ANODIC OXIDE GROWTH ON TANTALUM

by

TA-MING YANG

B.Sc., National Cheng Kung University, 1988
M.A.Sc., University of Waterloo, 1991
M.Sc., University of Waterloo, 1992

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

in

THE DEPARTMENT OF ELECTRICAL ENGINEERING

we accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

October 1996

© Ta-Ming Yang 1996
In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Electrical Engineering

The University of British Columbia
Vancouver, Canada

Date Oct 15, 1996
Abstract

In the growth of anodic oxide films on tantalum, the ionic conduction at a given current density depends on the electrical history of the film. In this thesis, this history effect has been investigated by examining the frequency response of the high field ionic conduction process involved in the growth of anodic oxidation on tantalum. The admittivity, defined as the complex ratio of the ac current density to the ac field, was measured. Ideally, constant dc and ac components of field would be applied. This was approximated by applying a ramp voltage plus an ac voltage, both increasing with the average film thickness. The measured current density includes displacement current. A series capacitance effect is caused by the ac component of the rate of change in film thickness. After correction for these effects, plots of the real vs. imaginary parts of the admittivity, which yielded arcs of circles with centers above the real axis, were fitted to a Cole-Cole type of expression: \( A + B/[1+(j\omega \tau_m)^{-\theta}] \), where \( 1/\tau_m \) is the angular frequency (\( \omega \)) at the maximum magnitude imaginary part, \( \pi(1-\theta) \) is the central angle subtended by the arc. This result was explained as being due to a distribution of relaxation times being involved in the history effect. This is consistent with the history effect being caused by changes in the structure of the semi-amorphous oxide film depending on formation conditions. \( \tau_m \) was found to be close to inversely proportional to the dc ionic current density and almost independent of temperature. This is consistent with a process driven by current density.

In anodic oxidation, there is the possibility of incorporation in the film of species from the electrolyte. Experiments were carried out to see if the curvature of the steady-state logJ vs. E relation and the dependence of the stepped-field line slope on the steady-state field can be caused by this incorporation. The steady-state log (J) vs. field (E) relation has been studied for
films grown in 0.1M sulphuric acid and 0.1M oxalic acid. The refractive index and thickness of the films were measured using ellipsometry. Films grown in oxalic acid showed 1-2% higher field. Measurement of the oxide capacitance at different film thickness showed the oxide permittivity to be higher for the films grown in sulphuric acid. An overshoot transient in the field (E), with a peak field (Ep), was observed when the ionic current was stepped from the steady state to a higher value (J2). The field as a function of time, E(t), was derived from the measured voltage divided by the thickness which was equivalent to the integration of ionic charge passed at each time t. Measurement of the initial slope of dE/dt showed the oxide permittivity agreed with the results from the capacitance measurement. The relation of logJ2 vs. Ep at a steady-state current density can approximately describe the stepped-field condition with the history effect held constant during the transient. The relation of logJ2 vs. Ep was shown to be linear and the slope of dlogJ2/dEp was shown to be dependent on the initial field for films grown in both electrolyte solutions. Further work is necessary before a definite conclusion can be reached about the contribution of electrolyte incorporation to curvature of the Tafel plot.
# Table of Contents

Abstract ii

Table of Contents iv

List of Figures vii

List of Tables x

Acknowledgement xi

Chapter 1 Introduction 1

Chapter 2 High Field Ionic Conduction 6

2.1 Phenomena 6
   2.1.1 Both Ions Move during Anodization on Tantalum 6
   2.1.2 Electrostriction Effect 7
   2.1.3 Faraday Law 8
   2.1.4 Overpotential and Overfield 9

2.2 Theories and Models of the History Effect 10
   2.2.1 Frenkel Defect Theory for the History Effect 10
   2.2.2 Dielectric Relaxation Model 12
   2.2.3 Structural Change Model 15

2.3 Conclusions 16

Chapter 3 Ellipsometry 17

3.1 Introduction 17

3.2 Sample Preparation 18

3.3 Steady State Studies 20

Chapter 4 Anodic Oxide Growth on Tantalum with ac+dc Field- Experiment and Results 28

4.1 Introduction 28

4.2 Experimental Procedure 29
   4.2.1 Specimen Preparation and the Anodization Cell 29
   4.2.2 Experimental Control and Data Acquisition 31
   4.2.3 Thickness Independence 33
   4.2.4 System Calibration 35

4.3 Data Analysis 40
   4.3.1 Corrections 40

4.4 Results 44
   4.4.1 Relaxation Times 44
Appendix B  Calibration of ac+dc Experimental Set-up  110
Appendix C  Impedance Measurement Programs  117
    C.1 Flowchart  117
    C.2 Source Codes of Imped Program  118
Appendix D  Ellipsometer Control Programs  131
    D.1 Introduction  131
    D.2 User’s Guide to the Ellipsometer Control Program  131
    D.3 Source Codes of Newell Program  132
    D.4 Source Codes of Ellsub Program  144
    D.5 Source Codes of Ellsubb  145
Appendix E  Parameters from the Cole-Cole Plots  159
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1(a) Delta vs. psi measured from ellipsometer. Data were measured at 5V intervals.</td>
<td>23</td>
</tr>
<tr>
<td>3.1(b) Deviation of psi from the fitted curve vs. delta.</td>
<td>24</td>
</tr>
<tr>
<td>3.1(c) Deviation of delta from the fitted curve vs. delta.</td>
<td>25</td>
</tr>
<tr>
<td>3.2(a) Potential drop between anode and a saturated calomel electrode vs. oxide thickness in a steady state condition at room temperature. The oxide thickness is the analysis result from the ellipsometric data.</td>
<td>26</td>
</tr>
<tr>
<td>3.2(b) The potential difference between the data point and the fitted results in Fig. 3.2(a) vs film thickness.</td>
<td>27</td>
</tr>
<tr>
<td>4.1 1/Current vs. time in a potentiostatic condition. Test results for silicone stop-off. The circles represent results at 20V. The squares represent results at 40V.</td>
<td>30</td>
</tr>
<tr>
<td>4.2 Ac+dc voltage/current measurement system.</td>
<td>32</td>
</tr>
<tr>
<td>4.3 Imaginary vs. real part of admittivity normalized by ( \frac{J_0}{kT} ). Thickness independence test at current density ( J=0.05 \text{mA/cm}^2 ) at room temperature in 0.1M sulphuric acid. Circles represents run at voltage interval 19-20V. Squares represents run at 30-31V. Triangles represents run at 37-38V.</td>
<td>34</td>
</tr>
<tr>
<td>4.4(a) Phase shift between ac current and ac voltage vs. amplitude of applied ac voltage.</td>
<td>36</td>
</tr>
<tr>
<td>4.4(b) The absolute value of admittivity normalized by ( \frac{J_0}{kT} ) vs. amplitude of applied ac voltage.</td>
<td>37</td>
</tr>
<tr>
<td>4.5(a) Real part of admittivity normalized by ( \frac{J_0}{kT} ) vs. amplitude of ac voltage with the same frequency and current density.</td>
<td>38</td>
</tr>
<tr>
<td>4.5(b) Imaginary part of admittivity normalized by ( \frac{J_0}{kT} ) vs. amplitude of applied voltage.</td>
<td>39</td>
</tr>
<tr>
<td>4.6(a) Imaginary vs. Real part of &quot;apparent&quot; admittivity divided by ( \frac{J_0}{kT} ). The current density was 0.089 mA/cm², the voltage range was 15 to 46 volts and the temperature was 25 °C. The data points represent results from different frequencies (which are, counter-clockwise from the top left, 32, 16, 8, 4, 2, 1, 0.5, 0.25, 0.125, 0.062, 0.031Hz). Units: qm, where q is proton charge.</td>
<td>45</td>
</tr>
<tr>
<td>4.6(b) Same as Fig. 4.6(a) after removal of contribution of displacement current density.</td>
<td>46</td>
</tr>
</tbody>
</table>
4.6(c) Same as Fig. 4.6(b) after correction also for Faradaic capacitance. .......... 47

4.7(a) Log $\tau_m$ vs. log $J_0$ at various temperatures. The line is $\chi/J_0$ with $\chi$ fitted using the whole data set. ................................................................. 49

4.7(b) Log $\tau_m$ vs. the field $E_0$. The solid lines are calculated from the fitted expression $\tau_m(E,T)$ (equation 4.4.2.2) and the broken lines from $\tau_m=\chi/J_0$ .... 50

5.1 F(s) vs. ln(s) with $\theta$ as parameter. .................................................. 58

6.1 Log $J$ vs. E in steady state conditions. The solid line shows the results predicted from Eq. 1.2 (Page 2). The smaller dashed line, fitted to the circles, represents the results measured using 0.1M sulphuric acid as the electrolyte solution. The larger dashed line, fitted to the squares, represents the results measured using 0.1M oxalic acid as the electrolyte solution. All the measurements were done at 25$^\circ$C. .............................................. 70

6.2 The slope of steady state line vs. field. The circles are the results measured in sulphuric acid and the squares are the results measured in oxalic acid using the ac+dc field method. Line 1 is the fitted results for the circles, line 2 is from Eq. 6.3.1.1, line 3 is from Eq. 6.3.1.2, and line 4 is from Eq.6.3.1.3. ................................................................. 73

6.3(a) Reciprocal capacitance vs. film thickness in a steady state condition. .... 75

6.3(b) Deviations from the fitted lines shown in Fig.6.3(a) vs. thickness. ....... 76

6.4(a) The reciprocal capacitance vs. the potential. ................................... 77

6.4(b) Deviations from the fitted lines in Fig.6.4(a) vs. potential. ............... 78

7.1(a) Voltage vs. time in a transient. Measured data for film grown in oxalic acid. ................................................................. 84

7.1(b) The current density vs. time and the field vs. charge passed converted from the data in Fig.7.1(a) after correction for the displacement current effect. .... 85

7.2 Log $J$ vs. field for steady state and stepped field conditions. The stepped field lines measured in sulphuric acid. The dashed curve represents the steady state line measured in sulphuric acid from Chapter 6. Triangles are the measured values from the stepped field experiment and circles are the measured data from the steady state measurement. .............................................. 86

7.3(a) Log $J$ vs. field for steady state and stepped field conditions. The stepped field lines measured in oxalic acid. The dashed curve is the steady state line. All the symbols are measured data. .................................................. 87

7.3(b) The deviations from the fitted lines in Fig. 7.2 and Fig. 7.3(a) vs. field. The symbols correspond to the symbols in Fig.7.2 and Fig.7.3(a). ............... 88
7.4 Stepped field slope vs. the steady state field. The triangles represent data measured in sulphuric acid and the squares represent data measured in oxalic acid. The big symbols represent results from the stepped current experiment. The small symbols represent results from the ac+dc field method. 

7.5(a) Field vs time of an initial transient. The solid line is the straight line fitted to the first 20 data. The circles are the data. 

7.5(b) The deviations from the fitted line in Fig. 7.5 (a) vs. time. 

A.1 The computer controlled ellipsometer consists of a laser source (L), a linear polarizer (P), a quarter wave phase retarder (Q.W.P) , a linear analyzer (A), and a photodetector (p.d.). The computer interfacing part includes: the shaft encoder that reads the angle of the polarizer and analyzer prisms, DT2828 ADC that reads in the photo signal, PXB721 digital I/O board that reads in the prism angle from the shaft encoder, and outputs electrical signal to move the stepping motor, as well to change the gain setting of the photodetector. 

A.2 Photosignal vs. angle of polarizer and analyzer. The photo signal was measured at the least gain of the photodetector interfaced with pc via the DT2828 ADC. 

A.3 Polarizer angle vs. analyzer angle. Fixed analyzer line was generated by fixing analyzer at an angle around 90° and balancing the polarizer at around 0°. The same principle was applied to the fixed polarizer line. 

B.1 X-axis represents the measured values from the differential voltmeter (d.v.). Y-axis represents the percentage differences between dac0/dac1 and differential voltmeter. * represents the data measured from dac0. □ represents the data measured from dac1. 

B.2 X-axis is the value measured by differential voltmeter. Y-axis is the percentage difference of adc1/adc0 and differential voltmeter measurement. 

B.3 X-axis is DC output voltage measured by the differential voltmeter. Y-axis is the percentage differences between dc input (□) / adc0 (*) and measurement from the differential voltmeter. 

B.4 X-axis is the voltage measured from adc1. Y-axis is the percentage differences between adc1 reading and d.v. 

B.5 DAC voltage vs. percentage differences. Calibration of the current signal. Dac voltage is the output voltage to resistors (100k □, and 1M *). The current signal through I/V converter was measured by adc1. The percentage difference calculates the difference between the output voltage value estimated by pc and the measured current multiplied by resistance.
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Data for Fig.4.6: real (R) and imaginary (I) parts of admittivity divided by ( J_0/kT ). Subscripts: 1, &quot;apparent&quot;; 2, corrected for displacement current term; 3, corrected for Faradaic capacitance; 4, calculated from fitted curve; 5, obtained using second analysis program. Units ( 10^{-10} ) qm.</td>
</tr>
<tr>
<td>5.1</td>
<td>( J_0 \tau_m ) over the experimental range of field and temperature, calculated from the fitted expression for ( \tau_m ) in Chapter 4 and from the reported expression for the steady state current density, normalized by value at ( 0 ) °C and field ( 6.6 \times 10^6 ) V/cm.</td>
</tr>
<tr>
<td>6.1</td>
<td>C.D. is current density in mA/cm². The unit for field in oxide, ( E ), is MV/cm. ( N_{sl} ) represents the refractive index of the tantalum substrate, ( N_{ni} ) is the refractive index of the tantalum oxide film, and ( E_1 ) is the field determined using method 1. ( N_n ) is the refractive index and ( E_2 ) is the field calculated by method 2 with ( N_{ni}=2.27-i2.50 ). ( E_3 ) is the field calculated by method 3 with ( N_{ni}=2.186 ), and ( N_{sl}=N_{sl} ). ( E_Y ) is the results from Young's work (1960) using spectrophotometric method.</td>
</tr>
<tr>
<td>A1</td>
<td>( P_{mf} ) represents the real zero of the polarizer and ( A_{ref} ) represents the real zero of the analyzer prism.</td>
</tr>
<tr>
<td>A2</td>
<td>The micrometer settings of the Babinet-Soleil compensator to result in quarter wave retardation.</td>
</tr>
<tr>
<td>A3</td>
<td>Zone differences measurements at different film thickness. ( A_1-A_3 ) represents the zone 1 and zone 3 analyzer angle difference. ( P_1-P_3 ) represents the zone 1 and zone 3 polarizer angle difference.</td>
</tr>
<tr>
<td>E1</td>
<td>The fitted parameters from Cole-Cole plot. ( T ) is the temperature, ( J ) is the steady state current density. ( A, B, \tau_m, ) and ( \theta ) correspond to the parameters shown in Eq. 4.4.1.1.</td>
</tr>
</tbody>
</table>
Acknowledgement

I thank my thesis supervisor Dr. L. Young for his profound knowledge in this field, consistent encouragement and patient guidance during the course of this research, Dr. C. Backhouse for major help in setting up the equipment and in programming, Dr. N. Jaeger for the use of a personal computer as well as a laser printer, and Mr. Hiroshi Kato for consultation in lab matters. Thanks are also given to Mr. T. Leugner, L. Kilby, D. Fletcher, D. Dawson, C. Dumont and A. Prince for the help with electronics equipment and for assistance in the machine shop. The financial support from the Natural Sciences and Engineering Research Council of Canada is gratefully acknowledged.
Chapter 1

Introduction

The objectives of this work are:

1. to study the history effect involved in the anodic oxide growth on tantalum. This effect refers to the dependence of the high field ionic conduction at a given current density on the history of the film.

2. to study how electrolyte incorporation in the growing film affects the Tafel slope (logJ (current density) vs. E (field in the oxide) relation) in both steady-state and stepped-field conditions.

Anodizing tantalum is a typical process for manufacturing high capacitance and low-loss electrolytic capacitors (Westwood 1975). Anodizing tantalum is a very reproducible process when proper chemical surface treatment of tantalum is performed (see sections 3.2 and 4.2). Anodic oxidation of tantalum involves an ionic conduction process at such high fields (1MV/cm
Chapter 1 Introduction

to 10MV/cm) that Ohm's law is not obeyed. In the classical theory of high field ionic conduction, the relation between ionic current density \( J \) and field \( E \) was given by

\[
J = J_0 \exp\left(-\frac{W(E)}{kT}\right),
\]

where \( J_0 \) is a constant and \( W(E) \), the activation barrier for ionic conduction, is \( W - \alpha E \), where \( W \) and \( \alpha \) are constants. In practice, the current at a given field depends on the history of film and temperatures previously applied. There are two limiting cases. After a certain amount of charge has been passed at constant current density a steady state condition is reached in which the field and current density are constant. In the other limiting case the field is stepped rapidly to a new value starting with steady state conditions. The plot of \( \log J \) vs \( E \) for these conditions is called the stepped field line. The assumption is that it describes the relation between \( J \) and \( E \) with whatever condition in the films which causes the history effect held constant.

Vermilyea (1953) obtained a linear steady state relation of \( \log J \) vs. \( E \) but found the Tafel slope (\( d\log J/dE \)) is independent of temperature, in contradiction with Eq. 1.1. In more accurate experimental work by Young (1960), it was found that the steady state \( \log J \) vs. \( E \) line was curved and could be described by

\[
J = J_0 \exp\left(-\frac{W}{kT}\right)\exp\left[\frac{Q}{kT} \left(\alpha + \beta E\right) E\right]
\]

where \( J_0 = 10^{8.24} \) A/cm\(^2\), \( W = 2.185 \) eV, \( Q = 5q \) (where \( q \) is proton charge), \( \alpha = 6.995 \) Å and \( \beta = -3.35 \) Å/(10\(^7\)V/cm). The films were grown in 0.1 M sulphuric acid and their thickness measured using a spectrophotometric method assuming constant film refractive index for various current densities. The idea of Eq. 1.2 was that the activation barrier \( W(E) \) in Eq. 1.1 could be expanded in a Taylor series and the second order term can not be neglected. However, Eq. 1.1 should really be
considered in relation to the stepped field line. Here it was found by Young (1961) that the initial slope $\frac{d\log J}{dE}$ of the stepped field line depended on the initial steady state field in a manner consistent with the stepped field line being expressed in similar form to Eq. 1.2. The conclusion was made that the stepped field line was also curved. Various models were advance to explain this physically, including electrostriction (Young 1963), and an activation distance dependent on field (Dignam 1981). In the theory of Bean et al. (1956) (see Chapter 2 page 10), both steady state and the stepped field lines, were predicted to depend on field and temperature as described by Eq. 1.1. Hence it seemed reasonable to assume the field dependence of the activation energy should also be of the same form.

The stepped field method is difficult to realize experimentally. However, when the current density is suddenly stepped to a new and higher value ($J_2$), the increase of the current density initially goes to charge up the oxide capacitor, inducing a rapid increase in the field. The peak field of the overshoot ($E_p$) can be an approximation of the stepped field which would give the stepped current density. Dewald (1957) had used this method and found linear $\log J$ vs. $E$ plots. Smith and Young (1981) also found that the stepped field lines ($\log J_2$ vs $E_p$) were straight using the stepped current method but found that the slope of $\frac{d\log J_2}{dE_p}$ was linearly dependent on the initial steady state field.

Young (1960) found that the field and the curvature of the steady state line was higher when the film was grown in electrolyte solution of higher concentration. Randall et al. (1965) used radio-tracers, $P^{32}$ and $S^{35}$, contained in phosphoric and sulphuric acid electrolyte solutions, respectively, to study the anodic oxide growth on tantalum. They found that the films consisted of two layers with an outer layer characteristic of the anodizing electrolyte. They also found that
the electrolyte incorporation increased with solution concentration and current density but decreased with temperature. Unlike films grown in dilute phosphoric acid, a two layer structure can not be observed using ellipsometry for films grown in sulphuric acid. Dell'Oca and Young (1970) proposed two effects associated with electrolyte incorporation to result in the curvature for film grown in dilute phosphoric acid. First, the metal ion transport number increases with J. Thus the higher J, the larger the fraction of film which has electrolyte incorporation and requires a higher field. Second, the concentration of the incorporated electrolyte is found to increase with J. The question is whether the curvature is due to some fundamental property of the ionic conduction process or is a result of electrolyte incorporation. Dell'Oca and Young also pointed out the fields in the two layers were roughly inversely proportional to their permittivity. Masing et al. (1961) found about 1% increase in the refractive index with 100 times increase in current density. It requires further work to determine if the different refractive index at different current density affects the value of the field and results in curvature in the relation of logJ vs. E.

In the present work, steady state and stepped current experiments (Chapter 6 and 7) were carried out using an electrolyte for which there was evidence of little incorporation into the film.

The small signal frequency response of the ionic conduction process is a powerful method of investigation. At very low frequencies, the current will keep up with the field and the ratio dJ/dE will be given by the slope of the steady state line. At intermediate frequencies, the history effect results in an inductive response, since at the same field, the current density is less when the field is increasing than when it is decreasing. At very high frequencies, the current is again
in phase with the field and the ratio $dJ/dE$ will correspond to the slope of the stepped field line. The frequency response has been investigated previously by Goad and Dignam (1972) and de Wit et al. (1979) for aluminum and by Taylor and Dignam (1973) for tantalum.

The phenomenon of high field ionic conduction and some important models that have been put forward in previous works which relate to the present work are reviewed in Chapter 2. In order to characterize the optical properties and the thickness of oxide films, *in situ* ellipsometry was used. The method used to analyze the ellipsometric data is described in Chapter 3. Details of the ac+dc field method relating to the experimental set-up, methods of analysis, and results are given in Chapter 4. Based on the results from Chapter 4, a structural change model which explains the history dependence of high field ionic conduction is shown in Chapter 5. The steady state log $J$ vs. $E$ relation measured in two electrolyte solutions is studied in Chapter 6. The stepped current method and results measured in two electrolyte solution are described in Chapter 7. The conclusions are given in Chapter 8.
Chapter 2

High Field Ionic Conduction

2.1 Phenomena

2.1.1 Both Ions Move during Anodization of Tantalum

Marker and tracer experiments were devised to study the mobile ions during the anodization process. Davies et al. (1962) implanted Xe$^{125}$ into tantalum substrate before anodizing tantalum. The Xe$^{125}$ marker was believed not to be charged during anodic oxidation, so the marker was expected not to move. The energy of $\beta$ rays emitted from the Xe$^{125}$ was measured using a beta ray spectrometer. Since the energy of the rays diminishes with distance in the medium, the energy of the beta ray measured by the detector could determine the location of the markers. If only oxygen ions moved, the markers would be located near the oxide surface.
It was found that both ions moved. Davies et al. (1965) determined the transport number of metal ions using the marker experiment. It was found that both ions moved while anodizing Al, Nb, W, and Ta. The transport number of those metal ions increased with ionic current density. The markers spread out after the oxide growth. However, the smallness of the spreading suggested that the oxide was not formed in the body of the existing film. Amsel and Samuel (1962) first used O\textsubscript{18} tracers. The tracers were introduced into the electrolyte solution. The tracers were detected by nuclear microanalysis. Pringle (1973) anodized tantalum first in a solution containing O\textsubscript{16}. Then, the specimen was anodized in a solution containing O\textsubscript{18}. O\textsubscript{18} was found to be near the solution side but with some mixing at the junction of the layers containing O\textsubscript{16} and O\textsubscript{18}. The results showed the movement of oxygen ions was in a series of small steps.

2.1.2 Electrostriction Effect

Ord et al. (1972), using \textit{in situ} ellipsometry, found that the thickness of oxide film increased with the electric field and the refractive index of the film decreased with the field. The changes were attributed to the variation in the density of the film with the applied field. They suggested that the local field is the control field in anodic oxidation process. The local field is given by

\[ E_{\text{local}} = \frac{E(\varepsilon + 2)}{3} \]

where \( E \) is the applied field, and \( \varepsilon \) is the dielectric constant of the oxide. Based on their results, they concluded that the dielectric constant varied linearly with the field, so the quadratic relation
between $\log(J)$ vs. $E$ could be explained with the local field control model. Cornish and Young (1973) suggested that the range of field applied in Ord et al.'s (1972) work is too small to conclude the linear dependence of thickness on the field. They showed that the oxide film is isotropic without an applied field but anisotropic with the field. It was found in the same work that film thickness and the refractive indices have a quadratic relation with the field as expected for an amorphous material.

2.1.3 Faraday Law

Ionic conduction obeys Faraday's law in the steady state. According to Faraday's law, the increase in the oxide thickness is given by

$$D = \frac{QMF}{10AF\rho}$$

2.1.3.1

where $Q$ is the charge passed, $A$ is the area, $\rho$ is the oxide density, $M$ is the molecular weight of tantalum oxide, $F$ is the Faraday, and $f$ is the current efficiency, which is close to 100%. In a steady state condition, the rate at which the potential across the oxide film changes, can be written

$$\frac{dV}{dt} = \frac{dV}{dD} \frac{dD}{dt} = \frac{EJMf}{10F\rho}$$

2.1.3.2

where $E$ is the steady state field and $J$ is the steady state current density.
Chapter 2 High Field Ionic Conduction

2.1.4 Overpotential and Overfield

A typical measured or applied voltage from the cell is the potential difference between the anode and a reversible reference electrode (we used either a saturated calomel electrode or hydrogen electrode). The potential of a reversible electrode relative to the hydrogen electrode in the same solution can be calculated from the free energy of the chemical reaction. Overpotential $V$ was defined as the potential of the tantalum with respect to the platinum electrode minus the calculated reversible potential of the cell (Ta/Ta$_2$O$_5$/Solution/H$_2$/Pt):

$$V = V_m - V_{rev} - IR_{electrolyte}$$

where $V_m$ is the potential drop from the anode to the reference reversible electrode, $V_{rev}$ is the reversible potential of the cell, $R_{electrolyte}$ is the electrolyte resistance, and $I$ is the current applied.

The overpotential includes the change from the values at the reversible potential across the oxide, plus those across the metal-oxide and solution-oxide interfaces. It has been shown by Young (1966, p.66) that the potential, extrapolated at zero 1/capacitance of the 1/capacitance vs. voltage plot, was within 0.2 V from the theoretical calculated value. The overfield $E$ is defined to be $V/D$, where $D$ is the thickness of oxide film. The overfield in the oxide is approximately equal to the differential field $dV/dD$ which is close to the actual field in the oxide.
2.2 Theories and Models of the History Effect

2.2.1 Frenkel Defect Theory for the History Effect

In classical theory, the activation barrier for an ion moving from one interstitial site to another available site is $W_0$ in the absence of a field. With an applied field $E$, the barrier would be reduced to $W_0 - QaE$ for an ion moving with the field (where $Q$ is the charge on the ion, and $a$ is half the distance between the sites). The barrier would be increased to $W_0 + QaE$ for an ion moving against the field. The total current can be written as

$$J = 2anv\exp\left[-\frac{(W - QaE)}{kT}\right] - 2a[n^+a\frac{\partial n}{\partial x}]v\exp\left[-\frac{(W + QaE)}{kT}\right]$$

where $v$ represents the frequency of the jump with no activation barrier, and $2a\frac{\partial n}{\partial x}$ takes into account the concentration gradient. The forward current is normally much larger than the backward current, so the second term can be ignored.

Bean et al.'s theory (1956) has the concentration of mobile interstitial ions dependent on history. It was proposed that high field strength can pull an ion out from a lattice site into an interstitial site. It was assumed that the vacant cation sites were immobile, and the interstitial ions were mobile. The oxide was assumed to be electrically neutral. The generation rate of the mobile ions was given by:

$$(N_0 - m)v_1\exp\left[-\frac{W_1 - Qa_1E}{kT}\right]$$

2.2.1.1

where $N_0$ is the concentration of lattice sites and $m$ is the concentration of mobile ions. The
subscript 1 indicates the process of lattice ions escaping from interstitial sites. The recombination rate (R), at which the ions were captured by the sites, was given by:

\[ R = i_0 m \]

where

\[ j = 2a_2 \nu v_2 \exp\left(\frac{W_2 - Qa_2E}{kT}\right) \]

where \( i \) is the ionic current, and \( \sigma \) is the cross section of vacant lattice sites for capture of passing interstitial ions. The subscript 2 refers to the mobility of ions. The rate of change of the mobile ion concentration was:

\[ \frac{\partial m}{\partial t} = (N_0 - m) \nu_1 \exp\left(\frac{W_1 - Qa_1E}{kT}\right) - i_0 m \]

When the generation rate and recombination rate of the mobile ions were equal, the ionic conduction process reached a steady state. In the steady state \( \partial m/\partial t = 0 \), with \( N_0 >> m \). Combining equations 2.2.1.3 and 2.2.1.4, the ionic current becomes:

\[ j = 2a_2 \nu_2 \left(\frac{\nu_1 N_0}{2\sigma a_2 v_2}\right)^{\frac{1}{2}} \exp\left[\frac{W_1 + W_2}{2kT}\right] \exp\left[\frac{QE}{kT}\right] \]

which could be simplified to: \( i = i_0 \exp\left[-(W - QaE)/kT\right] \).

