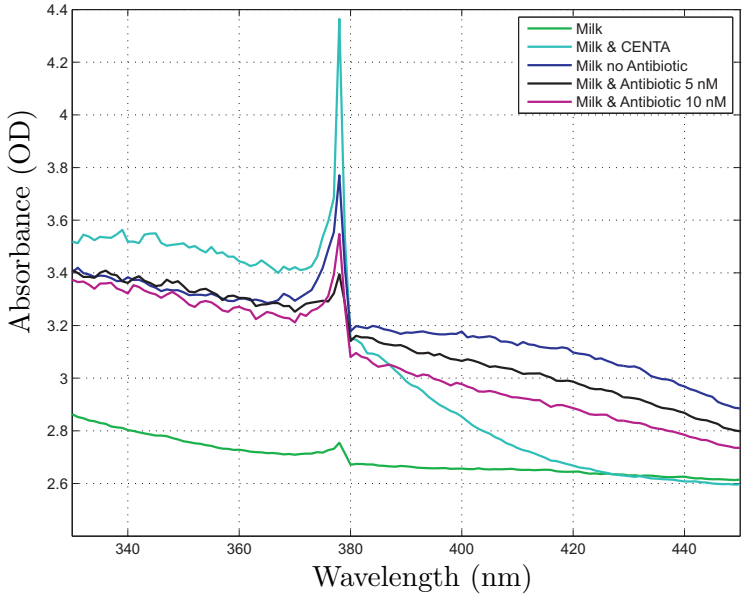


Figure 4.14: Absorbance settled curves of milk (green), milk and CENTA (cyan), milk and CENTA and protein (blue), milk and CENTA and protein and antibiotic (5 nM) (black), milk and CENTA and protein and antibiotic (10 nM) (purple). The measurements have been performed with a spectrometer (cuvette: 1 mm, scan-rate: 60 nm/min).

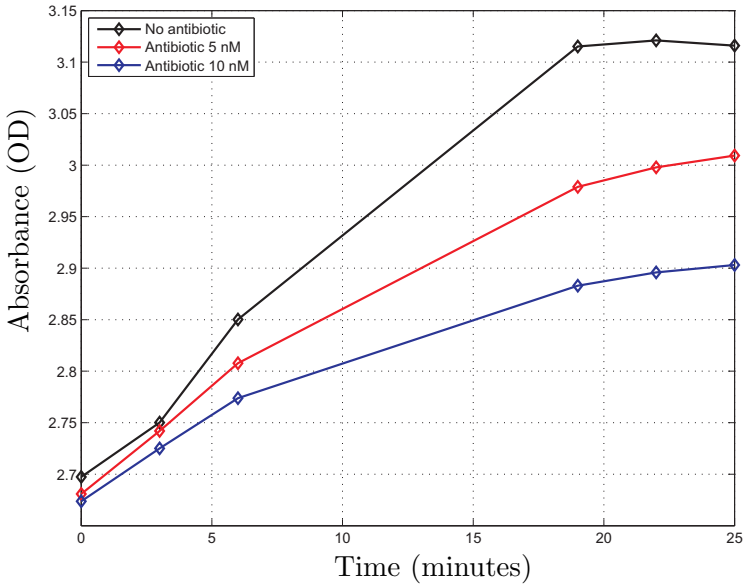
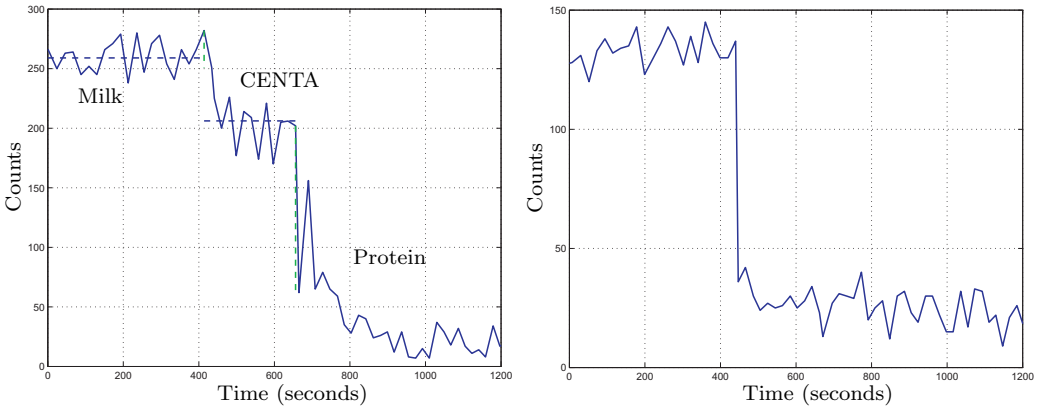


Figure 4.15: Absorbance recorded at 405 nm, without antibiotic (black), with antibiotic, concentration 5 nM, (red) and with antibiotic, concentration 10 nM (blue). The measurements have been performed with a spectrometer (cuvette: 1 mm, scan-rate: 60 nm/min).

4.4.2 Functional tests of the developed system

The measurements with the developed system have been performed following the protocol described in Section 4.3. A typical curve recorded with the developed system is shown in Figure 4.16, at the two different wavelength, 405 nm (Figure 4.16a) and 365 nm (Figure 4.16b) respectively. The first part represents the signal measured with only milk, the second one after having added CENTA and finally, the last one, after having added the protein.



(a) (b)

Figure 4.16: A typical curve recorded with the developed system at the two different wavelength, 405 nm and 365 nm respectively. The first part represents the signal measured with only milk, the second one after having added CENTA and finally, the last one, after having added the protein.

In the next sections, three different tests will be described. For each test, parameters A_∞ and $K_m^{'}$ were determined. In the presence of the antibiotic, A_∞ should decrease whereas $K_m^{'}$ should increase.

4.4.2.1 Test 1

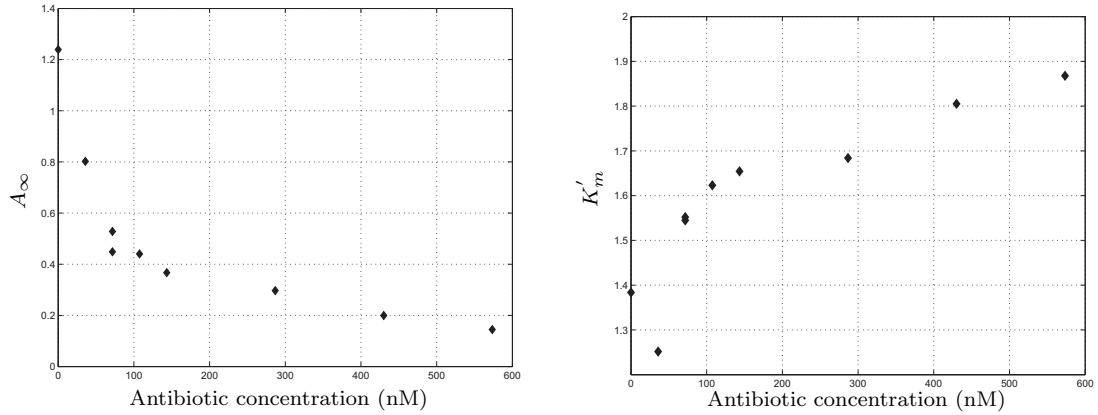
The first test has been performed with a cuvette with an optical path of 10 mm, using the developed system. The component concentrations are reported in Table 4.3.

Table 4.3: Component concentrations for the 1° test.

Component	**Stock concentration**	**Added volume**	**Cuvette concentration**
Milk		80 μl	
Buffer		720 μl	
CENTA	8 mM	27 μl	261 μM
Protein	13.85 μM	10 μl	165 nM
Oxacillina	50 μM	0.6 μl	35 nM ($\approx$ 14 $\mu g/kg$)[a]
		1.2 μl	70 nM ($\approx$ 28 $\mu g/kg$)
		1.8 μl	105 nM ($\approx$ 42 $\mu g/kg$)
		2.4 μl	140 nM ($\approx$ 56 $\mu g/kg$)
		4.8 μl	280 nM ($\approx$ 112 $\mu g/kg$)
		7.2 μl	420 nM ($\approx$ 168 $\mu g/kg$)
		9.6 μl	560 nM ($\approx$ 224 $\mu g/kg$)

[a] Oxacillina molecular weight: 401.436 g/mol.

In Figure 4.17a, A_∞ values versus antibiotic concentration are shown, whereas in Figure 4.17b the $K_m^{'}$ values versus antibiotic concentration are shown.



(a) (b)

Figure 4.17: A_∞ (a) and $K_m^{'}$ (b) versus antibiotic concentration, measured using the developed system, 1° test. CENTA 261 μM, protein 165 nM, Antibiotic: Oxacillina.

4.4.2.2 Test 2

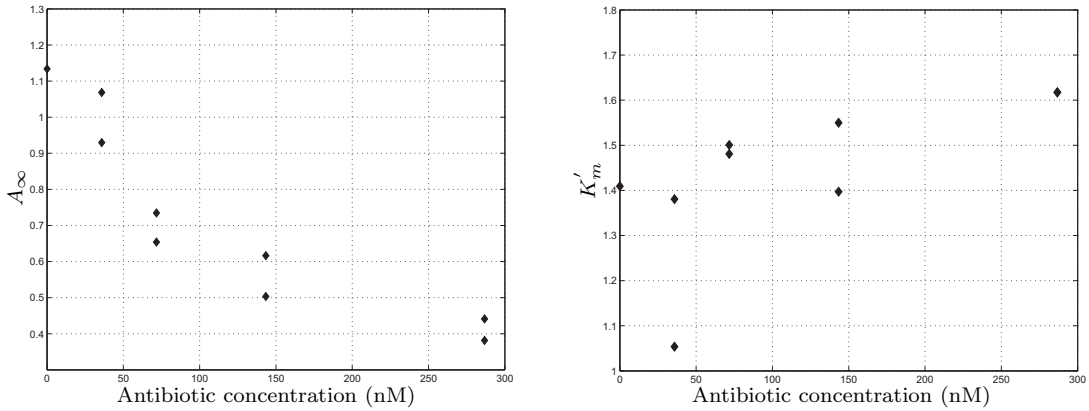
The second test has been performed with a cuvette with an optical path of 10 mm. The component concentrations are reported in Table 4.4.

Table 4.4: Component concentrations for the 2° test.

Component	**Stock concentration**	**Added volume**	**Cuvette concentration**
Milk		80 μl	
Buffer		720 μl	
CENTA	8 mM	27 μl	261 μM
Protein	13.85 μM	10 μl	165 nM
Oxacillina	50 μM	0.6 μl	35 nM ($\approx$ 28 $\mu g/kg$)[a]
		1.2 μl	70 nM ($\approx$ 14 $\mu g/kg$)
		2.4 μl	140 nM ($\approx$ 14 $\mu g/kg$)
		4.8 μl	280 nM ($\approx$ 14 $\mu g/kg$)

[a] Oxacillina molecular weight: 401.436 g/mol.

In Figure 4.18a, A_∞ values versus antibiotic concentration are shown, whereas in Figure 4.18b the $K_m^{'}$ values versus antibiotic concentration are shown.



(a) (b)

Figure 4.18: A_∞ (a) and $K_m^{'}$ (b) versus antibiotic concentration, measured using the developed system, 2° test. CENTA 261 μM, protein 165 nM, Antibiotic: Oxacillina.

4.4.2.3 Test 3

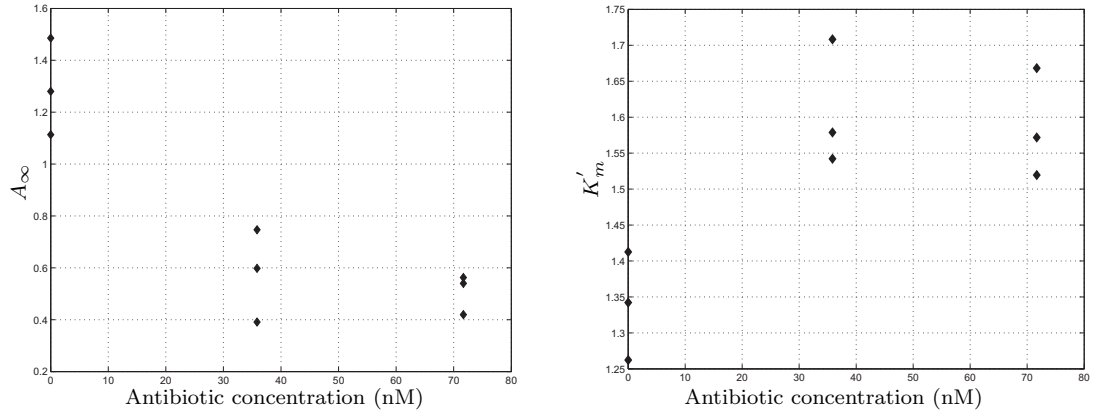
The third test has been performed with a cuvette with an optical path of 10 mm. The component concentrations are reported in Table 4.5.

Table 4.5: Component concentrations for the 3° test.

Component	**Stock concentration**	**Added volume**	**Cuvette concentration**
Milk		80 μl	
Buffer		720 μl	
CENTA	8 mM	27 μl	261 μM
Protein	13.85 μM	10 μl	165 nM
Oxacillina	50 μM	0.6 μl	35 nM ($\approx$ 28 $\mu g/kg$)[a]
		1.2 μl	70 nM ($\approx$ 14 $\mu g/kg$)

[a] Oxacillina molecular weight: 401.436 g/mol.

In Figure 4.19a, A_∞ values versus antibiotic concentration are shown, whereas in Figure 4.19b the $K_m^{'}$ values versus antibiotic concentration are shown.



(a) (b)

Figure 4.19: A_∞ (a) and $K_m^{'}$ (b) versus antibiotic concentration, measured using the developed system, 3° test. CENTA 261 μM, protein 165 nM, Antibiotic: Oxacillina.

4.5 Conclusions

A system for the fast determination of antibiotic concentrations in milk samples has been developed and preliminary functional tests have been performed. The system is based on an optical competitive method, deploying the competition between a particular substrate (CENTA) and the antibiotic for the protein active site. The optics and electronics have been designed and realized, together with a software interface. Preliminary tests have been performed with a spectrometer and the developed system, both on buffer solutions and milk samples. The system is able to discriminate different antibiotic concentrations, in particular a relation between A_∞ and antibiotic concentration has been observed; nevertheless, further experimental tests have to be performed in order to better characterize the system. Moreover, the experimental protocol needs to be standardized in order to diminish the uncertainty due to the manual addition of the components (e.g. CENTA, protein) to the sample under test. This can be achieved by using an automated system or a ready-to-use test kit.

4.6 Comparison with the state of the art

Name	Description	Principle	Method	Reaction time	Detection limit
Developed system	Optical sensor for the determination of antibiotic concentration in milk samples	Competition between CENTA and antibiotic	Absorbance variation at 405 nm	<3 min	Below MRLs
TwinsensorBT	Receptor-based assay in a dipstick format allowing rapid and simultaneous detection of β-lactams and tetracyclines present in a milk sample	Competitive method	Colourimetric test	≈6 min at 50°C	Qualitative test (Above or below MRLS)
IDEXX SNAP Beta-Lactam ST Test	Optical sensor for the determination of antibiotic concentration in milk samples, without incubation or heat	Competitive method	Colourimetric test	≈6 min	Qualitative test (Above or below MRLS)

Table 4.6: Characteristics comparison between the developed system and the main commercial systems.

Conclusions

Biomedical and agri-food industry always requires new sensors to monitor biological parameters and to check the quality and health of food. Nevertheless, the variety and complexity of the sensing environments make the sensing issue very challenging. For these reasons, different sensing methods have to be investigated.

In particular, in the present thesis, the design and realization of four different sensors for biomedical and agri-food industry has been described. In order to face the complexity and variety of the application fields, both optical and electrochemical methods have been used. In particular, two sensors for biomedical industry and two sensors for agri-food industry have been developed: a multi-analyte fluorescence based sensor platform and a combination of electrochemical and SPR techniques, an electrochemical sensor to determine the glycerol concentration in grape juice and an optical sensor to estimate the presence of antibiotics in milk.

The multi-analyte fluorescence based sensor platform consists of a polymer sensing element, including a fluorescent dye, placed in contact with the sample under test, an optical head to excite the fluorescent dye and collect the fluorescence signal and electronics to drive the light source, to collect and process the information of interest. The developed system has been used for the measurement of blood pH and pCO_2 during extracorporeal circulation. These two sensors differ only for the polymer sensing element.

In particular, the pH sensor is based on the pH-dependent fluorescence of a new kind of fluorescent monomer (fluorescein *O*-methacrylate 97%). Since the pK_a of the sensing element has been calculated to be 7.9, the sensor is suitable for measurement of near neutral solutions. Good performance in terms of linearity (in the range 7.0-8.0), stability and reproducibility were observed.

In the tests performed on blood, the sensor has shown a very high correlation (R of 0.9887 and 0.9936 in-vitro and R of 0.9431 and 0.9901 ex-vivo) between pH value determined by our device and pH measured by a commercial blood gas analyzer during in-vitro and ex-vivo experiments. After calibration, a mean square error lower than the 3% of the RoI was determined both for in-vivo and ex-vivo measurements.

Considering an ipothermic treatment at 32°C, the mean square error, calculated with the calibration curve obtained considering only the pH values recorded at 37°C, was lower than 3% of the measuring range. This means that this sensor is suitable for iphotermic treatment without the need of a new calibration curve.

However a different set of calibration coefficients was obtained for each sensing element. This variation is mainly associated to the reproducibility of the deposition of the fluorescence film on the substrate. Therefore a calibration procedure before each ECC treatment is required. This might be problematic, since at least two points of calibration are required. This procedure leads to two main disadvantages: (i) a commercial blood gas analyzer is always required in order to calibrate the sensor (limiting its field of application); (ii) calibrating the sensor on a human being may be difficult since, in normal conditions, pH has little variations. This means that the sensor is likely to be calibrated with two near points, with a high probability of committing a great error for pH values out of the range of calibration. These problems could be overcome calibrating the sensor with two buffer physiological solutions with an already-known pH. Nevertheless, this approach could be problematic for perfusionists who are accustomed to use only one priming physiological solution.

The fluorescence pCO_2 sensor, instead, is based on the pH-dependent fluorescence of a purposely developed polymer matrix including a fluorescent dye (HPTS). Preliminary tests have been performed both in air and on cow blood. The sensor response has been coherent with the pCO_2 variations, showing a linear behavior for pCO_2 concentrations between 0 and ca. 70 mmHg. The linear range depends both on the pCO_2 of the fluorescent indicator and on the concentration of the phase transfer agent. Thus, by properly tuning the phase transfer agent concentration, the linear range can be increased.

Nowadays, with the advent of the genomic era, nucleic acid testing technologies are becoming very important. However, it has been recognized that the molecular packing density within the DNA layer crucially determines the functionality of the nucleic acids. For instance, the ability of surface immobilized probe strands to capture complementary target sequences from solution is largely suppressed if the layer density is too high. As a consequence, electrical induced desorption of DNA probes is becoming a common technique to control the DNA surface density. At the same time, surface plasmon resonance is a powerful tool to monitor the changes in the refractive index at a solid-liquid interface. Thus, it has been widely used to characterize interactions between biomolecules. For these reasons, a combination of SPR and electrochemical techniques can widely improve the system, allowing for a simultaneous electrical DNA probes desorption and monitoring of the solid-liquid interface. Thus, a system combining electrochemical and SPR techniques has been

developed. It is based on a four-channel SPR system developed at the Institute of Photonics and Electronics AS CR, Prague. An in-situ electrochemical cell has been realized in each channel of the flow cell. Two different geometries have been proposed for the electrodes realization. Experimental tests have been performed with the whole system, comparing the electrochemical and SPR signals. Cyclic voltammetries performed in PBS and ferricyanide have been used to determine the pseudo-reference potential of the electrochemical cells. Preliminary tests have been performed to study the DNA probes desorption and regeneration. In particular, different negative potentials and different DNA probes have been used. In general, it has been noticed that it is possible to regenerate the DNA sensing chip through an electrical induced desorption. The desorption threshold potential and the desorption efficiency depend both on the applied negative potentials and on the DNA probes. It has been noticed that the longer DNA probes, the more negative potential is required to induce the desorption. Nevertheless, further experimental tests are required in order to better characterize the system and study the electrical induced desorption process.