During a stepped current transient, the generation and recombination rate of carriers inside the film were not equal. The rate at which vacancy concentration changes is expressed by Eq. 2.2.1.4. The time dependent field can be expressed by the combination of Poisson's equation and
the continuity equation for the stepped current (Dewald 1957):

\[
\left( \frac{\partial E}{\partial t} \right)_x = \frac{1}{\varepsilon_r \varepsilon_0} \left[ I \delta i(x,t) \right]
\]

where \( \varepsilon_r \) is the dielectric constant, \( \varepsilon_0 \) is the permittivity of free space, \( I \) is the constant current supplied, and \( i(x,t) \) is the ionic conduction current.

However, the Frenkel defect theory was proved to be in contradiction with experimental evidence by Young's transient experiment (1961). In the experiment, a specimen with a few hundred nm thickness of oxide film was annealed in boiling water for a few minutes followed by re-applying a constant voltage to the film in the electrochemical cell. The current was low initially, then, increased rapidly and finally levelled off. The initial current response to the field was found to have the relation: \( \frac{dJ}{dt} \sim J^2 \). Based on the Frenkel defect theory, the rise rate of the current should have been the highest initially and decreased thereafter. Thus the generation rate of mobile ions was apparently not solely a function of the field. Based on the observed phenomenon, Young suggested the idea of impact ionization which involved the kinetic energy transfer from moving ions.

2.2.2 Dielectric Relaxation Model

Dignam (1965), adopted Young's idea of current driven process and proposed a current driven dielectric polarization change model. The model proposed that the ionic conduction was controlled by the local field which was influenced by the ion distribution within each crystallite in the amorphous oxide. Later on, Taylor and Dignam (1973) proposed another model in which the ionic conduction process was controlled by the oxide electrolyte interface. The overpotential
(η) was divided into a film portion (ηₐ) and an electrolyte-oxide double layer portion (η₈). The normal component of electric displacement was constant throughout the film and the double layer. The electric displacement was

$$\varepsilon_0 E_0 P = \varepsilon_0 K_d E_d = \frac{\varepsilon_0 K_d \eta_d}{d}$$

where $\varepsilon_0$ is the permittivity of free space, $E$ and $P$ are the field and polarization in the oxide, $K_d$ is the effective dielectric constant of the double layer region, $d$ is the thickness of the double layer, and $E_d$ is the field in the double layer. The logarithm of current density is approximately linearly dependent on the field at steady state, so that

$$i = \alpha \exp(\beta_s E)$$

where $i$ is the steady state current density, and $\alpha$ and $\beta_s$ are constant at constant temperature. The general expression for ionic conduction could be written as

$$i = \alpha \exp(\beta E_d).$$

Under steady state conditions

$$P = \varepsilon_0 (K_s - 1) E$$

where $K_s$ is the steady-state dielectric constant of the film. So

$$\beta_s = (K_s / K_d) \beta.$$

$P$ is divided into two components: $P_1$ which follows the field, and $P_2$ which results in relaxation.
Chapter 2 High Field Ionic Conduction

process. So

\[ P = P_1 + P_2 \]  \hspace{1cm} 2.2.2.6

\[ P_1 = \varepsilon_0 (K_1 - 1) E \]  \hspace{1cm} 2.2.2.7

\[ \frac{dP_2}{dt} = \frac{1}{\tau} (\varepsilon_0 \chi_2 E - P_2) = Bi(\varepsilon_0 \chi_2 E - P_2) \]  \hspace{1cm} 2.2.2.8

where \( K_1 \) is the high frequency dielectric constant, \( \tau \) is the relaxation time which is inversely proportional to current density \( i \), and \( \chi_2 \) and \( B \) are constants. Equation 2.2.2.8 represents a single relaxation type process. In steady state, \( P_2 = \varepsilon_0 \chi_2 E \).

The mathematical form of the model could fit the experimental results to some extent, but required introducing an additional dielectric relaxation parameter \( P_3 \) to have more degrees of freedom, where \( P_3 \) was given by

\[ \frac{dP_3}{dt} = \frac{1}{\tau_3} (\varepsilon_0 \chi_3 E - P_3) \]  \hspace{1cm} 2.2.2.9

where \( \tau_3 = 1/B_3J \). This model attributed the ionic conduction to an interface controlled process. However, it is hard to understand why the ionic conduction is purely controlled by the double layer at the electrolyte oxide junction. It is also not clear how the ionic current changes the polarization.
Chapter 2 High Field Ionic Conduction

2.2.3 Structural Change Model

The structural change of tantalum oxide during oxidation is one of the important effects in ionic conduction. The change of structure changes the activation barrier and the concentration of mobile ions. Young and Smith (1979b, 1983) considered the structural change in tantalum to be the cause of the history effect. They used the results from Roth and Stephenson (1971)(1976) who studied the structure of thermally grown tantalum oxide. They claimed that the semi-amorphous form of Ta$_2$O$_5$ contains chains of pentagonal bipyramids with oxygen atoms at the corners and tantalum atoms in the centre. These chains with various lengths are joined randomly by sharing corners. Folding chains with different numbers of pentagons form a sequence of structures with different oxygen-metal ratios. The atomic arrangement surrounding the atoms of reduced coordination seems to be a defect-like situation in the ion transport process. The concentration of defects and potential variation caused by structural change may result in history dependence (Young and Smith 1979b, 1983).

Based on these observations, Young and Smith (1983) proposed the structural change model. They assumed that the factor, S, which characterizes the structure of oxide, was history dependent. The activation energy (W) of ionic conduction was a function of E and S. According to a Maclaurin expansion with small E and S

\[ W(S,E) = W(0,0) + \frac{\partial W}{\partial E} E + \frac{\partial W}{\partial S} S + \ldots = W_0 + aE + bS + \ldots \]

The ionic current at all time is given by
Chapter 2 High Field Ionic Conduction

\[ J = J_0 \exp\left( - \frac{W(S,E)}{kT} \right) \]

2.2.3.2

To satisfy the experimental evidence of \( \frac{dJ}{dt} \ll J^2 \) and saturation for steady state, they constructed an equation for the rate of change of \( S \) by assuming that \( S \) depended linearly on \( E \) in the steady state, i.e.,

\[ \frac{\partial S}{\partial t} = B(fE - S) \]

2.2.3.3

where \( B \) is a constant. In steady state conditions, \( fE = S \).

2.3 Conclusions

A few unsolved questions have been raised in Chapter 1: 1. The cause of the non-linear \( \log J \) vs. \( E \) relation under steady state conditions: In order to see if the curvature is caused by electrolyte incorporation, 0.1M oxalic acid, which was thought not to be incorporated into the film, was used. Ellipsometry was used to measure the film thickness and optical constants of the tantalum oxide film. 2. The cause for the dependence of the stepped field slope on the initial steady state field: 0.1M oxalic acid was used in the stepped field experiment to test if the electrolyte incorporation is the cause for the dependence of the stepped field slope on the steady state field. Since the history effect results in an inductive response of an ac current density to an ac field, the technique of small ac response of high field ionic conduction will be also used to investigate the history effect. 3. A phenomenological model for the history effect was formulated involving a distribution of relaxation times.
Chapter 3

Ellipsometry

3.1 Introduction

The details of our ellipsometer and alignment procedures are given in Appendix A. The ellipsometer measures the azimuth of polarizer and analyzer with a resolution of a few 0.01°. A phase change of about 360° is expected for an increase in film thickness of \( \frac{\lambda}{2n\cos \theta} \) (where \( \lambda \) is the wavelength (6328 Å) of laser light source, \( n \) is the refractive index of the measured film and \( \theta \) is the angle of refraction). A resolution of a few 0.01° can be translated into a resolution of sub-angstroms in thickness measurement. However, the high resolution does not mean high accuracy. To have meaningful ellipsometry results, the ellipsometer must be well aligned. There are a number of ways of aligning the instrument. The major steps of alignment include defining: 1. the laser alignment with respect to the pinholes, 2. the plane of incidence, 3. the compensator adjusted to give a quarter wave retardation, and 4. the cell windows and specimen. A complete description of the alignment has been provided in the thesis of Smith.
A potentially major source of error is due to the optical component imperfection (Azzam and Bashara 1977). The significance of the error is shown in the zone difference. A set of typical zone differences from one run is shown in Appendix A. The error caused by polarizer, analyzer, and compensator, can be eliminated by averaging the measurement at two zones.

3.2 Sample Preparation

In order to have an optical flat surface for the ellipsometer measurement, the tantalum specimens have to be mechanical polished and electropolished. A discussion of the electropolishing process can be found in Tegart's (1959) book. There are two processes associated with the process of electropolishing:

1. A viscous sublayer is generated on the surface of the specimen due to a constant current flow of electrolyte. It is believed that the macroscopic polishing process occurs due to a concentration gradient of the special reactants in the layer. Since the outer boundary of the viscosity layer is very flat, the mountains on the surface would have a higher concentration gradient than the valleys on the surface. The mountains will then polish faster than the valleys.

2. A thin oxide film is formed during the polishing process. Oxide growth will consume the metal on the surface. It is obvious that a microscopic notch or peak portion of the tantalum surface will be consumed faster than the flat surface during oxidation.

A few important factors are associated with the polishing process: initial surface preparations, electrolyte temperature, electrolyte agitation, electrolyte compositions, and potential-current density relations. A few small flag-shaped specimens about 1.5 centimeter square were
cut from a Fansteel poly-crystalline tantalum sheet using a hand saw. Two different single crystal tantalum pieces with <111> and <100> - orientations were also used. The sharp edges generated from the hand saw were removed using a 100 grit abrader. The samples were mechanically abraded using three different grits of silicon carbide paper (200, 400, 800). Each specimen was waxed onto the flat surface of a cylindrical metal weight. The surface was abraded along one direction using the lowest grit until the previous surface roughness disappeared. Then, the surface was rotated about 90 degrees and abraded in the new direction until the scratches from previous abrading disappeared. The same abrading scheme was repeated using papers with different grits. The specimens were cleaned thoroughly using tap water before being transferred to the paper with the next larger grit value. A mirror-like surface was achieved after abrading with 800 grit paper. All the specimens were spot welded onto tantalum wires, which could be supported by a specimen holder for ellipsometer alignment. The copper residue from the spot weld machine was removed by immersing the specimen in hot nitric acid. The specimens were cleaned in hot trichloroethylene and ultrasound before electropolishing.

The electropolishing process had to be handled with extreme care. The experimental apparatus consisted of a teflon polishing cell with a built-in propeller, a stirring rotor clamped to a stand, a platinized platinum cathode clamped to another stand and a specimen with a specimen holder clamped to a third stand. The whole experiment was carried out in a fume hood. The fume hood was cleared and cleaned. The voltage of the dc rotor was set to about 15V to 20V. The water level had to be just below the inside edge of the cell so it was low enough to avoid spilling of the liquid in the cell and high enough to have the specimen immersed completely. The cathode was connected to ground and the specimen was connected to the positive terminal of an Anatek dual regulated dc power supply. The polishing electrolyte was composed of 9 parts H$_2$SO$_4$(98%)
to 1 part HF(48%) by volume. It was very important to have the current-potential relation operated in the polishing zone (at 12-20 volts when the solution was warm). The specimen could be damaged if the process was in either the etching zone (at less than 12 volts) or the gas evolution zone (at over 20 volts). The adjustment of the Anatek was determined by the specimen. The open circuit voltage of the Anatek was set to 16V, and the short circuit current was set to 200 mA. It was found that only the adjustment of the voltage setting could move the voltage into the right zone for some of the specimens. But for other specimens, the adjustment of the dc current supply could move the voltage to 12-20 volts. Since it was difficult to calculate the current density, it was more effective to adjust both the current and voltage settings to reach the range of 12 to 20 volts. The electro-polishing time was about 5-10 minutes for each specimen. The longer the polishing time, the flatter and shinier the surface became.

3.3 Steady State Studies

The standard problem in studying anodic oxidation process using ellipsometry is to determine the thickness and refractive index of the tantalum oxide on the tantalum substrate, from measurements of $\psi$ and $\Delta$ at each null setting. A few standard techniques for solving the boundary conditions of Fresnel's equation and calculating the transmission and reflection coefficients for an isotropic film on a substrate can be found in Yeh (1988). It is an underdetermined problem to solve for the thickness and refractive index of the film and substrate from measurements of $\psi$ and $\Delta$. With our ellipsometer, there are two standard ways that we can convert an underdetermined problem into an overdetermined one. With the specimen state unchanged, we can vary the angle of incidence and measure the different nulls. For example, with three angles of incidence, we can numerically solve the problem of 6 equations with 4
unknowns (one thickness of the film, one refractive index of the isotropic film, two refractive
indices of the absorbing substrate). Measuring various film thickness on the same substrate is
the other way of solving the problem. In each experimental run, a few sets, say ten, of P
(polarizer) and A (analyzer) measurements were made at ten different film thicknesses grown at
a constant current density. The refractive index is taken to be constant during the course of the
measurement. This then becomes an overdetermined problem, with 13 unknowns (10
thicknesses, one refractive index of the film and two refractive indices of the substrate) and 20
equations, that can be solved numerically. The purpose of our experiment is to determine the
ionic conduction field at a constant current density. The field is determined from the slope of
the film thickness vs. the potential drop from anode to a saturated calomel reference electrode.

The numerical method we used to solve the overdetermined problem is the non-linear
Newton’s method given by Press et al. (1989). The method starts with a few guessed parameters,
film thickness \( l_j \) at the \( j \)th thickness, refractive index of the film \( n_f \) and the substrate \( n_s \) and
\( K_o \). The guessed values have to be close to the actual values. We can make matrices \( D \) and \( n \)
such that

\[
D = \sum_{j=1}^{n} D_j = [\psi_c, \psi_i, \Delta_c, \Delta_i]
\]

\[
n = \sum_{j=1}^{n-3} n_j = [l_j, n_s, K_o, n_f]
\]

where \( \psi_c \) and \( \Delta_c \) are values calculated from the known \( n_s, K_o, n_f \) and \( l_j \). We define the matrix
\( M_j = \partial D_j / \partial n_j \). The matrix \( D \) has to be close to zero for a good fit. The standard Newton’s method
used for finding the roots involves the iteration of the expression
\( D = M \Delta n \) or \( \Delta n = (M^T M)^{-1} \( M^T D \).\)
Chapter 3 Ellipsometry

1M^TD until the sum of D_i is minimized. The algorithm finds the thickness of the film at each data point, as well as the best fit optical constants for the data set.

Fig. 3.1 (a) shows a typical fitting curve and data point of our anodizing experiment at current density of 1.75 mA/cm² and a temperature of 25.1°C in 0.1M oxalic acid. As the film grew to a certain thickness at a certain voltage, the current source and the sample were electrically shorted, followed by ellipsometric measurements at two different zones. The measured data points, represented by circles, moved counter-clockwise along the closed curve as the film thickness increased. The solid curve is the fitted result. The refractive index of the film was found to be 2.18 and the substrate index was 2.23-j2.48. Figures 3.1 (b) and (c) shows the deviation of \( \psi \) and \( \Delta \), respectively, from the fitted curve. The standard deviations of \( \psi \) and \( \Delta \) are 0.0109° and 0.0473°, respectively. The electric field, which equals 6.49 MV/cm, is determined from the slope of the potential vs. thickness plot shown in Fig. 3.2(a). The potential deviation from the fitted curve is shown in Fig. 3.2(b). The standard deviation is 0.077 volts.
Figure 3.1(a) Delta vs. psi. The data points were measured at 5V intervals.
Figure 3.1 (b) Deviation of psi from the fitted curve vs. delta.
Figure 3.1(c) Deviation of delta from the fitted curve vs. delta.
Figure 3.2(a) The potential drop between anode and a saturated calomel electrode vs. thickness.

The oxide thickness is the analysis result from the ellipsometric data.
Figure 3.2(b) The potential difference between the data point and the fitted results vs. film thickness.
Chapter 4
AC+DC Field Studies
- Experiment and Results

4.1 Introduction

Several papers (Goad et al. 1973, de Wit et al. 1979, and Bojinov et al. 1992) have reported growing anodic oxide films with a small ac component of current or voltage superimposed on the dc component and measuring the ac impedance. de Wit et al. (1979), who give reference to this earlier work, noted that Baumann reported the inductive effect with aluminum as early as 1939. Dignam and collaborators (1972) used a dc plus ac current source with lock-in amplifier. They plotted an effective permittivity and found negative values (inductive response) with aluminum and tantalum at lower frequencies and ordinary positive values (capacitive response) at higher frequencies. De Wit et al. (1979) used a computer
controlled system on aluminum. They plotted impedance and admittance on the complex plane and found three regions of frequency. At low frequencies the component of growth due to the ac current gave a "Faradaic" capacitance effect (explained below). At intermediate frequencies the inductive effect due to the ionic conduction process dominated. A single time constant was found to be inversely proportional to the dc current. At higher frequencies the (capacitive) displacement current through the oxide dominated. Bojinov et al. (1992) found somewhat similar results with bismuth.

Expressing the results in terms of impedance or admittance is not ideal since these quantities depend on the thickness which is increasing as the experiment proceeds. In the present work, we use instead the "admittivity" defined as the (complex) ratio of current density and field, as distinct from the admittance, which relates current and voltage. It bears the same relation to admittance as conductivity does to conductance. (The name complex conductivity could be used.)

4.2 Experimental Procedure

4.2.1 Specimen Preparation and the Anodization Cell

The specimens were chemically polished. The active area was delineated using either a Teflon holder with an O-ring seal or a thick oxide covered with Dow Corning 732 RTV Sealant silicone stop-off. The exposed oxide was removed by dipping the specimen into HF solution before each run. The oxide on the specimen from previous run was removed by a few dips in 48% HF solution. A good stop-off can be sustained over 5 runs. A clean flat surface on which the silicone was applied is an important factor in obtaining a good stop-off. A test of the stop-off
Figure 4.1 Test results for silicone stop-off. The circles represent results at 20V. The squares represent results at 40V.
is shown in Fig 4.1. A constant voltage was applied to the anode and cathode, and the current through the cell was measured. Dignam (1981) suggested that an ideal anodic oxidation behaviour under potentiostatic conditions resulted in a linear relation of $1/J$ vs. time. The tantalum specimens demonstrate the anodic oxidation process in Fig.4.1 is not far from that of an ideal situation.

The glass cell was immersed in a thermostatic water bath. The electrolyte solution was saturated with hydrogen. The cathode was platinized platinum and was large enough that its potential was close to the "hydrogen potential in the same solution" at all times. The ac+dc voltage was applied to tantalum electrode with respect to the saturated hydrogen electrolyte.

4.2.2 Experimental Control and Data Acquisition

The experiments were run by an Intel Inc. 486 based personal computer with a Data Translation Inc. DT-2801 D/A (digital to analogue converter) and A/D (analogue to digital converter) card. The schematic diagram is shown in Fig. 4.2. In the operating program, the required current density and the operating temperature were first called for. The program then calculated the field required to give this current density using the expression for the steady state ionic current density in Eq.1.2. The results are based on the film thickness as determined in Young (1960) and used there to obtain the formula for the current density; that is, they are based on a value of the refractive index at 350 nm which was assumed to be independent of the current density and temperature at which the film was made. The relation to other methods for the thickness is discussed in Young (1961) and Smith et al. (1983).
Figure 4.2 Ac+dc voltage/current measurement system.
The program also calculated the rate of rise of the ramp voltage required to obtain this field (neglecting the ac current) using the rate of increase of thickness $D$ calculated using Faraday's law (Eq. 2.1.3.2). A voltage ramp was applied using a D/A converter followed by a high voltage operational amplifier. After growth of the film sufficient to give a preset voltage increase had occurred—thus establishing steady state conditions—each of a pre-chosen set of ac frequencies generated by a second D/A was superimposed in turn for at least 10 periods each. The program precalculated the ramp plus ac signal and later stepped through the array of calculated values. A Krohn-Hite model 3342 low pass filter was used to avoid noise from the digital origin of the ac voltage. The ac voltage was kept small enough in relation to the non-linear nature of the ionic conduction process. The program calculated an amplitude by using a non-linearity parameter based on the current calculated from Eq. 2.1.5.1 for the maximum and minimum fields. The ac component of field was normally less than 0.013 of the dc field. Experiments were done to check that the results were not sensitive to the amplitude of the ac signal in the range used. The effects due to non-linearity are different in our work from those in previous work because we used voltage and not current sources. The current taken by the specimen was measured by an A/D preceded by an EGG 181 current to voltage amplifier. The ac+dc current and voltage at the specimen were extracted and stored.

4.2.3 Thickness Independence

Three sets of runs at 0.05 mA/cm$^2$ current density at various film thickness are shown in Fig. 4.3. Four different frequencies (2Hz, 1Hz, 0.5Hz, and 0.25Hz) are recorded in each run. The admittivity in Fig.4.3 is normalized by $(J_0/kT)$, where $J_0$ is 0.05 mA/cm$^2$ and $T$ is 25°C. The data measured at different film thicknesses superimpose almost exactly, indicating that the
Figure 4.3 Thickness independence test at current density $J=0.05\text{mA/cm}^2$. The different symbols represent the results measured at different film thickness, where circles are the results measured at voltage interval 19-20V, squares are those measured at 30-31V and triangles are those measured at 37-38V.
admittivities are independent of the formation voltage, at least over the range of voltages tested.

4.2.4 System Calibration

The details used for calibrating each measurement device are shown in Appendix B. There are two considerations in determining the applied ac voltage amplitude: one is the signal to noise ratio, and the other is the non-linear relation of current density vs. field. One run with various ac voltage amplitudes (0.007-0.48 times the dc voltage) was done to test how the amplitude of the ac voltage affects the experimental results. Fig. 4.4(a) shows the phase difference between the ac current density and the ac field at various amplitudes. The variation of phase shift with amplitude is very random. The standard deviation is 0.0096 radian. Fig. 4.4(b) shows the amplitude ratio of current density to field with its standard deviation of $0.078 \times 10^{-10}$ qm. Figures 4.5(a) and (b) show the real and imaginary part of admittance vs. amplitude in the complex plane with a standard deviation $0.054 \times 10^{-10}$ qm in the imaginary part and $0.08 \times 10^{-10}$ qm in the real part. The results show that the admittance is independent of the field amplitude within the range (0.007-0.48 times dc voltage) tested. The actual current was measured; hence the deviation of the dc current from its target value due to non-linearity in the current density-field relation was not relevant. In the analysis programs, the ac current and voltage (or field) data were fitted to sine functions. This procedure eliminated effects from any harmonics in the current. These were not very obvious.
Figure 4.4(a) Relation between phase shift and amplitude of applied ac voltage.
Figure 4.4(b) Relation between (amplitude of ac current density/ amplitude of ac field) and amplitude of applied ac voltage.
Figure 4.5(a) Real part of admittance vs. amplitude of ac voltage with the same frequency and current density.
Figure 4.5(b) Imaginary part of admittance vs. amplitude of applied voltage.
Chapter 4 Anodic Oxide Growth on Tantalum with ac+dc Field

4.3 Data Analysis

4.3.1 Corrections

Considering a constant ac component of field, rather than a constant ac voltage, which would give a decreasing ac field as the film grows, makes the analysis much simpler.

Ideally, we wished to apply a field

$$E(t) = E_0 + E_1 e^{j\omega t}$$

where $E_0$ and $E_1$ are constants, and measure an ionic current density

$$J_{\text{ionic}} = J_0 + J_1 e^{j\omega t}$$

where $J_0$ is the dc ionic current density and $J_1$ is the (complex) amplitude of the ionic current density. The required admittivity is defined as the (complex) quantity $J_1/E_1$.

To obtain constant components of field, the voltage has to be increased in step with the thickness. The thickness of the oxide $D$ is given by Faraday's law applied to the total charge passed. The current efficiency of formation of these films is normally very close to 100%. The thickness is therefore given by integrating the ionic (not the total) current density with respect to time:

$$D(t) = \int_0^t \lambda (J_0 + J_1 e^{j\omega t}) dt$$

where, for convenience, zero time $t$ has been chosen to correspond with to zero thickness, where
\[ \lambda = \frac{MW}{10F\rho} \]

\[ MW \text{ is the molecular weight, and } \rho \text{ is the density of tantalum pentoxide (taken as } 8 \text{ g/cm}^3) \]

\[ F \text{ is the Faraday. As de Wit et al. (1979) first noted, appreciable modulation of the rate of growth of the film occurs due to the ac component of ionic current. Consequently, to obtain a field given by equation 4.3.1.1 the voltage } V(t) = ED \text{ which would have to be applied across the oxide would be rather complicated:} \]

\[ V(t) = E_0 \lambda J_0 t + E_1 \lambda J_0 e^{j\omega t} + \frac{\lambda J_1}{j\omega} (E_0 e^{j\omega t} - E_0 + E_1 e^{j\omega t} - E_1 e^{j\omega t}) \]

4.3.1.4

This is not very practical. In our experiments a ramp voltage was applied on which was superimposed a sinusoidal voltage increasing linearly with time, i.e.,

\[ V(t) = V_0(t) + V_1 e^{j\omega t} \]

4.3.1.5

where \( V_0 = E_0 \lambda J_0 t \), \( V_1 = E_1 \lambda J_0 t \), and \( E_2 \) is a constant. Thus the field, \( E = V/D \), after some algebra and a binomial expansion neglecting higher terms and the harmonic, is

\[ E = E_0 + E_1 e^{j\omega t} - \frac{J_1 e^{j\omega t}}{\lambda J_0 j\omega C_{\text{Faradaic}}} \]

4.3.1.6

so that

\[ E_1 = E_2 - \frac{J_1}{\lambda J_0 j\omega C_{\text{Faradaic}}} \]

4.3.1.7

where \( C_{\text{Faradaic}} = 1/\lambda E_0 \) is de Wit et al.'s Faradaic capacitance. De Wit et al. (1979) gave a simple argument for the existence of this capacitance but did not correct for it. Since it is a constant
capacitance, independent of thickness, its effect is inversely proportional to the thickness, which, neglecting the contribution of the ac current, is $\lambda J_0 t$. Physically, the ac component of the rate of growth is constant and therefore has less effect the thicker the film. Because of the $\omega$ in equation 4.3.1.7, this complication is more important at lower frequencies.

The measured current density includes a displacement current density (i.e. a capacitive current density through the film)

$$\frac{dD}{dt} = \varepsilon_0 \varepsilon_r \frac{dE}{dt} = \varepsilon_0 \varepsilon_r j\omega E_0 e^{j\omega t}$$

4.3.1.8

where $D$ (bold font) is the electric displacement, $\varepsilon_0$ is the permittivity of free space and $\varepsilon_r$ is the relative permittivity of the oxide at this frequency and during growth at this ionic current density. This complication is, of course, more important at the higher frequencies. To correct for it in the present work, a term $\omega \varepsilon_0 \varepsilon_r$ was subtracted from the imaginary part of the measured admittance based on the total measured current density:

$$J(t) = J_0 + J_1 e^{j\omega t} + \varepsilon_0 \varepsilon_r j\omega E_0 e^{j\omega t}$$

4.3.1.9

since, dividing by $E_1 e^{j\omega t}$, the measured admittance is seen to be $J_0/E_1 + \omega \varepsilon_0 \varepsilon_r$, where $J_0/E_1$ is the desired quantity.