The quality control of wine, since the grape harvest, is becoming a problem of great importance. Nevertheless, traditional food-inspection techniques rely on sample collection and subsequent off-line analysis in a laboratory through periodic chemical and microbiological tests. These methods do not allow an easily continuous monitoring, because they are expensive, slow, need well trained operators and in some cases require steps of extraction or sample pre-treatment. At the same time, glycerol concentration in grape juice has proven to be a reliable indicator for the future quality of wine. In particular, deviations from its normal value might indicate technological alterations during the process or deterioration of the harvested grapes.

In order to allow a quality control directly at the winery, during the grape harvest, a fast and reliable sensor is required. For all these reasons, a portable system for the fast estimation of glycerol concentration in grape juice has been developed.

Thus, an electrochemical sensor has been realized, together with an automated system through the use of micropumps. The system is able to automatically collect the samples from the corer, which takes some grape samples from the trucks, to dilute them with buffer solution and add all the reactants needed for the measurement.

Experimental tests have been performed both in laboratory and during the grape harvest. In order to perform the measurement in less than 3 minutes (time constraint defined by the winery), an external calibration curve has been used for all the grape kinds (white and red), previously determined in a mute grape juice, and only one replicate has been considered. Although this approach worsens the measurement accuracy, the instrument can correctly classifies grapes based on the content of glyc-

erol, providing an important tool for an objective classification of the grapes, not biased by the visual judgment.

Finally, milk is one of the most important food, both as a beverage and to produce milk derivatives, such as cheese and butter. For this reason, milk is one of the most regulated product in food industry. As a consequence, there is an increasing request of a method capable of a fast and multi-antibiotic sensing. In fact, lots of present test methods are laborious and time-consuming and can be performed only off-line.

For these reasons, a new system for the fast determination of antibiotic concentrations in milk samples has been studied. The system is based on an optical competitive method, deploying the competition between a particular substrate (CENTA) and the antibiotic for the protein active site. The optics and electronics have been designed and realized, together with a software interface. Preliminary tests have been performed with a spectrometer and the developed system, both on buffer solutions and milk samples. The system is able to discriminate different antibiotic concentrations, nevertheless, further experimental tests have to be performed in order to better characterize the system. Moreover, the experimental protocol needs to be standardized in order to diminish the uncertainty due to the manual addition of the reactants to the sample under test. This can be achieved by using an automated system or a ready-to-use test kit.

Future works will look at improving all the sensors in order to make them suitable for a direct implementation in their proper application field.

In particular, in the multi-analyte fluorescence based sensor platform, great effort will be done to improve the reproducibility of the sensing membranes, so that it could be possible to calibrate some sample sensors of the same production lot before the use on ECC and then using the same calibration curve for all the sensors of the lot. Thus, it could be possible to use a generic sensor on ECC just taking a point of calibration, in order to adjust the calibration offset.

Further tests are needed for the electrochemical-SPR system, in order to better control the DNA density and better understand the relation between applied potential and DNA desorption. The combination of these two techniques could lead to a very powerful instrument, which deploys the advantages of both techniques.

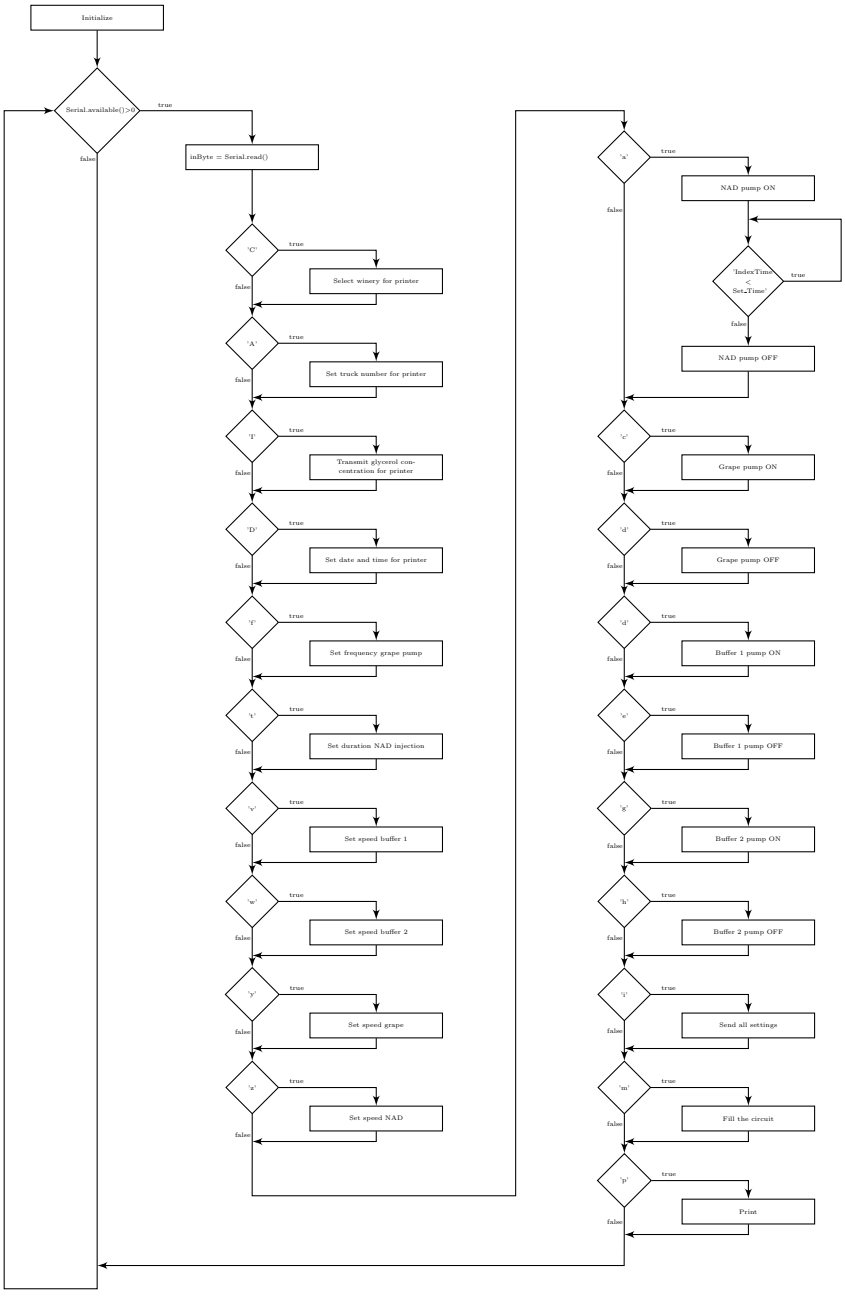
Finally, further field-tests are needed for the developed agri-food sensors. In fact, both grape juice and milk are very complex matrices and a great effort has to be done in order to take into account their great variety and to test the sensors reliability within a short test time (less than 3 minutes).

In conclusion, a great effort has been done to design and realize biomedical and agri-food sensors. The developed systems have placed the basis for future improvements in biomedical monitoring and in food quality control.

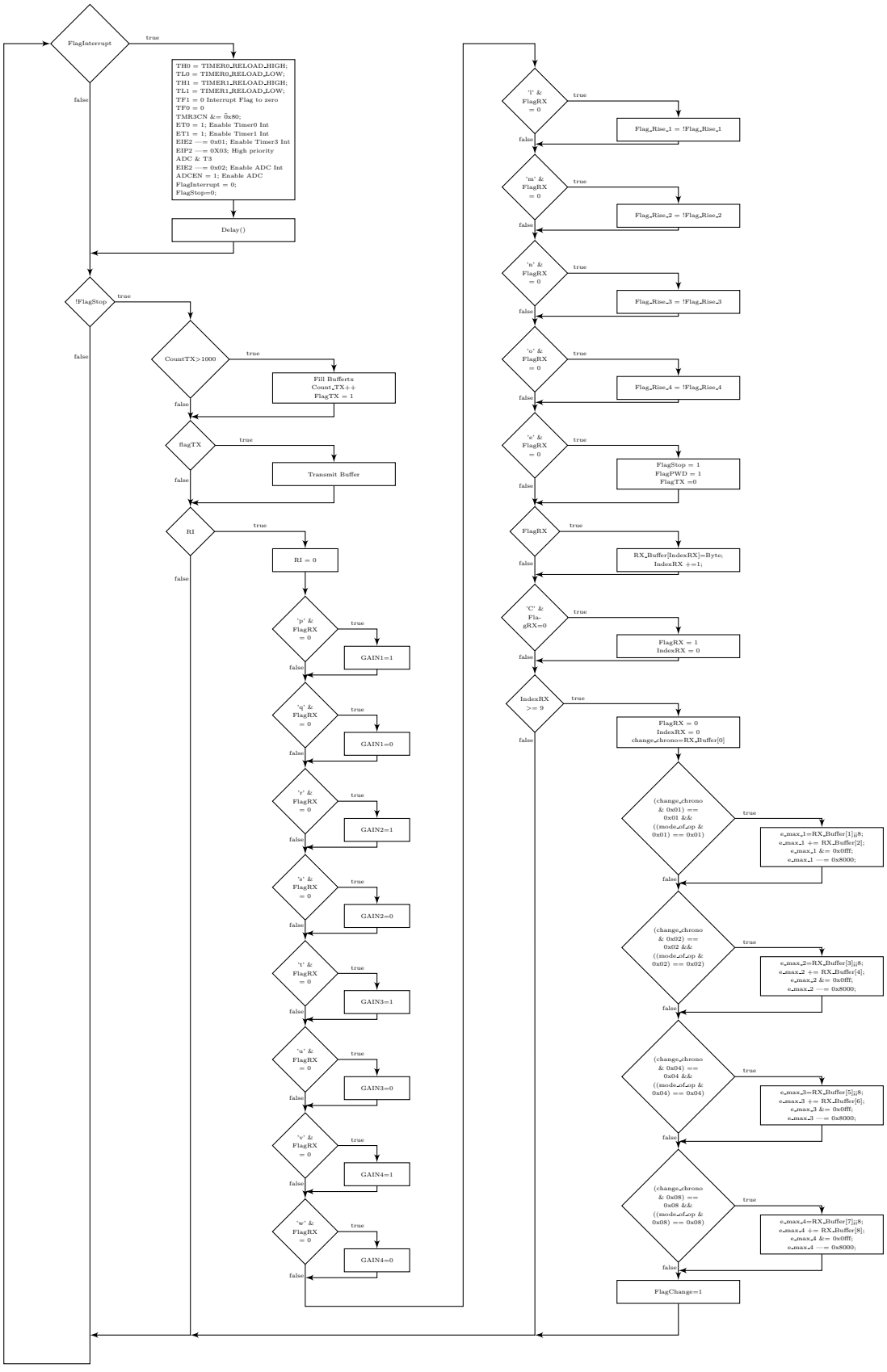
"You cannot manage what you do not measure"

Flow charts

A.1 Glycerol concentration sensor

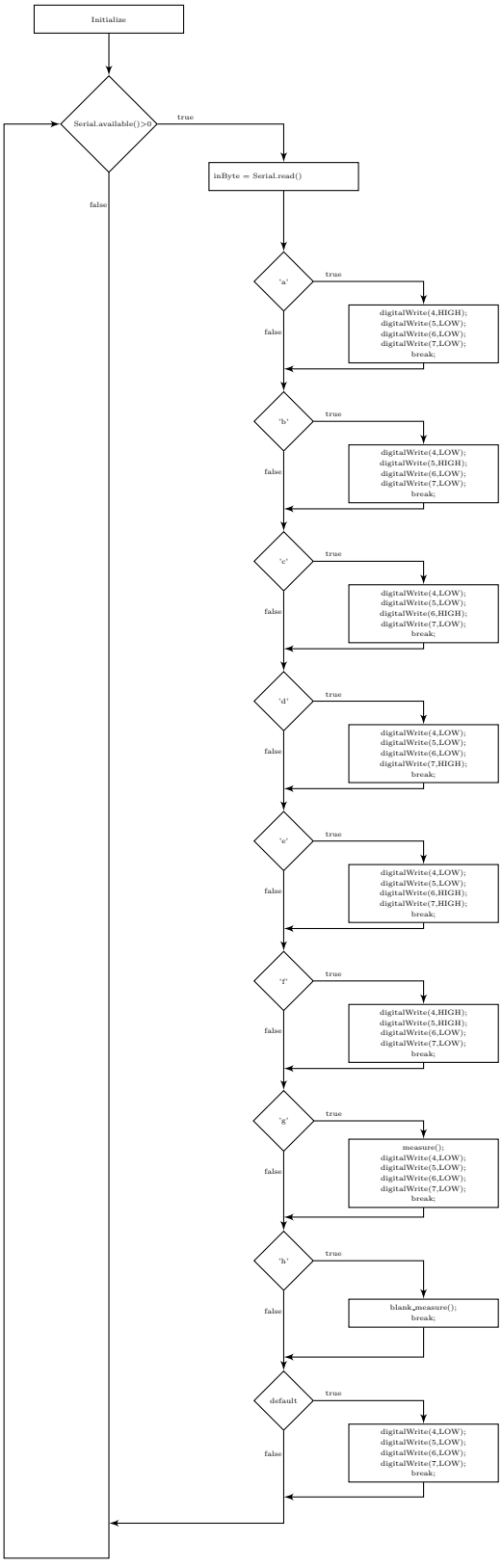


A.2 Electrochemical SPR system

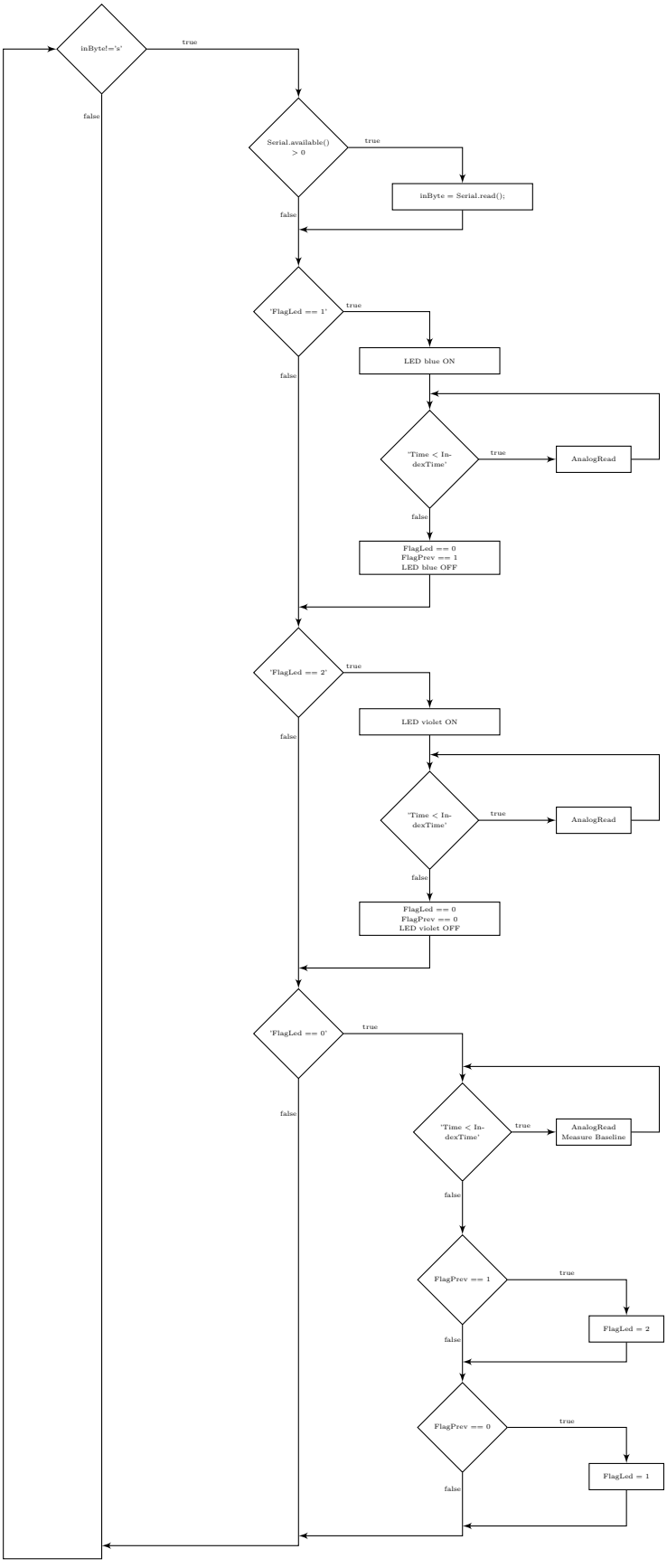


A.3 Antibiotic concentration sensor

A.3.1 Main



A.3.2 Measure()


inByte!='s'
true
false
Serial.available() > 0
true
inByte = Serial.read();
false
'FlagLed == 1'
true
LED blue ON
false
'Time < IndexTime'
true
AnalogRead
false
FlagLed == 0
FlagPrev == 1
LED blue OFF
'FlagLed == 2'
true
LED violet ON
false
'Time < IndexTime'
true
AnalogRead
false
FlagLed == 0
FlagPrev == 0
LED violet OFF
'FlagLed == 0'
true
false
'Time < IndexTime'
true
AnalogRead
Measure Baseline
false
FlagPrev == 1
true
FlagLed = 2
false
FlagPrev == 0
true
FlagLed = 1
false