4.3.2 Methods of Analysis

Two data analysis programs were used. The results are given in Table 4.1. The first program determined the apparent admittance by first fitting sinusoids to the measured ac current and voltages. The dc field was calculated from the measured dc current density using equation
<table>
<thead>
<tr>
<th>ω</th>
<th>R₁</th>
<th>I₁</th>
<th>I₂</th>
<th>R₃</th>
<th>I₃</th>
<th>R₄</th>
<th>I₄</th>
<th>R₅</th>
<th>I₅</th>
<th>λq navigation (nm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>4.71</td>
<td>6.73</td>
<td>-0.393</td>
<td>4.71</td>
<td>-0.394</td>
<td>4.74</td>
<td>-0.318</td>
<td>4.71</td>
<td>-0.39</td>
<td>27.32</td>
</tr>
<tr>
<td>8</td>
<td>4.79</td>
<td>3.18</td>
<td>-0.381</td>
<td>4.79</td>
<td>-0.382</td>
<td>4.85</td>
<td>-0.511</td>
<td>4.80</td>
<td>-0.384</td>
<td>27.37</td>
</tr>
<tr>
<td>4</td>
<td>5.04</td>
<td>0.876</td>
<td>-0.906</td>
<td>5.04</td>
<td>-0.910</td>
<td>5.05</td>
<td>-0.810</td>
<td>5.04</td>
<td>-0.909</td>
<td>27.47</td>
</tr>
<tr>
<td>2</td>
<td>5.38</td>
<td>-0.450</td>
<td>-1.34</td>
<td>5.37</td>
<td>-1.35</td>
<td>5.39</td>
<td>-1.25</td>
<td>5.38</td>
<td>-1.35</td>
<td>27.66</td>
</tr>
<tr>
<td>1</td>
<td>6.01</td>
<td>-1.33</td>
<td>-1.78</td>
<td>6.00</td>
<td>-1.80</td>
<td>5.99</td>
<td>-1.85</td>
<td>6.00</td>
<td>-1.79</td>
<td>28.04</td>
</tr>
<tr>
<td>0.5</td>
<td>6.91</td>
<td>-2.08</td>
<td>-2.30</td>
<td>6.87</td>
<td>-2.35</td>
<td>6.97</td>
<td>-2.54</td>
<td>6.88</td>
<td>-2.34</td>
<td>28.80</td>
</tr>
<tr>
<td>0.25</td>
<td>8.34</td>
<td>-2.68</td>
<td>-2.80</td>
<td>8.22</td>
<td>-2.94</td>
<td>8.43</td>
<td>-3.11</td>
<td>8.26</td>
<td>-2.91</td>
<td>30.02</td>
</tr>
<tr>
<td>0.125</td>
<td>10.3</td>
<td>-2.79</td>
<td>-2.84</td>
<td>10.1</td>
<td>-3.12</td>
<td>10.2</td>
<td>-3.25</td>
<td>10.1</td>
<td>-3.06</td>
<td>48.47</td>
</tr>
<tr>
<td>0.063</td>
<td>12.3</td>
<td>-2.29</td>
<td>-2.32</td>
<td>12.0</td>
<td>-3.06</td>
<td>11.8</td>
<td>-2.88</td>
<td>12.1</td>
<td>-2.89</td>
<td>54.62</td>
</tr>
<tr>
<td>0.031</td>
<td>13.8</td>
<td>-0.539</td>
<td>-0.553</td>
<td>13.5</td>
<td>-2.17</td>
<td>13.1</td>
<td>-2.21</td>
<td>13.6</td>
<td>-1.80</td>
<td>66.92</td>
</tr>
<tr>
<td>0.016</td>
<td>13.8</td>
<td>1.61</td>
<td>1.60</td>
<td>13.9</td>
<td>-1.17</td>
<td>13.8</td>
<td>-1.57</td>
<td>14.0</td>
<td>-0.278</td>
<td>78.63</td>
</tr>
</tbody>
</table>

Table 4.1 Data for Fig 1. Real (R) and imaginary (I) parts of admittivity divided by J₀/kT. Subscripts: 1, "apparent"; 2, corrected for displacement current term; 3, corrected for Faradaic capacitance; 4, calculated from fitted curve; 5, obtained using second analysis program. Units 10⁻¹⁰ qm.
1.2 (page 2). The measured voltage then allowed the thickness to be calculated. The measured ac voltage and this thickness then gave the ac field and hence the apparent admittivity. A term was then subtracted from the imaginary part of this corresponding to the displacement current density. This is correct for the higher frequencies for which this term is significant because at these frequencies the second term on the right in equation 4.3.1.7 is negligible. The corrected admittivity was then inverted to give the "impedivity" and a term correcting for the reciprocal of the Faradaic capacitance was subtracted. This is correct for the lower frequencies at which this term is important. Inversion then gave the desired ionic current density admittivity. In the second program the admittivity was calculated using the ac field calculated from the measured ac voltage and the thickness calculated point by point by integrating the total measured current and using Faraday's law. This eliminates the Faradaic capacitance effect correctly at lower frequencies but at the higher frequencies the displacement current is lumped with the ionic current in calculating thickness.

4.4 Results

4.4.1 Relaxation Times

Figure 4.6 shows some results for tantalum at 25°C and 0.089 mA/cm² in 0.1M sulphuric acid. Figure 4.6(a) shows the "apparent" admittivity (normalized by J₀/kT) as calculated from the total current density, using the first data analysis program. Fig 4.6(b) shows the results after removal of the displacement current term calculated using εᵣ = 27.6 which is the value reported by Young (1958) for measurements at 1 kHz, a fixed time after ceasing formation at 25 °C for films formed at 10 mA/cm² in the same electrolyte solution. As in Dignam et al.'s work on
Figure 4.6(a) Imaginary vs. real part of "apparent" admittance divided by $J_d/kT$. The current density was 0.089 mA/cm², the voltage range was 15 to 46 volts and the temperature was 25 °C. Units: qm, where q is proton charge.
Figure 4.6(b) Same after removal of contribution of displacement current density.
Figure 4.6(c) Same after correction also for Faradaic capacitance.
tantalum (1972), the data indicated a value close to this. Evidently the permittivity during actual film growth is not very different from that measured afterwards. Figure 4.6(c) shows the results corrected for the Faradaic capacitance. The points lie on an arc of a circle whose centre is above the real axis. The admittivity could thus be described the Cole and Cole (1941) type of expression which was given by

\[ A + \frac{B}{1 + (j\omega \tau_m)^{1-\theta}} \]  

where \( \frac{1}{\tau_m} \) is the angular frequency corresponding to the maximum imaginary part and \( \pi (1-\theta) \) is the central angle subtended by the arc. Table 4.1 shows the numerical data for Fig. 4.6 plus the values at each frequency calculated from the fitted modified Cole and Cole expression, showing the goodness of fit. Also included are the results obtained from the second analysis program applied to the same experimental data. These confirm the adequacy of the correction procedures in the first analysis program.

4.4.2 Temperature Dependence of Relaxation Time

Figure 4.7(a) shows a log-log plot of fitted values of \( \tau_m \) at different temperatures and current densities plotted vs. \( J_0 \). The line (of slope minus one) is

\[ \tau_m = \frac{\chi}{J_0} \]

where \( \chi \) was obtained by a least mean squares fit to the whole set of points.

Figure 4.7(b) shows the same data for log \( \tau_m \) plotted vs. the dc field in the oxide \( E_0 \). The
Figure 4.7(a) Log $\tau_m$ vs. log $J_0$ at various temperatures. The line is $\chi/J_0$ with $\chi$ fitted using the whole data set.
Figure 4.7(b) Same data plotted vs. the field $E_0$. The solid lines are calculated from the fitted expression $\tau_m(E,T)$ (equation 4.4.2.2) and the broken lines from $\tau_m = \chi/J_\phi$. 
Chapter 4 Anodic Oxide Growth on Tantalum with ac+dc Field

lines in Fig. 4.7(b) are calculated from the fitted expression, which is of similar form to the reciprocal of the expression used to describe the current density (equation 1.2)

\[ \tau_m = \tau_0 \exp\left(\frac{W - qE(a - bE)}{kT}\right) \]

It was found that the data required the quadratic term. The fitted parameters were \( \ln \tau_0 = -24.49 \), \( W = 1.718 \text{ eV} \), \( a = -1.99 \times 10^{-7} \text{ qcm} \), \( b = 3.328 \times 10^{-15} \text{ qcm}^2/\text{V} \) and \( q \) is proton charge. It is stressed that these parameters are not independent and that they are given in sufficient precision to describe the data rather than to represent their individual accuracies.

4.5 Conclusions

By correcting for the displacement current density and the Faradaic capacitance the results are brought into a clearer form than in previous work and it was then found that the locus of the admittivity is a circle with the centre off the real axis. The data could then be fitted with a modified Cole and Cole expression. Experiments were made to determine \( \tau_m \) over a range of steady state current densities and (for the first time) at different temperatures. Only the experimental results are given in this chapter. The significance of the results in terms of models of the ionic conduction process is investigated in the next chapter.
Chapter 5

Anodic Oxide Growth on Tantalum with ac+dc Field - Modelling and Theory

5.1 Introduction

A structural change model, which describes the ionic current density for all situations of field, temperature, and history, has been proposed based on the single relaxation time model of Young and Smith (1983). Since the experimental results from Chapter 4 showed that the history effect was due to a distribution of relaxation times, a modification of the structural change model based on a distribution of relaxation times is developed in this chapter.
5.2 Modelling

5.2.1 Single relaxation time

We consider first the ac response expected using the equations formulated by Young and Smith (1983, 1992) which attempt to predict the ionic current density for any conditions. These were based on the idea that the history effects are due to structural changes in the "amorphous" oxide. These equations are essentially equivalent to one of the simpler versions of the equations proposed by Dignam (1981) on the basis of his particular physical model, which involved an effective field driving ionic conduction with a contribution from a new form of current-driven dielectric polarization. De Wit et al. (1979) formulated yet another model which leads to similar equations in which the relaxation effect is ascribed to a build-up of surface charge at the metal-oxide interface. Since there is no agreement about the basic physics, it may be best to regard the equations as phenomenological, that is the result of an attempt merely to describe the experimental results rather than equations implying or being deduced from a particular physical model.

For simplicity we neglect both the curvature of the steady state logJ vs. E plots (Young 1960) and the dependence of dlogJ/dE for fields stepped to new values, starting from steady state conditions, on the initial steady state field (Young 1961). The ionic current density J is then given at all times by

\[ J = J_\text{ex} \exp\left[-\left(\frac{W_0 - \alpha E - bS}{kT}\right)\right] \]

5.2.1.1

where \( J_\text{ex} \) and \( W_0 \) are constants, \( E \) is the electrostatic field in the oxide, and \( S \) is a parameter which describes the effect of the history of the film on its ionic conduction properties. For example, \( \exp(bS/kT) \) might measure the concentration of the "defects", probably distributed defects rather than point defects, which are presumed to be involved in the ionic transport.
Chapter 5 Anodic Oxide Growth on Tantalum with ac+dc Field

process, in which both oxygen and metal move. Alternatively, the activation energy might be a function of S as well as E, with \( \alpha E \) and \( bS \) representing the linear terms in a MacLaurin expansion.

As discussed in Chapter 2 (page 12), Dignam assumed that ionic movement is due to an effective, or local, field which included a contribution from the dielectric polarization \( P_k \) from a special, current-driven process relaxation type

\[
\frac{dP_k}{dt} = \frac{1}{\tau} (\varepsilon \chi_k E - P_k) \tag{5.2.1.2}
\]

He assumed that \( \tau \propto 1/J \). His argument was that the ions sweep out a volume proportional to \( J \) in which the polarization process is expedited. This assumption is required to describe Young's (1960) experimental result that the initial rate of rise of current density for a constant field applied to films annealed in boiling water to reduce S to a low value is

\[
\frac{dJ}{dt} \propto J^2 \tag{5.2.1.3}
\]

In the present analysis Eq.5.2.1.2 becomes

\[
\frac{dS}{dt} = \frac{1}{\tau} (fE - S), \tag{5.2.1.4}
\]

where \( f \) is a constant. In the steady state, \( dS/dt=0 \) giving \( S=fE \) so that the steady state current density is given by

\[
J_s = J_0 \exp\left[-\frac{(W_0-(\alpha+bf)E)}{kT}\right]. \tag{5.2.1.5}
\]

If the field is stepped slowly enough to maintain steady state, then
Chapter 5 Anodic Oxide Growth on Tantalum with ac+dc Field

\[
\frac{d \ln J}{dE} = \frac{\alpha + \gamma}{kT} \tag{5.2.1.6}
\]

where

\[
\gamma = bf. \tag{5.2.1.7}
\]

If the field is stepped quickly enough that \( S \) cannot change appreciably, then

\[
\frac{d \ln J}{dE} = \frac{\alpha}{kT}. \tag{5.2.1.8}
\]

Consider a voltage \( V(t) \) applied so that the field in the oxide is the real part of

\[
E = E_0 + E_1 e^{j\omega t}. \tag{5.2.1.9}
\]

We seek the ac steady state response giving an ac current density

\[
J = J_0 + J_1 e^{j\omega t}. \tag{5.2.1.10}
\]

and \( S = S_0 + S_1 e^{j\omega t} \), where \( E_1 << E_0 \), \( J_1 << J_0 \) and \( S_1 << S_0 \). Substituting equations 5.2.1.9 and 5.2.1.10 in equations 5.2.1.4 and 5.2.1.5 and neglecting small quantities the admittivity is obtained as

\[
\frac{J_1 - J_0}{E_1} = \frac{J_0}{kT} \left[ \alpha + \gamma \left( \frac{1}{1 + j\omega \tau} - \frac{j\omega \tau}{1 + \omega^2 \tau^2} \right) \right]. \tag{5.2.1.11}
\]

According to this, the admittivity is normalised by \( J_0/kT \) (where \( J_0 \) is the dc current density), and gives a semi-circle on the Argand plane with its centre on the real axis at \( \alpha + (\gamma/2) \) and radius \( \gamma/2 \), independent of \( J_0 \). At very low frequencies compared to \( 1/\tau \) the ac current density and field are in phase and the intercept on the real axis corresponds to the steady state value of \( d \ln J/dE \).
At very high frequencies the current density and field are again in phase but the intercept corresponds to the stepped field value of $d\ln J/dE$. At intermediate frequencies the ac current density and field are out of phase in an inductive mode.

5.2.2 Distribution of Relaxation Times

The experimental results for the admittivity lie on an arc of a circle but its centre is above the real axis. This is analogous to the situation found by Cole and Cole (1941) for a wide range of liquids and solids, for which the classic model of Debye giving relaxation losses with a single relaxation time might have been expected. Cole and Cole showed that the experimental result may be represented by replacing the Debye term $1/(1+j\omega \tau)$ by $1/(1+(j\omega \tau_m)^{1-\theta})$, where $\omega = 1/\tau_m$ is the angular frequency corresponding to the maximum magnitude of the imaginary part of the response function, and where the angle subtended by the arc is $(1-\theta)\pi$. The anomaly was assumed by them to be due to a distribution of relaxation times. Using a formula derived using transform methods by Fuoss and Kirkwood (1941) to de-convolute the distribution of relaxation times from the observed frequency dependence, Cole and Cole showed that writing the distribution of relaxation times as $G(\tau)$, where

$$\int_0^\infty G(\tau) d\tau = 1$$

then introducing

$$s = \ln \left( \frac{\tau}{\tau_m} \right)$$
Chapter 5 Anodic Oxide Growth on Tantalum with ac+dc Field

and defining

\[ F(s) = \tau G(\tau), \]

yields

\[ F(s) = \frac{\frac{1}{2\pi} \sin(\theta \pi)}{\cosh[(1-\theta)s]-\cos(\theta \pi)}. \]

This function is shown in Fig. 5.1 with \( \theta \) as a parameter.

The Cole and Cole analysis may be modified as follows to fit the present situation. Let \( \delta S \) be the contribution to \( S \) due to the \( G(\tau)\delta \tau \) part of the distribution. The time evolution of \( \delta S \) is then given by

\[ \frac{d\delta S}{dt} = \frac{1}{\tau}[fE G(\tau)\delta \tau - \delta S] \]

so that with an ac field component \( E_1 e^{j\omega t} \), \( \delta S \) becomes \( \delta S_0 e^{j\omega t} \) where by substituting in equation 5.2.2.5 it is found that \( \delta S_0 = fE_1 G(\tau) \delta \tau/(1+j\omega \tau) \). This gives

\[ S(\omega) = E_1 e^{j\omega t} \frac{fG(\tau)}{1+j\omega \tau} d\tau \]

so that

\[ \frac{I_1}{E_1} = \frac{I_0}{kT(\alpha + \gamma)} \frac{G(\tau)}{1+j\omega \tau} d\tau \]

Cole and Cole's distribution function then gives the observed result.
Figure 5.1 $F(s)$ vs. $\ln(s)$ with $\theta$ as parameter.
When a single relaxation time is assumed, making it inversely proportional to $1/J$ reproduces equation 5.2.1.3. Goad and Dignam (1972) found a better fit to experimental data on the ac response with two relaxation times. De Wit et al. (1979) found (also for aluminium) that the frequency of the maximum ratio of imaginary to real components of the admittance was proportional to $J_0$. For a distribution of relaxation times, all of which depended inversely on $J$, $\tau_m$ would be inversely proportional to $J_0$. The experimental data diverge from this but not so far as to be sure that the divergence is not due to experimental error. Thus Table 5.1 shows the product $J_0\tau_m$ over the experimental range of field and temperature, calculated from the fitted expression for $\tau_m$ from Chapter 4 and from the expression for the steady state current density reported by Young (1983). As noted in Chapter 4, the expression for $\tau_m$ required the $E^2$ term as does the expression for $J_0$ (Eq. 1.2).

<table>
<thead>
<tr>
<th>T(°C)</th>
<th>0</th>
<th>25</th>
<th>50</th>
<th>75</th>
</tr>
</thead>
<tbody>
<tr>
<td>E/(10^6 V/cm)</td>
<td>4.8</td>
<td>1.19</td>
<td>1.43</td>
<td>1.68</td>
</tr>
<tr>
<td></td>
<td>5.4</td>
<td>1.69</td>
<td>1.98</td>
<td>2.27</td>
</tr>
<tr>
<td></td>
<td>6.0</td>
<td>1.60</td>
<td>1.88</td>
<td>2.16</td>
</tr>
<tr>
<td></td>
<td>6.6</td>
<td>1.00</td>
<td>1.22</td>
<td>1.45</td>
</tr>
</tbody>
</table>

Table 5.1 $J_0\tau_m$ over the experimental range of field and temperature, calculated from the fitted expression for $\tau_m$ in Chapter 4 and from the reported expression for the steady state current density, normalized by value at 0 °C and field 6.6×10^6 V/cm.
Chapter 5 Anodic Oxide Growth on Tantalum with ac+dc Field

5.4 Discussions

5.4.1 Gaussian Distribution

Since the oxide film is essentially amorphous, it would be expected that a distribution of relaxation times should be required to describe the time evolution of the change in the oxide which causes the history dependence and hence the "transients". As Cole and Cole pointed out, the distribution of relaxation times (Eq. 5.2.2.4) is close to a Gaussian. The true distribution is probably Gaussian and the fact that the function $1/(1+(j\omega \tau_m)^{-\beta})$ which fits the experimental results corresponds exactly to equation 5.2.2.4 is a convenient accident. However, the treatments of dielectrics by Elliott (1987) indicated that the Cole and Cole type of response may involve linked processes rather than independent processes with a distribution of relaxation times. Similar interpretations may apply here.

5.4.2 Comments on the Dielectric Polarization Model

When the accelerating build-up of the current density at constant field, described by Eq. 5.2.1.2, was first reported by Young (1961 page 109) it was concluded that the build-up was actually controlled by the current density, perhaps through some process of momentum transfer. An ion which has just surmounted an energy barrier clearly has an excess kinetic energy which must be disposed of. The idea that the build up was indeed controlled by the current density has hardly been discussed in the literature and may well have been regarded as unlikely. Also, since both ions move one has to ask which ion triggers the change in the oxide. The idea was evidently taken up by Dignam (1981) in his model in which the ions are driven by an effective
field which contains a contribution from dielectric polarization. Dignam's models are based on the idea that there was a Debye type polarization process in which the relaxation time was inversely proportional to the current density giving a current-driven component of dielectric polarization. Young and Smith (1983) took the position that Dignam's equations are more easily seen as contrived to fit the experiment data than as derived from a physical model using accepted methods. There seems to be no previous suggestion of such a current-driven polarization process, and, as regards the other feature of the model, the effective field (usually Lorentz field $(1/3)(\varepsilon_r+2)$), whilst it is true that the field for a given ionic current density in the oxides on different metals or in oxides on the one metal but with different electrolyte incorporation tends to go inversely as the relative dielectric permittivity $\varepsilon_r$ it is not clear whether this is because an effective field is involved or because the same properties of the oxide control both permittivity and ionic conductivity. The question of an effective field was discussed in Young (1961 pages 17 and 179).

In equation 5.2.1.4 the relaxation time is taken as inversely dependent on J. This could either be due to Dignam's model applying or to the passage of ions through the oxide somehow driving a change in the structure of the oxide which causes the history effect. There does not seem to have been any previous investigation of the temperature dependence and it is noteworthy that the experimental data show $\tau_m$ as being not only not far from being inversely proportional to $J_0$ at a single temperature but also quite close to being inversely proportional to $J_0$ independent of temperature. With a distribution of relaxation times one presumably has a distribution of activation energies and field parameters and unless all the $\tau$ were individually inversely proportional to J one would perhaps not expect an inverse dependence of $\tau_m$. Whether one favours the dielectric polarization model or the structural change model, as Young (1961)
indicated elsewhere, the conclusion that the current itself drives the process could be mistaken. As regards the variation of $\tau_m$ with $J_0$, the fitted expression for $\tau_m$ was chosen to have the same form as that for the steady state current density but the fitted parameters are distinctly different. However, the deviation of the experimental points from the hypothesis of a constant $\tau_m J_0$ may appear arguably not that large in Fig 4.7 (a) in Chapter 4, although going over a range of 1 to 2.54 in Table 5.1 (page 59), which shows the product $J_0 \tau_m$ calculated from the fitted expressions. $J$ itself is controlled at all times by $E$ and $S$. In equation 5.2.1.3, since in our experiments $J$ is basically the steady state current $J_0$, being modulated only in small degree by the ac field, $J$ could be eliminated using equation 5.2.1.4 and the rate of change of $S$ would then appear as dependent on the field and on the actual state of the oxide structure as measured by $S$, rather than being directly affected by the passage of ions through the oxide. One could argue that the relaxation time indeed depends on the field, the temperature and the condition of the oxide material as described by $S$, with parameters which accidentally give values not too far from inversely proportional to the steady state current density. If this position is not taken, then the problem of just how the current drives the process has to be faced.

5.5 Summary

The inductive phase relation between the ionic current and field is due to the dependence of the ionic current density on the previous history of fields and temperatures, etc. It seems probable that this history dependence is due to structural changes in the oxide and equations have previously been formulated to attempt to describe (or predict) the ionic current density for all situations of field, temperature and history. These equations were modified to account for the observed behaviour by transferring the analysis by Cole and Cole of an analogous situation with the dielectric permittivity. The conclusion in brief is that the change in the films which causes
the ionic current to depend on the history of the film may be described as involving a
distribution of relaxation times. The type of distribution found is not too different from a
Gaussian and this seems very consistent with the hypothesis that the history effects are due to
changes in the structure of these amorphous oxides. However, since other models have been
proposed which give similar equations, these could be regarded as phenomenological.

The relaxation times involved in the adjustment of the structure to a new field might be
expected to be a function of the field, temperature and the state of the oxide at any given time.
For the present experiments, since the state of the oxide corresponds to steady state (growth at
constant current and field for long enough), $\tau_m$ could be fitted by a function of field and
temperature alone. The implications of the fact that $\tau_m$ was not too far from being inversely
proportional to the ionic current density independent of temperature are discussed in relation to
the models of high field ionic conduction.
Chapter 6

Electrolyte Incorporation Studies

- Steady State

6.1 Introduction

Previous work on electrolyte incorporation effects has been described in section 2.1.4. In our unpublished work, the sulphur could not be detected in the film, when oxidized in 0.1 M sulphuric acid, by XPS. Dr. Seigfried Hofmann at the Institut für Werkstoffwissenschaften stated, in a personal communication, that he could not detect any incorporated species in films made in oxalic acid by measuring depth profiles using XPS and AES with a sensitivity of 0.5%.

In the present work, to test how the electrolyte incorporation affects the steady state lines, the relation of \( \log J \) vs. \( E \) for film grown in both 0.1M sulphuric acid and 0.1 M oxalic acid was measured. The optical properties of the oxide films were measured using ellipsometry. The
steady state fields were determined using the differential field approach. The dependence of the
film refractive index on the current density and that of the curvature of the steady state line on
the refractive index are analyzed. The permittivities of the films grown in the different
electrolytes were measured. The ac+dc field method described in Chapter 4 was used to study
the steady state curvature in both electrolytes. The results from the ac+dc field studies were
compared with the ellipsometry results.

6.2 Experimental

The anodization cell contained either 0.1M sulphuric acid or 0.1M oxalic acid. The cell
was rinsed thoroughly between each change of electrolyte. A thermostatic controller with a
stirring propeller was used to control the temperature inside the cell to be 25±0.05°C. The
cathode was made of a 4 cm$^2$ platinum foil. A saturated calomel electrode was used as a
reference electrode. The details of the tantalum surface treatment have been mentioned in section
3.2. The potential drop between the anode and the reference electrode was measured by a very
high input impedance (>10$^9$Ω) DANA volmeter to avoid drawing current from the cell. The
voltage reading was interfaced to a 486 personal computer via a pxb 721 digital I/O board.

6.3 Steady State Results

6.3.1 Steady State Line

The ellipsometric method of determining the field has been described in Chapter 3. The
thickness of the film was measured at 5-volt intervals up to 50 volts at one current density. The
field in the oxide was calculated using the differential field method. The ellipsometry data measured in the two different electrolytes were fitted to a model with a single oxide layer on a tantalum substrate. Three different numerical approaches were used in deriving the results shown in Table 6.1. In the first method, the refractive index of the film and substrate as well as the thickness of the film were varied to find a best fit of the ellipsometry data at each current density. The averaged refractive index of the substrate is $(2.27\pm0.049)-i(2.50\pm0.034)$. The average value excluded run 10 because the error function of the numerical fitting could not be minimized with the averaged values. The fitted refractive index of the substrate varied from run to run even with the same specimen. According to some previous works, the refractive index of the substrate measured using ellipsometry can vary from $3.02-i2.57$ (Ord 1972) to $2.45-i2.57$ (Cornish and Young 1973) with a 632.8 nm He-Ne laser as the light source. There are two possible causes for the differences in the substrate refractive indices. One is the numerical approaches, and the other is the fundamental differences caused by the surface treatment. If a bare tantalum substrate exists for ellipsometry measurement, the refractive index of the substrate can be calculated from a single set of polarizer (P) and analyzer (A) measurements based on the Fresnel equation. Nevertheless, a natural thin layer of oxide always exists though the tantalum has been chemically polished. So it is not possible to measure the refractive index for the bare substrate. Besides, both metal ions and oxygen ions moved to form oxide at both the metal-oxide and oxide-electrolyte interfaces during oxidation, so the interface conditions were different as the oxides grew thicker. Because the ellipsometer can resolve film thickness to sub-angstrom range, any microscopic changes in the oxide film will be included in the ellipsometry results. A series of measurements of P and A at various oxide thicknesses can be extrapolated to find the refractive index of the substrate using the statistical method mentioned in Chapter 3. The substrate refractive index was indirectly determined from a model which consists of an oxide film
### 0.1 M Oxalic Acid as the Electrolyte Solution

<table>
<thead>
<tr>
<th>Run#</th>
<th>$N_{s1}$</th>
<th>$N_n$</th>
<th>$N_{s2}$</th>
<th>C.D.</th>
<th>$E_1$</th>
<th>$E_2$</th>
<th>$E_3$</th>
<th>Specimen Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.23-2.48</td>
<td>2.182</td>
<td>2.181</td>
<td>1.75</td>
<td>6.49</td>
<td>6.49</td>
<td>6.52</td>
<td>Polycrystalline</td>
</tr>
<tr>
<td>2</td>
<td>2.24-2.49</td>
<td>2.183</td>
<td>2.181</td>
<td>0.70</td>
<td>6.27</td>
<td>6.27</td>
<td>6.30</td>
<td>Polycrystalline</td>
</tr>
<tr>
<td>3</td>
<td>2.24-2.49</td>
<td>2.186</td>
<td>2.183</td>
<td>0.176</td>
<td>6.02</td>
<td>6.01</td>
<td>6.03</td>
<td>Polycrystalline</td>
</tr>
<tr>
<td>4</td>
<td>2.25-2.50</td>
<td>2.189</td>
<td>2.185</td>
<td>0.071</td>
<td>5.82</td>
<td>5.82</td>
<td>5.82</td>
<td>Polycrystalline</td>
</tr>
<tr>
<td>5</td>
<td>2.25-2.49</td>
<td>2.187</td>
<td>2.185</td>
<td>0.0166</td>
<td>5.56</td>
<td>5.56</td>
<td>5.56</td>
<td>Polycrystalline</td>
</tr>
<tr>
<td>13</td>
<td>2.26-2.42</td>
<td>2.186</td>
<td>2.210</td>
<td>0.0030</td>
<td>5.30</td>
<td>5.31</td>
<td>5.23</td>
<td>Polycrystalline</td>
</tr>
</tbody>
</table>

### 0.1 M Sulphuric Acid as the Electrolyte Solution

<table>
<thead>
<tr>
<th>Run#</th>
<th>$N_{s1}$</th>
<th>$N_n$</th>
<th>$N_{s2}$</th>
<th>C.D.</th>
<th>$E_1$</th>
<th>$E_2$</th>
<th>$E_3$</th>
<th>$E_Y$</th>
<th>Specimen Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>2.28-2.52</td>
<td>2.187</td>
<td>2.184</td>
<td>1.75</td>
<td>6.35</td>
<td>6.39</td>
<td>6.34</td>
<td>6.28</td>
<td>Polycrystalline</td>
</tr>
<tr>
<td>7</td>
<td>2.30-2.51</td>
<td>2.184</td>
<td>2.186</td>
<td>0.70</td>
<td>6.19</td>
<td>6.18</td>
<td>6.18</td>
<td>6.12</td>
<td>Polycrystalline</td>
</tr>
<tr>
<td>8</td>
<td>2.35-2.53</td>
<td>2.180</td>
<td>2.185</td>
<td>0.176</td>
<td>5.91</td>
<td>5.91</td>
<td>5.88</td>
<td>Polycrystalline</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>2.37-2.54</td>
<td>2.177</td>
<td>2.180</td>
<td>0.071</td>
<td>5.75</td>
<td>5.75</td>
<td>5.78</td>
<td>5.73</td>
<td>Polycrystalline</td>
</tr>
<tr>
<td>10</td>
<td>2.98-2.65</td>
<td>2.172</td>
<td>N/A</td>
<td>0.0162</td>
<td>5.48</td>
<td>N/A</td>
<td>N/A</td>
<td>5.50</td>
<td>&lt;111&gt;</td>
</tr>
<tr>
<td>11</td>
<td>2.28-2.52</td>
<td>2.187</td>
<td>2.187</td>
<td>0.250</td>
<td>6.00</td>
<td>5.98</td>
<td>5.99</td>
<td>5.94</td>
<td>&lt;100&gt;</td>
</tr>
<tr>
<td>12</td>
<td>2.20-2.45</td>
<td>2.197</td>
<td>2.197</td>
<td>0.0030</td>
<td>5.26</td>
<td>5.26</td>
<td>5.21</td>
<td>5.24</td>
<td>Polycrystalline</td>
</tr>
</tbody>
</table>

Table 6.1 C.D. is current density in mA/cm². The unit for field in oxide, $E$, is MV/cm. $N_{s1}$ represents the refractive index of the tantalum substrate, $N_n$ is the refractive index of the tantalum oxide film, and $E_1$ is the field determined using method 1. $N_{s2}$ is the refractive index and $E_2$ is the field calculated by method 2 with $N_{s2}=2.27$-2.50. $E_3$ is the field calculated by method 3 with $N_{s3}=2.186$, and $N_{s3}=N_{s2}$. $E_Y$ is the results from Young's work (1960) using spectrophotometric method.
on a substrate, so any source of error from a sequence of measurements at various thicknesses will result in a deviation of the substrate refractive index. However, the choice of the substrate refractive index affects the estimation of film thickness. For example, in Table 6.1 run 13, a 7% increase in the $N_s$ would result in a 2-3% increase in oxide film thickness. The calculated reversible potential of the cell (Ta/Ta$_2$O$_5$/solution/KCl(sat.)/Hg$_2$Cl$_2$/Hg) is -1.17 V. The zeros of overpotential extrapolated from the zero thickness of the thickness vs. voltage relation, vary from -1.26 to -1.94. These variations may be caused by the deviations of the substrate refractive index from the true values and can be a result of thicknesses which are 0.4-2% higher than the actual thickness. In run 12 and run 13, the same specimen, under the same chemical treatment, was used to grow oxide. The ellipsometry data were measured on different sides of the specimen on different runs. The surface for run 12 was not as optically flat as that for run 13. The difference between the refractive index of the substrate was found to be about 2.7%. So surface roughness, which results in light scattering, may be one of the causes of the difference in the estimated refractive index of the substrate. The refractive index of the oxide film and substrate in run 10 was quite different from the other runs, but the field was found to be reasonable. The disagreement of the result from run 10 was suspected to be due to misalignment of the ellipsometer. The zone difference on Inconel slides has been checked again, and the difference was found to be about the same as the results of latest alignment. To verify if the crystal orientation affects ellipsometry measurement of the refractive index, a tantalum single crystal of $<100>$ orientation was used to repeat the experiment. The results show good agreement with the results measured using a polycrystalline specimen. Because tantalum has a body centered cubic crystal structure (Kittel 1986), the difference of the substrate refractive index on the tantalum of $<111>$ orientation cannot be due to the crystal orientation. Instead it must be due to some Ta/Ta$_2$O$_5$ interfacial phenomenon suggesting a difference in the surface roughness and/or structure.
The values of the substrate index for oxalic acid in table 6.1 are slightly smaller than those for sulphuric acid. The results of substrate index agreed better for oxalic acid than those for sulphuric acid. These results may relate to some special condition at the oxide and substrate interface. Smith and Young (1983) attributed the variation of the substrate index to an interfacial layer introduced during HF etching. Different amounts of a fluorine rich film left at the tantalum surface after etching in HF may result in an interfacial layer at the substrate - oxide interface. These absorbing films are about 2 nm thick.