Implemented code

B.1 Electrochemical SPR system

```
1
//-----------------------------------------------------------------------------
3 // Includes
//-----------------------------------------------------------------------------
5
#include <C8051F000.h>                  // SFR declarations
7 #include <stdio.h>

9 //-----------------------------------------------------------------------------
// User-defined types, structures, unions etc
11 //-----------------------------------------------------------------------------
#ifndef BYTE
13 #define BYTE unsigned char
#endif
15
//-----------------------------------------------------------------------------
17 // 16-bit SFR Definitions for 'F0xx
//-----------------------------------------------------------------------------
19
sfr16 RCAP2    = 0xCA;                  // Timer2 capture/reload
21 sfr16 TMR2     = 0xCC;                  // Timer2
sfr16 DAC0     = 0xd2;                  // DAC0 data
23 sfr16 ADC0     = 0xbe;                  // ADC0 data
sfr16 TMR3RL   = 0x92;                  // Timer3 reload value
25 sfr16 TMR3     = 0x94;                  // Timer3 counter

27 sbit SLAVE_SEL = P0^7; //NSS
sbit    GAIN1 = P1^3;           // GAIN channel 1 (1 --> lower gain)
29 sbit    GAIN2 = P1^7;           // GAIN channel 2 (1 --> lower gain)
sbit    GAIN3 = P2^3;           // GAIN channel 3 (1 --> lower gain)
31 sbit    GAIN4 = P2^7;           // GAIN channel 4 (1 --> lower gain)
//-----------------------------------------------------------------------------
33 // Global Constants
//-----------------------------------------------------------------------------
35
#define BAUDRATE     115200             // Baud rate of UART in bps
37
```

```
39 // SYSCLK = System clock frequency in Hz

41 #define SYSCLK        22118400L          // SYSCLK in Hz
#define SAMPLE_RATE  50000              // Sample frequency in Hz
43 #define INT_DEC       64                 // Integrate and decimate ratio
#define SAR_CLK       2500000            // Desired SAR clock speed
45
#define ANALOG_INPUTS 9                  // Number of AIN pins to measure
```

```
47                                          // (min=1, max=8)
    // SPI
49  #define  F_SCK_MAX          2400000      // Max SCK freq (Hz)
    #define  INCREMENT          1                    // Increment signal SPI
51
    //Timer 0
53  #define TIMER_PRESCALER        12  // Based on Timer CKCON settings

55  #define TIMER_RATE             8  // Interval time between interrupts (Timer0) in ms

57
    #define TIMER_TICKS_PER_MS  SYSCLK/TIMER_PRESCALER/1000
59
    // Note: TIMER_RATE*TIMER_TICKS_PER_MS should not exceed 65535 (0xFFFF)
61  // for the 16-bit timer

63  #define AUX1      TIMER_TICKS_PER_MS*TIMER_RATE
    #define AUX2      -AUX1
65  #define AUX3      AUX2&0x00FF
    #define AUX4      ((AUX2&0xFF00)>>8)
67
    #define TIMER0_RELOAD_HIGH        AUX4  // Reload value for Timer0 high byte
69  #define TIMER0_RELOAD_LOW         AUX3  // Reload value for Timer0 low byte

71  // Timer1
    #define TIMER1_RELOAD_HIGH  ((0xfbae&0xFF00)>>8)
73  // Reload value for Timer1 high byte (50 us)
    #define TIMER1_RELOAD_LOW      0xfbae&0x00FF
75  // Reload value for Timer1 low byte

77  //-----------------------------------------------------------------------------
    // Function Prototypes
79  //-----------------------------------------------------------------------------

81  void OSCILLATOR_Init (void);
    void DAC0_Init (void);
83  void PORT_Init (void);
    void UART0_Init (void);
85  void SPI_Init (void);
    void ADC0_Init (void);
87  void Timer0_Init(void);
    void Timer1_Init(void);
89  void TIMER3_Init (void);
    void ADC0_ISR (void);
91  void Fill_COUNTS_buffertx(unsigned Counts,unsigned short index);
    void Start_Measure(void);
93  void End_Measure(void);

95  void SPI_crono (unsigned int total);

97  void SPI_PWD (void);

99  void TX_data (void);

101 void Delay(void);
    //-----------------------------------------------------------------------------
103 // Global Variables
    //-----------------------------------------------------------------------------
105
    static char Byte;
107
    unsigned Result[ANALOG_INPUTS];      // ADC0 decimated value
109
    unsigned char IndexRX = 0;
111 unsigned char k;

113 unsigned long count;

115 xdata unsigned int IndexTimer = 0x0000;
    xdata unsigned int IndexTimer_bis = 0; //counts every IndexTimer overflow
117
    static char Byte;
119
    bit FlagRX = 0;
121 bit FlagTX = 0;
    bit FlagStop = 1;
123 bit FlagInterrupt = 0;
    bit FlagPWD = 0;
125 bit FlagChange = 0; //change chrono value
```

```
127 unsigned int count_tx = 0;

129
    //-----------------------------------------------------------------------------
131 // ADC Global Variables
    //-----------------------------------------------------------------------------
133 unsigned char amux_input = 0;          // index of analog MUX inputs
    unsigned char amux_convert = 0;
135 //-----------------------------------------------------------------------------
    // Uart0Commands Global Variables
137 //-----------------------------------------------------------------------------
    xdata volatile unsigned char buffertx[ 26 /*MAX_TX_SIZE*/];
139 xdata volatile unsigned char RX_Buffer[ 27 /*MAX_RX_SIZE*/];
    //-----------------------------------------------------------------------------
141 // DACs Global Variables
    //-----------------------------------------------------------------------------
143
    bit Flag_Rise_1 = 0;
145 bit Flag_Rise_2 = 0;
    bit Flag_Rise_3 = 0;
147 bit Flag_Rise_4 = 0;

149 unsigned int e_max_1 = 0;
    unsigned int e_min_1 = 0;
151 unsigned int e_max_2 = 0;
    unsigned int e_min_2 = 0;
153 unsigned int e_max_3 = 0;
    unsigned int e_min_3 = 0;
155 unsigned int e_max_4 = 0;
    unsigned int e_min_4 = 0;
157
    unsigned int dac1 = 0;
159 unsigned int dac2 = 0;
    unsigned int dac3 = 0;
161 unsigned int dac4 = 0;

163 xdata unsigned int IndexTimer1 = 0;
    xdata unsigned int IndexTimer2 = 0;
165 xdata unsigned int IndexTimer3 = 0;
    xdata unsigned int IndexTimer4 = 0;
167
    xdata unsigned int freq1 = 2000;
169 xdata unsigned int freq2 = 2000;
    xdata unsigned int freq3 = 2000;
171 xdata unsigned int freq4 = 2000;
    xdata unsigned int freq_square = 2000;
173
    unsigned char mode_of_op;  //decide for each channel if chronoamperometry or
175                                                                                  //
                                                                                       cyc

                                                                                       volt


    unsigned char type_of_wave; //decide type of wave
177 unsigned char change_chrono; //decide which channel change
    //-----------------------------------------------------------------------------
179 // main() Routine
    //-----------------------------------------------------------------------------
181
    void main (void)
183 {

185    WDTCN = 0xDE;        // Disable watchdog timer
       WDTCN = 0xAD;
187
       OSCILLATOR_Init (); // Initialize oscillator
189    PORT_Init ();        // Initialize crossbar and GPIO
       DAC0_Init ();
191
       UART0_Init ();       // Initialize UART0
193
       TIMER3_Init ();     // Initialize Timer3 to overflow at
195
       Timer0_Init();
197
       Timer1_Init();
199
       ADC0_Init ();       // Init ADC
```

```
201
    SPI_Init ();
203
    EA = 1;             // Enable global interrupts
205
    //ADCEN = 1;
207
    while (1)
209 {
      if (FlagInterrupt)
211   {
        TH0 = TIMER0_RELOAD_HIGH;            // Reinit Timer0 register
213     TL0 = TIMER0_RELOAD_LOW;
        TH1 = TIMER1_RELOAD_HIGH;          // Reinit Timer1 register
215     TL1 = TIMER1_RELOAD_LOW;

217     TF1 = 0;
        TF0 = 0;
219
        TMR3CN &= ~0x80;     // Flag Timer3 Interrupt =0
221
        ET0 = 1;       // Enable Timer0 interrupt
223     ET1 = 1;       // Enable Timer1 interrupt

225     EIE2  |= 0x01;       // Enable Timer3 Intterupt
      EIP2  |= 0X03;           // High priority ADC and Timer3
227
        EIE2 |= 0x02;                          // Enable ADC interrupt
229     ADCEN = 1;           // Enable ADC

231     FlagInterrupt = 0;
      FlagStop=0;
233     Delay ();
      }
235

237   if (!FlagStop)
      {
239           TX_data ();
      }
241

243 }
245
247 //-------------------------------------------------------------------------
    // Functions
    //-------------------------------------------------------------------------
249
    //-------------------------------------------------------------------------
251 // Start_Measure
    //-------------------------------------------------------------------------
253 //
    // This function sets  everything to start the measure
255 //
    //-------------------------------------------------------------------------
257
    void Start_Measure(void)
259 {

261             count_tx=0;

263             dac1 = e_min_1;  //Channel 1
                dac2 = e_min_2;  //Channel 2
265         dac3 = e_min_3;  //Channel 3
            dac4 = e_min_4;  //Channel 4
267
                if((mode_of_op & 0x01) != 0x01)
269             //if cyclic voltammetry, start at the positive value with ferricyanide
                {
271             dac1 = e_max_1;
                }
273
                if((mode_of_op & 0x02) != 0x02)
275             {
                dac2 = e_max_2;
277             }

279             if((mode_of_op & 0x04) != 0x04)
```

```
                    {
281                 dac3 = e_max_3;
                    }
283
                    if((mode_of_op & 0x08) != 0x08)
285                 {
                    dac4 = e_max_4;
287                 }

289                 Flag_Rise_1 = 0;  //start decreasing
                    Flag_Rise_2 = 0;
291                 Flag_Rise_3 = 0;
                    Flag_Rise_4 = 0;
293
                    IndexTimer1 = 0;
295                 IndexTimer2 = 0;
                    IndexTimer3 = 0;
297                 IndexTimer4 = 0;

299                 IndexTimer = 0;
                    IndexTimer_bis = 0;
301
                    ES = 0; //Disable UART interrupt
303                 IP  &= 0xef; //low priority UART

305                 amux_input = 0;

307                for(k=0; k<ANALOG_INPUTS; k++)
                   {
309                        Result[k] = 0;//Reset Result
                   }
311
                    SPI_crono (dac1);
313
                    SPI_crono (dac2);
315
                    SPI_crono (dac3);
317
                    SPI_crono (dac4);
319
                    if ((mode_of_op & 0x10) == 0x10)  //first channel ON
321                 {
                    P1 = (P1 & 0xf0) | 0x0b;  //open switch channel 1
323                 }

325                 if ((mode_of_op & 0x20) == 0x20)  //second channel ON
                    {
327                 P1 = (P1 & 0x0f) | 0xb0;
                    }
329
                    if ((mode_of_op & 0x40) == 0x40)  //third channel ON
331                 {
                    P2 = (P2 & 0xf0) | 0x0b;
333                 }

335                 if ((mode_of_op & 0x80) == 0x80)  //fourth channel ON
                    {
337                 P2 = (P2 & 0x0f) | 0xb0;
                    }
339
                    Delay ();
341
                    FlagInterrupt = 1;
343
    }
345
    //-------------------------------------------------------------------------
347 // Delay
    //-------------------------------------------------------------------------
349 //
    // Return Value : None
351 // Parameters   : None
    //
353 // Delay before starting the measurement
    //
355 //-------------------------------------------------------------------------
    void Delay (void)
357 {
       for (count = 50000; count > 0; count--);
```

```
359 }

361 //-------------------------------------------------------------------------------
    // End_Measure
363 //-------------------------------------------------------------------------------
    //
365 // This function sets  everything to start the measure
    //
367 //-------------------------------------------------------------------------------

369 void End_Measure(void)
    {
371         P1=0x44;          //Set the Switch
            P2=0x44;              //Set the switch
373 }

375 //-------------------------------------------------------------------------------
    // TX_data
377 //-------------------------------------------------------------------------------
    //
379 // This function transmits data
    //
381 //-------------------------------------------------------------------------------
    void TX_data (void)
383 {
            if (count_tx >= 25000)
385                 {
                            count_tx = 0;
387                         FlagTX=1;                       //Flag TX adc
                            buffertx[0]=0x50; //P
389                         buffertx[1]=0x51; //Q
                            buffertx[2]=0x52; //R
391                         buffertx[3]=0x53; //S
                            for(k=0;k<ANALOG_INPUTS; k++)
393                                 {
                                    Fill_COUNTS_buffertx(Result[k],k*2 + 4);
395                                 //Insert in the buffer TX the ADC value
                                    }
397
                                    Fill_COUNTS_buffertx(IndexTimer,22); //Timer
399                                 Fill_COUNTS_buffertx(IndexTimer_bis,24);

401                                 TI=1;
                    }
403         count_tx++;

405
            if(FlagTX)
407                 {
                    for(k=0;k<26;k++)
409                         {
                                    while ((SCON & 0x02) != 0x02) ;
411                                 // Check if transmit flag is set

413                                 SCON = (SCON & 0xFD);

415                                 SBUF=buffertx[k];
                            }
417                 FlagTX = 0;
                    }
419

421         if(RI)
            {
423           SCON = (SCON & 0xFE);      //RI1 = 0;
          Byte = SBUF;                  // Read a character
425
             //Change the gain
427
             if(Byte==0x70 && FlagRX==0)   //if TX starts with 'p'
              {
                     GAIN1 =1;
431           }

433           if(Byte==0x71 && FlagRX==0)   //if TX starts with 'q'
              {
435                   GAIN1 =0;
              }
437
```

```
            if(Byte==0x72 && FlagRX==0)   //if TX starts with 'r'
439         {
                GAIN2 =1;
441         }

443         if(Byte==0x73 && FlagRX==0)   //if TX starts with 's'
            {
445             GAIN2 =0;
            }
447

449         if(Byte==0x74 && FlagRX==0)   //if TX starts with 't'
            {
451             GAIN3 =1;
            }
453
            if(Byte==0x75 && FlagRX==0)   //if TX starts with 'u'
455         {
                GAIN3 =0;
457         }

459         if(Byte==0x76 && FlagRX==0)   //if TX starts with 'v'
            {
461             GAIN4 =1;
            }
463
            if(Byte==0x77 && FlagRX==0)   //if TX starts with 'w'
465         {
                GAIN4 =0;
467         }

469         if(Byte==0x6c && FlagRX==0)   //if TX starts with 'l'
            {
471             Flag_Rise_1 = !Flag_Rise_1;
            }
473
            if(Byte==0x6d && FlagRX==0)   //if TX starts with 'm'
475         {
                Flag_Rise_2 = !Flag_Rise_2;
477         }

479         if(Byte==0x6e && FlagRX==0)   //if TX starts with 'n'
            {
481             Flag_Rise_3 = !Flag_Rise_3;
            }
483
            if(Byte==0x6f && FlagRX==0)   //if TX starts with 'o'
485         {
                Flag_Rise_4 = !Flag_Rise_4;
487         }

489         if(Byte==0x65 && FlagRX==0)   //if TX starts with 'e'
            {
491             FlagStop=1;
                FlagTX = 0;
493             FlagPWD = 1;                    //DACs powerdownmode

495             //IP  |= 0x10;                  //high priority UART interrupt
            }
497
            if(FlagRX)
499         {
              RX_Buffer[IndexRX]=Byte;
501           IndexRX +=1;
            }
503
            if(Byte==0x43 && FlagRX==0)   //if TX starts with 'C'
505         {
                FlagRX=1;
507             IndexRX=0;
            }
509
            if(IndexRX >= 9)     //if 9 bytes have been received
511         {
                    FlagRX=0;
513                 IndexRX = 0;
                    change_chrono=RX_Buffer[0];
515
```

```
517                 if ((change_chrono & 0x01) == 0x01 && ((mode_of_op & 0x01) == 0x01))
                     //if potential has been chenaged and crono mode is active
519                         {
                                    e_max_1=RX_Buffer[1]<<8;
521                                 e_max_1 += RX_Buffer[2];
                                    e_max_1 &= 0x0fff;
523                                 e_max_1 |= 0x8000;
                             }
525
                            if ((change_chrono & 0x02) == 0x02 && ((mode_of_op & 0x02) == 0x02))
527                         {
                                    e_max_2=RX_Buffer[3]<<8;
529                                 e_max_2 += RX_Buffer[4];
                                    e_max_2 &= 0x0fff;
531                                 e_max_2 |= 0x9000;
                            }
533
                            if ((change_chrono & 0x04) == 0x04 && ((mode_of_op & 0x04) == 0x04))
535                          {
                                      e_max_3=RX_Buffer[5]<<8;
537                                   e_max_3 += RX_Buffer[6];
                                 e_max_3 &= 0x0fff;
539                                  e_max_3 |= 0xa000;
                            }
541
                            if ((change_chrono & 0x08) == 0x08 && ((mode_of_op & 0x08) == 0x08))
543                         {
                                     e_max_4=RX_Buffer[7]<<8;
545                                  e_max_4 += RX_Buffer[8];
                                     e_max_4 &= 0x0fff;
547                                  e_max_4 |= 0xb000;
                            }
549
                            FlagChange = 1;
551             }

553         }

555 }

557 //-------------------------------------------------------------------------
    // SPI_crono
559 //-------------------------------------------------------------------------
    //              This function write the SPI data for cronoamperometry (fixed potential)
561 //-------------------------------------------------------------------------
    void SPI_crono (unsigned int total)
563 {
        SLAVE_SEL = 0; //NSS
565
        SPI0DAT  = (BYTE)((total >> 8) & 0x00FF);
567     while (!SPIF);
        SPIF     = 0;
569
        SPI0DAT  = (BYTE)(total & 0x00FF);
571     while (!SPIF);
        SPIF     = 0;
573     SLAVE_SEL = 1;

575 }

577 //-------------------------------------------------------------------------
    // SPI_PWD
579 //-------------------------------------------------------------------------
    //
581 //
    //-------------------------------------------------------------------------
583 void SPI_PWD (void)
    {
585         SLAVE_SEL = 0;
      SPI0DAT = 0xf9;
587   while (!SPIF);
      SPIF     = 0;
589   SPI0DAT = 0x80;
      while (!SPIF);
591   SPIF     = 0;
      SLAVE_SEL = 1;
593
    }
595
```

```
//-----------------------------------------------------------------------------
// Fill_COUNTS_buffertx(
//-----------------------------------------------------------------------------
//
// This function fill the buffer TX
//
//-----------------------------------------------------------------------------

void Fill_COUNTS_buffertx(unsigned Counts,unsigned short index)
{

    buffertx[index] = (BYTE)((Counts >> 8) & 0x00FF); //MSB
    buffertx[index+1] = (BYTE)(Counts & 0x00FF);    //LSB

}

//-----------------------------------------------------------------------------
// Initialization Subroutines
//-----------------------------------------------------------------------------

//-----------------------------------------------------------------------------
// OSCILLATOR_Init
//-----------------------------------------------------------------------------
//
// Return Value : None
// Parameters   : None
//
// This function initializes the system clock to use an external 22.1184MHz
// crystal.
//
//-----------------------------------------------------------------------------
void OSCILLATOR_Init (void)
{
   int i;                              // Software timer

   OSCICN |= 0x80;                     // Enable the missing clock detector

   // Initialize external crystal oscillator to use 22.1184 MHz crystal

   OSCXCN = 0x67;                      // Enable external crystal osc.

   for (i = 0; i < 256; i++);          // Wait at least 1 ms
   while (!(OSCXCN & 0x80));           // Wait for crystal osc to settle

   OSCICN |= 0x08;                     // Select external clock source
   OSCICN &= ~0x04;                    // Disable the internal osc.
}

//-----------------------------------------------------------------------------
// PORT_Init
//-----------------------------------------------------------------------------
//
// Return Value : None
// Parameters   : None
//
// This function configures the crossbar and GPIO ports.
//
// Pinout:
//
// P0.0   digital   push-pull     UART TX
// P0.1   digital   open-drain    UART RX
//-----------------------------------------------------------------------------
void PORT_Init (void)
{
   XBR0     = 0x06;                    // Enable SPI and UART0
   XBR2     = 0x40;                    // Enable crossbar and weak pull-up

   PRT0CF |= 0x95;  // Set SCK,MOSI,Slaveselect and TX pin to push-pull
   PRT1CF |= 0xFF;      // enable P1 as a push-pull output
   PRT2CF |= 0xFF;      // enable P2 as a push-pull output

        P1=0x44;          //Set the Switch
        P2=0x44;              //Set the switch
}

//-----------------------------------------------------------------------------
// DAC0_Init
//-----------------------------------------------------------------------------
```

```
675 //
    // Return Value : None
677 // Parameters   : None
    //
679 // Enable the DAC and the VREF buffer.
    //
681 //-------------------------------------------------------------------------

683 void DAC0_Init(void)
    {
685    DAC0CN = 0x87;       // Enable DAC0 in left-justified mode

687    REF0CN |= 0x07;      // Enable the internal VREF (2.4v) and
                            // the Bias Generator
689
       DAC0 = 0xa000;  //1.5 volt    // Write to DAC0 (0xd550 = 1.9998 Volt)
691
    }
693
    //-------------------------------------------------------------------------
695 // UART0_Init   Variable baud rate, Timer 2, 8-N-1
    //-------------------------------------------------------------------------
697 //
    // Return Value : None
699 // Parameters   : None
    //
701 // Configure UART0 for operation at <baudrate> 8-N-1 using Timer2 as
    // baud rate source.
703 //
    //-------------------------------------------------------------------------
705 void UART0_Init (void)
    {
707    CKCON |= 0x20;        // Timer2 uses the system clock
       T2CON = 0x34;         // Timer2 used for TX and RX, enabled
709    RCAP2 = - ((long) (SYSCLK/BAUDRATE)/32);
       TMR2 = RCAP2;
711    TR2= 1;               // Start Timer2

713    SCON = 0x50;          // 8-bit variable baud rate;
                             // 9th bit ignored; RX enabled
715                          // clear all flags
       ES  = 1;    //interrupt enable
717    //IP  |= 0x10; //high priority
    }
719
    //-------------------------------------------------------------------------
721 // ADC0_Init
    //-------------------------------------------------------------------------
723 //
    // Configure ADC0 to use Timer3 overflows as conversion source, to
725 // generate an interrupt on conversion complete, and to use left-justified
    // output mode.  Enables ADC end of conversion interrupt. Leaves ADC disabled.
727 //
    //-------------------------------------------------------------------------
729
    void ADC0_Init (void)
731 {
       ADC0CN = 0x04;  // ADC0 disabled; normal tracking
733                    // mode; ADC0 conversions are initiated
                       // on overflow of Timer3; ADC0 data is
735                    // right-justified
       REF0CN = 0x07;  // enable temp sensor, on-chip VREF,
737                    // and VREF output buffer
       AMX0SL = 0x02;  // Select AIN0 as ADC mux output
739    ADC0CF = 0x86;  // ADC conversion clock = SYSCLK/16 && PGA gain = 0.5
       //ADC0CF &= ~0x07;
741
       //EIE2 |= 0x02; // enable ADC interrupts
743 }

745 //-------------------------------------------------------------------------
    // SPI_Init
747 //-------------------------------------------------------------------------
    //
749 //
    //-------------------------------------------------------------------------
751 void SPI_Init(void)
    {
753     SPI0CFG   = 0x47;   //clk idle high, phase 0, 8 bits data word
```

```
        SPI0CN    = 0x03;   //Master mode
755
            SPI0CKR     = (SYSCLK/(2*F_SCK_MAX))-1; //SPI frequency
757
            SLAVE_SEL = 1;
759
        EIE1      = 0x01;   //Enable SPI interrupt
761         //EIP1      = 0x01;   //High priority SPI interrupt
    }
763
    //-------------------------------------------------------------------------
765 // Timer0_Init
    //-------------------------------------------------------------------------
767 //
    // Return Value : None
769 // Parameters   : None
    //
771 //
    // Note: The Timer0 uses a 1:12 prescaler.  If this setting changes, the
773 // TIMER_PRESCALER constant must also be changed.
    //-------------------------------------------------------------------------
775 void Timer0_Init(void)
    {
777    TH0 = TIMER0_RELOAD_HIGH;           // Reinit Timer0 High register
       ET0 = 0;                            // Timer0 interrupt disabled
779    TMOD |= 0x01;                        // 16-bit Mode Timer0
       TCON |= 0x10;                        // Timer0 ON
781 }

783 //-------------------------------------------------------------------------
    // Timer1_Init
785 //-------------------------------------------------------------------------
    //
787 // Return Value : None
    // Parameters   : None
789 //
    //
791 // Note: The Timer1 uses a 1:12 prescaler.  If this setting changes, the
    // TIMER_PRESCALER constant must also be changed.
793 //-------------------------------------------------------------------------
    void Timer1_Init(void)
795 {
       TH1 = TIMER1_RELOAD_HIGH;           // Reinit Timer1 High register
797    ET1 = 0;                            // Timer1 interrupt disabled
       TMOD |= 0x10;                        // 16-bit Mode Timer1
799    CKCON |= 0x10;                                          // SYSCLK
       TCON |= 0x40;                        // Timer1 ON
801 }

803 //-------------------------------------------------------------------------
    // TIMER3_Init
805 //-------------------------------------------------------------------------
    //
807 // Return Value : None
    // Parameters   :
809 //   1)  int counts - calculated Timer overflow rate
    //                    range is postive range of integer: 0 to 32767
811 //
    // Configure Timer3 to auto-reload at interval specified by <counts> (no
813 // interrupt generated) using SYSCLK as its time base.
    //
815 //-------------------------------------------------------------------------

817 void Timer3_Init (void)
    {
819    TMR3CN = 0x02;                      // Stop Timer3; Clear TF3;
                                           // use SYSCLK as timebase
821    TMR3RL  = 65535 -(SYSCLK / 23255);  // reload at 43 us
       TMR3    = 0xffff;                   // set to reload immediately
823    //EIE2   |= 0x01;                     // enable Timer3 interrupts
       TMR3CN |= 0x04;                     // start Timer3
825 }

827
    //-------------------------------------------------------------------------
829 // Interrupt Service Routines
    //-------------------------------------------------------------------------
831
    //-------------------------------------------------------------------------
```

```
// Timer0_ISR
//-------------------------------------------------------------------------------
//
// Here we process the Timer0 interrupt
//
//-------------------------------------------------------------------------------
void Timer0_ISR (void) interrupt 1
{
   TH0 = TIMER0_RELOAD_HIGH; // Reinit Timer0 register
   TL0 = TIMER0_RELOAD_LOW;
   IndexTimer++;             // Increase the timer
   if (IndexTimer > 65533)
   {
    IndexTimer_bis++;
        IndexTimer = 0;
    }
}

//-------------------------------------------------------------------------------
// Timer0_ISR
//-------------------------------------------------------------------------------
//
// Here we process the Timer1 interrupt
//
//-------------------------------------------------------------------------------
void Timer1_ISR (void) interrupt 3
{
   TH1 = TIMER1_RELOAD_HIGH;  // Reinit Timer1 register
   TL1 = TIMER1_RELOAD_LOW;

   if((type_of_wave & 0x01)==0x01)  //if chrono or cyclic voltammetry
        {
                if((mode_of_op & 0x01) != 0x01) //Increase the counter only in cyclic
                     voltammetry
                {
         IndexTimer1++;                // Increase timer channel1
                }
                if((mode_of_op & 0x02) != 0x02)
                {
                 IndexTimer2++;      // Increase timer channel2
        }
                if((mode_of_op & 0x04) != 0x04)
                {
                 IndexTimer3++;      // Increase timer channel3
                }
                if((mode_of_op & 0x08) != 0x08)
                {
         IndexTimer4++;          // Increase timer channel4
                }

                if(freq1<IndexTimer1)
                {
                        SLAVE_SEL = 0; //NSS

                        if (Flag_Rise_1 && ((dac1+INCREMENT) > e_max_1)) //
                        {
                                Flag_Rise_1=0; //segnale cala
                        }

                        if (!Flag_Rise_1 && ((dac1-INCREMENT) < e_min_1)) //
                        {
                                Flag_Rise_1=1;
                        }

                        if (Flag_Rise_1)
                        {
                                dac1 += INCREMENT;
                        }

                        if (!Flag_Rise_1)
                        {
                                dac1 -= INCREMENT;
                        }

                        IndexTimer1 = 0;

                        SPI0DAT  = (BYTE)((dac1 >> 8) & 0x00FF);
                        while (!SPIF);
                        SPIF     = 0;
```

```
911
                        SPI0DAT  = (BYTE)(dac1 & 0x00FF);
913                     while (!SPIF);
                        SPIF     = 0;
915                     SLAVE_SEL = 1;
                }
917
                if(freq2<IndexTimer2)
919             {
                        SLAVE_SEL = 0; //NSS
921
                        if (Flag_Rise_2 && ((dac2+INCREMENT) > e_max_2)) //
923                     {
                                Flag_Rise_2=0; //segnale cala
925                     }

927                     if (!Flag_Rise_2 && ((dac2-INCREMENT) < e_min_2)) //
                        {
929                             Flag_Rise_2=1;
                        }
931
                        if (Flag_Rise_2)
933                     {
                                dac2 += INCREMENT;
935                     }

937                     if (!Flag_Rise_2)
                        {
939                             dac2 -= INCREMENT;
                        }
941
                        IndexTimer2 = 0;
943
                        SPI0DAT  = (BYTE)((dac2 >> 8) & 0x00FF);
945                     while (!SPIF);
                        SPIF     = 0;
947
                        SPI0DAT  = (BYTE)(dac2 & 0x00FF);
949                     while (!SPIF);
                        SPIF     = 0;
951                     SLAVE_SEL = 1;
                }
953
                if(freq3<IndexTimer3)
955             {
                        SLAVE_SEL = 0; //NSS
957
                        if (Flag_Rise_3 && ((dac3+INCREMENT) > e_max_3)) //
959                     {
                                Flag_Rise_3=0; //segnale cala
961                     }

963                     if (!Flag_Rise_3 && ((dac3-INCREMENT) < e_min_3)) //
                        {
965                             Flag_Rise_3=1;
                        }
967
                        if (Flag_Rise_3)
969                     {
                                dac3 += INCREMENT;
971                     }

973                     if (!Flag_Rise_3)
                        {
975                             dac3 -= INCREMENT;
                        }
977
                        IndexTimer3 = 0;
979
                        SPI0DAT  = (BYTE)((dac3 >> 8) & 0x00FF);
981                     while (!SPIF);
                        SPIF     = 0;
983
                        SPI0DAT  = (BYTE)(dac3 & 0x00FF);
985                     while (!SPIF);
                        SPIF     = 0;
987                     SLAVE_SEL = 1;
                }
989
```

```
                if(freq4<IndexTimer4)
991             {
                        SLAVE_SEL = 0; //NSS
993
                        if (Flag_Rise_4 && ((dac4+INCREMENT) > e_max_4)) //
995                     {
                                Flag_Rise_4=0; //segnale cala
997                     }

999                     if (!Flag_Rise_4 && ((dac4-INCREMENT) < e_min_4)) //
                        {
1001                            Flag_Rise_4=1;
                        }
1003
                        if (Flag_Rise_4)
1005                    {
                                dac4 += INCREMENT;
1007                    }

1009                    if (!Flag_Rise_4)
                        {
1011                            dac4 -= INCREMENT;
                        }
1013
                        IndexTimer4 = 0;
1015
                        SPI0DAT  = (BYTE)((dac4 >> 8) & 0x00FF);
1017                    while (!SPIF);
                        SPIF     = 0;
1019
                        SPI0DAT  = (BYTE)(dac4 & 0x00FF);
1021                    while (!SPIF);
                        SPIF     = 0;
1023                    SLAVE_SEL = 1;
                }
1025        }

1027        if((type_of_wave & 0x02)==0x02)  //if square
            {
1029            IndexTimer1++;
                IndexTimer2++;
1031            IndexTimer3++;
                IndexTimer4++;
1033
                if(freq1<IndexTimer1)
1035            {
                        SLAVE_SEL = 0; //NSS
1037
                        if (Flag_Rise_1) //
1039                    {
                                dac1 = e_min_1;
1041                    }

1043                    if (!Flag_Rise_1) //
                        {
1045                            dac1 = e_max_1;
                        }
1047                    Flag_Rise_1=!Flag_Rise_1;

1049                    IndexTimer1 = 0;

1051                    SPI0DAT  = (BYTE)((dac1 >> 8) & 0x00FF);
                        while (!SPIF);
1053                    SPIF     = 0;

1055                    SPI0DAT  = (BYTE)(dac1 & 0x00FF);
                        while (!SPIF);
1057                    SPIF     = 0;
                        SLAVE_SEL = 1;
1059            }

1061            if(freq2<IndexTimer2)
                {
1063                    SLAVE_SEL = 0; //NSS

1065                    if (Flag_Rise_2) //
                        {
1067                            dac2 = e_min_2;
                        }
```

```
1069
                        if (!Flag_Rise_2) //
1071                    {
                                dac2 = e_max_2;
1073                    }
                        Flag_Rise_2=!Flag_Rise_2;
1075
                        IndexTimer2 = 0;
1077
                        SPI0DAT  = (BYTE)((dac2 >> 8) & 0x00FF);
1079                    while (!SPIF);
                        SPIF     = 0;
1081
                        SPI0DAT  = (BYTE)(dac2 & 0x00FF);
1083                    while (!SPIF);
                        SPIF     = 0;
1085                    SLAVE_SEL = 1;
                }
1087
                if(freq3<IndexTimer3)
1089            {
                        SLAVE_SEL = 0; //NSS
1091
                        if (Flag_Rise_3) //
1093                    {
                                dac3 = e_min_3;
1095                    }

1097                    if (!Flag_Rise_3) //
                        {
1099                            dac3 = e_max_3;
                        }
1101                    Flag_Rise_3=!Flag_Rise_3;

1103                    IndexTimer3 = 0;

1105                    SPI0DAT  = (BYTE)((dac3 >> 8) & 0x00FF);
                        while (!SPIF);
1107                    SPIF     = 0;

1109                    SPI0DAT  = (BYTE)(dac3 & 0x00FF);
                        while (!SPIF);
1111                    SPIF     = 0;
                        SLAVE_SEL = 1;
1113            }

1115            if(freq4<IndexTimer4)
                {
1117                    SLAVE_SEL = 0; //NSS

1119                    if (Flag_Rise_4) //
                        {
1121                            dac4 = e_min_4;
                        }
1123
                        if (!Flag_Rise_4) //
1125                    {
                                dac4 = e_max_4;
1127                    }
                        Flag_Rise_4=!Flag_Rise_4;
1129
                        IndexTimer4 = 0;
1131
                        SPI0DAT  = (BYTE)((dac4 >> 8) & 0x00FF);
1133                    while (!SPIF);
                        SPIF     = 0;
1135
                        SPI0DAT  = (BYTE)(dac4 & 0x00FF);
1137                    while (!SPIF);
                        SPIF     = 0;
1139                    SLAVE_SEL = 1;
                }
1141
        }
1143
        if (FlagChange)
1145    {
            FlagChange = 0;
1147
```

```
                if (((change_chrono & 0x01) == 0x01) && ((mode_of_op & 0x01) == 0x01))  //
                     potential has been changed and crono mode
1149            {
                        SPI_crono (e_max_1);
1151            }

1153            if (((change_chrono & 0x02) == 0x02) && ((mode_of_op & 0x02) == 0x02))
                {
1155                    SPI_crono (e_max_2);
                }
1157
                if (((change_chrono & 0x04) == 0x04) && ((mode_of_op & 0x04) == 0x04))
1159            {
                        SPI_crono (e_max_3);
1161            }

1163            if (((change_chrono & 0x08) == 0x08) && ((mode_of_op & 0x08) == 0x08))
                {
1165                    SPI_crono (e_max_4);
                }
1167
        }
1169
        if (FlagPWD)
1171      {
        FlagPWD = 0;
1173      SPI_PWD ();

1175      ET0 = 0;                          // Disable Timer0 interrupt
        ET1 = 0;                          // Disable Timer1 interrupt
1177
        ADCEN = 0;              // disable ADC
1179      EIE2 &= 0xfd;               // disable adc interrupt

1181      EIE2 &= 0xfe;           // Disable Timer3 interrupt

1183      TI=0;
        ES = 1;                         // Enable UART interrupt
1185
        End_Measure ();         // Close all the switches
1187      }

1189  }

1191  //-------------------------------------------------------------------------
      // ADC0_ISR
1193  //-------------------------------------------------------------------------
      //
1195  // This ISR is called when the ADC0 completes a conversion.  Each value is
      // added to a running total <accumulator>, and the local decimation counter
1197  // <int_dec> decremented. When <int_dec> reaches zero, we post the decimated
      // result in the global variable <Result[]>.
1199  //
      // The analog input is sampled, held, and converted on a Timer3 overflow.  To
1201  // maximize input settling time, the analog mux is also advanced to the next
      // input on the Timer3 overflow.  Two different indices are held globally:
1203  //    amux_convert:  index of the analog input undergoing conversion
      //    amux_input:    index of the analog input selected in the analog
1205  //                   multiplexer
      //
1207  //
      //-------------------------------------------------------------------------
1209  void ADC0_ISR (void) interrupt 15
      {
1211
        static unsigned int_dec=INT_DEC;    // Integrate/decimate counter
1213                                        // we post a new result when
                                            // int_dec = 0
1215    unsigned char k;
        static long accumulator[ANALOG_INPUTS] ={0L};
1217                                        // Here's where we integrate the
                                            // ADC samples from input AIN0.0
1219
       ADCINT = 0;                          // clear ADC conversion complete
1221                                        // indicator

1223    accumulator[amux_convert] += ADC0;  // Read ADC value and add to running
                                            // total
1225
```

```
    if(amux_convert == (ANALOG_INPUTS-1))// Reset input index if the last input
1227                                     //was just read
    {
1229    int_dec--;                       // Update decimation counter
                                         // when last of the analog inputs
1231                                     // sampled
    }
1233
    if (int_dec == 0)                    // If zero, then post result
1235{
    int_dec = INT_DEC;                   // Reset counter
1237
    for(k=0; k<ANALOG_INPUTS; k++)
1239  {
      Result[k] = accumulator[k] >> 6; //Copy decimated values into Result
1241  accumulator[k] = 0L;               // Reset accumulators
    }
1243}

1245amux_convert = amux_input;           // Now that conversion results are
                                         // stored, advance index to the analog
1247                                     // input currently selected on the mux


1249
}
1251
//-----------------------------------------------------------------------------
1253// TIMER3_ISR
//-----------------------------------------------------------------------------
1255//
// The timer3 overflow triggers the ADC0 conversion on the analog MUX input
1257// previously selected.  It is permissable to change the analog MUX
// input once conversion has started, as the ADC has an internal sample
1259// and hold.
//
1261// This ISR routine will then select the next analog MUX input so as to
// maximize the settling time.
1263//
//-----------------------------------------------------------------------------
1265
void TIMER3_ISR(void) interrupt 14
1267{

1269TMR3CN &= ~0x80;                     // Acknowledge interrupt

1271amux_input ++;                      // Step to the next analog mux input

1273if(amux_input == ANALOG_INPUTS)     // Reset input index if the last input
    {                                    // was just read
1275   amux_input=0;                     // reset input index back to AIN0.0
    }
1277
   AMX0SL = amux_input;                  // Select the next input on the analog
1279                                     // multiplexer

1281}

1283//-----------------------------------------------------------------------------
// UART1_Interrupt
1285//-----------------------------------------------------------------------------
//
1287// This routine is invoked whenever a character is received
//
1289//-----------------------------------------------------------------------------

1291void UART0_Interrupt (void) interrupt 4
{
1293
        if ((SCON & 0x02) == 0x02)  //flag byte received
1295      {
          TI = 0;
1297      }


1299
    if ((SCON & 0x01) == 0x01)  //flag byte received
1301{

1303   SCON = (SCON & 0xFE);          //RI1 = 0;
       Byte = SBUF;                    // Read a character
```

```
1305
            if(FlagRX)
1307        {
            RX_Buffer[IndexRX]=Byte;
1309        IndexRX +=1;
            }
1311

1313        if(Byte==0x63 && FlagRX==0)   //if TX starts with 'c'
            {
1315            FlagRX=1;
                IndexRX=0;
1317        }

1319
            if(Byte==0x65 && FlagRX==0)   //if TX starts with 'e' stop measure
1321        {
                FlagStop=1;
1323            FlagPWD = 1;    //DACs powerdownmode
            }
1325
            if(IndexRX >= 26)     //if 25 bytes have been received
1327        {
            FlagRX=0;
1329        IndexRX = 0;
            e_max_1=RX_Buffer[0]<<8;
1331        e_max_1 += RX_Buffer[1];
            e_max_1 &= 0x0fff;
1333
            e_max_1 |= 0x8000;
1335
            e_min_1=RX_Buffer[2]<<8;
1337        e_min_1 += RX_Buffer[3];
            e_min_1 &= 0x0fff;
1339
            e_min_1 |= 0x8000;
1341
            e_max_2=RX_Buffer[4]<<8;
1343        e_max_2 += RX_Buffer[5];
            e_max_2 &= 0x0fff;
1345
            e_max_2 |= 0x9000;
1347
            e_min_2=RX_Buffer[6]<<8;
1349        e_min_2 += RX_Buffer[7];
            e_min_2 &= 0x0fff;
1351
            e_min_2 |= 0x9000;
1353
            e_max_3=RX_Buffer[8]<<8;
1355        e_max_3 += RX_Buffer[9];
            e_max_3 &= 0x0fff;
1357
            e_max_3 |= 0xa000;
1359
            e_min_3=RX_Buffer[10]<<8;
1361        e_min_3 += RX_Buffer[11];
            e_min_3 &= 0x0fff;
1363
            e_min_3 |= 0xa000;
1365
            e_max_4=RX_Buffer[12]<<8;
1367        e_max_4 += RX_Buffer[13];
            e_max_4 &= 0x0fff;
1369
            e_max_4 |= 0xb000;
1371
            e_min_4=RX_Buffer[14]<<8;
1373        e_min_4 += RX_Buffer[15];
            e_min_4 &= 0x0fff;
1375
            e_min_4 |= 0xb000;
1377
            freq1=RX_Buffer[16]<<8;
1379        freq1 += RX_Buffer[17];

1381        freq2=RX_Buffer[18]<<8;
            freq2 += RX_Buffer[19];
1383
```

```
            freq3=RX_Buffer[20]<<8;
1385        freq3 += RX_Buffer[21];

1387        freq4=RX_Buffer[22]<<8;
            freq4 += RX_Buffer[23];
1389
            mode_of_op = RX_Buffer[24];
1391
            type_of_wave = RX_Buffer[25];
1393
            }
1395
          if(Byte==0x64 && FlagRX==0)   //if TX starts with 'd' start meaure
1397        {
                Start_Measure ();
1399
            }
1401
      }
1403
  }
1405
  //-------------------------------------------------------------------------------
1407 // End Of File
  //-------------------------------------------------------------------------------
```