In the second method, the variables in the numerical fitting are the refractive index of the film and the thickness of the film. The refractive indices of the substrate used in the fitting were the averaged value, except the results from the <111> - oriented specimen. In this method, the averaged indices may differ from the true value, but all the specimens were assumed to have the identical substrate surface, which is closer to reality. However, there is a trade-off between the least deviation by varying all the parameters and a higher standard deviation by varying fewer parameters. The differential fields determined using the second method are listed under $E_2$ in Table 6.1. The values of $E_1$ and $E_2$ are almost identical. The maximum difference in run 13 is less than 1%.

The film refractive indices ($N_{qr}$) shows random variations within 1% over a current density range of 500. In order to check how significantly the fixed refractive index of the film will affect the results, in the third method, both the refractive index of the film and substrate are the averaged values, except the results for the <111> - oriented sample which were obtained from
Figure 6.1 Steady state lines. The solid line shows the results predicted from Eq. 1.2 (Page 2). The smaller dashed line, fitted to the circles (○), represents the results measured using 0.1M sulphuric acid as the electrolyte solution. The larger dashed line, fitted to the squares (□), represents the results measured using 0.1M oxalic acid as the electrolyte solution.
Chapter 6 Electrolyte Incorporation Studies - Steady State

the first method. Only the thicknesses of the film are the variables in the fitting. The field determined using this method is shown as \( E_3 \) in Table 6.1. Again, \( E_3 \) was found to be identical to \( E_1 \) or \( E_2 \). Since the fields determined from method 1 to method 3 agree with one another, the relation of \( \log_{10} J \) vs. \( E \) shown in Fig. 6.1 is plotted based on \( E_1 \).

The data points in Fig. 6.1 have to be fitted to a quadratic expression. The data with square and circular symbols are the results in oxalic acid and sulphuric acid, respectively. The data fitted to quadratic relations are shown as the dashed lines. The solid line represents the results from Young (1960) and Young’s field results are listed as \( E_Y \) in Table 6.1. The fitted quadratic relations are:

Young’s results:

\[
\log_{10}\left( \frac{J}{mA/cm^2} \right) = -0.284 E^2 + 5.93 E - 25.8
\]

Present results using 0.1M sulphuric acid:

\[
\log_{10}\left( \frac{J}{mA/cm^2} \right) = -0.412 E^2 + 7.25 E - 29.2
\]

Present results using 0.1M oxalic acid:

\[
\log_{10}\left( \frac{J}{mA/cm^2} \right) = -0.414 E^2 + 7.20 E - 29.04
\]

where \( E \) is in MV/cm. According to the report by Young (1958), the dispersion relation for a tantalum oxide film formed at 25°C with a current density 10 mA/cm\(^2\) is given by \( N_f = 2.14 + 0.292/(\lambda/10^3\text{Å}-2.305)^{1/2} \). Our refractive index for the film is \( 2.185 \pm 0.005 \), which agrees well with the dispersion relation, \( N_f = 2.195 \pm 1\% \) (in the range of 2.173 to 2.217) with \( \lambda = 6328\text{Å} \). The present steady state line for sulphuric acid falls on the higher field side of Young’s curve at
higher current density. However, the measured fields for sulphuric acid agree to within about 1% of Young's results, which is about the tolerance of the dispersion relation. The interesting phenomenon is that the electric fields measured in oxalic acid are always larger than those measured in sulphuric acid by about 1-2%.

6.3.2 Comparison with Results Using ac+dc Field Method

Based on Eq. 5.2.2.7, the endpoint of the arc at the low frequency end shown in Fig. 4.6(c) is equal to the \( \frac{\text{dlog}J}{\text{dE}} \). The endpoints of arcs at the low frequency ends, measured in Chapter 4 using 0.1 M sulphuric acid, are shown as circles in Fig. 6.2. The data are fitted by line 1 in Fig. 6.2. The fitted expression for Young's steady state results (1960), Eq. 6.3.1.1, is plotted as line 2. Line 3 is from Eq. 6.3.1.2 and line 4 is from Eq. 6.3.1.3. The length of each line represents the range of the measurement. The four square symbols in Fig. 6.2 represent the results at four different dc current densities using the ac+dc field method in oxalic acid. Line 1 can be expressed (in the same units as Eq. 6.3.1.1) as:

\[
\log_{10}(\frac{J}{mA/cm^2}) = -0.407 E^2 + 7.28 E + \text{constant}
\]

6.3.3 Permittivities of the Films

A Wien capacitance bridge, which means that the measured impedance is balanced against a series of resistors and capacitors, was built to measure the series capacitance from the tantalum anode to the platinized platinum cathode. The construction details are shown in Smith's thesis (1981, p138). A standard capacitor of 0.1 \( \mu \)F with \( \pm 1\% \) error was measured by both the bridge
Figure 6.2 Comparison of the steady state curvature. The circles (○) are the results measured in sulphuric acid and the squares (□) are the results measured in oxalic acid using the ac+dc field method. Line 1 is the fitted results for the circles, line 2 is from Eq.6.3.1.1, line 3 is from Eq.6.3.1.2, and line 4 is from Eq.6.3.1.3.
and a Phillips PM6303 digital RCL meter. The measured value of the standard capacitor is 0.0993 μF by the bridge and 0.0999 μF by the RCL meter at 1kHz. The series resistance ($R_s$) measurement from the RCL meter fluctuated substantially due to the very low electrolyte resistance (within 10Ω) which was close to the measurement limit of the meter. The difference in series capacitance measurement were within 0.5% by the bridge and the RCL meter during the run. The series capacitance ($C_s$) and resistance were measured by the bridge with 500 Hz, 1kHz, 2kHz frequency. The electrolyte resistances extrapolated from the $R_s$ vs. 1/f plots were (4.63±0.34)Ω for sulphuric acid and (8.16±0.74)Ω for oxalic acid. The loss tangent ($\tan\delta$) of the film grown in sulphuric acid is 0.009±0.002 and that in oxalic acid is 0.009±0.002. The dielectric constants were determined from the slope of $1/C$ vs. $d$ plots shown in Fig.6.3, where C is the capacitance at 1KHz and d is the film thickness. The dielectric constant for the film grown in sulphuric acid is 26.9(2)±0.2 and that for film grown in oxalic acid is 26.8(5)±0.3, by using the thickness determined using method 1. The dielectric constant of the film grown in sulphuric acid seems to be higher than that of the film grown in oxalic acid. The plot of voltage vs. reciprocal capacitance (at 1kHz) is shown in Fig. 6.4. The zeros of the overpotential extrapolated at zero reciprocal capacitance from Fig.6.4 are -0.99 V and -1.09 V from the results measured in oxalic acid and sulphuric acid, respectively.

6.4 Summary

The estimated substrate refractive indices varied within 2% from run to run. The cause of the difference might be attributed to the chemical surface treatment of the substrate, surface roughness, and the statistical approach used in extrapolating ellipsometry data. However within the range of variations, the substrate refractive index did not affect the values of differential
Figure 6.3(a) Dependence of reciprocal capacitance on the film thickness.
Figure 6.3(b) Deviations from the fitted lines shown in Fig6.3(a).
Figure 6.4(a) Dependence of the reciprocal capacitance on the potential.
Fig. 6.4(b) Deviations from the fitted lines in Fig.6.4(a).
fields. The differential fields determined using ellipsometry with a film grown in 0.1M sulphuric acid showed less than 1% difference from Young's results (1960). The curvature of the steady state line was not caused by the variation in refractive index of the oxide film, which changes less than 0.5% at different current densities. The relations of logJ vs. E for films grown in diluted sulphuric acid and oxalic acid exhibit curvature. The films grown in oxalic acid consistently show higher field than those grown in sulphuric acid. The dielectric constants for films grown in oxalic acid and sulphuric acid at 0.003 mA/cm$^2$ are about the same. The curvature of the steady state lines measured using ellipsometry agrees with that measured using the ac+dc method to within 1% in sulphuric acid. The curvature determined using the ac+dc method in oxalic acid shows a higher curvature. The steady state curvature is slightly higher for films grown in oxalic acid than for those grown in sulphuric acid. In the previous work of Dell'Oca and Young (1970), phosphate incorporated films tended to have higher fields at the same current density. Young's work (1960) also has shown that the field in the oxide tended to become larger as the film grown in more concentrated sulphuric acid. Based on these previous results, the present work showed that the oxalic acid was incorporated into the film. However, the optical properties and dielectric properties of the films do not show resolved difference for films grown in either electrolyte solution.
Chapter 7
Electrolyte Incorporation Studies
- Stepped Current Method

7.1 Motivation

Bean et al.'s theory (section 2.2.1) suggested that the ionic conduction was due to interstitial metal ions whose concentration changed with field. The ions were pulled out of lattice sites and placed in interstitial sites during high field ionic conduction, i.e. by the high field production of Frenkel defects. The stepped field line, defined in section 2.1.6, should represent the ionic current with a fixed concentration of mobile ions based on this model. As explained in Chapter 1 (page 2), Young (1961) studied the initial slope of the stepped field lines starting at different initial steady state fields. He found that the slope depended on the initial field. Since both the production of Frenkel defects and the mobility process are essentially similar, it was
natural to fit both by an expression of the type of Eq. 2.1.5.1. However in later work by Smith and Young (1982), it was found by using the stepped current method (described in section 2.1.6) that the relation between peak field and stepped current was linear, as had been reported by Dewald (1957) using this method. The slope of peak field vs. stepped current was dependent on the initial steady state field. Meanwhile, Dell'Oca and Young (1970) proposed that the curvature of the steady state line was due to the two layer structure of the films. The outer layer grew by metal ion movement and contained incorporated electrolyte which modified the permittivity and hence the effective field causing ionic transport. The question is whether the dependence of the slope of the stepped field line on the initial steady state field is not also due to the effects of incorporated electrolyte. The test for this is to use an electrolyte which is not incorporated. In the present work, the stepped current experiments were repeated in 0.1 M sulphuric acid and 0.1 M oxalic acid.

7.2 Experimental

The stepped current experiment is similar to Smith and Young's work (1982). The schematic diagram is the same as that in Smith (1981, p.184). The DANA digital volt-meter with input impedance $>10^4$ MΩ was used as a high input impedance buffer. In the present set-up, the analog output from the DANA was connected to the A/D channel of a DT2828 board via an isolator with unity gain. The stepped current transients were measured using the DT2828 A/D board with 12-bit resolution. A pair of back-to-back 10-volt Zener diodes was used to protect the board. A standard cell was used to calibrate the system. The accuracy of the system was within ±0.1%. Current stepping was controlled by a mercury-wetted relay with a response time of less than 1 ms, which was connected with an external resistor in parallel with the internal
resistor of the current source. The clock speed of the DT2828 board was 4 MHz ±0.01%. The direct memory access (DMA) function of the DT2828 board was used to trigger a transient with a speed of 1000 data/second. A typical triggering time was 1 second.

7.3 Stepped Field Results

7.3.1 Method of Analysis

The measured potential and time were converted to fields and charge passed. The increase in the film thickness during the transient could not be calculated by using Faraday's Law, because the displacement current was involved in the process. The equation for the oxide thickness, $x(t)$, proposed by Smith et al. (1982b) is given by

$$x(t) = x(0) + \left( \frac{\eta M}{10 F \rho} \right) \int (J_2 - J_c) \, dt$$

7.3.1.1

where $J_c = \frac{d[C(0)V(t)]}{dt}$, $C(0) = \varepsilon_0 \varepsilon_x x(t)$, $x(0)$ is the oxide thickness just before stepping the current, $\rho$ is the density of the oxide (we use 8 g/cm$^3$), $M$ is the molecular weight of tantalum pentoxide (441.76 g/mole). $\eta$ is the current efficiency (taken as 1), $F$ is Faraday's constant, $J_c$ is the displacement current, and $(J_2 - J_c)$ is the actual ionic current. Since the sampling interval was very small, integration and differentiation could be approximated discretely by rewriting Eq. 7.3.1.1 as

$$x(t) = x(0) + \frac{\eta M}{10 F \rho} \sum \left[ J_2 - \frac{C_k V_k - C_{k+1} V_{k+1}}{\Delta t} \right] \Delta t$$

7.3.1.2
where \( C_k = \varepsilon_c \varepsilon_0 / x_{k+1}(t) \).

The values of \( V_k \) were from the transient data. The initial oxide thickness, \( x(0) \), was determined by \( x(0) = (V(0) - V_{rev}) / E_1 \) and \( E_1 \) was calculated from the steady state line from Fig. 6.1. The reversible potential of the cell (same as section 6.3.1), \( V_{rev} \), is -1.17V.

7.3.2 History Effect

A set of typical measurements is shown in Fig. 7.1 with \( J_1 = 0.0067\text{mA/cm}^2 \) and \( J_2 = 0.854\text{mA/cm}^2 \), at 25°C in oxalic acid. The initial 30 milliseconds in Fig.7.1(a) represents the steady state stage. The current was stepped at 30 milliseconds after the DMA was started. The potential of the cell responded to the abrupt increase of current density with an overshoot then eased to a new steady state value. The first 100 data points during the transient were converted to field (E) and ionic current density (J) vs charge passed using the numerical method outlined in section 7.3.1. The results, shown in Fig.7.1(b), demonstrate a delayed response of the ionic current density to field.

7.3.3 Stepped Field Lines

The relationships of \( \log J_2 \) vs. \( E_p \) measured in two electrolyte solutions are shown in Figs. 7.2 and 7.3. The steady state field used in the analysis was calculated from the fitted steady state results in Eqs. 6.3.1.2 and 6.3.1.3. The slopes of the stepped field lines shown in both Figs.7.2 and 7.3 depend on the initial steady state field as reported by Smith et al. (1982). The slopes of the stepped field lines vs. the steady state fields (\( E_i \)) are plotted in Fig. 7.4 as the larger
Figure 7.1(a) Measured data for film grown in oxalic acid.
Figure 7.1(b) The current density and field response during the stepped field experiment.
Figure 7.2 The stepped field lines measured in sulphuric acid. The dashed curve represents the steady state line measured in sulphuric acid from Chapter 6. Triangles are the measured values from the stepped field experiment and circles are the measured data from the steady state measurement.
Figure 7.3(a) The stepped field lines measured in oxalic acid. The dashed curve is the steady state line. All the symbols are measured data.
Figure 7.3(b) The deviations from the fitted lines in Fig. 7.2 and Fig. 7.3(A). The symbols correspond to the symbols in Fig.7.2 and Fig.7.3(A).
Figure 7.4 The dependence of stepped field slope on the steady state field. The triangles represent data measured in sulphuric acid and the squares represent data measured in oxalic acid. The big symbols represent results from the stepped current experiment. The small symbols represent results from the ac+dc field method.
symbols. The stepped field slopes calculated from the ac+dc field experiment are plotted as the smaller symbols. Both results show a linear dependence on the steady state field. The results exhibit no differences for oxide grown in oxalic acid solution (shown as squares) and for oxide grown in sulphuric acid solution (shown as triangles). The results obtained using the ac+dc field method show lower values of the stepped field slope.

7.3.4 Permittivity of the Oxide Film

The permittivity of the oxide can be determined from the stepped field results, since the initial change of voltage is due to the displacement current charging up the capacitor. The displacement current ($J_c$) is given by

$$J_c = J_2 - J_1 = \left. \frac{dCV}{dt} \right|_{t=0} = \varepsilon_r \varepsilon_0 \left. \frac{dE}{dt} \right|_{t=0}$$

7.3.4.1

where $C$ is the capacitance of the oxide film at $t=0$, $\varepsilon_r$ is the dielectric constant of the film, and $\varepsilon_0$ is the permittivity of free space. Figure 7.5(a) shows one of the data sets used to extrapolate and obtain the dielectric constant of an oxide film which has been grown in sulphuric acid. The estimate of the dielectric constant was 26.5 in Fig. 7.5(a). Figure 7.5(b) shows the deviation from the fitted line. The dielectric constant of film grown in sulphuric acid was estimated as $27.2 \pm 1.5$ and that of film grown in oxalic acid as $26.5 \pm 1.4$. The dielectric constants are, on average, 2% higher for films grown in sulphuric acid than for the films grown in oxalic acid. The results agree with the dielectric constants measured in chapter 6.
Figure 7.5(a) The solid line is the straight line fitted to the first 20 data points. The circles are the data.
Figure 7.5(b) The deviations from the fitted line in Fig. 7.5(a).
7.4 Summary

Stepped current experiments were performed in 0.1 M sulphuric acid and 0.1 M oxalic acid. It was found that the relationships between $\log J_2$ and $E_p$ are linear and their slopes depend linearly on the steady state field in both electrolytes. No difference was found in the dependence of the stepped field slope on the steady state field for these electrolytes. In comparison with the stepped field results with the results using the ac+dc field method, it is found that the stepped current slopes have a stronger linear dependence on the steady state field. The stronger dependence of the steady state field may be due to the non-linearity between field and current density, which does not appear in the ac+dc field experiments. The linear $\log J_2$ vs. $E_p$ relation suggests that the quadratic term in the exponential function of Eq. 2.1.5.1 is caused by electrolyte incorporation.
Chapter 8

Summary and Conclusions

The frequency response of the high field ionic conduction process involved in the growth of anodic oxide films on tantalum has been studied in this work. The results are given as admittivity, which is defined as the complex ratio of ac current density to ac field. After correcting for the displacement current density and the Faradaic capacitance, plots of the real vs. the imaginary part of the admittivity were arcs of circles with centres above the real axis. These were fitted to a Cole-Cole expression: \( A + B / [1 + (j \omega \tau_m)^{1-\alpha}] \). This result was interpreted as demonstrating a distribution of relaxation times in the process which produces the history effect rather than a single relaxation time or two distinct relaxation times as proposed in previous work. A distribution of relaxation times is considered to be a reasonable expectation for the model by Young et al. (1992, 1979 and 1983) in which the history effect is due to structural change in an amorphous oxide. The relaxation time \( \tau_m \) was measured over a range of steady state current densities at different temperatures and was found to be close to inversely proportional to the
current density independent of temperature. This is consistent with a current density-driven process.

Experiments were done to test whether the curvature of the steady state line, and the dependence of the slope of the stepped field lines on the initial steady state field, are due to electrolyte incorporation. Oxalic acid was used since there was evidence from XPS that it is not incorporated. Films grown in both 0.1M sulphuric acid and 0.1M oxalic acid showed single layer structure by ellipsometry. The electric fields for films grown in 0.1M sulphuric acid showed less than 1% difference from the previous work. Curvatures of the logJ vs. E relationships were demonstrated for films grown in both 0.1M oxalic acid and 0.1M sulphuric acid. Films grown in oxalic acid result in higher field at the same current density. The lower oxide permittivity for films grown in oxalic acids indicates that in fact oxalic acid was incorporated into the film. The results were inconclusive as to whether the curvature of the steady state line and the dependence of the stepped field line slope on the steady state field are caused by the electrolyte incorporation. However, Shimizu et al. (1996) observed no incorporation in films grown in 1M halide electrolytes using a SIMS depth profiling technique. Further tests using halide electrolyte solutions will yield more information about the Tafel slope with no electrolyte incorporated into the films. The previously reported linear stepped field logJ vs. E relation and the dependence of the slope (dlogJ/dE) on the steady state field has been confirmed using the stepped current method with films grown in 0.1M sulphuric acid and 0.1M oxalic acid in the present work. The linearity of the logJ vs. E stepped field line is surprising in view of the curved logJ vs. E steady state line, and further work is required to test whether the linearity of the stepped field logJ vs. E relation is due to a failure of the approximation involved in deducing stepped field data from stepped current experiments.
References


Davies J.A. and Domeij B. J. Electrochem. Soc. 110, 849 (1963)

Davies J.A., Domeij B., Pringle J.P.S. and Brown F. ibid. 112, 675 (1965)

Dell'Oca C.J. and Young L. Applied Physics Letters 13, 228 (1968)
References

Dell'Oca C.J. and Young L. Surface Science, 16, 331 (1969)


Dell'Oca C.J. and Young L. ibid, 117, 1548 (1970b)


Dignam M.J. ibid. 112, 729 (1965)


Elliott S.R. Advances in Physics, 36, p.164 (1987)


Goad D.G. and Dignam M.J. ibid., 50, 3267 (1972)


References


Pringle J.P.S. J. Electrochem. Soc. 119, 482 (1972)

Pringle J.P.S. ibid. 120, 1391 (1973)

Pringle J.P.S. ibid. 120, 398 (1973)


Smith D.J. and Young L. J. Electrochem. Soc. 129, 2513 (1982a)

Smith D.J. and Young L. ibid. 129, 2518 (1982b)


Taylor D.F. and Dignam M.J. J. Electrochem. 120, 1299 (1973)

Vermilyea D.A. J. Electrochem. Soc. 102, 655 (1955)


Vermilyea D.A. ibid. 104, 212 (1957b)
References


Young L. Tran. Faraday Soc. 50, 153 (1954)


Young L. Tran. Faraday Soc. 55, 632 (1959)


Young L. J. Electrochem. Soc., 111, 1289 (1964)


Young L. Can. J. Chem. 50, 574 (1972)

Young L. J. Electrochem. Soc. 124, 528 (1977)

Young L. and Smith D.J. ibid. 126, 1972 (1979a)

Young L. and Smith D.J. ibid. 126, 765 (1979b)

Young L. and Smith D.J. ibid. 130, 408 (1983)

Young L. Trends in Electrochemistry, 1, 259 (1992)


Young L., Yang T., and Backhouse C. ibid., 142, 3483 (1995)
Appendix A

Ellipsometer Alignment and Zone Differences

The optical components in an ellipsometer are shown in figure A.1. The laser emits a linearly polarized light which is converted into circularly polarized light after passing through a quarter wave plate. The circularly polarized light passes through a polarizer and becomes linearly polarized. Elliptically polarized light is generated after the linearly polarized light passes through another quarter wave plate. The elliptical light is reflected from the surface and becomes linearly polarized, at null, which enters into the analyzer prism at 90°. Therefore, no light signal is sensed on the photodetector and the ellipsometer reaches its null setting. The following equation, which represents the state of the light after passing through each optical component, can be derived using the Jones matrices, i.e.,

\[
\begin{pmatrix}
    0 \\
    0
\end{pmatrix}
= \begin{pmatrix}
    \cos^2 A & \cos A \sin A \\
    \cos A \sin A & \sin^2 A
\end{pmatrix}
\begin{pmatrix}
    \frac{r_p}{r_s} & 0 \\
    0 & \frac{r_s}{r_p}
\end{pmatrix}
\begin{pmatrix}
    1 & -i & \frac{\cos^2 P \sin P \cos P}{1} & -i \\
    -i & 1 & \frac{\sin P \cos P}{\sin^2 P} & 1
\end{pmatrix}
\begin{pmatrix}
    1 & -i \\
    i & 1
\end{pmatrix}.
\[
\frac{r_p}{r_s} = \tan \psi e^{i\Delta}
\]

where \(r_p\) is the reflection coefficient for light polarized parallel to the plane of incidence, and \(r_s\) is the reflection coefficient for light polarized perpendicular to plane of incidence. \(\tan \psi\) is the relative amplitude diminution, and \(\Delta\) is the relative phase retardation. The matrices, that were shown on the right hand side of the equation, represent, from right to left, laser, quarter wave plate at -45° with respect to the plane of incidence, polarizer, quarter wave plate, specimen surface, and analyzer, respectively. The light to the photodetector is extinguished when the values of \(P, A, \psi, \Delta\) satisfy the following relations:

\[
\psi = A \\
\delta = 2P \pm \frac{1}{2} \pi
\]

Our self-nulling ellipsometer shown in Fig. A.1, determines the null using a statistical method. The stepping motor (s.m.) of the polarizer prism first makes a few degrees of sweep, while in the mean time, the pc reads in a photo signal from the detector. The null is found by fitting the photo signal vs. the angle of prism using Chebyshev polynomials. The same rules are applied to the analyzer prism.
Figure A.1 The computer controlled ellipsometer consists of a laser source (L), a linear polarizer (P), a quarter wave phase retarder (Q.W.P), a linear analyzer (A), and a photodetector (p.d.). The computer interfacing part includes: the shaft encoder that reads the angle of the polarizer and analyzer prisms, DT2828 ADC that reads in the photo signal, PXB721 digital I/O board that reads in the prism angle from the shaft encoder, and outputs electrical signal to move the stepping motor, as well to change the gain setting of the photodetector.
Appendix A Ellipsometer Alignment and Zone Differences

A.1 Alignment of Laser

The analyzer arm was set to a straight through position. The He-Ne laser is on top of an adjustable stand with three screws which are used to adjust the tilting of the laser beam. There are two apertures on the analyzer arm: one is attached to the compensator, the other is attached to the polarizer. The apertures were opened wide. A piece of white paper was used to see the light from the laser after passing through the two apertures. The periphery of the aperture can be seen on the paper due to the reflection of light from the laser source. The laser stand has to be adjusted so that the brightest spot of the laser beam is on the center of the aperture. When the spot is located at the center of both apertures, the laser beam is very sharp with two apertures turned into pin holes and the beam is aligned with the polarizer's arm.

The quarter wave plate (Q.W.P.) attached to the laser was adjusted by aligning the line scratch on the laser and that on the compensator. A minor adjustment around the scratch has to be done. The goodness of the circularly polarized light was detected by rotating both polarizer and analyzer prism 360° without the compensator (Q.W.P.). The result is demonstrated in Fig. A.2. The photo-signal in arbitrary units was fitted to a sine wave. There are two maxima and two minima with a maximum to minimum ratio of about 1.4. The results show an elliptically polarized light which is not far from a circularly polarized light. The cause of the deviation from circular polarization may be the non-ideality of the laser source and the Q.W.P. However, this deviation should not affect null positions, but only change the intensity of the light at various
Figure A.2 The photo signal was measured at the least gain of the photodetector interfaced with pc via the DT2828 ADC.
Appendix A Ellipsometer Alignment and Zone Differences

polarizer angles.