B.2 Glycerol concentration sensor

```
/*
LCD:
//8 bit: RS, RW, EN, D0, D1, D2, D3, D4, D5, D6, D7
//RS-->PC0
//RW-->PC1
//EN-->PC2

Micropump scheme
D1=4; PWM1=12; RED --> BUFFER 2
D2=5; PWM2=11; RED --> BUFFER 1
D3=6; PWM3=3; BLUE --> NAD
D4=7; PWM4=2; GREEN --> MOSTO
*/

#include <LiquidCrystal.h>
#include <SoftwareSerial.h>
#include <Thermal.h>
#include<math.h>

int printer_RX_Pin = 22;  //TX della stampante (verde)
int printer_TX_Pin = 24;  //RX pin della stampante (giallo)

int freq_tone=100*4; //frequenza tone()

int day=1;
int month = 1;
int year = 20;
int year1 = 12;

int carro=0;
int carro_lb=0;
int carro_hb=0;

int hour = 0;
int minute = 10;
int seconds = 0;

int cantina=1;

int glic_int = 0;
int glic_dec1 = 0;
int glic_dec2 = 0;
int glic_dec3 = 0;

String shour = String(hour);
String sminute = String(minute);
String sseconds = String(seconds);
String sday = String(day);
String smonth = String(month);
String syear = String(year);
String syear1 = String(year1);
String scarro = String(carro);
String sglic_int = String(glic_int);
String sglic_dec1 = String(glic_dec1);
String sglic_dec2 = String(glic_dec2);
String sglic_dec3 = String(glic_dec3);

String string_ora;
char char_ora[34];

String string_numero;
char char_num[15];

String string_conc_glic;
char char_glic[21];

int time_set = 90;  //dichiaro il tempo di accensione,poi lo cambio tramite seriale
int v_1=0;  //Buffer 2
int v_2=60;  //Buffer 1
int v_3=60;  //MOsto
int v_4=65;  //NAD
int val; //analog read (sense sull'alimentazione)
int time_fill=13000;
int k=0;  //sense per l'alimentazione
int l=0;
```

```
int inByte1;

Thermal printer(printer_RX_Pin, printer_TX_Pin);

LiquidCrystal lcd(54, 55, 56, 50, 26, 48, 28, 46, 30, 44, 32);
/*inizializzo LCD*/

void setup() {
  // initialize serial communication:
  Serial.begin(9600);
  lcd.begin(8, 2);
  //cambio i timer 1 (controlla pin 12,11) e 2 (controlla pin 10,9)
  //http://arduino.cc/forum/index.php?topic=72092.0
  int myEraser = 7;      // this is 111 in binary and is used as an eraser

  TCCR1B &= ~myEraser;   // this operation (AND plus NOT),  set the three bits in TCCR2B to 0
  TCCR3B &= ~myEraser;

  int myPrescaler = 1;          // this could be a number in [1 , 6].
  TCCR1B |= myPrescaler;  //this operation (OR), replaces the last three bits in TCCR2B with our
       new value 011
  TCCR3B |= myPrescaler;  //freq PWM diventa 31 kHz

  // inizializzo i pin di controllo
   pinMode(7, OUTPUT);
   pinMode(6, OUTPUT);
   pinMode(5, OUTPUT);
   pinMode(4, OUTPUT);

   //Inizializzo LCD
   lcd.clear();
   lcd.setCursor(0,0);
   lcd.print(" Cevico ");
   lcd.setCursor(0,1);
   lcd.print(" Group ");

   delay(500);

   start_up();
   }

void start_up(){
  analogWrite(12,0);
  analogWrite(11,0);
  analogWrite(3,0);
  analogWrite(2,0);
  digitalWrite(7,HIGH);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" Mosto ");
  delay(500);
  digitalWrite(7,LOW);
  digitalWrite(6,HIGH);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("  NAD   ");
  delay(500);
  digitalWrite(6,LOW);
  digitalWrite(5,HIGH);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Buffer 1");
  delay(500);
  digitalWrite(5,LOW);
  digitalWrite(4,HIGH);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Buffer 2");
  delay(500);
  digitalWrite(4,LOW);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" Cevico ");
  lcd.setCursor(0,1);
  lcd.print(" Group ");

}

void loop() {
```

```
    int i;
156 int inByte;

158 String stringOne;
    String stringTotal;
160 unsigned long time_pump;
    unsigned long time_total;
162 unsigned long time_remaining;

164   val=analogRead(4);
    // Serial.println(val);
166   //controllo che alimentazione sia collegata
  if(val>100 && k==0){
168     lcd.clear();
      lcd.setCursor(0,0);
170   lcd.print(" Cevico ");
      lcd.setCursor(0,1);
172   lcd.print(" Group ");
      k=1;
174   l=0;}
  if(val<101 && l==0){
176     lcd.clear();
      lcd.setCursor(0,0);
178   lcd.print("Connect");
      lcd.setCursor(0,1);
180   lcd.print(" Power! ");
    k=0;
182   l=1;}

184
  if (Serial.available() >0) {
186   inByte = Serial.read();

188     //seleziono la cantina
      if (inByte == 'C'){
190     inByte1=Serial.read();
      while (inByte1<0){
192     inByte1=Serial.read();
      }
194     cantina=inByte1;
        }
196
    //imposto numero carro HB
198     if (inByte == 'A'){
        inByte1=Serial.read();
200   while (inByte1<0){
        inByte1=Serial.read();
202   }
        carro_hb=inByte1;
204     }

206   //imposto numero carro LB
      if (inByte == 'B'){
208     inByte1=Serial.read();
      while (inByte1<0){
210     inByte1=Serial.read();
      }
212     carro_lb=inByte1;
        carro=carro_lb+carro_hb*256;
214     scarro=String(carro);
      }
216 //trasmetto concentrazione glicerolo
    //numero intero
218     if (inByte == 'I'){
        inByte1=Serial.read();
220   while (inByte1<0){
        inByte1=Serial.read();
222   }
        glic_int=inByte1;
224     sglic_int=String(glic_int);
      }
226
    //1° cifra decimale
228     if (inByte == 'J'){
        inByte1=Serial.read();
230   while (inByte1<0){
        inByte1=Serial.read();
232   }
        glic_dec1=inByte1;
```

```
234           sglic_dec1=String(glic_dec1);
          }
236
      //2° cifra decimale
238       if (inByte == 'K'){
            inByte1=Serial.read();
240       while (inByte1<0){
            inByte1=Serial.read();
242       }
            glic_dec2=inByte1;
244         sglic_dec2=String(glic_dec2);
          }
246
      //3° cifra decimale
248       if (inByte == 'L'){
            inByte1=Serial.read();
250       while (inByte1<0){
            inByte1=Serial.read();
252       }
            glic_dec3=inByte1;
254         sglic_dec3=String(glic_dec3);
          }
256
      //imposto giorno
258       if (inByte == 'D'){
            inByte1=Serial.read();
260       while (inByte1<0){
            inByte1=Serial.read();
262       }
            day=inByte1;
264         sday=String(day);
          }
266
          //imposto mese
268       if (inByte == 'M'){
            inByte1=Serial.read();
270       while (inByte1<0){
            inByte1=Serial.read();
272       }
          month=inByte1;
274       smonth=String(month);
          }
276
          //imposto anno millennio e centenario
278       if (inByte == 'Y'){
            inByte1=Serial.read();
280       while (inByte1<0){
            inByte1=Serial.read();
282       }
          year=inByte1;
284       syear=String(year);
          }
286
                //imposto anno decina e unità
288       if (inByte == 'W'){
            inByte1=Serial.read();
290       while (inByte1<0){
            inByte1=Serial.read();
292       }
          year1=inByte1;
294       syear1=String(year1);
          }
296
          //imposto ora
298       if (inByte == 'H'){
            inByte1=Serial.read();
300       while (inByte1<0){
            inByte1=Serial.read();
302       }
          hour=inByte1;
304       shour=String(hour);
          }
306
          //imposto minuti
308       if (inByte == 'N'){
            inByte1=Serial.read();
310       while (inByte1<0){
            inByte1=Serial.read();
312       }
```

```
        minute=inByte1;
314     sminute=String(minute);
        }
316
              //imposto secondi
318     if (inByte == 'S'){
          inByte1=Serial.read();
320     while (inByte1<0){
          inByte1=Serial.read();
322     }
        seconds=inByte1;
324     sseconds=String(seconds);
        }
326   //imposto freq mosto

328    if (inByte == 'q'){
       //Serial.println(inByte);
330       inByte1=Serial.read();
        while (inByte1<0){
332       inByte1=Serial.read();
        }
334     freq_tone=inByte1*4;
        lcd.clear();
336     lcd.setCursor(0,0);
        lcd.print(" Freq. ");
338     lcd.setCursor(0,1);
        lcd.print(freq_tone/4);
340     //lcd.setCursor(1,1);
        //lcd.write('i');
342     lcd.setCursor(4,1);
        lcd.write('H');
344     lcd.setCursor(5,1);
        lcd.write('z');
346     //lcd.setCursor(3,1);
        //lcd.write('e');
348
        digitalWrite(7, HIGH);
350     delay(300);
        digitalWrite(7, LOW);
352     delay(300);
        digitalWrite(7, HIGH);
354     delay(300);
        digitalWrite(7, LOW);
356     lcd.clear();
        digitalWrite(7, HIGH);
358     delay(300);
        digitalWrite(7, LOW);
360     lcd.clear();
        lcd.setCursor(0,0);
362     lcd.print(" Cevico ");
        lcd.setCursor(0,1);
364     lcd.print(" Group ");

366   }

368   //imposto time iniezione NAD

370   if (inByte == 't'){
        //Serial.println(inByte);
372       inByte1=Serial.read();
        while (inByte1<0){
374       inByte1=Serial.read();
        }
376     time_set=inByte1;
        lcd.clear();
378     lcd.setCursor(0,0);
        lcd.print("  NAD  ");
380     lcd.setCursor(0,1);
        lcd.write('T');
382     //lcd.setCursor(1,1);
        //lcd.write('i');
384     lcd.setCursor(2,1);
        lcd.write('=');
386     //lcd.setCursor(3,1);
        //lcd.write('e');
388     lcd.setCursor(4,1);
        lcd.print(time_set*50);
390     digitalWrite(6, HIGH);
        delay(300);
```

```
392         digitalWrite(6, LOW);
            delay(300);
394         digitalWrite(6, HIGH);
            delay(300);
396         digitalWrite(6, LOW);
            lcd.clear();
398         lcd.setCursor(0,0);
            lcd.print(" Cevico ");
400         lcd.setCursor(0,1);
            lcd.print(" Group ");
402
        }
404
       // memorizzo dati velocitÃ  1 Buffer
406
          if (inByte == 'v'){  // velocitÃ  1
408          inByte1=Serial.read();
              while (inByte1<0){
410           inByte1=Serial.read();
            }
412         v_1=inByte1;
            lcd.clear();
414         lcd.setCursor(0,0);
            lcd.print("Buffer 2");
416         lcd.setCursor(0,1);
            lcd.write('V');
418         lcd.setCursor(1,1);
            lcd.write('e');
420         lcd.setCursor(2,1);
            lcd.write('l');
422         lcd.setCursor(3,1);
            lcd.write('.');
424         lcd.setCursor(5,1);
            lcd.print(v_1);
426         digitalWrite(4, HIGH);
            delay(300);
428         digitalWrite(4, LOW);
            delay(300);
430         digitalWrite(4, HIGH);
            delay(300);
432         digitalWrite(4, LOW);
            lcd.clear();
434         lcd.setCursor(0,0);
            lcd.print(" Cevico ");
436         lcd.setCursor(0,1);
            lcd.print(" Group ");
438
        }
440
       // memorizzo dati velocitÃ  2 Buffer
442
        if (inByte == 'w'){ // velocitÃ  2
444           inByte1=Serial.read();
              while (inByte1<0){
446           inByte1=Serial.read();
            }
448         v_2=inByte1;
            //Serial.println(v_2);
450         lcd.clear();
            lcd.setCursor(0,0);
452         lcd.print("Buffer 1");
            lcd.setCursor(0,1);
454         lcd.write('V');
            lcd.setCursor(1,1);
456         lcd.write('e');
            lcd.setCursor(2,1);
458         lcd.write('l');
            lcd.setCursor(3,1);
460         lcd.write('.');
            lcd.setCursor(5,1);
462         lcd.print(v_2);
            digitalWrite(5, HIGH);
464         delay(300);
            digitalWrite(5, LOW);
466         delay(300);
            digitalWrite(5, HIGH);
468         delay(300);
            digitalWrite(5, LOW);
470         lcd.clear();
```

```
        lcd.setCursor(0,0);
472     lcd.print(" Cevico ");
        lcd.setCursor(0,1);
474     lcd.print(" Group ");

476   }

478  // memorizzo dati velocitÃ  3 mstoD

480   if (inByte == 'y'){  // velocitÃ 3
        inByte1=Serial.read();
482     while (inByte1<0){
          inByte1=Serial.read();
484     }
        v_3=inByte1;
486     //Serial.println(v_3);
        lcd.clear();
488     lcd.setCursor(0,0);
        lcd.print(" Mosto");
490     lcd.setCursor(0,1);
        lcd.write('V');
492     lcd.setCursor(1,1);
        lcd.write('e');
494     lcd.setCursor(2,1);
        lcd.write('l');
496     lcd.setCursor(3,1);
        lcd.write('.');
498     lcd.setCursor(5,1);
        lcd.print(v_3);
500     digitalWrite(7, HIGH);
        delay(300);
502     digitalWrite(7, LOW);
        delay(300);
504     digitalWrite(7, HIGH);
        delay(300);
506     digitalWrite(7, LOW);
        lcd.clear();
508     lcd.setCursor(0,0);
        lcd.print(" Cevico ");
510     lcd.setCursor(0,1);
        lcd.print(" Group ");
512
    }
514
  // memorizzo dati velocitÃ  4 NAD
516
      if (inByte == 'z'){ // velocitÃ 4
518       inByte1=Serial.read();
        while (inByte1<0){
520       inByte1=Serial.read();
        }
522     v_4=inByte1;
        //Serial.println(v_4);
524     lcd.clear();
        lcd.setCursor(0,0);
526     lcd.print("  NAD");
        lcd.setCursor(0,1);
528     lcd.write('V');
        lcd.setCursor(1,1);
530     lcd.write('e');
        lcd.setCursor(2,1);
532     lcd.write('l');
        lcd.setCursor(3,1);
534     lcd.write('.');
        lcd.setCursor(5,1);
536     lcd.print(v_4);
        digitalWrite(6, HIGH);
538     delay(300);
        digitalWrite(6, LOW);
540     delay(300);
        digitalWrite(6, HIGH);
542     delay(300);
        digitalWrite(6, LOW);
544     lcd.clear();
        lcd.setCursor(0,0);
546     lcd.print(" Cevico ");
        lcd.setCursor(0,1);
548     lcd.print(" Group ");
```

```
550   }

552    switch (inByte) {
       case 'a': //NAD
554      time_pump=millis();
         time_total=time_pump+time_set*50;
556      //Serial.println(time_total,'DEC');
         lcd.clear();
558      lcd.setCursor(0,0);
         lcd.print("  NAD   ");
560      while (millis()<time_total){
         digitalWrite(6, HIGH);
562      analogWrite(3,v_4);
         lcd.setCursor(0,1);
564      lcd.write('T');
         lcd.setCursor(2,1);
566      lcd.write('=');
         time_remaining = (time_total-millis());
568      if (time_remaining<1000){  //sposto di una posizione a destra se va sotto il 100
           lcd.setCursor(4,1);
570        lcd.write('0');
           lcd.setCursor(5,1);}
572        else{
             lcd.setCursor(4,1);}
574      lcd.print(time_remaining);
         }
576      digitalWrite(6, LOW);
         analogWrite(3,0);
578      lcd.clear();
         lcd.setCursor(0,0);
580      lcd.print(" Cevico ");
         lcd.setCursor(0,1);
582      lcd.print(" Group ");
         break;
584
       case 'b':  //spengo NAD
586      digitalWrite(6, LOW);
         analogWrite(3,0);
588      break;

590    case 'c':    //mosto
         digitalWrite(7, HIGH);
592      tone(9,freq_tone);
         analogWrite(2,v_3);
594      break;

596    case 'd':  //spengo mosto
         digitalWrite(7, LOW);
598      noTone(9);
         analogWrite(2,0);
600      break;

602    case 'e':    //buffer 1
         digitalWrite(5, HIGH);
604      analogWrite(11,v_2);
         break;
606
       case 'f':  //spengo buffer 1
608      digitalWrite(5, LOW);
         analogWrite(11,0);
610      break;

612    case 'g':    //buffer 2
         digitalWrite(4, HIGH);
614      analogWrite(12,v_1);
         break;
616
       case 'h':  //spengo buffer 2
618      digitalWrite(4, LOW);
         analogWrite(12,0);
620      break;

622    case 'i':  //riepilogo
       stringOne = "S" ;
624    stringTotal = stringOne + 'V' + v_1 + 'W' + v_2 + 'Y' + freq_tone/4 + 'Z' + v_4 + 'X' +
           time_set + 'T' + val + 'U';
       //Serial.println(v_1);
626    Serial.println(stringTotal);
         //buffer 1
```

```
628         lcd.clear();
            lcd.setCursor(0,0);
630         lcd.print("Buffer 1");
            lcd.setCursor(0,1);
632         lcd.write('V');
            lcd.setCursor(1,1);
634         lcd.write('e');
            lcd.setCursor(2,1);
636         lcd.write('l');
            lcd.setCursor(3,1);
638         lcd.write('.');
            lcd.setCursor(5,1);
640         lcd.print(v_2);
            digitalWrite(5, HIGH);
642         delay(500);
            digitalWrite(5, LOW);
644         delay(500);
            digitalWrite(5, HIGH);
646         delay(500);
            digitalWrite(5, LOW);
648
            //buffer 2
650         lcd.clear();
            lcd.setCursor(0,0);
652         lcd.print("Buffer 2");
            lcd.setCursor(0,1);
654         lcd.write('V');
            lcd.setCursor(1,1);
656         lcd.write('e');
            lcd.setCursor(2,1);
658         lcd.write('l');
            lcd.setCursor(3,1);
660         lcd.write('.');
            lcd.setCursor(5,1);
662         lcd.print(v_1);
            digitalWrite(4, HIGH);
664         delay(500);
            digitalWrite(4, LOW);
666         delay(500);
            digitalWrite(4, HIGH);
668         delay(500);
            digitalWrite(4, LOW);
670
            //mosto
672         lcd.clear();
            lcd.setCursor(0,0);
674         lcd.print(" Mosto");
            lcd.setCursor(0,1);
676         lcd.write('F');
            lcd.setCursor(1,1);
678         lcd.write('r');
            lcd.setCursor(2,1);
680         lcd.write('e');
            lcd.setCursor(3,1);
682         lcd.write('q');
            lcd.setCursor(5,1);
684         lcd.print(freq_tone/4);
            digitalWrite(7, HIGH);
686         delay(500);
            digitalWrite(7, LOW);
688         delay(500);
            digitalWrite(7, HIGH);
690         delay(500);
            digitalWrite(7, LOW);
692
            //NAD
694         lcd.clear();
            lcd.setCursor(0,0);
696         lcd.print("  NAD");
            lcd.setCursor(0,1);
698         lcd.write('V');
            lcd.setCursor(1,1);
700         lcd.write('e');
            lcd.setCursor(2,1);
702         lcd.write('l');
            lcd.setCursor(3,1);
704         lcd.write('.');
            lcd.setCursor(5,1);
706         lcd.print(v_4);
```

```
        digitalWrite(6, HIGH);
708     delay(500);
        digitalWrite(6, LOW);
710     delay(500);
        digitalWrite(6, HIGH);
712     delay(500);
        digitalWrite(6, LOW);
714
        //NAD tempo
716     lcd.clear();
        lcd.setCursor(0,0);
718     lcd.print("  NAD  ");
        lcd.setCursor(0,1);
720     lcd.write('T');
        lcd.setCursor(2,1);
722     lcd.write('=');
        lcd.setCursor(4,1);
724     lcd.print(time_set*50);
        digitalWrite(6, HIGH);
726     delay(500);
        digitalWrite(6, LOW);
728     delay(500);
        digitalWrite(6, HIGH);
730     delay(500);
        digitalWrite(6, LOW);
732     lcd.clear();
        lcd.setCursor(0,0);
734     lcd.print(" Cevico ");
        lcd.setCursor(0,1);
736     lcd.print(" Group ");

738     break;

740     case 'l':
        Serial.println(val);
742
        break;
744
        case 'm': //riempio il circuito
746
        //NAD
748
        time_pump=millis();
750     time_total=time_pump+time_fill +6000;
        //Serial.println(time_total,'DEC');
752     lcd.clear();
        lcd.setCursor(0,0);
754     lcd.print("  NAD   ");
        while (millis()<time_total){
756     digitalWrite(6, HIGH);
        analogWrite(3,60);
758     lcd.setCursor(0,1);
        lcd.write('T');
760     lcd.setCursor(2,1);
        lcd.write('=');
762     time_remaining = (time_total-millis())/1000;
        if (time_remaining<10){  //sposto di una posizione a destra se va sotto il 100
764       lcd.setCursor(4,1);
          lcd.write('0');
766       lcd.setCursor(5,1);}
          else{
768       lcd.setCursor(4,1);}
        lcd.print(time_remaining);
770      }
        digitalWrite(6, LOW);
772     analogWrite(3,0);

774
        //riempio buffer 1
776
        lcd.clear();
778     lcd.setCursor(0,0);
        lcd.print("Buffer 1");
780
        time_pump=millis();
782     time_total=time_pump+time_fill;

784     //buffer 1
        digitalWrite(5, HIGH);
```

```
786         analogWrite(11,60);

788         while (millis()<time_total){
            lcd.setCursor(0,1);
790         lcd.write('T');
            lcd.setCursor(2,1);
792         lcd.write('=');
            time_remaining = (time_total-millis())/1000;
794         if (time_remaining<10){  //sposto di una posizione a destra se va sotto il 100
              lcd.setCursor(4,1);
796           lcd.write('0');
              lcd.setCursor(5,1);}
798           else{
              lcd.setCursor(4,1);}
800         lcd.print(time_remaining);
            }
802
            //buffer 1
804         digitalWrite(5, LOW);
            analogWrite(11,0);
806 /*
            //riempio buffer 2
808         time_pump=millis();
            time_total=time_pump+time_fill;
810
            lcd.clear();
812         lcd.setCursor(0,0);
            lcd.print("Buffer 2");
814
            //buffer 2
816         digitalWrite(4, HIGH);
            analogWrite(12,60);
818
            while (millis()<time_total){
820         lcd.setCursor(0,1);
            lcd.write('T');
822         lcd.setCursor(2,1);
            lcd.write('=');
824         time_remaining = (time_total-millis())/1000;
            if (time_remaining<10){  //sposto di una posizione a destra se va sotto il 100
826           lcd.setCursor(4,1);
              lcd.write('0');
828           lcd.setCursor(5,1);}
              else{
830           lcd.setCursor(4,1);}
            lcd.print(time_remaining);
832         }

834         //buffer 2
            digitalWrite(4, LOW);
836         analogWrite(12,0);
 */
838         //riempio mosto
            time_pump=millis();
840         time_total=time_pump+time_fill;

842         lcd.clear();
            lcd.setCursor(0,0);
844         lcd.print(" Mosto");

846         //mosto
            digitalWrite(7, HIGH);
848         tone(9,freq_tone);
            analogWrite(2,60);
850
            while (millis()<time_total){
852         lcd.setCursor(0,1);
            lcd.write('T');
854         lcd.setCursor(2,1);
            lcd.write('=');
856         time_remaining = (time_total-millis())/1000;
            if (time_remaining<10){  //sposto di una posizione a destra se va sotto il 100
858           lcd.setCursor(4,1);
              lcd.write('0');
860           lcd.setCursor(5,1);}
              else{
862           lcd.setCursor(4,1);}
            lcd.print(time_remaining);
864         }
```

```
866         //mosto
            digitalWrite(7, LOW);
868         noTone(9);
            analogWrite(2,0);
870
            lcd.clear();
872         lcd.setCursor(0,0);
            lcd.print(" Cevico ");
874         lcd.setCursor(0,1);
            lcd.print(" Group ");
876
            break;
878
            case 'p':
880
          printer.setSize('S'); // set type size, accepts 'S', 'M', 'L'
882       printer.println("--------------------------------"); //print line
          printer.setSize('S'); // set type size, accepts 'S', 'M', 'L'
884       printer.justify('C');
          printer.println("Gruppo Cevico");
886       printer.println("Vini Romagnoli");
         //printer.printBitmap(243, 126, cevico5);
888     //  printer.printBitmap(155, 155, cevicoqr);

890       printer.setSize('S'); // set type size, accepts 'S', 'M', 'L'
          printer.print("--------------------------------"); //print line
892
          printer.setSize('S'); // set type size, accepts 'S', 'M', 'L'
894       printer.justify('R');
          printer.boldOn(); // Turn bold on
896       printer.inverseOn(); //set to print white on black

898       string_ora=String("Ora & Data:"+shour+"."+sminute+"."+sseconds+" - "+sday+"/"+smonth+"/"+
              syear+syear1);
          string_ora.toCharArray(char_ora,34);
900
          printer.println(char_ora); //print line
902       string_numero=String("Numero: "+scarro);
          string_numero.toCharArray(char_num,15);
904       printer.feed(); //advance one line
          printer.println(char_num); //print line
906       printer.feed(); //advance one line
          printer.inverseOff(); //set to print black on white (default)
908
          printer.setSize('M'); // set type size, accepts 'S', 'M', 'L'
910       printer.justify('C');

912       string_conc_glic=String("Glicerolo: "+sglic_int+"."+sglic_dec1+sglic_dec2+sglic_dec3+" g/l")
              ;
          string_conc_glic.toCharArray(char_glic,21);
914
          //Serial.print(char_glic);
916
          printer.println(char_glic); //print line
918       printer.feed(); //advance one line
          //printer.setSize('L'); // set type size, accepts 'S', 'M', 'L'
920       //printer.justify('C');
          //printer.println("0.92 g/l"); //print line
922
          printer.setSize('S'); // set type size, accepts 'S', 'M', 'L'
924       printer.print("--------------------------------"); //print line

926       printer.justify('C');
          if(cantina){
928         printer.println("Cantina di Bagnacavallo"); //print line}
        }
930       else if(cantina==0){
            printer.println("Cantina di Imola");
932       }
          printer.print("--------------------------------"); //print line
934
          printer.boldOff(); //Rurn bold
936
          printer.feed(); //advance one line
938       printer.feed(); //advance one line
          printer.feed(); //advance one line
940       break;
```

```
942       default:
            // turn all the LEDs off:
944         digitalWrite(4, LOW);
            digitalWrite(5, LOW);
946         digitalWrite(6, LOW);
            digitalWrite(7, LOW);
948         analogWrite(12,0);
            analogWrite(11,0);
950         analogWrite(3,0);
            analogWrite(2,0);
952         noTone(9);
            break;
954          }
      }
956 }
```