A.2 Alignment with the Plane of Incidence

There are a few methods that were used by Smith (1981, page 78) to align the optical devices with the plane of incidence. An inconel slide was used to reflect the light. The results are shown in Figure A.3. The same method was used at a few different angles of incidence. The plane of incidence at various angles of incidence is shown in Table A1. The plane of incidence was found to be parallel to the fast axis of the compensator at the compensator angle of 1.13° by Cornish (1976). The polarizer is set to 0° and the analyzer is set to 90°, so that the photodetector detects a null when the fast axis is parallel to the plane of incidence at an arbitrary angle of incidence.

<table>
<thead>
<tr>
<th>Angle of incidence</th>
<th>P_{ref}</th>
<th>A_{ref}</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>-0.025</td>
<td>2.82</td>
</tr>
<tr>
<td>62.77</td>
<td>0.13</td>
<td>2.68</td>
</tr>
<tr>
<td>65</td>
<td>0.21</td>
<td>2.62</td>
</tr>
<tr>
<td>70</td>
<td>0.43</td>
<td>2.45</td>
</tr>
</tbody>
</table>

Table A1  \( P_{ref} \) represents the real zero of the polarizer and \( A_{ref} \) represents the real zero of the analyzer prism.

A.3 Alignment of the Compensator

The next step in the alignment is to adjust the micrometer attached to the Babinet-Soleil compensator to give an exact quarter wave retardation. The zone difference between zones 1 and 3 is assumed to be due to the errors in the quarter wave retardation. Therefore, the transmittance
Figure A.3 Fixed analyzer line was generated by fixing analyzer at an angle around 90° and balancing the polarizer at around 0°. The same principle was applied to the fixed polarizer line.
(T_c) and phase retardation (Δ_c) can be estimated using the zone difference according to p87 in Smith’s thesis (1981). Table A2 shows the positions of the micrometer and the value of T_c and Δ_c.

<table>
<thead>
<tr>
<th>Angle of incidence</th>
<th>Micrometer setting</th>
<th>T_c</th>
<th>Δ_c</th>
</tr>
</thead>
<tbody>
<tr>
<td>70°</td>
<td>26.8</td>
<td>1.015</td>
<td>78.25</td>
</tr>
<tr>
<td></td>
<td>26.9</td>
<td>1.016</td>
<td>83.78</td>
</tr>
<tr>
<td></td>
<td>27.0</td>
<td>1.017</td>
<td>89.73</td>
</tr>
<tr>
<td></td>
<td>27.1</td>
<td>1.004</td>
<td>90.87</td>
</tr>
<tr>
<td>62.77°</td>
<td>27.03</td>
<td>1.011</td>
<td>89.76</td>
</tr>
<tr>
<td></td>
<td>27.03</td>
<td>1.011</td>
<td>90.40</td>
</tr>
<tr>
<td></td>
<td>27.0</td>
<td>1.011</td>
<td>90.93</td>
</tr>
<tr>
<td></td>
<td>27.0</td>
<td>1.011</td>
<td>89.94</td>
</tr>
<tr>
<td></td>
<td>27.0*</td>
<td>1.004*</td>
<td>89.88*</td>
</tr>
</tbody>
</table>

Table A2 The status of the quarter wave retardation. (* The latest check.)

A.4 Alignment of the Cell Windows and Specimen

The most time consuming part of the alignment is to align both the cell and the specimen. It often happened that the specimen was aligned but the cell window was not. Then the results will not be meaningful. One of the technical difficulties is due to the non-transparent cell; so the light travelling inside the cell is not visible. The other technical difficulty is that the flexibility of moving the cell is very small due to the small size of the window which is only about 1.5cm in diameter. The arc between the two cell windows is 125.54°, so the angle of incidence is 62.77°. The alignment of the cell windows starts with the laser on. There are two degrees of freedom for the cell. One involves motion along the normal direction to the specimen surface. The other involves rotation around the direction of the plane of incidence. The light incident on the window facing the polarizer arm has to be at normal incidence. The reflected
light from the window has to be nearly incident on the pin hole of the compensator aperture. By adjusting the two degrees of freedom, the center of the other window can roughly be along the analyzer arm. Then, the position of the cell is roughly settled. The specimen is mounted onto the stand in the cell. The hole of the cell cap for the specimen insertion is about 1.5 cm in diameter. The position of the specimen has to be in about the center of the hole which removes an additional two degrees of freedom. The specimen can only have very minor adjustment along the direction normal to the specimen surface and the direction parallel to the plane of incidence. The other two degrees of freedom are rotation around the direction of the plane of incidence and tilting the surface about the direction of parallel to the incidence plane. If major changes were required in these two directions, the rod, which supported the specimen, was twisted to reach the alignment until the reflected light could be detected in the photodetector. The final adjustment can be done by putting cell and specimen in place.

A.5 Zone Differences

Table A.3 shows the ellipsometry results from a series of oxide growth experiments at about 5 volt intervals with a current density 1.75 mA/cm². $A_1-A_3$ represents the zone 1 and zone 3 analyzer angle difference. $P_1-P_3$ represents the zone 1 and zone 3 polarizer angle difference.
### Appendix A Ellipsometer Alignment and Zone Differences

<table>
<thead>
<tr>
<th>Phi</th>
<th>A₁-A₃</th>
<th>Del</th>
<th>P₁-P₃</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>29.500</td>
<td>-0.490</td>
<td>101.994</td>
<td>-0.022</td>
<td>5.460</td>
</tr>
<tr>
<td>31.457</td>
<td>-0.589</td>
<td>94.179</td>
<td>0.094</td>
<td>10.366</td>
</tr>
<tr>
<td>34.299</td>
<td>-0.729</td>
<td>87.793</td>
<td>0.123</td>
<td>15.358</td>
</tr>
<tr>
<td>37.864</td>
<td>-0.616</td>
<td>83.892</td>
<td>0.056</td>
<td>19.880</td>
</tr>
<tr>
<td>44.336</td>
<td>-0.621</td>
<td>82.439</td>
<td>0.201</td>
<td>25.171</td>
</tr>
<tr>
<td>55.482</td>
<td>-0.488</td>
<td>86.173</td>
<td>0.296</td>
<td>30.280</td>
</tr>
<tr>
<td>71.935</td>
<td>-0.691</td>
<td>104.777</td>
<td>0.627</td>
<td>35.280</td>
</tr>
<tr>
<td>77.380</td>
<td>-1.236</td>
<td>199.224</td>
<td>0.621</td>
<td>40.117</td>
</tr>
<tr>
<td>58.658</td>
<td>-1.767</td>
<td>235.679</td>
<td>0.113</td>
<td>45.320</td>
</tr>
<tr>
<td>45.843</td>
<td>-1.976</td>
<td>240.651</td>
<td>-0.073</td>
<td>50.250</td>
</tr>
</tbody>
</table>

Table A3 Zone differences measurement.
Appendix B

Calibration of ac+dc Experimental Set-up

A step by step calibration of the system, shown in Fig. 4.2, was done. Dac channels 0 and 1 were calibrated by sending voltages ranging from -10V to 10V, to each channel. The output from each channel was checked using a differential voltmeter (d.v.). The percentage differences between input and output signals were compared in Fig. B.1. Since the denominator was close to zero, the percentage differences rose at around zero. The calibrations of adc channel 0 and 1 were done by applying voltages between -10 to 10 volts to these channels. The input signals were measured by a d.v. The readings from the adc channels were measured by a pc. The potential differences were plotted in Fig. B.2. The signals sent from the dac channel were amplified by the op amp circuit board. The output signals from the board were the potential applied to the cell. The output signals were checked by both a d.v. and the readings from the adc0. The differences were compared in Fig. B.3 as * symbols. The data □ were the percentage differences between the signal predicted by the pc and the actual output voltage. Since the lower

110
voltage results were in greater error, our experiments were run from 10 to about 45 V. Voltages in the range of 0 to 10 V were applied to the adc1. The applied signals were checked by the d.v. The voltage readings converted by the program were compared in Fig. B.4. A range of currents, from 0.01mA to 1mA were measured by way of an A/V converter. The measured voltage values were compared to the applied voltages to the test resistor as shown in Fig. B.5.
Appendix B Calibration of ac+dc Experimental Set-up

Figure B.1 X-axis represents the measured values from the differential voltmeter (d.v.). Y-axis represents the percentage differences between dac0/dac1 and differential voltmeter.

* represents the data measured from dac0. □ represents the data measured from dac1.
Figure B.2. X-axis is the value measured by d.v. Y-axis is the percentage difference of adc1/adc0 and d.v. measurement.
Figure B.3 X-axis is DC output voltage measured by the differential voltmeter. Y-axis is the percentage differences between dc input (□) / adc0 (*) and measurement from the d.v.
Figure B.4 X-axis is the voltage measured from adcl. Y-axis is the percentage differences between adcl reading and d.v.
Figure B.5 Calibration of the current signal. Dac voltage is the output voltage to resistors (100k \(\square\), and 1M \(\diamondsuit\)). The current signal through I/V converter was measured by adc1. The percentage difference calculates the difference between the output voltage value estimated by pc and the measured current multiplied by resistance.
Appendix C

Impedance Measurement Programs

C.1 Flowchart

A description of the program has been given in Chapter 4. The program starts executing by typing a command line: " imped (startV) (stopV) (Idens) (Vprel) (linc) (Vrev)".

The ac signal starts to be superimposed on the dc voltage at "startV". The measurements stop at voltage "stopV". Idens is a given dc current density. Vprel is the given voltage range before the ac signal is superimposed. Linc is the given linearity of the ac current density vs. the ac field. Vrev is the reversible potential of the cell.

Flow of the routine:

1. Calculate the overfield, thicknesses, ramp rate and the approximate time (D) required for the run specified. Print out the maximum and minimum frequencies that are accessible. \( F_{\text{max}} \approx 8000/(D \times 6) \) where the 6 results is from the requirement of 3 x Nyquist sampling; \( F_{\text{min}} \approx 10/D \).
2. Perform a precise determination of the run duration by performing a dummy run in the
Appendix C Impedance Measurement Program

subroutine "Tune". Adjust the ACStopV in order to maintain the originally intended ramp rate.
The degree of correction is small for low frequencies but becomes large as the computing
overhead per point become comparable to the desired sampling time (~100Hz).

3. Calculate the DC and AC signals and store them in integer form. The AC frequencies are
taken from a list in the program or from a list entered from the keyboard. The program will
ignore any frequencies for which it cannot achieve 3x Nyquist sampling for 10 periods, i.e. Fmin
< f < Fmax.

4. Ramp up from ACStartV-Vprel to ACStartV without any AC signals and record the voltage
and current during this time in the file 'preramp.dat'.

5. Loop through the values: Output: DAC0, DAC1. Input: ADC0, ADC1, ADC2, ADC7.

6. Store the data in the file IM#.dat. Where # in the run # and is updated and stored in the file
'impruns.dat'. A record of the run and its parameters is appended to the log file 'implog.dat'.

C.2 Source Codes of Imped program

/* IMPEDnx : perform impedance measurements
Usage:
command line: imped StartV StopV Idens Area Vprel Line Vrev
Given the above parameters the program will estimate the duration of
the run and hence the minimum and maximum frequencies which can be
applied. The user will be shown these and asked whether to proceed.
ACStartV: Voltage at which to turn on the AC signal
ACStopV: " " " " off " "
Idens: Current density (A/cm²)
Area: cm²
Vprel: Starts DC ramp at ACStartV - Vprel
-Linc: Linearity factor: (Jmax + Jmin) / (2 * Javg)
Vrev: Reversible potential
definitions:
Vps: is the output of the circuit which adds the AC + DC signals
DAC0: the DAC output connected to the DC input to the op-amp circuit
DAC1: " " " " " " AC " " " "
ADC0: the ADC connected to the voltage divider attached to Vps
ADC1: " " " " " " EGG (i.e. the current)
ADC2: optional
ADC7: optional

Appendix C Impedance Measurement Program

special cases:
If ACStartV is -1 then the subroutine QuickCal records Vps for each value of DAC0 from 0 to -10.0V (corresponding to Vps being 70V to 60V) in intervals of 0.1V. These are written to the file 'caldat.dat'. After this the program exits.
If ACStartV is -2 the program is initiated with preset values of the above parameters. Intended for use with calibration runs.

NOTES:
If electrometer is to be used, return must be connected to virtual ground of EGG, not to cathode (else ground loop leads to large noise signal). Also: keep varicap set to about 50% so that its inductance damps out SCR-induced noise spikes from the heater.
If the expected current is too large (or nearly so) for the EGG to handle (i.e. on a setting of 10^-4 the maximum current that can be detected is 1 mA, generating 10V on ADC1 ), then a shunt will be called for. The shunt consists of Rh in series with the EGG current input with Rlow in parallel with both. Rh ~ 1k, Rlow ~ 100 ohms.

Pt electrode—VVVV—(EGG)—> GND
<table>
<thead>
<tr>
<th>Rh</th>
<th>Rlow</th>
</tr>
</thead>
</table>

***************************************************************

FILES:
"dacvids.scr","w" - identifies when each frequency was applied
"preramp.dat","w" - V, Vad0, Vad1 during ramp up
"implog.dat","a" - logs the parameters (including time, date) of run
"im%d.dat","w" - records the data for current run
"impruns.dat","r/w" - records the number of the last run made

Other terms:
Icrit - maximum current load that is acceptable (A), corresponds to fuse in box containing AC + DC cct
Dummy - flag whether false run (at Home) or not
Runnum - number of this run
NentrMax - number of entries in the data vectors
GN0,1 - gains to be used on ADC channels 0 and 1
ITim - initial # of seconds to pause before starting prelude ramp
Kb - Boltzmann's constant (J/K)
Temp - Absolute T
Q - proton charge */

#include <dos.h>
#include <time.h>
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#define Version 10
#define Home 0
#define TuneF 1
#define sqr(x) ((x)*(x))
#define Pi 3.14159
#define TimeLoop 0.0005 /* estimated sampling time (minimum) sec*/
#define SampIntDef 0.002
Appendix C Impedance Measurement Program

/* Numerical recipes helper-routines */
char *cgets(char *txt);
void nrerror(char error_text[]); /* flag an error */
int *ivector(int nl, int nh); /* allocate a vector */
void free_ivector(int *v, int nl, int nh); /* free a vector */
#include "nrutil.c"

/*
 ** VARIABLES AND PARAMETERS
 **
** Some calibration data follows: (see HBp54) */
int NCal; /* Number of calibration measurements */
#define NCalMx 101 /* Maximum of above */
#define CalStep 0.1
float CalV[NCalMx]; /* no more DG */
float AG; /* AC gain, about 0.1 */
float XD; /* voltage divider ratio on ADC0 (VR2) see HB-44, about 0.2 */
define Icrit 0.030

int Dummy=0, Runnum=0, NentrMax = 8000, GN0 = 0, GN1 = 0, ITim;
double Kb = 1.38e-23, Temp = 298.2, Q= 1.602e-19;
float Area = 1.0, Vrev, Linc, Rlow, Rhigh, Linearity;

/* List of frequencies to be used in each run (if possible) */
int NFreq = 34;
float FMin, FMax, Freq[40] = {
100.0, 80.0, 64.0, 51.0, 41.0, 32.0, 26.0, 21.0, 17.0, 13.0,
11.0, 9.0, 7.0, 5.0, 4.0, 3.0, 2.2, 1.8, 1.4, 1.2,
0.9, 0.74, 0.58, 0.47, 0.36, 0.3, 0.24, 0.18, 0.15, 0.12,
0.09, 0.07, 0.05, 0.03};

/*
 ** Prototype Declarations
*/
void exit(int status);
void QuickCal(void); /* check and adjust hardware calibrations */
void Prelude(int it, float Vsta, float Vsto, float Dura, float Vprel);
float SetV(float outx); /* find needed DAC0 voltage for given RAMP output */
float VSetV(float outx); /* find needed DAC0 voltage for given RAMP output */
float VFluct(float outx);/* find needed DAC1 voltage for given AC output */
void DelayLoop(float s); /* delay for desired # of ms, considering overhead */
int GetRunnum(void); /* Find the last run's number */
/* Write all the data for the current run */
void SetRunDat(int run, int NFrqs, int nen, int accQ, int ram[], int av[],
    int bv[], int cv[], int dv[], float startv, float stopv, float id, float ar,
    float vrv, float magn, float samp, float etime, float rat,
    float Vprel, float Din, float Dfin, float E);
void Mps(char txt[]); /* Pause for operator */
float VThickns(float V); /* rough estimate of thickness from V, from DS thesis */
float Thickns(float Qch); /* calculate thickness from Q using Faraday's law */
void MyDelay(int s); /* delay s seconds */
int main(int argc, char *argv[]);
void Mwt(int i, int j, int k); /* wait loop for interfacing to ATLAB */
void DA_Imm(int chan, int val); /* output integer value 'val' to DAC# 'chan' */
int AD_Imm(int chan, int gain); /* get integer value from ADC channel 'chan' with 'gain' */
Appendix C Impedance Measurement Program

float VAD_Imm(int chan, int gain); /* as above but convert to Volts */
float CVolt(int gain, int ival); /* convert an integer value to V */
float SVAD_Imm(int chan); /* smart VAD_Imm, figures out best gain */
void DAC(int i, float x); /* set DAC # i to 'x' Volts */
int DacVal(float V); /* pre-calculate integer value for 'V' volts */
float DacEqv(int i); /* calculate V equivalent of i */
float InADC(int n); /* like VAD_Imm */
int FInADC(int n, int g); /* as above but leaves as integer (faster) */
void FDAC(int i, int j); /* as above, but use pre-calculated integer value */
/* routine to perform IO for each IO channel */
void IOS(float DA1, float DA2, float *AD0, float *AD1, float *AD7); /* as above, but use pre-calculated values for DACs, calculate ADC voltages later */
void FIOS(int DA1, int DA2, int *AD0, int *AD1, int *AD7, int *AD2); /* Calculate current (A/cm2) from field (V/cm) using Zobel's data */
double JffZ(double E); /* Calculate overfield (V/cm) from current (A/cm2) from Zobel's data */
double EoverZ(double J); /* Calculate current_ss see LY: PRS 258 p496 */
double Jff(double E); /* Calculate overfield (V/cm) from current (A/cm2) using LY's data */
double Eover(double J); /* guess the correct root! */
float SetV(float outx); /* calculate+set needed DAC0 voltage for given RAMP output */
float VSetV(float outx); /* find needed DAC0 voltage for given RAMP output */
float VFhict(float outx); /* find needed DAC1 voltage for given AC output */
float NBits(float I, float EGn, int ADGn); /* calculate how many AD bits used to resolve signal */
float DrYT(float J); /* Apply DrY's test PRS 258 p497, 3V/s at 10 ma/cm2 */
void Tune(int N, float T, float StaV, float Vprel, float *StoV, float Dur); /* guess the correct root! */
void QuickCal(void); /* Derived Functions *******************/

void IOS(float DA1, float DA2, float *AD0, float *AD1, float *AD7) {
    DAC(0, DA1); DAC(1, DA2);
    *AD0 = InADC(0); *AD1 = InADC(1); *AD7 = InADC(7);
}

void FIOS(int DA1, int DA2, int *AD0, int *AD1, int *AD7, int *AD2) {
    FDAC(0, DA1); FDAC(1, DA2);
    *AD0 = SFInADC(0, GN0); *AD1 = SFInADC(1, GN1);
    *AD7 = FInADC(7); *AD2 = SFInADC(2, 2);
}

float SVAD_Imm(int chan) { int i, j; float x, y[10];
    x = VAD_Imm(chan, 0);
Appendix C Impedance Measurement Program

if (fabs(x)>0.01) { i = fabs(x/1.25); } else i=0; SVADStd = x = 0.0;
for (j=0;j<10;j++) {
  switch (i) {
    case 0: GlobGn = 3; x += y[j] = VAD_Imm(chan,3);
      /* in +/- 1.25V */
      break;
    case 1: GlobGn = 2; x += y[j] = VAD_Imm(chan,2);
      /* in +/- 1.25 -> 2.5V */
      break;
    case 2: case 3: GlobGn = 1;
      x += y[j] = VAD_Imm(chan,1); break; /* in +/- 2.5 to 5V */
    default: GlobGn = 0;
      x += y[j] = VAD_Imm(chan,0); break; /* +/- 5 to 10V */
  }
  x /= 10.0;
  for (j=0; j<10; j++) SVADStd += sqr(x - y[j]);
  SVADStd = sqrt(SVADStd)/10.0;
  return(x);
}

double JffZ(double E) { /* Calculate current (A/cm2) from field (V/cm) */
  double rv, W0 = 2.56, Gamma = 7.65e-4, J0; J0 = pow(10.0,8.05);
  rv = J0 * exp(-(W0 - (Gamma * sqrt(E)))/(Kb * Temp/Q)); return(rv); }

double EoverZ(double J) { /* Calculate overfield (V/cm) from current (A/cm2)*/
  double rv, W0 = 2.56, Gamma = 7.65e-4, J0; J0 = pow(10.0,8.05);
  rv = (W0 + (Kb * Temp * log(J/J0)/Q))/Gamma; rv = sqr(rv); return(rv); }

double Jff(double E) { /* Calculate current_ss see LY: PRS 258 p496 */
  double rv, W0 = 2.185, Alpha, Beta, J0;
  Alpha = 5.0 * 6.995e-8; Beta = -5.0 * 3.35e-15; J0 = pow(10.0,8.24);
  rv = J0 * exp((-W0 + (Alpha * E) + (Beta * sqr(E)))/(Kb * Temp/Q));
  return(rv); }

double Eover(double J) { /* Steady state */
  double disc, rv, a, b, c, W0 = 2.185, Alpha, Beta, J0;
  Alpha = 5.0 * 6.995e-8; Beta = -5.0 * 3.35e-15; J0 = pow(10.0,8.24);
  rv = (Temp/1.1605E4) * log(J / J0);
  c = -W0 - rv; b = Alpha; a = Beta;
  disc = sqrt(sqr(b) - (4.0*a*c));
  printf("Alternate E is : %g Primary is : %g\n",
    (-b - disc)/(2.0 * a), (-b + disc)/(2.0 * a));
  rv = (-b + disc)/(2.0 * a);
  return(rv); }

/*
Use predetermined table to find needed DACO value for stated Vps
*/
float LookUp(float x) { float rv; int i; i = NCal - 1;
  while ((CalV[i] > 998.0) || (i >= 0) && (x < CalV[i])) i--;
  if ((i == (NCal - 1)) && (CalV[i] == CalV[i+1]))
    rv = -i * CalStep;
  else {
    rv = CalStep * (i + ((x - CalV[i])/(CalV[i+1] - CalV[i])));
  }
  return(rv); }

/* find and set needed DACO voltage for stated voltage */
float SetV(float outx) { /* find needed DACO voltage for given RAMP output */

Appendix C Impedance Measurement Program

int ia, ib, ic, id; /* and then set it */
float rx = -LookUp(outx);
if ((rx > 10.0) || (rx < -10.0)) {
    printf("Warning: SetV returning too large a number ( %6.3f
",rx);
    rx = 0.0;
}
Fios(DacVal(rx), DacVal(0.0), &ia, &ib, &ic, &id);
return(rx); }

/* as above but do not set the value */
float VSetV(float outx) { /* find needed DAC0 voltage for given RAMP output */
    float rx = -LookUp(outx);
    if ((rx > 10.0) || (rx < -10.0)) {
        printf("Warning: SetV returning too large a number ( %6.3f
",rx);
        rx = 0.0;
    }
    return(rx); }

float VFluct(float outx) { /* find needed DAC1 voltage for given AC output */
    float rx = outx / AG;
    if (rx > 10.0) {
        printf("Warning: VFluct returning too large a number ( %6.3f
",rx);
        rx = 9.9;
    } else if (rx < -10.0) {
        printf("Warning: VFluct returning too small a number ( %6.3f
",rx);
        rx = -9.9;
    }
    return(rx); }

/* End of transfer functions */

/* Find how many bits of the ADC are required to span the stated range */
float NBits(float I, float EGn, int ADGn) {
    float x, v, ADGnV[4] = {10.0, 5.0, 2.5, 1.25};
    v = I * EGn;
    x = 2048.0 * v/ADGnV[ADGn]; return(x); }

/* Return thickness in nm from voltage */
float VThickns(float V) { float t;
    t = (V - Vrev) * 1000.0 / (55 - Vrev); return(t); }

/* Return thickness in nm from passed charge */
float Thickns(float Qch /* Coul/cm2 */) {
    double rho = 8.0 /* g/cm3 */;
    mwt = 441.76 /* g/Mole */;
    zy = 10.0 /* e/molecule */;
    eff = 1.0 /* unitless */;
    Avo = 6.02e23 /* #/Mole */;
    rv;
    rv = Qch * mwt * eff / (zy * Area * Avo * Q * rho); /* in cm */
    rv *= 1e7; /* put as nm */ return(rv); }

/* Apply DrY's test PRS 258 p497, 3V/s at 10 ma/cm2 */
float DrYT(float J) { float dVdd, ddt;
    J /= (1.0e3 /* mA to A */);
    dVdd = Eover(J);
    ddt = Thickns(J /* for 1 sec per cm2 */)/1.0e7; /* cm/s */
    printf("dV/dt = %g (Eover %g ddt %g)dVdd/ddt,dVdd,ddt); return(0.0); }


/* A rough delay function which attempts to take into account the  
software overhead */
void DelayLoop(float s) { delay((int) ((1000 * (s - TimeLoop)))); }
/* A long-term delay (seconds) */
void MyDelay(int s) { int i; for (i=0;i<s;i++) delay(1000); }
/* Look up the number of the last run */
int GetRunnum(void) {FILE *fp; int i = -1; j;
if ((fp = fopen("impruns.dat","r")) == NULL) {
printf("IMPRUNS not found - setting Runnum == 0\n");
} else { j = fscanf(fp," %d ",&i);
if (j == 1) printf("01d Runnum found, is %d\n",i);
else { printf("Old Runnum not readable\n: set to 999\n"); i = 998; }
fclose(fp); return(i+1); }
/* Calculate the prelude duration and ramp up to ACStartV */
void Prelude(int it, float Vsta, float Vsto, float Dura, float Vprel) {
time_t start, stop; int j, k; FILE *fp; float Vad0, Vad1, Vad7, x;
float V = Vsta - Vprel, Vdel;
printf("Starting Prelude!\nFirst a pause of %d seconds:\n",it);
MyDelay(it);
fp = fopen("preramp.dat","w");
printf("End of initial delay, now starting preliminary ramp\n");
start = clock(); k=0;
Vdel = (Vsto - Vsta)/(Dura * CLK_TCK); /* calculate ramp */
while (V < (Vsta-Vdel)) { /* ramp up until just short of starting voltage */
j = clock()-start; et = ((float) j)/ CLK_TCK;
V = (Vsta - Vprel) + (Vdel*j);
Vad0 = SVAD_Imm(0); Vad1 = SVAD_Imm(1); Vad7 = SVAD_Imm(7);
if (k++ == 10) {
if (fp != NULL) fprintf(fp,"%f %f %f %f %f\n",et,V,Vad0,Vad1,Vad7);
printf("t=%f V=%f Vad0=%f Vad1=%f Vad7=%f\n",et,V,Vad0,Vad1,Vad7);
k=0; }
SetVfV); }
stop = clock();
x = (stop-start)/CLK_TCK;
printf("It takes %f seconds\n",x);
printf("Prelude Done\n"); if (fp != NULL) fclose(fp); }
/* Update the run # file, write the IM# file, update the log file */
void SetRunDat(int run, int NFrqs, int nen, int acc[ ], int ram[ ], int av[ ],
int bv[ ], int cv[ ], float startv, float stopv, float id,
float ar, float vrv, float magn, float samp, float etim, float rat,
float Vprel, float Din, float Dfin, float E) { float et;
FILE *fp, *gp; char txt[90]; int i; time_t ss;
if ((fp = fopen("impruns.dat","w")) == NULL) {
printf("IMPRUNS not opened\n"); exit(99); }
else fprintf(fp," %d ",&run);
fclose(fp);
#define ZL 2048
sprintf(txt,"im%d.dat",run);
if ((fp = fopen(txt,"w")) == NULL) {
printf("%s not opened\n",txt); exit(99); }
/* for (i=0; i<nen; i++) {
fprintf(fp,"%d %d %d %d %d %d %d %d %d
", i, acc[i] - ZL, ram[i] - ZL,
Appendix C Impedance Measurement Program

```c

for (i=0; i<nen; i++) {
    et = i * etim / ((float) nen);
    fprintf(fp, "%5.3f %f %f %f %f\n", et, DacEqv(acc[i]), DacEqv(ram[i]),
            CVolt(GN0, av[i]), CVolt(GN1, bv[i]), CVolt(0, cv[i]), CVolt(2, dv[i]));
/* printf("%d %d %d %d\n", acc[i], ram[i], av[i], bv[i]); */
}
fclose(fp);
if ((fp = fopen("implog.dat", "a")) == NULL) {
    printf("IMPLOG not opened\n"); exit(99); }
time(&ss);
fprintf(fp, "%4d %5d %5.3f %5.3f %6.4f %6.4f\n"
        "%5.3f %6.4f %6.4f %f %f %s\n",
        run, nen, startv, stopv, id * 1000.0, ar,
        vrv, magn, samp, etim, rat,
        GN0, GN1, fTim, Rlow, Rhigh, Linearity,
        AG, XD, Vprel, Din, Dfm, E, ctim(e&ss));
while (kbhitO) { getchO; } /* get trailing /n's from stream */
printf("Please describe the run: "); txt[0]=90;
do cgets(txt); /* ??? */
while (strlen(txt)<4);
    fputs(txt+2, fp); /* returned string starts at txt[2] */
fprintf(fp, "\n\t { %d \nFrqs
    if ((gp = fopen("dacvids.scr", "r")) == NULL) { printf("FER\n"); exit(1); }
while (feof(gp) == 0) fputc(fgetc(gp), fp);
fprintf(fp, "%d\n", NCal); while (i < NCal) {
printf("Setting %f V on DA0 \n", Vx);
    DAC(0, -Vx); delay(1000); Vx = SVAD_Imm(0);
    DAC(1, 0.0); delay(2000); Vx = SVAD_Imm(0);
    DAC(1, 0.0); delay(2000); Vx = SVAD_Imm(0);
    DAC(0, 0.0); DAC(1, 0.0); AG = (Vx - Vx)/(8.0 * XD);
    printf("AC gain was: %5.3f\n", AG);
    printf("Now I will calibrate the power supply\n"); if ((fp = fopen("caldat.dat", "w")) == NULL) { printf("FOErr\n"); exit(2); }
i = 0; Vx = 0.0; NCal = 10.0 / CalStep;
fprintf(fp, "%f\n", NCal); while (i < NCal) {
    printf("Setting %f\n on DA0 \n", Vx);
    DAC(0, -Vx); delay(1000); Vx = SVAD_Imm(0);
fprintf(fp, "%f %f %f %f\n", Vx, Vy, XD);
    if (i < NCal) CalV[i] = Vy; else {
        if (i == 0) fprintf(fp, "%f %f %f\n", AG); else {
            if (i == 1) fprintf(fp, "%f %f %f\n", AG); else {
                if (i == 2) fprintf(fp, "%f %f %f\n", AG, Vox); else fprintf(fp, "%f\n");
            }
        }
    }
    i++; Vx += CalStep; }}
```