B.3 Antibiotic concentration sensor

```
1
  // digital 6 = 365 nm XSL-365-e Roithner
3  // digital 7 = 365 nm XSL-365-e Roithner

5 #include<math.h>

7   int inByte;
    unsigned long time_end;
9   unsigned long time_start;
    // unsigned long time_remaining;
11
    int mean_blu = 0;
13  int mean_violet = 0;
    int mean_blank = 0;
15  int mean_blank_old = 0;
    int mean_violet_blank = 0;
17  int mean_blu_blank = 0;
    int mean_blank_tot;
19  unsigned long int baseline=0;
    int mean_baseline = 0;
21
    unsigned long int a_count = 1;
23  unsigned long int a_blu = 0;
    unsigned long int a_violet = 0;
25  unsigned long int blank = 0;
    unsigned long int count_blank = 1;
27  unsigned long int a_count_v = 1;
    unsigned long int count_baseline;
29
    int led_blu =0;
31  int flag_set=1;
    int k=0;
33  int flag_wait=1;
    int led_on_prev=0;
35  int flag_blank=0;

37  int soglia=2000;
    int time_on=3000;
39
    int base_max=150;
41  int base_min=100; //limiti baseline

43  int time_wait=1500;
    unsigned long int time_wait_1=3000;
45  int val_pwm=150;

47  char char_string[4];
    String String_total;
49  String String_mean_blu;
    String String_mean_violet;
51  String String_tot_blank;
    String String_blank;
53  String String_tot_blu_blank;
    String String_blu_tot;
55  String String_tot_violet_blank;
    String String_violet_tot;
57
void setup() {
59  // initialize serial communication:
    Serial.begin(9600);
61
    //cambio i timer 1 (controlla pin 12,11) e 2 (controlla pin 10,9)
63  int myEraser = 7;     // this is 111 in binary and is used as an eraser

65  TCCR1B &= ~myEraser;   // this operation (AND plus NOT), set the three bits in TCCR2B to 0
    TCCR2B &= ~myEraser;
67
    int myPrescaler = 1;         // this could be a number in [1 , 6].
69  TCCR1B |= myPrescaler;  //this operation (OR), replaces the last three bits in TCCR2B with our
        new value 011
    TCCR2B |= myPrescaler;  //freq PWM diventa 31 kHz
71
    // inizializzo i pin di controllo
73   pinMode(9, OUTPUT);
     pinMode(7, OUTPUT);
75   pinMode(6, OUTPUT);
```

```
    pinMode(5, OUTPUT);
77  pinMode(4, OUTPUT);

79  digitalWrite(4,LOW);
    digitalWrite(5,LOW);
81  digitalWrite(6,LOW);
    digitalWrite(7,LOW);
83
    analogWrite(9,val_pwm);
85
    }
87
void measure(){
89
          while (inByte!='s'){
91
          if (Serial.available() > 0) {
93        inByte = Serial.read();}

95        //Serial.println(flag_wait);

97        if(flag_wait==1){//aspetto time_wait prima di riaccendere i LED

99          time_start=millis();
            time_end=time_start+time_wait_1;
101 //          Serial.println(time_end);
    //          Serial.println(time_start);
103           while(millis()<time_end){};  //tengo tutto spento per time_wait_1

105         flag_wait=0;
            flag_blank=1;
107         while(flag_blank==1){
            time_start=millis();
109         time_end=time_start+time_wait;
            baseline=0;
111         count_baseline=0;

113         while(millis()<time_end){
              baseline=baseline+analogRead(1);
115           count_baseline=count_baseline+1;}

117           mean_baseline=baseline/count_baseline;

119           //Serial.println(baseline);
              //Serial.print('A');Serial.println(count_baseline);
121           //Serial.println(mean_baseline);
              Serial.print('K');
123           Serial.print(val_pwm);
              Serial.println('L');
125

127           if (mean_baseline>base_max){val_pwm=val_pwm+1;
                if(val_pwm>255){val_pwm=255;}
129             analogWrite(9,val_pwm);}

131             else if(mean_baseline<base_min){val_pwm=val_pwm-1;
                if(val_pwm<0){val_pwm=0;flag_blank=0;}
133           analogWrite(9,val_pwm);}

135           else if(mean_baseline>=50 && mean_baseline<160){flag_blank=0;}
                }
137
            time_start=millis();
139         time_end=time_start+time_wait;
            while(millis()<time_end){}; //aspetto che si stabilizzi il sistema
141

143           }

145         //Serial.println(led_blu);

147           //cambio i LED accesi

149         if(led_blu==1){
              led_blu=0;
151           flag_set=1;
              led_on_prev=1;} //era acceso quello blu
153           else if(led_blu==2){
                led_blu=0;
```

```
155                     flag_set=1;
                        led_on_prev=0;
157                   flag_wait=1;}//controllo la baseline
                        else if(led_blu==0&&led_on_prev==0){
159                       led_blu=1;
                          flag_set=1;}
161                       else if(led_blu==0&&led_on_prev==1){
                            led_blu=2;
163                         flag_set=1;}

165
                 time_start=millis();
167              time_end=time_start+time_on;

169             /* Serial.println(time_start);
                 Serial.println(time_end);*/
171
                 while(millis()<time_end){
173                //accendo i LED di misura

175                if(led_blu==1 && flag_set==1){
                     digitalWrite(4,LOW);
177                  digitalWrite(5,HIGH);//per accendere i led blu devo mettere high questo
                     digitalWrite(6,LOW);
179                  digitalWrite(7,HIGH);}//per accendere i led blu devo mettere high questo
                   else if(led_blu==2 && flag_set==1){
181                  digitalWrite(4,HIGH);  //per accendere i led uv devo mettere high questo
                     digitalWrite(5,LOW);
183                  digitalWrite(6,LOW);
                     digitalWrite(7,LOW);}
185                else if(led_blu==0 && flag_set==1){
                     digitalWrite(4,LOW);
187                  digitalWrite(5,LOW);
                     digitalWrite(6,LOW);
189                  digitalWrite(7,LOW);
                   //flag_wait=1;
191              }

193                flag_set=0; //ho impostato i LED

195                //acquisisco le info di misura

197                if(led_blu==1){
                     if(millis()>time_start+soglia){
199                    a_blu=a_blu+analogRead(1);
                       a_count=a_count+1;
201                     }
                        // Serial.println(led_blu);
203                  //Serial.println(analogRead(1));
                     }
205
                       else if(led_blu==2){
207                      if(millis()>time_start+soglia){
                         a_violet=a_violet+analogRead(1);
209                      a_count_v=a_count_v+1;
                         }
211                      // Serial.println(led_blu);
                       //Serial.println(analogRead(1));
213                    }

215                      else if(led_blu==0){
                           if(millis()>time_start+soglia){
217                         blank=blank+analogRead(1);
                            count_blank=count_blank+1;
219                         }
                          //Serial.println(led_blu);
221                     //   Serial.println(analogRead(1));
                            }
223
                 }
225
                 //calcolo i valori medi dei segnali acquisiti
227
                 if(led_blu==1){
229                mean_blu=a_blu/a_count;
                   Serial.print('B');
231                Serial.print(mean_blu);
                   Serial.println('C');
233                /*String_mean_blu=String(mean_blu);
```

```
                String_total="B"+String_mean_blu+"C";
235             Serial.println(String_total);*/
                a_blu=0;
237             a_count=0;
                a_violet=0;
239             a_count_v=0;
                blank=0;
241             count_blank=0;
                k=1;}
243
                else if(led_blu==2){
245             mean_violet=a_violet/a_count_v;
                Serial.print('V');
247             Serial.print(mean_violet);
                Serial.println('W');
249             //Serial.println(a_count_v);
                //Serial.println(mean_violet);
251             /*String_mean_violet=String(mean_violet);
                String_total="V"+String_mean_violet+"W";
253             Serial.println(String_total);*/
                /*Serial.println(a_count_v);
255             Serial.println(a_violet);*/
                a_blu=0;
257             a_count=0;
                a_violet=0;
259             a_count_v=0;
                blank=0;
261             count_blank=0;
                k=1;}
263
                else if(led_blu==0){//medio il bianco
265                 mean_blank_old=mean_blank;
                    mean_blank=blank/count_blank;
267                 mean_blank_tot=(mean_blank+mean_blank_old)/2;
                    //Serial.println(mean_blank_tot);
269                 //Serial.println(mean_blank);
                    /*String_blank=String(mean_blank);
271                 String_tot_blank="D"+String_blank+"E";
                    Serial.println(String_blank);
273                 Serial.println(String_tot_blank);*/
                Serial.print('D');
275             Serial.print(mean_blank);
                Serial.println('E');
277
                   if(led_on_prev==1&&k==1){//era acceso il blu  //metto il controllo ke non sia
                       il rpimo giro
279                  mean_blu_blank=mean_blu-mean_blank_tot;
                    //Serial.println(mean_blu_blank);
281                 //if(mean_blu_blank>0){
                      //String_blu_tot=String(mean_blu_blank,DEC);
283                   //String_tot_blu_blank='P'+String_blu_tot+'Q';
                      //Serial.println(String_tot_blu_blank);}
285                     Serial.print('P');
                        Serial.print(mean_blu_blank);
287                     Serial.println('Q');
                    }
289
                      else if(led_on_prev==0&&k==1){//era accceso il viola
291                   mean_violet_blank=mean_violet-mean_blank_tot;
                    //Serial.println(mean_violet_blank);
293                   //if(mean_violet_blank>0){
                      //String_violet_tot=String(mean_violet_blank,DEC);
295                   //String_tot_violet_blank='R'+String_violet_tot+'S';
                      //Serial.println(String_tot_violet_blank);}//
297                     Serial.print('R');
                        Serial.print(mean_violet_blank);
299                     Serial.println('S');
                  }
301                 a_blu=0;
                    a_count=0;
303                 a_violet=0;
                    a_count_v=0;
305                 blank=0;
                    count_blank=0;
307                 k=1;}

309
            }
311 }
```

```
313 void blank_measure() {
    time_start=millis();
315 time_end=time_start+2000;
    blank=0;
317 count_blank=0;
    while(millis()<time_end){
319   blank=blank+analogRead(1);
      count_blank=count_blank+1;
321 }
    mean_blank=blank/count_blank;
323 String_blank=String(mean_blank,DEC);
    String_tot_blank='D'+String_blank+'E';
325 Serial.println(String_tot_blank);
}
327
  void loop() {
329  int i;

331 if (Serial.available() > 0) {
      inByte = Serial.read();
333   // do something different depending on the character received.
      // The switch statement expects single number values for each case;
335   // in this exmaple, though, you're using single quotes to tell
      // the controller to get the ASCII value for the character.  For
337   // example 'a' = 97, 'b' = 98, and so forth:
    //Serial.println(inByte);
339
    switch (inByte) {
341   case 'a': //accendo LED D1
     digitalWrite(4,HIGH);
343  digitalWrite(5,LOW);
     digitalWrite(6,LOW);
345  digitalWrite(7,LOW);
        break;
347
      case 'b':  //spengo NAD
349  digitalWrite(4,LOW);
     digitalWrite(5,HIGH);
351  digitalWrite(6,LOW);
     digitalWrite(7,LOW);
353     break;

355   case 'c':    //365 nm
     digitalWrite(4,LOW);
357  digitalWrite(5,LOW);
     digitalWrite(6,HIGH);
359  digitalWrite(7,LOW);
        break;
361
      case 'd':  // 365 nm
363  digitalWrite(4,LOW);
     digitalWrite(5,LOW);
365  digitalWrite(6,LOW);
     digitalWrite(7,HIGH);
367     break;

369  case 'e':  // 365 nm
     digitalWrite(4,LOW);
371  digitalWrite(5,LOW);
     digitalWrite(6,HIGH);
373  digitalWrite(7,HIGH);
        break;
375
     case 'f':  // 415 nm
377  digitalWrite(4,HIGH);
     digitalWrite(5,HIGH);
379  digitalWrite(6,LOW);
     digitalWrite(7,LOW);
381     break;

383    case 'g':
      measure();
385
     //Serial.println('a');
387
     digitalWrite(4,LOW);
389  digitalWrite(5,LOW);
     digitalWrite(6,LOW);
```

```
391       digitalWrite(7,LOW);

393       break;

395          case 'h':

397          blank_measure();

399          break;

401        default:
             // turn all the LEDs off:
403       digitalWrite(4,LOW);
          digitalWrite(5,LOW);
405       digitalWrite(6,LOW);
          digitalWrite(7,LOW);
407          break;
             }
409     }
    }
```

Bibliography

[1] J. Ponmozhi, C. Frias, T. Marques, and O. Frazao, "Smart sensors/actuators for biomedical applications: Review," *Measurement*, vol. 45, no. 7, pp. 1675 – 1688, 2012.

[2] Y. Gu, Z. Qian, B. Tao, and G. Wang, "A new fiber optic sensor for detecting in situ the concentration of pharmaceuticals in blood," *Sensors and Actuators B: Chemical*, vol. 66, no. 1-3, pp. 197 – 199, 2000.

[3] J.-C. Chen, H.-H. Chung, C.-T. Hsu, D.-M. Tsai, A. Kumar, and J.-M. Zen, "A disposable single-use electrochemical sensor for the detection of uric acid in human whole blood," *Sensors and Actuators B: Chemical*, vol. 110, no. 2, pp. 364 – 369, 2005.

[4] S. R., A. P.B., H. B., M. C., and M. R., "The physiologic cipher at altitude and real-time monitoring of climbers on Mount Everest," *Telemedicine Journal and e-Health*, vol. 6, no. 3, pp. 303 – 313, 2004.

[5] I. Korhonen, J. Parkka, and M. Van Gils, "Health monitoring in the home of the future," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 22, no. 3, pp. 66 – 73, 2003.

[6] A. Gandsas, K. Montgomery, D. McKenas, R. Altrudi, and Y. Silva, "In-flight continuous vital signs telemetry via the internet.," *Aviat Space Environ Med.*, vol. 71, 2000.

[7] M. Moy, S. Mentzer, and J. Reilly, "Ambulatory monitoring of cumulative free-living activity," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 22, no. 3, pp. 89 –95, 2003.

[8] D.-Y. Fei, X. Zhao, C. Boanca, E. Hughes, O. Bai, R. Merrell, and A. Rafiq, "A biomedical sensor system for real-time monitoring of astronauts physiological parameters during extra-vehicular activities," *Computers in Biology and Medicine*, vol. 40, no. 7, pp. 635 – 642, 2010.

[9] J. D. Bronzino, *The Biomedical Engineering Handbook: Second Edition*. CRC Press LLC, 2000.

[10] Y. Kim, R. Evans, and W. Iversen, "Remote sensing and control of an irrigation system using a distributed wireless sensor network," *Instrumentation and Measurement, IEEE Transactions on*, vol. 57, no. 7, pp. 1379 –1387, 2008.

[11] G. J. Hallman, "Control of stored product pests by ionizing radiation," *Journal of Stored Products Research*, vol. 52, no. 0, pp. 36 – 41, 2013.

[12] Y.-P. Wang, K.-W. Chang, R.-K. Chen, J.-C. Lo, and Y. Shen, "Large-area rice yield forecasting using satellite imageries," *International Journal of Applied Earth Observation and Geoinformation*, vol. 12, no. 1, pp. 27 – 35, 2010.

[13] F. Winquist, P. Wide, and I. Lundstrom, "An electronic tongue based on voltammetry," *Analytica Chimica Acta*, vol. 357, no. 1-2, pp. 21 – 31, 1997.

[14] G. Volpe, N. H. Ammid, D. Moscone, L. Occhigrossi, and G. Palleschi, "Development of an immunomagnetic electrochemical sensor for detection of bt-cry1ab/cry1ac proteins in genetically modified corn samples," *Analytical Letters*, vol. 39, no. 8, pp. 1599–1609, 2006.

[15] S. Mannino and J. Wang, "Electrochemical methods for food and drink analysis," *Electroanalysis*, vol. 4, no. 9, pp. 835 – 840, 1992.

[16] A. Erdem and M. Ozsoz, "Electrochemical dna biosensors based on dna-drug interactions," *Electroanalysis*, vol. 14, no. 14, pp. 965–974, 2002.

[17] J. Wang, G. Rivas, X. Cai, E. Palecek, P. Nielsen, H. Shiraishi, N. Dontha, D. Luo, C. Parrado, M. Chicharro, P. Farias, F. Valera, D. Grant, M. Ozsoz, and M. Flair, "Dna electrochemical biosensors for environmental monitoring. a review," *Analytica Chimica Acta*, vol. 347, no. 1-2, pp. 1 – 8, 1997.

[18] L. Civit, A. Fragoso, and C. O'Sullivan, "Electrochemical biosensor for the multiplexed detection of human papillomavirus genes," *Biosensors and Bioelectronics*, vol. 26, no. 4, pp. 1684 – 1687, 2010.

[19] J. Wang, "Electrochemical biosensors: Towards point-of-care cancer diagnostics," *Biosensors and Bioelectronics*, vol. 21, no. 10, pp. 1887 – 1892, 2006.

[20] R. Narayanaswamy and O. S. Wolfbeis, *Optical Sensors: Industrial, Environmental and Diagnostic Applications*. Springer, 2004.

[21] A. Pesatori, M. Norgia, and L. Rovati, "Self-mixing laser doppler spectra of extracorporeal blood flow: a theoretical and experimental study," *IEEE Sensors Journal*, vol. 12, pp. 552 – 557, 2012.

[22] A. Pesatori, M. Norgia, and L. Rovati, "Low-cost optical flowmeter with analog front-end electronics for blood extracorporeal circulators," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 5, pp. 1233 – 1239, 2010.

[23] M. Ushizima, S. Muuhlen, and I. Cestari, "A low-cost transmittance transducer for measurement of blood oxygen saturation in extracorporeal circuits," *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 4, pp. 495 – 499, 2001.

[24] D. Myklejord, M. Pritzker, D. Nicoloff, A. Emery, and R. Emery, "Clinical evaluation of the on-line $sensicath^{TM}$ blood gas monitoring system," *The Heart Surgery Forum*, vol. 1, pp. 60 – 64, 1998.

[25] S. Perov, N. Korotkov, V. Kuzemko, S. Zakharov, and V. Simanov, "Principles of optical oximetry in extracorporeal circulation systems," *Biomedical Engineering*, vol. 26, no. 5, pp. 243 – 247, 1992.

[26] F. Gamero, M. Ushizima, and I. Cestari, "Real time monitoring of oxygen saturation in extracorporeal circulation using an optical reflectance transducer," *Artificial Organs*, vol. 11, no. 25, pp. 890 – 894, 2001.

[27] J. Jiang, L. Gao, W. Zhong, S. Meng, B. Yong, Y. Song, X. Wang, and C. Bai, "Development of fiber optic fluorescence oxygen sensor in both in vitro and in vivo systems," *Respiratory Physiology & Neurobiology*, vol. 161, pp. 160 – 166, 2008.

[28] B. K. Lavine, D. J. Westover, N. Kaval, N. Mirjankar, L. Oxenford, and G. K. Mwangi, "Swellable molecularly imprinted polyn-(n-propyl)acrylamide particles for detection of emerging organic contaminants using surface plasmon resonance spectroscopy," *Talanta*, vol. 72, no. 3, pp. 1042 – 1048, 2007.

[29] L. I. Silva, F. D. Ferreira, A. C. Freitas, T. A. Rocha-Santos, and A. Duarte, "Optical fiber biosensor coupled to chromatographic separation for screening of dopamine, norepinephrine and epinephrine in human urine and plasma," *Talanta*, vol. 80, no. 2, pp. 853 – 857, 2009.

[30] G. C. Gil, R. J. Mitchell, S. T. Chang, and M. B. Gu, "A biosensor for the detection of gas toxicity using a recombinant bioluminescent bacterium," *Biosensors and Bioelectronics*, vol. 15, no. 1-2, pp. 23 – 30, 2000.

[31] J. Kumar, S. Jha, and S. D'Souza, "Optical microbial biosensor for detection of methyl parathion pesticide using Flavobacterium sp. whole cells adsorbed on glass fiber filters as disposable biocomponent," *Biosens Bioelectron*, vol. 21, no. 11, pp. 2100–5, 2006.

[32] A. Pasic, H. Koehler, I. Klimant, and L. Schaupp, "Miniaturized fiber-optic hybrid sensor for continuous glucose monitoring in subcutaneous tissue," *Sensors and Actuators B: Chemical*, vol. 122, no. 1, pp. 60 – 68, 2007.

[33] B. Polyak, E. Bassis, A. Novodvorets, S. Belkin, and R. Marks, "Optical fiber bioluminescent whole-cell microbial biosensors to genotoxicants," *Water Science & Technology*, vol. 42, no. 1-2, pp. 305 – 311, 2000.

[34] C. R. Schroeder, G. Neurauter, and I. Klimant, "Luminescent dual sensor for time-resolved imaging of pco2 and po2 in aquatic systems," *Microchimica Acta*, vol. 158, pp. 205–218, 2007.

[35] E. Eltzov, S. Cosnier, and M. R.S., "Biosensors based on combined optical and electrochemical transduction for molecular diagnostics," *Expert Rev Mol Diagn.*, vol. 11, pp. 533–546, 2011.

[36] I. Johnson and M. T. Z. Spence, *Molecular Probes Handbook, A Guide to Fluorescent Probes and Labeling Technologies, 11th Edition*. Life Technologies, Invitrogen, 2010.

[37] H. Xu and O. A. Sadik, "Design of a simple optical sensor for the detection of concentrated hydroxide ions in an unusual ph range," *Analyst*, vol. 125, pp. 1783–1786, 2000.

[38] M. F. Choi, "Spectroscopic behaviour of 8-hydroxy-1,3,6-pyrenetrisulphonate immobilized in ethyl cellulose," *Journal of Photochemistry and Photobiology A: Chemistry*, vol. 104, pp. 207 – 212, 1997.

[39] A. Richter, G. Paschew, S. Klatt, J. Lienig, K.-F. Arndt, and H.-J. P. Adler, "Review on hydrogel-based ph sensors and microsensors," *Sensors*, vol. 8, no. 1, pp. 561–581, 2008.

[40] M. Cajlakovic, A. Lobnik, and T. Werner, "Stability of new optical pH sensing material based on cross-linked poly(vinyl alcohol) copolymer," *Analytica Chimica Acta*, vol. 455, pp. 207 – 213, 2002.

[41] D. Wencel., B. MacCraith, and C. McDonagh, "High performance optical ratiometric sol-gel-based pH sensor," *Sensors and Actuators B: Chemical*, vol. 139, pp. 208 – 213, 2009.

[42] A. Dybko, W. Wroblewski, E. Rozniecka, K. Pozniak, J. Maciejewski, R. Romaniuk, and Z. Brzozka, "Assessment of water quality based on multiparameter fiber optic probe," *Sensors and Actuators B: Chemical*, vol. 51, pp. 208 – 213, 1998.

[43] M. Schirrmann, R. Gebbers, E. Kramer, and J. Seidel, "Soil pH mapping with an on-the-go sensor," *Sensors*, vol. 11, no. 1, pp. 573–598, 2011.

[44] W.-J. Cai, P. Zhao, and Y. Wang, "ph and pco2 microelectrode measurements and the diffusive behavior of carbon dioxide species in coastal marine sediments," *Marine Chemistry*, vol. 70, no. 1-3, pp. 133 – 148, 2000.

[45] H. Keith and S. Wong, "Measurement of soil co2 efflux using soda lime absorption: both quantitative and reliable," *Soil Biology and Biochemistry*, vol. 38, no. 5, pp. 1121 – 1131, 2006.

[46] H. R. Kermis, Y. Kostov, P. Harms, and G. Rao, "Dual excitation ratiometric fluorescent pH sensor for noninvasive bioprocess monitoring: development and application," *Biotechnology Progress*, vol. 18, pp. 1047 – 1053, 2002.

[47] I. Gannot, I. Ron, F. Hekmat, V. Chernomordik, and A. Gandjbakhche, "Functional optical detection based on pH dependent fluorescence lifetime," *Lasers in surgery and medicine*, vol. 35, pp. 342 – 348, 2004.

[48] A.-M. Kelly, D. Kerr, and P. Middleton, "Validation of venous pco2 to screen for arterial hypercarbia in patients with chronic obstructive airways disease," *The Journal of Emergency Medicine*, vol. 28, no. 4, pp. 377 – 379, 2005.

[49] D. Schneditz, C. Ronco, A. Brendolan, and N. L. (eds.), "Extracorporeal sensing techniques,' in Cardiovascular Disorders in Hemodialysis," *Contrib. Nephrol., Karger*, vol. 149, pp. 35 – 41, 2001.

[50] J. Tusa and H. He, "Critical care analyzer with fluorescent optical chemosensors for blood analytes," *J. Mater. Chem.*, vol. 15, pp. 2640 – 2647, 2005.

[51] C. Morgan, S. Newell, D. Ducker, J. Hodgkinson, D. White, C. Morley, and J. Church, "Continuous neonatal blood gas monitoring using a multiparameter intra-arterial sensor," *British Journal of Anaesthesia*, vol. 80, pp. 93 – 98, 1999.

[52] M. Ganter and A. Zollinger, "Continuous intravascular blood gas monitoring: development, current techniques, and clinical use of a commercial device," *British Journal of Anaesthesia*, vol. 91, pp. 397 – 407, 2003.

[53] H. Endoh, T. Honda, S. Oohashi, Y. Nagata, C. Shibue, and K. Shimoji, "Continuous intra-jugular venous blood-gas monitoring with the Paratrend 7 during hypothermic cardiopulmonary bypass," *British Journal of Anaesthesia*, vol. 87, no. 2, pp. 223–8, 2001.

[54] S. Marxer and M. Schoenfisch, "Sol-gel derived potentiometric ph sensors," *Anal. Chem.*, vol. 77, pp. 848 – 853, 2005.

[55] J. Lin, "Recent development and applications of optical and fiber-optic pH sensors," *Trends in analytical chemistry*, vol. 19, no. 9, 2000.

[56] L. Bilro, N. Alberto, J. L. Pinto, and R. Nogueira, "Optical sensors based on plastic fibers," *Sensors*, vol. 12, no. 9, pp. 12184–12207, 2012.

[57] J. W. Severinghaus and A. F. Bradley, "Electrodes for blood po2 and pco2 determination," *Journal of Applied Physiology*, vol. 13, no. 3, pp. 515–520, 1958.

[58] T. J. Manuccia and J. G. Eden, "Infrared optical measurement of blood gas concentrations and fiber optic catheter," *Patent ADD011836,*, 1985.

[59] X. Ge, Y. Kostov, and G. Rao, "High-stability non-invasive autoclavable naked optical co2 sensor," *Biosensors and Bioelectronics*, vol. 18, no. 7, pp. 857 – 865, 2003.

[60] Z. Liu, J. Liu, and T. Chen, "Phenol red immobilized PVA membrane for an optical pH sensor with two determination ranges and long-term stability," *Sensors and Actuators B: Chemical*, vol. 107, no. 1, pp. 311 – 316, 2005.

[61] P. Hashemi and R. Zarjani, "A wide range pH optical sensor with mixture of neutral red and thionin immobilized on an agarose film coated glass slide," *Sensors and Actuators B: Chemical*, vol. 135, no. 1, pp. 112 – 115, 2008.

[62] L. Rovati, P. Fabbri, L. Ferrari, and F. Pilati, "Construction and evaluation of a disposable pH sensor based on a large core plastic optical fiber," *Review of Scientific Instruments*, vol. 82, no. 2, pp. 023106 – 023106–7, 2011.

[63] D. Wencel, B. MacCraith, and C. McDonagh, "High performance optical ratiometric sol-gel based pH sensor," *Sensors and Actuators B: Chemical*, vol. 139, no. 1, pp. 208 – 213, 2009.

[64] Z. Li, C. Niu, G. Zeng, Y. Liu, P. Gao, G. Huang, and Y. Mao, "A novel fluorescence ratiometric pH sensor based on covalently immobilized piperazinyl-1,8-napthalimide and benzothioxanthene," *Sensors and Actuators B: Chemical*, vol. 114, no. 1, pp. 308 – 315, 2006.

[65] I. Sánchez-Barragán, J. Costa-Fernández, A. Sanz-Medel, M. Valledor, F. Ferrero, and J. Campo, "A ratiometric approach for ph optosensing with a single fluorophore indicator," *Analytica Chimica Acta*, vol. 562, no. 2, pp. 197 – 203, 2006.

[66] M. Uttamlal and D. R. Walt, "A fiber-optic carbon dioxide sensor for fermentation monitoring," *Nat Biotech*, vol. 13, pp. 597–601, 1995.

[67] Q. Chang, L. Randers-Eichhorn, J. R. Lakowicz, and G. Rao, "Steam-sterilizable, fluorescence lifetime-based sensing film for dissolved carbon dioxide," *Biotechnology Progress*, vol. 14, no. 2, pp. 326–331, 1998.

[68] L. Ferrari, P. Fabbri, L. Rovati, and F. Pilati, "Photobleaching effects in organic thin film sensing probes," *Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International*, pp. 1235 – 1239, 2012.

[69] Y. Kostov and G. Rao, "Low-cost optical instrumentation for biomedical measurements," *Review of Scientific Instruments*, vol. 71, pp. 4361 – 4374, 2000.

[70] L. Ferrari, L. Rovati, P. Fabbri, and F. Pilati, "Disposable fluorescence optical pH sensor for near neutral solutions," *Sensors*, vol. 13, no. 1, pp. 484–499, 2012.

[71] L. Ferrari, L. Rovati, P. Fabbri, and F. Pilati, "Continuos haematic pH monitoring in extracorporeal circulation using a disposable florescence sensing element," *Journal of Biomedical Optics*, vol. 18, no. 2, pp. –, 2013.

[72] J. K. Tusa and M. J. P. Leiner, "Optodes fluorescentes pour analytes de l'urgence," *Annales de Biologie Clinique, Dossier : 2e symposium international 'Gazométrie sanguine, biocapteurs et méthodes optiques'*, vol. 61, no. 2, pp. 183–191, 2002.