Calibrate the power supply

```c
*/
void QuickCal(void) { float Vox, Vx, Vy; int i; FILE *fp; DAC(1, 0.0);
    printf("QuickCal, set power supplies as for a run, disconnect sample\n");
    printf("First we need to check XD, the ratio of the ADCO reading to Vout\n");
    printf("Now with Vx+ ~ 45V, and +/-15V, measure Vps and enter\n");
    DAC(0.0, 0.0); DAC(1.0, 0.0); scanf("%f", &Vox); Vy = SVAD_Imm(0);
    printf("I read ADCO = %6.4f, indicating that XD is %6.4f\n",
            Vy, XD = fabs(Vy/Vox));
    DAC(0.0, 0.0); DAC(1, 0.0); delay(2000); Vx = SVAD_Imm(0);
    DAC(1, 0.0); delay(2000); Vx = SVAD_Imm(0);
    DAC(0.0, 0.0); DAC(1, 0.0); AG = (Vx - Vx)/(8.0 * XD);
    printf("AC gain was: %5.3f\n", AG);
    printf("Now I will calibrate the power supply\n"); if ((fp = fopen("caldat.dat", "w")) == NULL) { printf("FOErr\n"); exit(2); }
i = 0; Vx = 0.0; NCal = 10.0 / CalStep;
    fprintf(fp, "%f\n", NCal); while (i < NCal) {
        printf("Setting %f V on DA0 \n", Vx);
        DAC(0, -Vx); delay(1000); Vx = SVAD_Imm(0);
        fprintf(fp, "%f %f %f %f\n", Vx, Vy, XD, AG); if (i < NCal) CalV[i] = Vy; else {
            if (i == 0) fprintf(fp, "%f %f %f\n", AG); else {
                if (i == 1) fprintf(fp, "%f %f %f\n", AG); else {
                    if (i == 2) fprintf(fp, "%f %f %f\n", AG, Vox); else fprintf(fp, "%f\n");
                }
            }
        }
        i++; Vx += CalStep; }
```

Cahbrate the power supply

```c
*/
```
Appendix C Impedance Measurement Program

printf("Well, I think we are done!\n"); DAC(0,0.0); DAC(1,0.0); }

/*
Now read previous calibration data for the power supply
*/
void QkCalR(void) { float Vox, Vx; int i, j; FILE *fp;
printf("Now reading previous calibration data for the power supply\n");
if ((fp = fopen("caldat.dat","r")) == NULL) { printf("FOErr\n"); exit(2); }
scanf(fp,"%d",&NCal);
i = 0; Vx = 0.0; while ((Vx < 10.0) && (i < NCal)) {
    scanf(fp,"%f %f", &Vox, &Vx, &CalV[i]);
    if (fabsfVx- Vox) > 0.001) {
        printf("Warning: Calibration data is too old\n - redo\n"); exit(3); }
    i++;
}
printf("Well, I think we are done!\n"); DAC(0,0.0); DAC(1,0.0); }

/*
Perform a dry run: scale voltage range so as to properly
maintain the ramp rate
*/
void Tune(int N, float T, float StaV, float Vpre, float *StoV, float Pur) {
    int ij; long int start, stop; float x;
    int ramvec[i], acvec[i], av[i], bv[i], cv[i], dv[i];
    start = clock(); j = 2;
    ramvec[2] = DacVal(VSetV(StaV-Vpre));
    for (i=0; i<N; i++) {
        FIOS(ramvec[j], acvec[j], &av[j], &bv[j], &cv[j], &dv[j]);
        DelayLoop(T); }
    stop = clockO;
    x = (stop-start)/CLK_TCK;
    printf("Elapsed time was %d ticks, or %5.3f sec (Ratio of %5.3f)
", (int) (stop-start), x, x/Dur);
    *StoV += (StaV - *StoV)*(1.0 - (x/Dur)); }

/* ********************************************************** */
int main(int argc, char *argv[]) {
    float a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z;
    FILE *fp; unsigned long start, stop;
    printf("IMPED%d - Impedance measurement program\n", Version);
    /* First settle on the parameters */
    if (argc > 1) sscanf(argv[1], "%f", &ACStartV); else ACStartV = 0.0;
    if (ACStartV == -1.0) { QuickCalR(); exit(0); }
    if (argc > 2) sscanf(argv[2], "%f", &ACStopV); else ACStopV = 1.0;
    / Now determine what Gain we can use for ADC0 (Voltage) */
Appendix C Impedance Measurement Program

\[
x = ACStopV \times XD;
\]

if \((x > 5.0)\) \(GNO = 0;\) else { if \((x > 2.5)\) \(GNO = 1;\) else \(GNO = 3;\) }

if \((ACStartV == -3.0)\) { for \((i=0; i<10000; i++)\) { DAC(1, ACStopV); DAC(0, 0.0); printf("%f %f\n", CVolt(0, SFInADC(0, 0)), CVolt(0, SFInADC(1, 0))); delay(10000); exit(0); }

if \((ACStartV == -4.0)\) { for \((n=0; n<100; n++)\) { FIOS(DacVal(VSetV(5.0)), DacVal(VFluct(ACStopV)), &i, &j, &k, &l); printf("%f \n", CVolt(GNO, i)/XD); delay(1000); exit(0); }

if \((\text{argc} > 3)\) sscanf(argv[3], "%f", &Idens); else \(Idens = 0.001;\)

if \((\text{argc} > 4)\) sscanf(argv[4], "%f", &Area); else \(Area = 1.0; /* cm² */\)

if \((\text{argc} > 5)\) sscanf(argv[5], "%f", &Vprel); else \(Vprel = 5.0; /* prelude Voltage range*/\)

if \((\text{argc} > 6)\) sscanf(argv[6], "%f", &Linc); else \(Linc = 1.2;\)

if \((\text{argc} > 7)\) sscanf(argv[7], "%f", &Vrev); else \(Vrev = -0.71; /* CRC + DS Thesis*/\)

if \((\text{argc} > 8)\) sscanf(argv[8], "%f", &Magn); else \(Magn = 1.0; /* Magn is the maximum magnitude of the AC signal applied to the sample */\)

if \((\text{argc} > 9)\) sscanf(argv[9], "%d", &ITim); else \(ITim = 45;\)

if \((ACStartV == -2.0)\) { printf("Setting up for dummy resistor run: \n"); ACStartV = 5.0; ACStopV = 10.0; Idens = 0.06; Area = 1.0; Vprel = 5.0; Linc = 2.0; }

printf("Usage: imped ACStartV ACStopV Idens Area Vprel Linc Vrev Magn ITim\n"); printf("(for Calibration use 'imped -1 or -2')\n"); if \((\text{argc} < 2)\) exit(0); if \((\text{Home} == 1) \| (\text{Dummy} == 1)\) printf("Warning: Dummy is \(\%d\) and Home is \(\%d\) — 
");

printf("Setting initial voltage of %5.3fV. ACStartV - Vprel; IOS(daca = SetV(ACStartV - Vprel), dacb = 0.0, &a, &b, &c); Runnum = GetRunnum(); printf("New run will be number \(\%d\n", Runnum);

printf("Starting from %5.3fV to %5.3fV will ramp up for \(\%5.3f\ mA/cm²\n", ACStartV, ACStopV, Idens); J = Idens; /* convert to A*/

printf("This implies an overfield of %g (Z) or %g (Y) V/cm \n", EoverZ(J), E = Eover(J));

// E=4.098E6;

printf("Given that Vrev = %5.3f\n", Vrev); Din = (ACStartV - Vrev)/(E * 1000.0 /* convert cm to m */); Dfin = (ACStopV - Vrev)/(E * 1000.0 /* convert cm to m */); printf("%5.3f and %5.3f nm\n", Din*1.0e9, Dfin*1.0e9); printf("(By comparison from V-t plots: %5.3f and %5.3f nm)\n", 179)
Appendix C Impedance Measurement Program

\[ V_{\text{Thickns}}(AC_{\text{StartV}}), V_{\text{Thickns}}(AC_{\text{StopV}}); \text{nm per coul} = \text{Thickns}(1.0); \]
\[
\text{printf("Growth of \%5.3f nm per Coulomb, ramp will ",\text{nm per coul});
\]
\[
\text{printf("give \%5.3f nm of growth ",(Df\text{in-Df\text{in}})*1.0e9);}
\]
\[
\text{printf("\%5.3f Coul.in", Q=(D\text{f\text{in-Df\text{in}})*1.0e9/\text{nm per coul}}));}
\]

//Ta-Ming changes the expression for duration of the ramp
\[
V_{\text{T_slope}}=E*J*441.76/(10.0*96488.*8.);\text{Dura}=(AC_{\text{StopV}}-AC_{\text{StartV}})/V_{\text{T_slope}};
\]
\[
\text{printf("The V-t slope is \text{g v/s}\text{\"},V_{\text{T_slope}});
\]
\[
\text{printf("Hence the ramp with ac+dc will take \%5.3f seconds\text{\"},Dura);
\]

/* Now we come up with an estimate of the number of points we can have and, with that number, what sampling rate we can manage */
\[
\text{NentrFl = \text{Dura} / \text{SamplInt};
\]
\[
\text{printf("Since the sampling interval is \%5.3f, would like \%f entries\text{\"},
\]
\[
\text{SamplInt, NentrFl); \text{ Nentr = NentrFl;}
\]
\[
\text{if (NentrFl > NentrMax) \{\n\]
\[
\text{SamplInt = (SamplInt * NentrFl) / NentrMax; Nentr = NentrMax;}
\]
\[
\text{printf("But that is too many points, trimming to \%d\text{\"},Nentr);
\]
\[
\text{printf("and a sampling interval of \%5.3f will be used\text{\"},SamplInt); \}}
\]

/* Verify that this is the desired state of affairs */
\[
F_{\text{Max}} = 1.0 / (6.0 * \text{SamplInt}); /* 3 x Nyquist sampling */
\]
\[
F_{\text{Min}} = 1.0 / (0.5 * \text{Nentr} * \text{SamplInt}); /* an estimate - assumes lowest frequency gets half the vector */
\]
\[
\text{printf("Estimates: Fmax: \%5.3f Fmin \%5.3f\text{\"},F_{\text{Max}}, F_{\text{Min}});
\]
\[
\text{printf("Is this acceptable? (0/1)\text{\"});
\]
\[
\text{scanf("%d","&i);
\]
\[
\text{if (i == 0) \{\n\]
\[
\text{printf("Restart program with altered parameters\text{\"}); exit(0); \}}
\]

/* But we now need to make a dry-run to check the timing, with that information we can, for the # of entries and the sampling interval, settle on the desired ramping rate */

/* Ta-Ming keeps the same ACStopV to the input of Prelude routine*/
\[
\text{OldVsto=AC_{\text{StopV}};}
\]
\[
\text{#if \text{TuneF}
\]
\[
\text{printf("Entering Tuning subroutine\text{\"});
\]
\[
\text{Tune(\text{Nentr, SamplInt, AC_{\text{StartV}}, V_{\text{prel}}, &AC_{\text{StopV, Dura}});}
\]
\[
\text{#else}
\]
\[
\text{printf("Warning\text{\"}; Tuning turned off by Compile-time flag\text{\"});
\]
\[
\text{#endif}
\]
\[
\text{printf("Fine tuning of the sampling time and number of points\text{\"});
\]
\[
\text{printf(" indicates that the Stop Voltage should be revised to: \%5.3f\text{\"},}
\]
\[
\text{AC_{\text{StopV}});}
\]
\[
\text{AC_{\text{StopV}} *=1.004;}
\]
\[
\text{// printf("ratio:\text{\"});
\]
\[
\text{// scanf("\%f","&rat);}
\]
\[
\text{ramvec = ivector(0,\text{Nentr}); acvec = ivector(0,\text{Nentr});}
\]
\[
\text{av = ivector(0,\text{Nentr}); bv = ivector(0,\text{Nentr});
\]
\[
\text{cv = ivector(0,\text{Nentr}); dv = ivector(0,\text{Nentr});
\]
\[
\text{/* First make the DC RAMP */
\]
\[
\text{for (i=0; i<\text{Nentr}; i++) \{\n\]
\[
\text{a = (AC_{\text{StartV}} + ((AC_{\text{StopV}} - AC_{\text{StartV}})*i/\text{Nentr});
\]
\[
\text{ramvec[i] = DacVal(V\text{SetV}(a)); \}}
\]
Appendix C Impedance Measurement Program

/* and now the AC signals: */
/* -- first find the AC signal strength such that the signal is 
sufficiently linear, i.e. (Jhigh + Jlow)/Jmiddle < Linc */
printf("Now estimating magnitude of AC signal\n");
do {
    Jlow = JfffE * (1.0 - Magn));
    Jhigh = Jff(E * (1.0 + Magn));
    Linearity = (Jhigh+Jlow)/(2.0 * J);
    if (Linearity > Linc) {
        printf("Need to reduce Magn to %f since Linc = %fn",Magn,Linc);
        Magn /= 1.5;
    }
    printf("Fluctuation in field by %.5f% will lead 
",Magn*100.0);
    printf("Low: %.5f Centre: %.5f High: %.5f (mA/cm^2)\n", Jlow*le3, J*le3, Jhigh*le3);
    printf("Linearity factor is: %.5fn",Linearity);
    MPse("Press spacebar to continue\n");
}

/* Now check that the maximum current is an acceptable load */
if ((x=(Jhigh * Area))>Icrit) { prmtf("Warning: too high a current load (%5.3fA)\n",x); exit(0); }

/* now allow entry of frequencies */
printf("Do you wish to enter the frequencies (0/1)? : ");
scanf("%d",&i);
if (i == 1) {
    printf("How many frequencies (must be < 40) ?"); scanf("%d",&NFreq);
    i=0; while (i<NFreq) {
        printf("Enter f%d (%5.3f < f < %5.3f): ",i,FMin,FMax);
        scanf("%f",&(Freq[i]));
        if ((Freq[i] <= FMax)&&(Freq[i] >= FMin)) i++; else
            printf("Unacceptable frequency\n");
    }
}

/* Now select a set of frequencies and make up a superimposed signal */
if ((fp = fopen("dacvids.scr","w")) == NULL) { printf("FER\n"); exit(1); }
j = i = 0; NFrqs = cycle = 0;
do {
    acvec[j] = 0;
    do {/* Find next valid frequency */
        printf("Now using Freq[%d] = %8.6f\n",i,Freq[i]);
        x = 10.0 / Freq[i]; /* time for 10 periods */
        k = x / SampInt; /* corresponding number of samples we have time for */
        if (k < 60) { i++; printf("Too fast - skipped\n"); }
    /* require at least 3x Nyquist sampling before really worth while */
        if (i == NFreq) i = 0; } while (k < 60);
    js = j; NFrqs++;
    /* Now set put 10 periods of this frequency into AC vector */
    while ((j < Nentr) && ((j-js) < k)) {
        y = ACStartV + ((ACStopV - ACStartV)*j/Nentr);
        y *= Magn * sin(2.0 * Pi * (j-js) * SampInt);
        acvec[j] = DacVal(VFluct(y));
        j++;
    }fprintf(fp,
        printf("%d %d %d %8.6f",i, js, j-1, Freq[i]);
    if (i < (NFreq-1)) i++; else { i=0; cycle++;
    if (((i+1) % 20)==0) MPse("\nPress Space Bar When Ready ");}
while (j < Nentr);
fclose(fp); cycle++;
printf("Data vector contains %d cycles (%d Freq) of the frequency set\n", cycle,NFrqs);
printf("Is this acceptable? (0/1) : ");
scanf("%d",&i);
if (i==0) exit(0);

Ilow = 1.0e3 * Jlow * Area; Ihigh = 1.0e3 * Jhigh * Area;
printf("From the above, the current should range from ");
printf("%5.3f to %5.3f mA\n", Ilow,Ihigh);
printf("Assuming 10e-4 A/V on the EGG, maximum signal will be %5.3fV\n", Ihigh = 10 * Ihigh);
if (Ihigh > 9.0) { GN1 = 0; printf("A current shunt will be required - please enter Rlow, Rhigh: ");
scanf("%f %f",&Rlow,&Rhigh); } else {
printf("No shunt required\n"); Rlow = Rhigh = 0.0;
if (Ihigh > 4.0) GN1 = 0; else { /* be a bit pessimistic here */
if (Ihigh > 1.8) GN1 = 1; else {
if (Ihigh > 1.0) GN1 = 2; else GN1 = 3; } } }
printf("ADGains are Gain(0) = %d and Gain(1) = %d\n",GN0,GN1);

printf("Setting initial voltage of %5.3fV and pausing %5.3f\n", ACStartV-Vprel,ITim);
/* Start initial ramp from ACStartV-Vprel to ACStartV, without AC */
Prelude(ITim, ACStartV, OldVsto, Dura, Vprel);
/* Now perform the run with a minimum of overhead, i.e. a maximum
of speed */
printf("Now starting run\n"); start = clock();
for (i=0; i<Nentr; i++) {
FIOS(ramvec[i], acvec[i], &av[i], &bv[i], &cv[i], &dv[i]);
DelayLoop(SampInt); }
stop = clock(); x = (stop-start)/CLK_TCK;
DacVal(VSetV(0.0));

printf("Elapsed time was %d ticks, or %5.3f sec (Ratio of %5.3f)\n",
(int) (stop-start), x, x/Dura);
printf("Write result table? (0/1) : ");
scanf("%d",&i);
if (i==0) exit(0);
Dfin=(ACStopV-Vrev)/( 100.0*E);
/* Write our results */
SetRunDat(Runnum, NFrqs, Nentr, acvec, ramvec, av, bv, cv, dv, ACStartV,
ACStopV, Idens, Area, Vrev, Magn, SampInt, x, x/Dura,
Vprel, Din * 1.0e9, Dfin * 1.0e9,E);
/* Release our vectors */
free_ivector(dv, 0, Nentr); free_ivector(cv, 0, Nentr);
free_ivector(bv, 0, Nentr); free_ivector(av, 0, Nentr);
free_ivector(acvec,0,Nentr); free_ivector(ramvec,0,Nentr);
printf("\nDone\n"); return(O); }
Appendix D

Ellipsometer Control Programs

D.1 Introduction

The C program for ellipsometer control is named newell. There are two programs concluded in newell; ellsub and ellsubb. Ellsub lists the channels number of DT2828 and ports of PXB 721 board used in the newell program. Ellsubb includes all the subroutine used in the newell program.

D.2 User's Guide to the Ellipsometer Control Program

After getting into the menu by typing "newell", the screen shows first the hardware menu. The menu contains various commands to control the ellipsometer. This menu was used while the hardware did not function properly.

Hardware interface test:

0. End Hrdwr Tsts  Swith to user interface tests menu
1. Step            Step a motor which controls either P or A prizm angles
2. PhotoSig        Read the photosignal from the photodetector
Appendix D Ellipsometry Control Programs

3. Monitor Continually read an adc channel
4. SelGain Select gain for the amplifier of the photodetector
5. Sweep Make a sweep of the stepping motor and print the photosignals
6. Fitted Sweep Fit result from a sweep
7. Set Flags Set sweep width, ramping etc.
8. Skip Check Check if the stepping motors are skipping
9. DAC0 Set DAC0 to a given voltage
10. DAC0 (int) Set DAC0 to a given value
11. ADC1 Read adc1
12. MultRelay Set/reset relays 0 to 3
13. Characterise Turn on and off the relays
14. Test Gains Test Gains on the amplifier
15. Set delay Set delay before any command is executed
16. Read DANA Read the digital output from DANA

User interface test
1. Grow oxide film to a preset voltage
2. Set parameters of sweep width and ramping
3. Set data-set number
4. Semi-manual zone changing
5. An automated rough balance
6. Move to the other zone
7. Read the shaft encoder for value of P or A
8. Set sweep width, ramping etc.
9. Move the prism P and A to given angles
10. Balance P and A in one zone
11. Balance P and A into two zones
12. Repeating balance in one zone till "-" key being pressed
13. Repeating balance 'n' times in one zone.
14. Balance in 2 zones and determined Tc and Dc
15. Summarize archived data
16. Run an open circuit experiment
17. Run a stepped current experiment
18. Run a stepped current experiment with single critical voltage
19. Run a stepped current experiment with two critical voltages
20. Run a steady-state experiment.

D.3 Source Codes of Newell Program

/* NEWELLn.c : program for ELLIPSOMETER <--> 486 interface board */
/* based on NDEL (13 March 1993) */
/* Note that:
P(true) is taken as P(dial) = P(shaft)
A(true) is taken as A(dial) = 181.90 - A(shaft)
This is due to two factors:
1) shaft encoder for A is reversed and
2) the bellows coupling between the A prism and encoder introduces 1.9 deg difference between the dial reading and the shaft encoder 180 degree position.

All ANA readings are corrected to dial values in ReadO */

#define Version 1
#define Home 0
#include <math.h>
#include <stdio.h>
#include <graphics.h>
#include <time.h>
#include <ctype.h>
#include <conio.h>
#include <stdlib.h>
#include <dos.h>
#include <string.h>
#include <alloc.h>
#include <sys/timeb.h>
#define sqr(x) ((x)*(x))
#include "c:\newell\ellsub.c"
#define LMaxcnt 100
char far *seg, *segor;
char mychar, samnam[80];
unsigned int numdat = 398;
int Ramp = 1, MovIt = 1, grp = 1, ssz = 2,
    dbg = 0, Graph = 2, channels[16], gains[16], GainL, *buffer, buffer_number;
long int MyLoc[2] = {0, 0}, PosIn, PosFin, mytime,
    DANAVec[2][MaxDANA], TAR[Maxcnt], OpenClk[LMaxcnt], CloseClk[LMaxcnt];
float VolAr[Maxcnt], Gain, Theta = 70.07, data[MaxD], f0[MaxD], f1[MaxD],
    f2[MaxD], yfTMaxD], LastFit[2], NLastFit[2],
    LastDANA, II, j2j1, adcvecf[MaxAD];
#include "c:\newell\ellsubb.c"

int main(int argc, char *argv[]) {
    int i, j, k, rl, selc = 1, mydel = 0, nrep, nchan;
    long int p, a, jl; FILE *fp; float flp, fa, V;
    printf("Direct Ellipsometer Control (NEWELL%d)\n", Version);

    printf("Initializing DIO Board\n"); InitDIO();
    RelayOn(RLY0); RelayOff(RLY1); RelayOff(RLY2); RelayOff(RLY3);
    for (i=0;i<16;i++) { gains[i] = 1; channels[i] = 0; }
    rv = AL_INITIALIZE0; MTest(rv);
    rv = AL_SELECT_BOARD(1); MTest(rv);
    rv = AL_RESET0; MTest(rv);
    rv = AL_GET_CONFIGURATION(&config); MTest(rv);
    DUMP_CONFIGURATION0;
    AL_SETUP_DAC(0 /* Internal Timing */, 0 /* DAC0 */);
    rv = AL_SETUP_ADC(0, 1, channels, gains); MTest(rv);
    printf("INIT of DIO of ADC\n");
    rv = AL_ENABLE_FOR_OUTPUT(1); MTest(rv);
    rv = AL_ENABLE_FOR_INPUT(0); MTest(rv);
    rv = AL_SET_FREQUENCY(1.0); MTest(rv);
SelPA(POL);
if (argc > 1) { exit(0); } // this was to initialise for boot-up
Stupidity(); // check for obvious system faults
MPsc();
dnum = ArchNum();
if ((segor = (char far *) farmalloc((long)0x20000)) == NULL)
    printf("No Go\007 - Don't use DMA\n");
else {
    seg = (char far *) (((unsigned long) segor) >> 28);
    seg = (char far *) (((unsigned long) seg) + 1) << 28;
    printf("Allocation successful \tseg: %lx segor: %lx\n", seg, segor);
}
// HARDWARE LEVEL TESTS
while (selc > -1) {
    selc = 1; while (selc > 0) {
        printf("HARDWARE INTERFACE TESTS\n\n");
        DispFlgs();
        printf("Select:
O: End Hrdwr Tsts\n");
        printf("2: PhotoSig\t		3: Set initial delay\n");
        printf("4: Monitor channel\t	5: SelGain\n");
        printf("6: Sweep\t	7: Fitted sweep\n");
        printf("8: Set Flgs/Pars\n");
        printf("9: Repeating skip check\t10: DAC0 (V)\n");
        printf("15: Test gains\n");
        printf("17: Characterise the relays\n");
        printf("18: DANA read\n");
        printf("19: MultRelay tests\n");
        printf("20: Channel 0 and DANA compare\n");
        scanf("%d", &selc);
        if ((mydel > 0) && (selc == 6) || (selc == 7)) {
            printf("Starting delay\n");
            delay(mydel);
            if ((i >= 0) && (i < 8) {
                My_AL_ADC_VALUE(i, 1, &jl);
                if (i == -1) {
                    printf("DANA %d = %lf\n", ijl / 1000.0);
                    delay(102);
                }
            }
        }
    }
    switch(selc) {
    case 0: break;
    case 2: printf("Signal is %d\n", PhotoSig());
    case 3: printf("Enter delay (ms)\n");
        scanf("%d", &mydel);
    case 4: printf("Which channel? \n");
        scanf("%d", &i);
        printf("Here we go\007\n");
        Sweep(i, numdat, data);
        for (i = 0; i < numdat; i++) {
            printf("%2d %6.3f %6.3f %6.3f %6.3f\n", i,
                data[i], data[i + 1], data[i + 2], data[i + 3]);
        }
    case 5: printf("Select gain: (0-7) \n");
        scanf("%d", &i);
    case 6: printf("Select P=0, A=1: \n");
        scanf("%d", &i);
        printf("Here we go\007\n");
        Sweep(i, numdat, data);
        for (i = 0; i < numdat; i++) {
            printf("%2d %6.3f %6.3f %6.3f %6.3f\n", i,
                data[i], data[i + 1], data[i + 2], data[i + 3]);
        }
    case 7: printf("Fitted sweep\nSelect P=0, A=1: \n");
        scanf("%d", &i);
        Sweep(i, numdat, data);
        printf("Offset of minimum: %f\n", Fit(numdat, data, i));
        printf("Initial position: %ld Final position: %ld\007\n", PosIn, PosFin);
        MPsc();
    case 8: SetFlgs();
    case 9: printf("Repetitive skip check\n");
Appendix D Ellipsometry Control Programs