[73] Y. A. Povrozin, L. I. Markova, A. L. Tatarets, V. I. Sidorov, E. A. Terpetschnig, and L. D. Patsenker, "Near-infrared, dual-ratiometric fluorescent label for measurement of pH," *Analytical Biochemistry*, vol. 390, pp. 136 – 140, 2009.

[74] A. S. Vasylevska, A. A. Karasyov, S. M. Borisov, and C. Krause, "Novel coumarin-based fluorescent pH indicators, probes and membranes covering a broad pH range," *Analytical and Bioanalytical Chemistry*, vol. 387, pp. 2131 – 2141, 2007.

[75] M. Szabelski, K. Guzow, A. Rzeska, J. Malicka, M. Przyborowska, and W. Wiczk, "Acidity of carboxyl group of tyrosine and its analogues and derivatives studied by steady-state fluorescence spectroscopy," *Journal of Photochemistry and Photobiology A: Chemistry*, vol. 152, pp. 73 – 78, 2002.

[76] E. Braunwald, A. S. Fauci, D. L. Kasper, S. L. Hauser, D. L. Longo, and J. L. Jameson, *Harrison's Principles of internal medicine*. McGraw Hill, 2002.

[77] L. Li and L. J. Lee, "Photopolymerization of HEMA/DEGDMA hydrogels in solution," *Polymer*, vol. 46, pp. 11540 – 11547, 2005.

[78] Y. K. Son, Y. P. Jung, J. H. Kim, and D. J. Chung, "Preparation and properties of PEG-Modified PHEMA hydrogel and the morphological effect," *Macromolecular Research*, vol. 14, pp. 394 – 399, 2006.

[79] Y. Tian, F. Su, W. Weber, V. Nandakumar, B. R. Shumway, Y. Jin, X. Zhou, M. R. Holl, R. H. Johnson, and D. R. Meldrum, "A series of naphthalimide derivatives as intra and extracellular ph sensors," *Biomaterials*, vol. 31, no. 29, pp. 7411 – 7422, 2010.

[80] H. Lu, Y. Jin, Y. Tian, W. Zhang, M. R. Holl, and D. R. Meldrum, "New ratiometric optical oxygen and ph dual sensors with three emission colors for measuring photosynthetic activity in cyanobacteria," *J. Mater. Chem.*, vol. 21, pp. 19293–19301, 2011.

[81] N. Klonis and W. H. Sawyer, "Spectral properties of the prototropic forms of fluorescein in aqueous solution," *Journal of Fluorescence*, vol. 6, no. 3, pp. 147 – 157, 1996.

[82] O. S. Wolfbeis, "Materials for fluorescence-based optical chemical sensors," *J. Mater. Chem.*, vol. 15, pp. 2657–2669, 2005.

[83] A. Mills, Q. Chang, and N. McMurray, "Equilibrium studies on colorimetric plastic film sensors for carbon dioxide," *Analytical Chemistry*, vol. 64, no. 13, pp. 1383–1389, 1992.

[84] A. Mills and Q. Chang, "Fluorescence plastic thin-film sensor for carbon dioxide," *Analyst*, vol. 118, pp. 839–843, 1993.

[85] C.-S. Chu and Y.-L. Lo, "Highly sensitive and linear optical fiber carbon dioxide sensor based on sol-gel matrix doped with silica particles and HPTS," *Sensors and Actuators B: Chemical*, vol. 143, no. 1, pp. 205 – 210, 2009.

[86] C.-S. Chu and Y.-L. Lo, "Fiber-optic carbon dioxide sensor based on fluorinated xerogels doped with HPTS," *Sensors and Actuators B: Chemical*, vol. 129, no. 1, pp. 120 – 125, 2008.

[87] F. Yu, D. Yao, and W. Knoll, "Oligonucleotide hybridization studied by a surface plasmon diffraction sensor (SPDS)," *Nucleic Acids Research*, vol. 32, no. 9, p. e75, 2004.

[88] R. Georgiadis, K. P. Peterlinz, and A. W. Peterson, "Quantitative measurements and modeling of kinetics in nucleic acid monolayer films using SPR spectroscopy," *Journal of the American Chemical Society*, vol. 122, no. 13, pp. 3166–3173, 2000.

[89] T. G. Drummond, M. G. Hill, and J. K. Barton, "Electrochemical DNA sensors," *Nat Biotech*, vol. 21, no. 10, pp. 1192–1199, 2003.

[90] C. Fan, K. W. Plaxco, and A. J. Heeger, "Electrochemical interrogation of conformational changes as a reagentless method for the sequence-specific detection of DNA," *Proceedings of the National Academy of Sciences*, vol. 100, no. 16, pp. 9134–9137, 2003.

[91] K. A. Marx, "Quartz crystal microbalance: A useful tool for studying thin polymer films and complex biomolecular systems at the solution-surface interface," *Biomacromolecules*, vol. 4, no. 5, pp. 1099–1120, 2003.

[92] J. Hihath, B. Xu, P. Zhang, and N. Tao, "Study of single-nucleotide polymorphisms by means of electrical conductance measurements," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 47, pp. 16979–16983, 2005.

[93] J. Fritz, E. B. Cooper, S. Gaudet, P. K. Sorger, and S. R. Manalis, "Electronic detection of DNA by its intrinsic molecular charge," *Proceedings of the National Academy of Sciences*, vol. 99, no. 22, pp. 14142–14146, 2002.

[94] F. Uslu, S. Ingebrandt, D. Mayer, S. Bocker-Meffert, M. Odenthal, and A. Offenhausser, "Labelfree fully electronic nucleic acid detection system based on a field-

effect transistor device," *Biosensors and Bioelectronics*, vol. 19, no. 12, pp. 1723 – 1731, 2004.

[95] A. W. Peterson, R. J. Heaton, and R. M. Georgiadis, "The effect of surface probe density on DNA hybridization," *Nucleic Acids Research*, vol. 29, no. 24, pp. 5163–5168, 2001.

[96] A. W. Peterson, L. K. Wolf, and R. M. Georgiadis, "Hybridization of mismatched or partially matched DNA at surfaces," *Journal of the American Chemical Society*, vol. 124, no. 49, pp. 14601–14607, 2002.

[97] K. Arinaga, U. Rant, J. Knezevic, E. Pringsheim, M. Tornow, S. Fujita, G. Abstreiter, and N. Yokoyama, "Controlling the surface density of DNA on gold by electrically induced desorption," *Biosensors and Bioelectronics*, vol. 23, no. 3, pp. 326 – 331, 2007.

[98] U. Rant, K. Arinaga, S. Fujita, N. Yokoyama, G. Abstreiter, and M. Tornow, "Electrical manipulation of oligonucleotides grafted to charged surfaces," *Org. Biomol. Chem.*, vol. 4, pp. 3448–3455, 2006.

[99] S. O. Kelley, J. K. Barton, N. M. Jackson, L. D. McPherson, A. B. Potter, E. M. Spain, M. J. Allen, and M. G. Hill, "Orienting DNA helices on gold using applied electric fields," *Langmuir*, vol. 14, no. 24, pp. 6781–6784, 1998.

[100] T. Kawaguchi, H. Yasuda, K. Shimazu, and M. D. Porter, "Electrochemical quartz crystal microbalance investigation of the reductive desorption of self-assembled monolayers of alkanethiols and mercaptoalkanoic acids on Au," *Langmuir*, vol. 16, no. 25, pp. 9830–9840, 2000.

[101] D.-F. Yang, C. P. Wilde, and M. Morin, "Studies of the electrochemical removal and efficient re-formation of a monolayer of hexadecanethiol self-assembled at an Au(111) single crystal in aqueous solutions," *Langmuir*, vol. 13, no. 2, pp. 243–249, 1997.

[102] C.-J. Zhong and M. D. Porter, "Fine structure in the voltammetric desorption curves of alkanethiolate monolayers chemisorbed at gold," *Journal of Electroanalytical Chemistry*, vol. 425, no. 1-2, pp. 147 – 153, 1997.

[103] D. A. Barrett, M. S. Hartshorne, M. A. Hussain, P. N. Shaw, and M. C. Davies, "Resistance to nonspecific protein adsorption by poly(vinyl alcohol) thin films adsorbed to a poly(styrene) support matrix studied using surface plasmon resonance," *Analytical Chemistry*, vol. 73, no. 21, pp. 5232–5239, 2001.

[104] A. T. A. Jenkins, T. Neumann, and A. Offenhäusser, "Surface plasmon microscopy measurements of lipid vesicle adsorption on a micropatterned self-assembled monolayer," *Langmuir*, vol. 17, no. 2, pp. 265–267, 2001.

[105] A. I. K. Lao, X. Su, and K. M. M. Aung, "SPR study of DNA hybridization with DNA and PNA probes under stringent conditions," *Biosensors and Bioelectronics*, vol. 24, no. 6, pp. 1717 – 1722, 2009.

[106] R. Wang, S. Tombelli, M. Minunni, M. M. Spiriti, and M. Mascini, "Immobilisation of DNA probes for the development of SPR-based sensing," *Biosensors and Bioelectronics*, vol. 20, no. 5, pp. 967 – 974, 2004.

[107] X. Su, H. F. Teh, K. M. M. Aung, Y. Zong, and Z. Gao, "Femtomol SPR detection of DNA-PNA hybridization with the assistance of DNA-guided polyaniline deposition," *Biosensors and Bioelectronics*, vol. 23, no. 11, pp. 1715 – 1720, 2008.

[108] H. Dong, X. Cao, C. M. Li, and W. Hu, "An in-situ electrochemical surface plasmon resonance immunosensor with polypyrrole propylic acid film: Comparison between spr and electrochemical responses from polymer formation to protein immunosensing," *Biosensors and Bioelectronics*, vol. 23, no. 7, pp. 1055 – 1062, 2008.

[109] M. Bart, P. van Os, B. Kamp, A. Bult, and W. van Bennekom, "Development of a confined wall-jet flow-through cell for simultaneous electrochemical and surface plasmon resonance applications," *Sensors and Actuators B: Chemical*, vol. 84, no. 2-3, pp. 129 – 135, 2002.

[110] R. Kurita, Y. Yokota, A. Ueda, and O. Niwa, "Electrochemical Surface Plasmon Resonance measurement in a microliter volume flow cell for evaluating the affinity and catalytic activity of biomolecules," *Analytical Chemistry*, vol. 79, no. 24, pp. 9572–9576, 2007.

[111] D. D. Schlereth, "Characterization of protein monolayers by surface plasmon resonance combined with cyclic voltammetry in situ," *Journal of Electroanalytical Chemistry*, vol. 464, no. 2, pp. 198 – 207, 1999.

[112] J. Homola, *Surface Plasmon Resonance Based Sensors*. Springer, 2006.

[113] J. Homola, "Surface plasmon resonance sensors for detection of chemical and biological species," *Chemical Reviews*, vol. 108, no. 2, pp. 462–493, 2008.

[114] E. Kretschmann and H. Raether, "Radiative decay of nonradiative surface plasmons excited by light," *Z. Naturforsch. A*, vol. 23, p. 2135, 1968.

[115] C. G. Zoski, *Handbook of Electrochemistry*. Amsterdam: Elsevier, 2007.

[116] D. Harvey, *Analytical Chemistry 2.0*. McGraw-Hill, 2008.

[117] Anon, *Wenking Electronic Potentiostats - operating manual*. Gerhard Bank Elektronik, Goettingen, 1968.

[118] T. Pearson, A. Cetin, A. Tewfik, and V. Gokmen, "An overview of signal processing for food inspection [applications corner]," *Signal Processing Magazine, IEEE*, vol. 24, pp. 106 –109, may 2007.

[119] A. R. Mohd Syaifudin, S. C. Mukhopadhyay, and P. L. Yu, "Novel sensors for food inspection," *Sensors and Transducers Journal*, vol. 114, pp. 1 –40, 2010.

[120] S. Li, A. Simonian, and B. A. Chin, "Sensors for agriculture and the food industry," *The Electrochemical Society Interface*, pp. 41 –46, 2010.

[121] A. Suprem, N. Mahalik, and K. Kim, "A review on application of technology systems, standards and interfaces for agriculture and food sector," *Computer Standards & Interfaces*, no. 0, pp. –, 2012.

[122] L. D. Mello and L. T. Kubota, "Review of the use of biosensors as analytical tools in the food and drink industries," *Food Chemistry*, vol. 77, no. 2, pp. 237 – 256, 2002.

[123] P. J. O'Connell, C. K. O'Sullivan, and G. G. Guilbault, "Biosensors for food analysis," *Irish Journal of Agricultural and Food Research*, vol. 39, no. 2, pp. 321 – 329, 2000.

[124] A. Eftekhari, "Glycerol biosensor based on glycerol dehydrogenase incorporated into polyaniline modified aluminum electrode using hexacyanoferrate as mediator," *Sensors and Actuators B: Chemical*, vol. 80, no. 3, pp. 283 – 289, 2001.

[125] M. Gerard, A. Chaubey, and B. Malhotra, "Application of conducting polymers to biosensors," *Biosensors and Bioelectronics*, vol. 17, no. 5, pp. 345 – 359, 2002.

[126] N. K. Guimard, N. Gomez, and C. E. Schmidt, "Conducting polymers in biomedical engineering," *Progress in Polymer Science*, vol. 32, no. 8-9, pp. 876 – 921, 2007.

[127] M. Rocchia, M. Ellena, and G. Zeppa, "Determination of ethyl alcohol content in red wines with an optical alcohol meter based on nanostructured silicon," *Journal of Agricultural and Food Chemistry*, vol. 55, no. 15, pp. 5984–5989, 2007.

[128] M. Larrain, A. Guesalaga, and E. Agosin, "A multipurpose portable instrument for determining ripeness in wine grapes using NIR spectroscopy," *Instrumentation and Measurement, IEEE Transactions on*, vol. 57, pp. 294 –302, feb. 2008.

[129] M. Gamella, S. Campuzano, A. J. Reviejo, and J. M. Pingarrón, "Electrochemical estimation of the polyphenol index in wines using a laccase biosensor," *Journal of Agricultural and Food Chemistry*, vol. 54, no. 21, pp. 7960–7967, 2006.

[130] F. Gallarta, F. Sáinz, and C. Sáenz, "Fluorescent sensing layer for the determination of l-malic acid in wine," *Analytical and Bioanalytical Chemistry*, vol. 387, pp. 2297–2305, 2007.

[131] G. Palleschi, G. Volpe, D. Compagnone, E. L. Notte, and M. Esti, "Bioelectrochemical determination of lactic and malic acids in wine," *Talanta*, vol. 41, no. 6, pp. 917 – 923, 1994.

[132] K. R. B. Silva, I. M. Raimundo, I. F. Gimenez, and O. L. Alves, "Optical sensor for sulfur dioxide determination in wines," *Journal of Agricultural and Food Chemistry*, vol. 54, no. 23, pp. 8697–8701, 2006.

[133] M. Niculescu, R. Mieliauskiene, V. Laurinavicius, and E. Csaregi, "Simultaneous detection of ethanol, glucose and glycerol in wines using pyrroloquinoline quinone-dependent dehydrogenases based biosensors," *Food Chemistry*, vol. 82, no. 3, pp. 481 – 489, 2003.

[134] S. Rebe Raz, M. G. E. G. Bremer, W. Haasnoot, and W. Norde, "Label-free and multiplex detection of antibiotic residues in milk using imaging surface plasmon resonance-based immunosensor," *Analytical Chemistry*, vol. 81, no. 18, pp. 7743–7749, 2009.

[135] D. Gendrel, M. Chalumeau, F. Moulin, and J. Raymond, "Fluoroquinolones in paediatrics: a risk for the patient or for the community?," *The Lancet Infectious Diseases*, vol. 3, no. 9, pp. 537 – 546, 2003.

[136] M. Barza and R. Scheife, "Drug therapy reviews: Antimicrobial spectrum, pharmacology and therapeutic use of antibiotics–part 4: aminoglycosides," *American Journal of Health-System Pharmacy*, vol. 34, no. 7, pp. 723–737, 1977.

[137] M. Okolo, "Bacterial drug resistance in meat animals: a review," *International journal of zoonoses*, vol. 13, no. 3, pp. 143–152, 1986.

[138] B. Shaikh and W. A. Moats, "Liquid chromatographic analysis of antibacterial drug residues in food products of animal origin," *Journal of Chromatography A*, vol. 643, no. 1-2, pp. 369 – 378, 1993.

[139] A. Székács, "Development of enzyme-linked immunosorbent assay (elisa) systems for environmental monitoring," *International journal of zoonoses*, vol. 45, no. 1, pp. 77–80, 1994.

[140] J. Adrian, D. Pinacho, B. Granier, J.-M. Diserens, F. Sánchez-Baeza, and M.-P. Marco, "A multianalyte ELISA for immunochemical screening of sulfonamide, fluoroquinolone and β-lactam antibiotics in milk samples using class-selective bioreceptors," *Analytical and Bioanalytical Chemistry*, vol. 391, pp. 1703–1712, 2008.

[141] F. Davis and S. Higson, "Label-free immunochemistry approach to detect and identity antibiotics in milk," *Pediatric Research*, vol. 67, pp. 476–480, 2010.

[142] A. A. Torriero, J. J. Ruiz-Diaz, E. Salinas, E. J. Marchevsky, M. I. Sanz, and J. Raba, "Enzymatic rotating biosensor for ciprofloxacin determination," *Talanta*, vol. 69, no. 3, pp. 691 – 699, 2006.

[143] G. Tsekenis, G.-Z. Garifallou, F. Davis, P. A. Millner, D. G. Pinacho, F. Sanchez-Baeza, M.-P. Marco, T. D. Gibson, and S. P. J. Higson, "Detection of fluoroquinolone antibiotics in milk via a labeless immunoassay based upon an alternating current impedance protocol," *Analytical Chemistry*, vol. 80, no. 23, pp. 9233–9239, 2008.

[144] J. Adrian, S. Pasche, G. Voirin, J. Adrian, D. G. Pinacho, H. Font, F. Sanchez-Baeza, M.-P. Marco, J.-M. Diserens, and B. Granier, "Wavelength-interrogated optical biosensor for multi-analyte screening of sulfonamide, fluoroquinolone, β-lactam and tetracycline antibiotics in milk," *TrAC Trends in Analytical Chemistry*, vol. 28, no. 6, pp. 769 – 777, 2009.

[145] S. Rebe Raz, M. G. E. G. Bremer, W. Haasnoot, and W. Norde, "Label-free and multiplex detection of antibiotic residues in milk using imaging surface plasmon resonance-based immunosensor," *Analytical Chemistry*, vol. 81, no. 18, pp. 7743–7749, 2009.

[146] P. P. Dillon, S. J. Daly, J. G. Browne, B. M. Manning, E. Loomans, A. Van Amerongen, and R. O'Kennedy, "Application of an immunosensor for the detection of the ß-lactam antibiotic, cephalexin," *Food and Agricultural Immunology*, vol. 15, no. 3-4, pp. 225–234, 2003.

[147] A. Rogers and Y. Gibon, "Enzyme kinetics: Theory and practice," in *Plant Metabolic Networks* (J. Schwender, ed.), pp. 71–103, Springer New York, 2009.

[148] M. Dixon and E. Webb, "Enzymes, 2nd edition," pp. 54–116, London, UK: Longmans, 1964.

[149] L. F. Ferreira, D. M. Hueber, and T. J. Barstow, "Effects of assuming constant optical scattering on measurements of muscle oxygenation by near-infrared spectroscopy during exercise," *Journal of Applied Physiology*, vol. 102, no. 1, pp. 358–367, January 2007.

[150] C. Bebrone, C. Moali, F. Mahy, S. Rival, J. D. Docquier, G. M. Rossolini, J. Fastrez, R. F. Pratt, J.-M. Frère, , and M. Galleni, "CENTA as a chromogenic substrate for studying β-lactamases," *Antimicrob Agents Chemother*, vol. 45, no. 6, pp. 1868 – 1871, 2001.