printf(" StUpDelDf: %d StlnDelDf %d StUpDelMn %d StlnDelMn %d\n", StUpDelDf, StlnDelDf, StUpDelMn, StlnDelMn);
Display = 0;
for (i=0;i<10;i++) {
  Sweep(POL,numdat,data);
  printf("P: Initial position: %ld Final position: %ld\n", PosIn,PosFin);
for (i=0;i<10;i++) {
  Sweep(ANA,numdat,data);
  printf("A: Initial position: %ld Final position: %ld (Ase: %ld )\n", PosIn,PosFin,AShaf(PosFin));
}
break;
}
case 10: { float xx; int i; printf("Set DAC to: "); scanf("%f",&xx);
i = VtoDac(xx); AL_DAC_VALUE(0 /* DACO */, &i); } break;
case 15: printf("Test gains\n"); for (i=0; i<8; i++) {
  SelGain(i); delay(100); j = PhotoSigO;
  printf("GainSet: %2d GainVal: %7.2f Signal: %4d Value: %5.3f\n", i, GainL/(4.096 * Gain), GainL, GainL); MPseO;
break;
case 17: { int del; printf("Select a delay: ");
  scanf("%d ",&del); printf("Working\n");
  while (kbbitO) getchO;
  while (!kbhitO) { MultRelay(0,0,0,0); printf("Off\n"); delay(del); MultRelay(1,1,1,1); printf("On\n"); delay(del); }
break;
case 18: { long int ii; printf("Will assume DANA is on 100V scale\n");
  while (kbhitO) getchO; while (!kbhitO) {
    ReadDANA(&ii); printf("DANA: %6.3f \n",ii/1000.0); }
break;
case 19: { int i, j, k, l; printf("MultRelay: enter pattern (EG 0 0 1 0): ");
  scanf("%d %d %d %d ",&i,&j,&k,&l);
  MultRelay(i,j,k,l); } break;
case 20: { float x; VReadDANA(&xx,10);
  while (kbhitO) getchO; while (!kbhitO) {
    printf("Compare: %5.3f %5.3f\n",x,ADC(0)); delay(100); MPseO;
break;
default: printf("Case not known\n"); break; }
} // USER INTERFACE TESTS

sele = 1; while (sele > 0) { printf("USER INTERFACE TESTS\n"); DispFlgsO;
  printf("Select:\n0: End User Tests\n1: End All\n2: Grow a thick oxide with logged V and time\n3: Set parameters (ndat, Ramping)\n4: Set DSN\n5: SemiMan find other zone\n6: Flip to other zone\n7: ReadPA\n8: Set Flags/Pars\n9: Goto\n10: Balance in 1 Zone\n11: Balance in 2 Zones\n12: Repeating 1Z balances\n13: RepBallZ\n14: Calculate Tc,Dc\n15: Archive summary\n16: Open circuit experiment\n17: Switched current experiment\n");
Appendix D Ellipsometry Control Programs

printf("18: Switched current experiment with two critical voltages\n");
printf("19: Switched current experiment with single critical voltage\n");
printf("20: Consecutive reads of DANA\n");
printf("21: Steady State Measurements\n");
scanf("%d", &selc);
switch (selc) {
  case 0: break;
  case 1:
    { int DanRan, i; long int iniclk, myclk;
      float FinV, lastV, Volts, LocLastV, x; FILE *gp;
      char fnam[50];
      printf("Growth of thick oxide\n");
      printf("Ensure that Relay 1 and 2 are in series across the sample\n");
      // Short out the sample (R1, R2 on)
      MultRelay(ON, OFF, OFF, OFF); delay(1000);
      printf("Turn the current supply on now\007\n");
      printf("What range is the DANA on? (10, 100 or 1000)\n");
      scanf("%d", &DanRan);
      printf("What is the maximum voltage\n");
      scanf("%f", &FinV);
      printf("File title to be used for the slope data? (max 7 chars)\n");
      scanf("%s", fnam);
      if ((gp = fopen(fnam, "w")) == NULL) {
        printf("FOErr\007\n");
        exit(99);
      }
      printf("Here we go:\nType '-' to finish.\n");
      // NBal2Z("Growth of oxide: initial 2Z\n");
      printf("Now opening relays\n");
      MultRelay(OFF, ON, ON, OFF);
      Volts = LocLastV = 0.0; iniclk = clock();
      do {
        VReadDANA(&Volts, DanRan);
        printf("DANA interim voltage: %6.4f\n", Volts);
        if (LocLastV != Volts) {*/ x = myclk = clock() - iniclk;
        x /= CLK_TCK;
        printf("%f %ld %6.4f\n", Volts, myclk, x);
        fprintf(gp, "%f %ld %6.4f\n", Volts, myclk, x);
        // LocLastV = Volts;
        delay(1000);*/}
      } while (Volts < FinV);
      printf("Now closing relays\n");
      MultRelay(ON, OFF, OFF, OFF);
      delay(1000); // allow sample to relax
      printf("Turn off the current source\007\n");
      fclose(gp);
      printf("Done\n");
      MPseO; break;
  case 2: printf("How many data points per sweep? (Presently %d) \n");
    scanf("%d", &numdat);
    printf("Do you wish to Ramp the stepper motors? (0/1) \n");
    scanf("%d", &Ramp);
    printf("Do you wish to store GRAPHER files? (0/1) \n");
    scanf("%d", &Graph); printf("Done\n");
    MPseO; break;
  case 3: printf("Enter data set number (presently %d) \n");
    scanf("%d", &dsnum);
    if (Arch) { /* start an entry in the archive */
      // Arch = TRUE;
    }
Appendix D Ellipsometry Control Programs

printf("Now archiving\n");
if ((fp = fopen("PAF","a"))==NULL) { printf("Open error\n");
exit(99); } 
fprintf(fp,"%d \n",dsnum); fclose(fp); break;

case 4: ReadPA(&p,&a); flp = p/100.0; fa = a/100.0;// ZoneaP=p; ZoneaA=a;
printf("Assuming NULLED at p= %6.3f, a= %6.3f ase= %6.3f\n",flp,fa,AShaf((long) (fa * 100))/100.0);
printf(" another null is at p= %6.3f, a= %6.3f ase= %6.3f\n",flp+90.0,180.0-fa,AShaf((long) ((180.0 - fa) * 100))/100.0);
printf("Please set to a NULL near that position\n"); MPse();
ReadPA(&p,&a); /* ZonebP=p; ZonebA=a;*/ break;

case 5: { float minv, x; int ossz; long minp, mina;
printf("Rough balance\n"); ossz = 3;
minv = SmPhotoSigQ; ReadPA(&minp,&mina);
for (j=POL; j<= ANA; j++) {
for (i=0; i< 12000; i++) { StepG, CW);
x = SmPhotoSigQ;
if (x < minv) { minv = x; ReadPA(&minp,&mina);
printf("min at p=%ld a= %ld with signal of %f\n",minp,mina,x);} }
Goto(minp,mina); MPse(); ssz = ossz; } break;

case 6: ReadPA(&p,&a);
printf("(ooo)\n"); MPse(); break;

case 7: ReadPA(&p,&a);
printf("P=%ld A=%ld Ase=%ld\n",p,a,AShaf(a)); MPseO; break;

case 8: SetFlgsO; break;

case 9: printf("Goto:\nP,A?\nEnter as integers: ");
scanf("%ld %ld",&p,&a); Goto(p,a); MPse(); break;

case 10: printf("1 Zone Balance\n"); Bal1ZO; break;

case 11: { char memo[80];
float Volts;
printf("2 Zone Balance (NBal2Z)\n");
Theta=62.77;
printf("What is the voltage?\n");
scanf("%.3f",&Volts);
printf("2Z at %5.3f\n",Volts);
NBal2Z(memo); }
MPse(); break;

case 12: printf("Infinite Repeating 1 Zone Balance\nPress key to end\n");
Running = 1; while (kbhit()) getchO; while (kbhitO) { Bal1Z();
printf("Key pressed - end of balancing\n"); MPse(); Running = 0; break;

case 13: printf("RepBal1Z -\n Enter #repeats, nchannel: ");
scanf("%d %d",&nrep, &nchan); RepBal1Z(nrep, nchan, "test");
printf("Finish\n"); MPse(); break;

case 14: { double A1, P1, A3, P3, SF, Tc, Dele, B, Q;
int i; float Aref, Pref;
printf("Calculate Tc and Dc\n";
Enter 1 for data entry, 0 for measurement: ");
scanf("%.3f",&i);
SF = Pi / 180.0; Q = -Pi / 4.0;
if (i == 1) { printf("Enter P1 A1 P3 A3 (in degrees)\n");
scanf("%.3f %.3f %.3f %.3f",&Pprime, &Aprime, &Pthree, &Athree); }
else NBal2Z("Tc Dc calc");
printf("What are the offsets for the prisms: Pref Aref ?\n");
printf("(These are from 'CALIB' command #3)\n");
}
Appendix D Ellipsometry Control Programs

```c
scanf("%f %f", &Pref, &Aref);
A1 = (Aprime - Aref) * SF; A3 = (Athree - Aref) * SF;
P1 = (Pprime - Pref) * SF; P3 = (Pthree - Pref) * SF;
Tc = -tan(P1-Q)*tan(P3-Q);
if (Tc < 0.0) {printf("Problem Tc\007 Val %f P1 %f P3 %f\n", (float)Tc, (float)P1, (float)P3); Tc = -Tc; }
Tc = 1.0 / sqrt(Tc); /* See P24 of NBS paper */
B = tan(A1)*((tan(P3 - Q)*tan(Q)) - (tan(P1-Q)/tan(Q)));
B -= tan(A3)*((tan(P1 - Q)*tan(Q)) - (tan(P3-Q)/tan(Q)));
B /= 2.0 * (tan(A1) - tan(A3));
Dele = (-sqrt(B)) - (tan(P1-Q)*tan(P3-Q));
if (Dele < 0.0) {printf("Problem Delc\007 Val %f Pl.%f P3 %f\n", (float)Dele, (float)Pl, (float)P3); Dele *= -1.0; }
Dele = sqrt(Dele)/B;
Dele = atan(Dele); if (Dele<0.0) Dele += Pi;
printf("Tc %7.3f DelC %7.3f ", Tc, Dele/SF);

case 15: ArchSum(stdout); break;

```

// Connect the sample. (R0 on)
// Disconnect the sample (R0 off)
```
Appendix D Ellipsometry Control Programs

// Short out the sample (R0 on), leave the resistance in parallel (R3 off)
MultRelay(ON, OFF, OFF, OFF);
printf("Turn the current supply on now\n");
printf("How many reads: initially, at current1, at current2 and finally?\n");
printf("(IE enter 4 integers): ");
scanf("%d %d %d %d", &Nin, &None, &Ntwo, &Nfin);
printf("Here we go:");
RepBallZ(Nin, ChanR, "Stepped current experiment (initial)"); // initial reads of parameters
// pass the first current (R0 off) (no longer short out the sample)
MultRelay(OFF, OFF, OFF, OFF);
RepBallZ(None, ChanR, "1st evolving"); // read voltage as grown under voltage/current
// short out resistance (R3 on)
MultRelay(OFF, OFF, OFF, ON); // turn off first current and on second
RepBallZ(Ntwo, ChanR, "2nd evolving"); // read voltage
// short out sample
MultRelay(ON, OFF, OFF, ON); // short out sample again
printf("Now turn current supply to standby\n");
RepBallZ(Nfin, ChanR, "finished"); // read voltage
printf("Experiment complete - recorded in data set number %d, dsnnum); } break;
case 18: { int Nin, Nfin; float CritV, CritV2;
printf("Stepped Current Experiment Based on Critical Voltage\n");
printf("Ensure that Relay0 is connected across the sample, and\n");
printf("that Relay3 connects or disconnects the external resistance.\n");
// Short out the sample (R0 on), leave the resistance in parallel (R3 off)
MultRelay(ON, OFF, OFF, OFF);
printf("Turn the current supply on now\n");
printf("How many reads: initially, and finally?\n");
scanf("%d %d", &Nin, &Nfin);
printf("What are the two critical voltages? ");
scanf("%f %f", &CritV, &CritV2);
printf("Here we go:");
RepBallZ(Nin, ChanR, "Stepped current (CritV) (initial)"); // initial reads of parameters
// pass the first current (R0 off) (no longer short out the sample)
MultRelay(OF, OFF, OFF, OFF);
do { 
RepBallZ(1, ChanR, "11"); /* read voltage as grown under voltage/current */ } while (LastDANA < fabs(CritV));
// short out resistance (R3 on)
MultRelay(OFF, OFF, OFF, ON); // turn off first current and on second
do { 
RepBallZ(1, ChanR,"12"); /* read voltage */ } while (LastDANA < fabs(CritV2));
// short out sample
MultRelay(ON, OFF, OFF, ON); // short out sample again
printf("Now turn current supply to standby\n");
RepBallZ(Nfin, ChanR, "finished"); // read voltage
printf("Note that current supply should now be on standby\n");
printf("Experiment complete - recorded in data set number %d, dsnnum); } break;
case 19: { int DelDANA, DanRan, freq, nsam, count, i, j; char mychar;
float CritV, Volts; FILE *fp; char fnam[50], fulfnam[80];
// struct timeb t;
printf("Stepped Current Experiment Based on Single Critical Voltage\n");
printf("Ensure that Relay0 is connected across the sample, and Relay3 connects or disconnects the external resistance.\n");
// Short out the sample (R0 on), leave the resistance in parallel (R3 off)
MultRelay(ON, ON, OFF, OFF);
printf("Turn the current supply on now\n");
// printf("How many ms delay between the initial DANA reads? \n");
// scanf("%d",&DelDANA);
DelDANA = 1000;
printf("What range is the DANA on? (10, 100 or 1000) \n");
scanf("%d",&DanRan);
printf("What frequency for sampling? (integer please)? \n");
scanf("%d",&freq);
printf("What is the critical voltage? ");
scanf("%f",&CritV);
printf("What is II (in mA): "); scanf("%f",&Il); . .
printf("Which sample (eg: sample1): "); scanf("%s",samnam);
printf("What is the file title to be used? (max 7 chars) ");
scanf("%s",fham);
printf("Here we go:\n");
// ellipsometer disabled here..
// RepBal1Z(Nin, ChanR); // initial reads of parameters
// pass the first current (R0 off) (no longer short out the sample)
MultRelay(OFF, OFF, OFF, OFF);
// now clear kbd buffer
// do { getchO; } while (kbhitO);
do {
    // RepBal1Z(1, ChanR); /* read V as grown under voltage/current */
    VReadDANA(&Volts, DanRan);
    // ftime(&t);
    printf("Seconds since 1/1/1970 GMT: %ld\n", t.time);
    // printf("Thousands of a second: %d\n", t.millisecond);
    printf("DANA Reading #\n Voltage: %6.4f\n\n", count, Volts);
    VolArj = (count++ % Maxcnt] = Volts;
    Tarj = clock();
    if (kbhitO) { mychar = (char) getchO; }
delay(DelDANA); }
while ((Volts < CritV)&&(mychar != '-')) ; // assumes that V increases with time
// short out resistance (R3 on)
printf("Switching currents and starting DMA\n");
MultRelay(OFF, OFF, OFF, ON); // turn off first current and on second
// do {
    // RepBal1Z(1, ChanR); /* read voltage */
    // while (LastDANA < fabs(CritV2));
    Grab(freq /* Hz */, nsam /* # */);
    printf("DMA completed\n");
    // shorts the sample right after the DMA completed.
delay(2000);
    MultRelay(ON, OFF, OFF, OFF);
// print out DANA data
printf("Printing DANA data\n");
sprintf(fulfnam,"%sd.dat",fnam);
if ((fp = fopen( fulfnam,"w")) ==NULL) {
    printf("FOPEN Error\007\n"); exit(88); }
if (count > Maxcnt) {
    for (i=count-1;i>count-Maxcnt-1;i--) {
        j = i % Maxcnt;
        printf("%d %f %ld\n",VolAr[j],TAr[j]);
        fprintf(fp,"%d %f %ld\n",j,VolAr[j],TAr[j]);
    }
else {
    for (i=0;i<count;i++) {
        printf("%d %f %ld\n",i,VolAr[i],TAr[i]);
        fprintf(fp,"%d %f %ld\n",i,VolAr[i],TAr[i]);
    }
    fclose(fp);
}
// now display the ADC data
//
printf("Saving ADC data\n");
for (i=0;i<nsam;i++) adcvec[i] = CVolt(0,GetVal(i));
SavDat(fnam ,nsam, (float) freq, adcvec);
Running = 0; DispDat(nsam-1,adcvec,adcvec);
// short out sample
// MultRelay(ON, OFF, OFF, ON); // short out sample again
printf("Now turn current supply to standby\n");
// RepBallZ(Nfin, ChanR); // read voltage
print("Experiment complete\n");
MPseO; break;
case 20: { int DelDANA, DanRan, freq, nsam, count, i, j; char mychar;
    float CritV, Volts; FILE *fp; char fnam[50], fulfnam[80];
    printf("Record DANA\n");
    printf("Ensure that Relay0 is connected across the sample, and\n");
    printf(" Relay3 connects or disconnects the external resistance.\n");
    Short out the sample (R0 on), leave the resistance in parallel (R3 off)
    MultRelay(ON, OFF, OFF, ON); // short out sample again
    printf("Now turn current supply to standby\n");
    // RepBallZ(Nfin, ChanR); // read voltage
    printf("Experiment complete\n");
    MPseO; break;
    case 20: { int DelDANA, DanRan, freq, nsam, count, i, j; char mychar;
        float CritV, Volts; FILE *fp; char fnam[50], fulfnam[80];
        // struct timeb t;

        printf("Record DANA\n");
        printf("Ensure that Relay0 is connected across the sample, and\n");
        printf(" Relay3 connects or disconnects the external resistance.\n");
        // Short out the sample (R0 on), leave the resistance in parallel (R3 off)
        MultRelay(ON, OFF, OFF, OFF);
        printf("Turn the current supply on now\007\n");
        DelDANA = 500;
        printf("What range is the DANA on? (10, 100 or 1000) \n");
        scanf("%d",&DanRan);
        printf("What is the critical voltage? ");
        scanf("%f",&CritV);
        printf("What is the file title to be used? (max 7 chars) ");
        scanf("%s",fnam);
        printf("Here we go:\nType ' ' to start DMA before critical voltage.\n");
        // pass the first current (R0 off) (no longer short out the sample)
        MultRelay(OFF, OFF, OFF, OFF);
        count = 0; mychar = "";
        do {
            VReadDANA(&Volts, DanRan);
            printf("DANA Reading #%d Voltage: %6.4f\n",count,Volts);
            VolAr[j] = (count++ % Maxcnt)) = Volts;
            TAr[j] = clock();
        }
Appendix D Ellipsometry Control Programs

if (kbhit()) { mychar = (char) getch(); }
    delay(DelDANA); }
while ((Volts < CritV) && (mychar != '-')) ;
// shorts the sample right after the DMA completed.
MultRelay(ON, OFF, OFF, OFF);
// print out DANA data
    printf("Printing DANA data\n");
    sprintf(fulfham,"%sd.dat",fnam);
if ((fp = fopen(fulfham,"w")) == NULL) {
    printf("OPEN Error\n"); exit(88); }
    if (count > Maxcnt) {
        for (i=count-1;i<count-Maxcnt-1;i--) {
            j = i % Maxcnt;
            printf("%d %f %ld\n",VolAr[j],TAr[j]);
            fprintf(fp,"%d %f %ld\n",VolAr[j],TAr[j]); } }
    else {
        for (i=0;i<count;i++) {
            printf("%d %f %ld\n",i,VolAr[i],TAr[i]);
            fprintf(fp,"%d %f %ld\n",i,VolAr[i],TAr[i]); }
    }
    fclose(fp);
    printf("Now turn current supply to standby\n");
    printf("Experiment complete\n"); }
MPseO; break;
case 21: { int DanRan, count, i, Stop; char mychar, memo[80];
    float StepV, lastV, Volts, LocLastV, x; FILE *fp, *gp;
    char fnam[50]; long int myclk, iniclk;
    printf("Steady State Measurements\n");
    printf("Ensure that Relays 1 and/or 2, in series, lead to the
    sample
    and that Relay 0 shorts out the current supply\n");
    MultRelay(ON, OFF, OFF, OFF); delay(1000);
    printf("Turn the current supply on now\n");
    scanf("%f",&Theta);
    printf("What is the angle of incidence? \n");
    scanf("%d",&DanRan);
    printf("Do you wish to pause after each zone's balance? (0/1): \n");
    scanf("%d",&StopBal);
    printf("Do you wish to pause before turning on currents? (0/1): \n");
    scanf("%d",&Stop);
    printf("What is the desired voltage step? \n");
    scanf("%f",&StepV);
    printf("File title to be used for the PA data? (max 7 chars) \n");
    scanf("%s",fnam);
    if ((fp = fopen(fnam,"w")) == NULL) { printf("OPEN Error\n"); exit(99); }
    printf("File title to be used for the slope data? (max 7 chars) \n");
    scanf("%s",fnam);
    if ((gp = fopen(fnam,"w")) == NULL) { printf("OPEN Error\n"); exit(99); }
    printf("Here we go:\n");
    NBal2Z(memo); // may not have time for ZZ
    printf("Have completed initial 2 zone balances\n");
    count = 0; mychar = ' '; Volts = 0.0;
if (kbhit()) { mychar = (char) getch(); }
iniclk = clock();
do {
    // pass the first current (R1 off) (no longer short out the sample)
    printf("Now opening relay0, closing R1,R2\n");
    OpenClk[count] = clock();
    MultiRelay(OFF, ON, ON, OFF);
    LocLastV = lastV = Volts;
    do { myclk = clock(); x = ((float) (myclk - iniclk))/CLK_TCK;
        VReadDANA(&Volts, DanRan);
        printf("DANA interim voltage: %6.4f\n",Volts);
        if (LocLastV != Volts) {
            fprintf(gp,"%5.3f %ld %f\n",Volts,myclk,x); LocLastV = Volts
            while (Volts < (lastV + StepV));
        VolAr[count] = Volts; // Short out the supply (R0 on)
    CloseClk[count++] = clock();
    MultiRelay(ON, OFF, OFF, OFF);
    delay(1000); // allow sample to relax
    if (Stop) { printf("Press the space-bar to continue\n"); MPse(); }
    sprintf(memo,"2Z at %5.3f\n",Volts);
    NBal2Z(memo);
    fprintf(fp,"%bc %lx %5.3f %5.3f %5.3f %5.3f %5.3f\n",
        OpenClk[count-1],CloseClk[count-1],VolAr[count-1],
        lastPavg, lastPstd, lastAavg, lastAstd);
    printf("Press any key (within 5s) to pause or - to end\n");
    while ((kbhit())&&(mychar != '-')) { mychar = getch(); } delay(5000);
    if ((kbhit()) && (mychar != '-')) {
        mychar = (char) getch(); MPse(); }
    } while ((count < LMaxcnt) && (mychar != '-'));
    // Finished, with sample shorted (R1, R2 on)
    printf("Turn off the current source\007\n");
    fprintf(fp,"%lx %lx %5.3f %5.3f %5.3f %5.3f %5.3f
",
        OpenClk[count-1],CloseClk[count-1],VolAr[count-1],
        lastPavg, lastPstd, lastAavg, lastAstd);
    fflush(fp);
    for (i=0; i<count; i++) {
        fprintf(fp,"%lx %lx %f\n",OpenClk[i],CloseClk[i],VolAr[i]);
        printf("%d: Open: %lx Close: %lx %f\n",i,
            OpenClk[i],CloseClk[i],VolAr[i]);
    fclose(fp); fclose(gp); printf("Done\n"); }
    MPse(); break;
}
case 22: { int nsam, freq;
    printf("Test of Grab/DMA\n"); nsam = 200;
    freq = 500; Display = 2;
    Grab(freq /* Hz */, nsam /* #. */); printf("DMA completed\n");
    for (i=0;i<nsam;i++) {
        adcvec[i] = CVolt(0,GetVal(i));
        printf("%d \%f\n",i,adcvec[i]);
        if (((i+1)%20) == 0) MPse(); }
    DispDat(nsam-1,adcvec,adcvec); } break;
Appendix D Ellipsometry Control Programs

    case -1: break;
    default: printf("Case not known\007\n"); break; }
}

printf("Done\n");
AL_TERMINATE(); return(0); }

D.4 Source Codes of Ellsub Program

// ELLSUB.c (Prologue)
/* Contains the subroutines used by CALIB and NEWELL  19 Mar 1993 */
#include <math.h>
int NWB;    // Toggle to indicate whether balances were 'good'
int RBALOO = 0;  // do not use Rough Balancing routine
int StopBal = 0; // do not stop after each RepBal1Z
float lastAavg, lastPavg, lastAstd, lastPstd;
float AInt, PInt, APrime, PPrime, Athree, Pthree;
float DA, DC, t1c, t2c, t1p, t2p, DP, npc;
#define APArSz 20
float PAr[APArSz], AArf[APArSz];
#define MXPL 300
#define Pi 3.14159
#define MaxD 400
#define MaxDANA 1000
#define StUpDelDf 7
#define StUpDelMn 3
#define E_NORMAL 0
#define E_GAIN -10
#define E_CHANNELS -29
#define E_UNEXP -100
#define E_ADERR 8
#define QTR 9000
#define HALF 18000
#define FULL 36000
// PXB-721 Definitions, assuming base address is 300
#define ControlReg 0x303 /* used for ellipsometer board */
#define PortA 0x304
#define PortB 0x305
#define PortC 0x306
#define ControlReg 0x30B // for reading Shaft Encoders
#define PortA3 0x308
#define PortB3 0x309
#define PortC3 0x30A
// Note that for reasons unknown,
// DT2828 channel 2 is DT707 channel 4
// DT2828 channel 1 is DT707 channel 2
// DT2828 channel 0 is DT707 channel 0
#define ChanA 1
Appendix D Ellipsometry Control Programs

// define channel number for DMA (DANA analogue output)
#define ChanD 0
// define channel number for reading experimental voltages
// note that a negative value will read the DANA rather than an ADC channel
#define ChanR -1
#define MinSig 400
#define MaxSig 2800
#define CritStep 1000.0 /* max tolerable step in sweep */
#define CritDif 2 /* " " difference in P, A readings */
#define CritR 3 /* number of repeats in P, A readings */
#define ON 1
#define OFF 0

enum prisms {POL,ANA};
enum direct {CW.CCW};

// Default values for rough balancing in the two zones
#define ACOR (long) 18190
#define MaxAD 5000
#define Maxcnt 1000

int MyCode[2][2] = {{PCW, PCCW}, {ACW, ACCW}};

D.5 Source Codes of Ellsubb

// ELLSUBB.c (Epilogue)
/* Contains the subroutines used by CALIB and NEWELL 19 Mar 1993 */
#define Klugy 0
time_t PT, AT;
#define RPD (3.1415926 / 180.0)

//******* Enter prototypes *******//
float msin(float x);
float mcos(float x);
void AdjPos(float x, int pas); // adjust prism angle
char *myctime(time_t *t); // return Date and Time w/o ln
void BalP(float *p, time_t *PT);
void BalA(float *a, time_t *AT);
void VReadDANA(float *x, int DanRan);
void DetermGain(void); // select suitable gain for sweep
void DispFlgs(void); // display flags
void DFlip(void); // Go to other zone
void SetFlgs(void); // set parameters for ELLIP control
void Selln(int i); // select input octal latch
void SelPA(int pas); // select shaft encoder to be read for P or A
long INorm(long prp); // normalise angle (long)
long AShaf(long ii); // calculate shaft encoder reading for A
float Norm(long prp); // normalise angle (float)
long int RawRead(int pas); // read shaft encode value (no correction)
long int Read(int pas); // read shaft encode value for P or A
void Chek(int i); // check for DT-2828 error condition
int PhotoSig(void); // read photodetector signal on ADC0
Appendix D Ellipsometry Control Programs

void SelGain(int i); // select gain for photodetector amplifier
float SmPhotoSig(void); // read photodetector with optimal gain
void RdySt(void); // set default delays for stepping
int WriteOut(int i, int j, int k); // set outputs of set 3
int ReadDANA(long int *kk); // read the BCD digits from the DANA
void RelayOn(int i); // turn on a single relay, #0 to 5
void RelayOff(int i); // turn off a single relay
void MultRelay(int R0, int R1, int R2, int R3); // set/reset several relays
void Step(int pas, int cc); // step P or A, CW or CCW
int Stupidity(void); // check Detector, SE are set properly
void MyMov(int pas, int count); // move P or A count steps
void Sweep(int pas, int ndat, float dat[]); // perform sweep
void DispDat(int np, float y[], float y[]); // display sweep data
float Fit(int n, float y[], int pas); // fit Chebyshev polynomials
int VtoDac(float x); // given a voltage, compute value for DAC
void DAC(float V); // set DAC to given voltage
float ADC(int chan); // read ADC channel
int My_AL_ADC_VALUE(int chan, int gain, long int *k);
int NotSetEnd(FILE *fp); // detect end of PAF data
int AL_TERMINATE(void); // see DT-2828 and ATLAB manuals
int AL_INITIALIZE(void);
int AL_SELECT_BOARD(int i);
int AL_SET_FREQUENCY(float freq);
int AL_SET_TIMEOUT(mHz);
int AL_SETUP_ADC(int timsrc, int scancnt, int chanar[], int gainar[]);
int AL_RESET(void);
int AL_GET_CONFIGURATION(void *i);
int AL_FTND_DMA_LENGTH(unsigned *i, unsigned *j);
void DUMP_CONFIGURATION(void);
int AL_SETUP_DAC(int i, int j);
int AL_ENABLE_FOR_OUTPUT(int i);
int AL_ENABLE_FOR_INPUT(int i);
int AL_DAC_VALUE(int i, int j);
int AL_ADC_VALUE(int chan, int gain, int *k);
int AL_INPUT_DIGITAL_VALUE(int i, int j, int k);
int AL_OUTPUT_DIGITAL_VALUE(int i, int j, int k);
int AL_DECLARE_BUFFER(int *buffer_number, int *buffer, int NConv);
int AL_RELEASE_BUFFER(int *unitnum, int *bumum);
int AL_LINK_BUFFER(int buffer_number);
int AL_BURST_ADC(void);

****** User Interface Subroutines ******
void MPse(void); // pause until spacetbar pressed
int ArchNum(void); // find latest data set number in PAF
void ArchSum(FILE *gp); // summarise archive (PAF)
void RelayOn(int i); // turn Relay i on
void RelayOff(int i); // off
void ReadPA(long *p, long *a); // read P and A
void Goto(long p, long a); // goto P and A position
float CVolt(int gain, int ival); // DMA helper
int GetVal(int j); // DMA helper
void Grab(int frq, int NConv); // DMA
void Bal(float *p, float *a, time_t *PT, time_t *AT); // Balance P,A
Appendix D Ellipsometry Control Programs

void BallZ(void); // 1 zone balance with time stamps
void RepBallZ(int nrep, int nchan, char y[]); // nrep x BallZ with nchan reading
void NBal2Z(char x[]);
int main(int argc, char *argv[]);

// ******

#if Home // Some dummy declarations for debugging at home
#define GLOC "d:\bc\bgi\"
int AL_TERMINATEO {return(0);}  
int AL_INITIALIZEO {return(0);}  
int AL_SELECT_BOARD(int i) {i++; return(0);} 
int AL_RESETO {return(0);}  
int AL_GET_CONFIGURATION(int *i) {return(*i);}  
void DUMP_CONFIGURATIONO {'delay(1); }  
int AL_SETUP_DAC(int i, int j) {i=i+j; return(0);} 
int AL_ENABLE_FOR_OUTPUT(int i) {i++; return(0);} 
int AL_ENABLE_FOR_INPUT(int i) {i++; return(0);} 
int AL_DAC_VALUE(int i, int *j) {i+= *j; return(0);} 
int AL_ADC_VALUE(int i, int j, int *k) {i += j + *k; return(0);} 
int AL_INPUT_DIGITAL_VALUE(int i, int j, int *k) {i ++= j + *k;return(0);} 
int AL_OUTPUT_DIGITAL_VALUE(int i, int j, int k) {i += j + k;return(0);} 
#define AL_CONFIGURATION int
#else
#include "c:\atlc\atlexsub.c"
define GLOC "c:\bc\bgi\"
#endif
jj *******

AL_CONFIGURATION config;
float msin(float x) { return(sin(x * RPD)); } 
float mcos(float x) { return(cos(x * RPD)); } 

int GetVal(int j) { long i; i = j; return( ((unsigned) (seg[2*i]&0xff)) | ((unsigned) ((seg[(2*i)+l]&0xff)«8)) )};

float CVolt(int gain,int ival) { float scr; float unip;  
unip = ival * (10.0 / (4096.0 * GainVal[gain])); return(unip); }

void MTest(int i) { if (i != 0) printf("Error code %d\n",i); else if (dbg) printf("Passed -\n"); }

void Grab(int frq, int NConv)  
{ int unit = 0, rv; buffer = (int *) seg; 
  printf("Now GRAB with %d Hz\n",frq); 
  rv = AL_SET_FREQUENCY((float) frq); MTest(rv); 
  rv = AL_DECLARE_BUFFER(&buffer_number, buffer, NConv); MTest(rv); 
  rv = AL_SET_TIMEOUT(20 ); /* 12.6 seconds. */ MTest(rv); 
  rv = AL_LINK_BUFFER(buffer_number); MTest(rv); 
  rv = AL_BURST_ADC(); MTest(rv); 
  rv = AL_WAIT_FOR_COMPLETION(buffer_number); MTest(rv); delay(2000); 
  rv = AL_RELEASE_BUFFER(&unit,&buffer_number); MTest(rv); } 
void MPse(void) {printf("Pausing\n"); 
do { mychar=(char) getch(); } while ((mychar != '0)&&(mychar != 'e'));
Appendix D Ellipsometry Control Programs

if (mychar == 'e') exit(0);}

char *myctime(time_t *t) { char *x; x=ctime(t); x[24] = '\0'; return(x); }

int ArchNum(void) { /* open PAF archive and find latest ds# */
FILE *fp; int i, rv;
if ((fp = fopen("PAF","r")) == NULL) {
  printf("PAF does not exist hence data set number is to be 0\n"); rv = 0;
  if ((fp = fopen("PAF","w")) == NULL) {
    printf("could not initialise either\n"); exit(8); }
  fprintf(fp,"0 { \n"); fclose(fp); }
else { /* expect at least one # */
  printf("PAF archive exists - now scanning for latest entry \n");
  while (!feof(fp)) {
    i = fscanf(fp,"%d", &rv);
    if (i != 1) { printf("Unexpected condition\n"); exit(9); }
    printf("Latest entry was %d\n", rv); fclose(fp); }
  return(rv); }

int NotSetEnd(FILE *fp) { int i, rv = ''; // check that there is still data
while (!feof(fp) || (!isdigit(rv)) && (((char) rv) != '}
rv = fgetc(fp); if (isdigit(rv)) { ungetc(rv, fp); i=1; }
else i=0; return(i); }

void ArchSum(FILE *gp) { /* open PAF archive and summarize */
FILE *fp; int i, rv, nz, fmt, val; time_t PT, P2T, AT, A2T;
float P, A, P2, A2, thet; if (gp == NULL) gp = stdout;
if ((fp = fopen("PAF","r")) == NULL) {
  printf("PAF does not exist\n"); }
else { /* expect at least one # */
  printf("PAF archive exists - scanning \n");
  while (!feof(fp)) {
    i = fscanf(fp,"%d", &rv);
    if (i != 1) { printf("Unexpected condition on 'iV
"); exit(9); }
    while (!feof(fp)) && (((char) fgetc(fp)) != '}
while (!feof(fp)) && (((char) fgetc(fp)) != '\n')
rv = fgetc(fp); if (isdigit(rv)) { ungetc(rv, fp); i=1; }
else i=0; return(i); }

void ByteVal[3] = {0, 0, 0}, PortAr[3] = {PortA1, PortB1, PortC1};
void MyBitOn(int name) { int i, j;
Appendix D Ellipsometry Control Programs

void MyBitOff(int name) { int i, j;
i = name / 8; j = name % 8;
ByteVal[i] &= (0xff * (1 << j));
if (Klugy) {
    outportb(ControlReg1,0x80);
    outportb(PortA1,ByteVal[0]);
    outportb(PortB1,ByteVal[1]);
    outportb(PortC1,ByteVal[2]); }
else outportb(PortAr[i],ByteVal[i]); }

void MyBit(int name, int value) {
    if (value != 0) MyBitOn(name); else MyBitOff(name); }

void RelayOn(int i) { MyBitOff(i); }

void RelayOff(int i) { MyBitOn(i); }

void MultRelay(int i, int j, int k, int l) {
    if (i > 0) RelayOn(RLY0); else RelayOff(RLY0);
    if (j > 0) RelayOn(RLY1); else RelayOff(RLY1);
    if (k > 0) RelayOn(RLY2); else RelayOff(RLY2);
    if (l > 0) RelayOn(RLY3); else RelayOff(RLY3); }

void SelPA(int pas) {
    if (pas == ANA) { MyBitOn(SEA); MyBitOff(SEP); } else
    { MyBitOff(SEA); MyBitOn(SEP); }
delay(100); }

long INorm(long prp) { int cnt; // normalise
cnt = prp / FULL; prp -= cnt * FULL; if (prp < -HALF) prp += FULL;
if (prp > HALF) /* gtr than 180deg, thus go other way */ prp = prp - FULL;
if (abs(prp) > HALF) { printf("Error likely\007\n"); exit(99); }
return(prp); }

long AShaf(long ii) { return(INorm(18190 - ii)); } // cnvrt Adial to Ase

float Norm(long prp) { return((float) INorm(prp)); } // normalise

long int RawRead(int pas) { int a,b,c,d,f,i; long tot; // read an encoder
SelPA(pas); delay(10); tot = 0;
i = inportb(PortA3); a = i & 0xOff; b = (i & 0x0f0) >> 4;
i = inportb(PortB3); c = i & 0x0f; d = (i & 0xf0) >> 4;
i = inportb(PortC3); /* e = i & 0x0f; */ f = (i & 0xf0) >> 4;
tot = (a * 10) + (b * 100) + f; tot *= 100; tot += (d * 10) + c;
// printf("a,b,c,d,f: %d %d %d %d %d\n",a,b,c,d,f);
return(tot); }
long int Read(int pas) { long tot; tot = RawRead(pas);
if (pas == ANA) tot = INorm(ACOR - tot); return(tot); }

#define TotCritR 10
void ReadPA(long *p, long *a) { long ii, jj; int j, k; // read both encoders
  /* resists noise */ delay(500);
  k = 0; j=0; ii = Read(POL);
  while ((j < CritR) && (k < TotCritR)) {
    jj = Read(POL); delay(30); k++; if (abs(jj - ii) < CritDif) j++;
    else { j=0; ii = jj; } *p=ii; // INorm(ii);
  }
  k=0; j=0; ii = Read(ANA);
  while ((j < CritR) && (k < TotCritR)) {
    jj = Read(ANA); delay(1); k++; if (abs(jj - ii) < CritDif) j++;
    else { j=0; ii = jj; } *a = ii; // INorm(ii);
  }
}

void Chek(int i) { switch (i) { // check AD card for errors
  case E_ADERR: printf("A/D error detected\n"); break;
  case E_GAIN : printf("Illegal gain\n"); break;
  case E_CHANNL: printf("Illegal channel\n"); break;
  case E_NORMAL: break;
  default: printf("Unexpected case\n"); break; }
}

int PhotoSig(void) { int i,ii; // read photodetector signal
  ii = AL_ADC_VALUE(ChanA, 1, &i); Chek(ii); return(i-2048); }

void InitDIO(void) { // see QUATECH PXB-721 manual p6
  outportb(ControlReg3, (char) 0x9b); // config set 3, mode 0, all input
  outportb(ControlReg, (char) 0x9b); // config set 2, mode 0, all input
  outportb(ControlRegl, (char) 0x80); // config setl for all output
  outportb(PortAl,0); outportb(PortBl,0);
  MyBitOff(SEP); MyBitOn(SEA); }

int WriteOut1(int i, int j, int k) {
  outportb(PortA1, (char)i);
  outportb(PortB1, (char)j);
  outportb(PortC1, (char)k);
  return(0); }

int ReadDANA(long *tot) { int i, j, k, m; long a,b,c,d,e,f,cntr;
  // if the bit 0x20 is set in C then was not ready
  if (dbg) { int cnt = 100;
    do { i = inportb(PortA); j = inportb(PortB); k = inportb(PortC);
      printf("Read: %2x %2x %2x\n",i,j,k&0x20); } while (((cnt--) != 0); }
  else { cntr = 0;
    do { i = inportb(PortA);
      j = inportb(PortB);
      k = inportb(PortC);
      cntr++;
      if (cntr >= 10000) {
        printf("DANA not responding - check DATA output is on\n");
        cntr = 0; } } while ((k & 0x20) != 0);
    m = k >> 6;
    k &= 0xf; // mask off DRdy and 2 unused bits
Appendix D Ellipsometry Control Programs

```c

a = (i & 0xf); *tot = a;
b = ((i & 0xfU)>>4); *tot += b * 10;
c = (j & 0xf); *tot += c * 100;
d = ((j & 0xfU)>>4); *tot += d * 1000;
a = ((k & 0x10)>>4) + (k & 0x0c); *tot += a * 10000;
f = (k & 0x01); *tot += f * 100000;
if (m & 0x1) {
    if (dbg) { printf("Negative\n"); MPseO; }
    *tot *= -1; } return(0); }

void VReadDANA(float *x, int DanRan) { long tot;
    ReadDANA(&tot); *x = ((float) tot) * DanRan / 100000.0; }

int My_AL_ADC_VALUE(int chan, int gain, long int *k) { int rv, ks; long kl;
    if (chan >= 0) { rv = AL_ADC_VALUE(chan, gain, &ks); *k = (long int)ks-2048; }
    else { rv = ReadDANA(&kl); *k = kl; } return(rv); }

void SelGain(int i) { // select gain for photodetector amplifier
    Gain = i; switch (i) {
    case 0: Gain = 1.0; break; case 1: Gain = 3.3; break;
    case 2: Gain = 10.0; break; case 3: Gain = 33.0; break;
    case 4: Gain = 100.0; break; case 5: Gain = 333.0; break;
    case 6: Gain = 1000.0; break; case 7: Gain = 3333.0; break;
    default: printf("Invalid gain for ellipsometer amplifier\n"); exit(99); break; }
    i = 7 - i; // new ccT inverts!
    MyBit(DG0, i & 0x1); MyBit(DG1, i & 0x2); MyBit(DG2, i & 0x4);
    delay(100); }

float SmPhotoSig(void) { int i,ii,j = 0; float x; // autoranging read
    SelGain(j); ii = AL_ADC_VALUE(ChanA, 1, &i); Chek(ii);
    if (i-2048 > MaxSig) {
        printf("Warning[007 - amplifier output near Vrail\n");
    }
    while ((i-2048 < MinSig)&&(j < 7)) { j++;
        SelGain(j); ii = AL_ADC_VALUE(ChanA, 1, &i); Chek(ii); }
    x = i-2048/(4.096 * Gain); return(x); }

void RdySt(void) {StUpDel=StUpDelDf;SdnDel=StInDelDf;} //ready for ramp

void Step(int pas, int cc) { static int lpas = -9, lcc = -9; // mv step mtr
    int j, i = MyCode[pas][cc]; for (j=0; j<ssz; j++) {
    if (Ramp) |
        if ((lcc == cc) && (lpas == pas)) {
            if (StUpDel > StUpDelMn) StUpDel -= 1;
            if (StInDel > StInDelMn) StInDel -= 1; }
            else { StUpDel = StUpDelDf; StInDel = StInDelDf; } }
    if (cc == CW) (MyLoc[pas])++; else (MyLoc[pas])--;
    MyBitOn(i); delay(StInDel); MyBitOn(i); delay(StUpDel); MyBitOff(i); }

void MyMov(int pas, int count) { int kssz; kssz = ssz; RdySt();
    if (count < 0) { ssz = -count; Step(pas, CW); }
    else { ssz = count; Step(pas, CCW); } ssz = kssz; }

// check with McCrackin/63
```
Appendix D Ellipsometry Control Programs

int Zone(long p, long a) { int i; p = p; // get rid of irritating warning
  printf("Calculating Zone\n");
  if (tan(a*Radian/100.0) > 0.0) i = 1; else i=3;
  printf("Zone %d\n",i);
  return(i); }

void DetermGain(void) { // select suitable gain for sweep
  int g; SmPhotoSig(); g = GainL;
  printf("Agreement reached on a gain of %d for the sweep\n",g); }

void Sweep(int pas, int ndat, float dat[]) { int i, j, ORamp;
  float x, xl = -999.0, gl; time_t TimIni, TimFi; ORamp = Ramp; Ramp = 0;
  ndat = ndat & 0xfffe; time(&TimIni); PosIni = Read(pas); RdyStO;
  for (i=0; i<ndat/2; i++) Step(pas,CW); RdyStO; DetermGain();
  for (i=0; i<ndat; i++) { x = 0.0;
    for (j=0; j<grp; j++) x += PhotoSigO; x = (dat[i] = x/j);
    if ((xl > 0.0) && ((fabs(xl - x) > CritStep)) { printf("Possible glitch at %d \n\n\t\t(last %f with %f)",i,x,gl);
      /* dat[i] = xl; */ else { xl = x; gl = Gain; }
      Step(pas,CCW); } } 
  x = 0.0; for (j=0; j<grp; j++) x += PhotoSigO; dat[ndat] = x/j;
  RdyStO; for (i=0; i<ndat/2; i++) Step(pas,CW);
  time(&TimFi); PosFin = Read(pas); Ramp = ORamp;
  mytime = (TimFi + TimIni)/2; }

void DispDat(int np, float y[], float yff[]) { float miny, maxy, scy, scx;
  int detect = 0, gmode, i, a, b, c, d, top; maxy = miny = y[0];
  for (i=0;i<=np;i++) {
    if (Display > 1) { if (miny > yf[i]) miny = yf[i];
      if (maxy < y[i]) maxy = y[i];
    } else { if (maxy != miny) {
      intgraph(&detect, &gmode, GLOC); setcolor(LIGHTBLUE);
      scy = ((float) (top = getmaxy())) / (maxy - miny);
      scx = ((float) getmaxxO) / (np + 1);
      for (i=0; i<np; i++) { a = i * scx; b = (y[i] - miny)*scy; putpixel(a,top - b,MAGENTA); }
      if (Display > 1) { b = top - ((yf[np] - miny)*scy); LineCounter++; }
      for (i=np-1; i>=0; i--) { c = i * scx; d = top - ((yf[i] - miny)*scy); LineCounter++; } 
      printf("Miny: %f Maxy: %f\n",miny,maxy);
      if (Running) { printf("\n"); fflush(stdout); delay (1000); } else MPseO; closegraphO; } }

void AdjPos(float x, int pas) { int i, kssz = ssz; i = numdat/2;
  if (x > i) { printf("+Clipping the move from %f (numdat=%d)\n",x,numdat);
    x = i; } else { if (x < -i) { printf("-Clipping the move from %f (numdat=%d)\n",x,numdat);
      x = -i; } }
Appendix D Ellipsometry Control Programs

float Fit(int n, float y[], int pas) { int i, iminy; FILE *fp;
float N, Sn[3], Cn[3], A[3], x, miny;
char Typ[2][20] = {"Polariser","Analyser"};
N = n; miny = y[0]; iminy = 0;
for (i=0; i<=n; i++) { /* beware indexing here !!!*/
if (y[i] < miny) { miny = y[i]; iminy = i; } // keep min for poor fits
f0[i] = 1.0; f1[i] = 1 - ((2.0 * i)/N);
f2[i] = (3 * (N - (2.0 * i)) * f1[i]) - (N+2);
f2[i] /= 2*(N-1);
}
for (i=0;i<3;i++) { A[i] = Cn[i]/Sn[i];
// now calculate f'(x)=0, ( rather than work with f(x) )
x = (A[1]*(N-1)/(6*A[2])) + (N/2.0);
// check that f'(x) != 0, if so then take minimum
if (A[2] < 0.0) { // then concave down -> maximum! so take min of data
printf("Detected maximum rather than minimum: ")
"moving to minimum of data\n007\n");
NWB = 1; x = (float) iminy; }
if ((x < (0.3*N)) || (x > (0.7*N)))
{ printf("Not well balanced\n007\n"); NWB = 1; }
for (i=0; i<=numdat; i++)
{ printf("x was %f\n",x); delay(2000); }
LastFitfpas[] = x;
}
if (Movlt) AdjPos(x,pas);
if (Graph) { char fnam[80];
sprintf(fnam,"%lx.dat",(long) Oxffffffff & mytime);
printf("Now writing the file: %s
",fnam);
if ((fp = fopen(fnam,"w"))==NULL) { printf("Open error\n007\n"); exit(99); } 
for (i=0;i<6;i++)
{ fprintf(fp,"%3d %6.4f %f
",i,y[i],yf[i]); }
fclose(fp);
}
if (((fp = fopen(fnam,"w"))==NULL) { printf("Open error\n007\n"); exit(99); } 
fprintf(fp,"1244
"DEFAULT" "DEFAULT" 84 l
"2.000e+000 6.000e+000 0.000e+000 2.000e-001
4
"0 27 "(DS %d) Balance of %s"
",dsnum,Typ[pas]);
fprintf(fp,"1 27 "near %6.3f degrees"
",PosIn/100.0);
fprintf(fp,"2 27 "recorded at: %lx"
",mytime);
fprintf(fp,"3 27 "%22s "
"myctime(&mytime));
fclose(fp); }

void Stats(int nd, float nstd, float *av, float *std, float vect[]) {
int i, lcnt; float oav; *av = oav = 0.0; *std = 0.0; lcnt = 0;
for (i=0; i<nd; i++) oav += vect[i]; oav /= nd;
for (i=0; i<nd; i++) *std += sqr(oav - vect[i]);
}
Appendix D Ellipsometry Control Programs

*std = sqrt(*std / nd);
for (i=0; i<nd; i++) if (fabs(vect[i] - oav) < (nstd * *std))
    { *av += vect[i]; lcnt++; }
if (lcnt < 1) { printf("Too much scatter in Stats!"); exit(99); }
*av /= lcnt;
printf("Av: %5.3f St: %5.3f\n", *av, *std);

void Goto(long p, long a) { long prp, pra; int cnt, ORamp;
    ReadPA(&prp, &pra); ORamp = Ramp; Ramp = 1;
    printf(" presently at P=%ld A=%ld (Ramp=%ld DelU=%ld DelD=%ld)\n", 
        prp, pra, Ramp, StUpDel, StInDel);
    printf(" moving to ");
    if (p < 99998) {
        printf(" P=%ld", p);
        prp -= p;
        cnt = prp / FULL; prp -= cnt * FULL; if (prp < -HALF) prp += FULL;
        if (prp > HALF) /* gtr than 180deg, thus go other way */
            prp = prp - FULL;
        if (fabs(prp) > HALF) /* (abs(pra) > HALF) */
            printf("Error likely\n"); exit(99); 
        MyMov(POL, prp); } else prp = 0;
    if (a < 99998) {
        printf(" A=%ld\", a);
        pra = a; /* now find shortest path */
        cnt = pra / FULL; pra -= cnt * FULL; if (pra < -HALF) pra += FULL;
        if (pra > HALF) /* gtr than 180deg, thus go other way */
            pra = pra - FULL;
        printf("stepping P=%ld A=%ld\", prp, pra);
        MyMov(ANA, pra); ReadPA(&prp, &pra); } 
    printf(" presently at P=%ld A=%ld\n", prp, pra);
    printf("differences P:%ld A:%ld\n", prp-p, pra-a);
    Ramp = ORamp; 
}

void GotoP(long p) { Goto(p, (long) 99999); }
void GotoA(long a) { Goto((long) 99999, a); } 

void RBal(float x) { // roughly null P, A within x deg
    long int p,a,postn,offs, mnpos; float rdng, mnrdng;
    if (RBALOO) {
        ReadPA(&p, &a); mnrdng = 9999.0; offs = 0.5 *(x * 100.0);
        for (postn=p-offs; postn < p+offs; postn += 10) {
            GotoP(postn); rdng = SmPhotoSig();
            if (rdng < mnrdng) { mnrdng = rdng; mnpos = postn; } }
        GotoP(mnpos);
        mnrdng = 9999.0;
        for (postn=a-offs; postn < a+offs; postn += 10) {
            GotoA(postn); rdng = SmPhotoSig();
            if (rdng < mnrdng) { mnrdng = rdng; mnpos = postn; } }
        GotoA(mnpos); } else printf("Rough Balancing turned off\n"); }

void DFlip() { long int p; long int a;
    /* determine which zone, update position of NULL, move to other NULL */
    float dp, da, disA, disB;
    ReadPA(&p, &a); 
    printf("Now in DFLIP with p=%ld a=%ld\", p, a);
    dp = p/100.0; da = a/100.0;
printf("I'd like to have: 
 p=%10.3f a=%10.3f\n",dp+90.0,180.0-da);
p = (dp + 90.0) * 100; a = (180.0 - da)*100.0; Goto(p,a);
RBal(5.0); }

void BalP(float *p, time_t *PT) {
Sweep(POL,numdat,data); *PT = mytime; lastPstd = 0.0;
*p = lastPavg = (PosIn - (ssz * (Fit(numdat, data, POL)))) /100.0; 
}

void BalA(float *a, time_t *AT) {
Sweep(ANA,numdat,data); *AT = mytime;
*a = lastAavg = (PosIn - (ssz * (Fit(numdat, data, ANA)))) /100.0;}

void Bal(float *p, float *a, time_t *PT, time_t *AT) { int cnt; cnt = 0;
do { NWB = 0; BalP(p,PT); BalA(a,AT);
if ((++cnt >= 10)&&(NWB != 0)) {
  printf("Warning: need assist!\007\007\n"); MPseO; cnt = 0; }
} while (NWB > 0); printf("NULL: p= %f a= %f\n",*p,*a); }

void BallZ(void) { float P, A; FILE *fp;
Bal(&P,&A,&PT,&AT); if (Arch) {
printf("Now archiving\n");
if ((fp = fopen("PAF","a")==NULL) { printf("Open error\007\n"); exit(99); }
fprintf(fp,"%6.3f 1 1 %6.3f %lx %6.3f %lx\n",Theta,P,PT,A,AT);
}
fclose(fp); }

void RepBallZ(int nrep, int nchan, char *y[] ) { // nrep x BalZ with nchan reading
int i, ii, j, fmtcd; long int jl; float P, A; time_t PT, AT; FILE *fp;
Running = 1; if (nchan > -2) fmtcd = 2; else fmtcd = 1;
for (i=0;i<nrep;i++) {
if (Predictive && (i > 0)) {
  for (j=0; j<2; j++) NLastFit[j] = LastFit[j];
  AdjPos(LastFit[POL],POL);
  AdjPos(LastFit[ANA],ANA); }
Bal(&P,&A,&PT,&AT);
if (i >= APArSz) { printf("Error - too many repetitions\007\n"); exit(8);}
PA[i] = P; AAr[i] = A;
if (Predictive) for (j=0; j<2; j++) LastFit[j] = NLastFit[j];
ii = My_AL_ADC_VALUE(nchan, 1, &jl); Chek(ii);
if (Arch) { printf("Now archiving\n");
if ((fp = fopen("PAF","a") ==NULL) { printf("Open error\007\n"); exit(99); }
if (nchan > -2) {
  if (i == 0) { dsnum++; fprintf(fp,"\n%d \(\%s\) \{\n",dsnum,y[0];
  fprintf(fp,"%6.3f 1 %ld %6.3f %lx %6.3f %lx \%f\n",
    Theta,fmtcd,P,PT,A,AT,jl/1000.0);
  fprintf("%6.3f 1 %ld %6.3f %lx %6.3f %lx \%f\n",
    Theta,fmtcd,P,PT,A,AT,LastDANA = jl/1000.0);
  else { LastDANA = 0.0;
  if (i == 0) { dsnum++; fprintf(fp,"\n%d \(\%s\) \{\n",dsnum++,y[0];
  fprintf(fp,"%6.3f 1 %ld %6.3f %lx %6.3f %lx \%f\n",
    Theta,fmtcd,P,PT,A,AT);
  fprintf("%6.3f 1 %ld %6.3f %lx %6.3f %lx \%f\n",Theta,fmtcd,P,PT,A,AT);}}
}

Appendix D Ellipsometry Control Programs

printf("Ideally would have: 
 p=%10.3f a=%10.3f\n",dp+90.0,180.0-da);
p = (dp + 90.0) * 100; a = (180.0 - da)*100.0; Goto(p,a);
RBal(5.0); }

void BalP(float *p, time_t *PT) {
Sweep(POL,numdat,data); *PT = mytime; lastPstd = 0.0;
*p = lastPavg = (PosIn - (ssz * (Fit(numdat, data, POL)))) /100.0; 
}

void BalA(float *a, time_t *AT) {
Sweep(ANA,numdat,data); *AT = mytime;
*a = lastAavg = (PosIn - (ssz * (Fit(numdat, data, ANA)))) /100.0; 

void Bal(float *p, float *a, time_t *PT, time_t *AT) { int cnt; cnt = 0;
do { NWB = 0; BalP(p,PT); BalA(a,AT);
if ((++cnt >= 10)&&(NWB != 0)) {
  printf("Warning: need assist!\007\007\n"); MPseO; cnt = 0; }
} while (NWB > 0); printf("NULL: p= %f a= %f\n",*p,*a); }

void BallZ(void) { float P, A; FILE *fp;
Bal(&P,&A,&PT,&AT); if (Arch) {
printf("Now archiving\n");
if ((fp = fopen("PAF","a") ==NULL) { printf("Open error\007\n"); exit(99); }
fprintf(fp,"%6.3f 1 1 %6.3f %lx %6.3f %lx\n",Theta,P,PT,A,AT);
}
fclose(fp); }

void RepBallZ(int nrep, int nchan, char *y[] ) { // nrep x BalZ with nchan reading
int i, ii, j, fmtcd; long int jl; float P, A; time_t PT, AT; FILE *fp;
Running = 1; if (nchan > -2) fmtcd = 2; else fmtcd = 1;
for (i=0;i<nrep;i++) {
if (Predictive && (i > 0)) {
  for (j=0; j<2; j++) NLastFit[j] = LastFit[j];
  AdjPos(LastFit[POL],POL);
  AdjPos(LastFit[ANA],ANA); }
Bal(&P,&A,&PT,&AT);
if (i >= APArSz) { printf("Error - too many repetitions\007\n"); exit(8);}
PA[i] = P; AAr[i] = A;
if (Predictive) for (j=0; j<2; j++) LastFit[j] = NLastFit[j];
ii = My_AL_ADC_VALUE(nchan, 1, &jl); Chek(ii);
if (Arch) { printf("Now archiving\n");
if ((fp = fopen("PAF","a") ==NULL) { printf("Open error\007\n"); exit(99); }
if (i >= APArSz) { printf("Error - too many repetitions\007\n"); exit(8);}
PA[i] = P; AAr[i] = A;
if (Predictive) for (j=0; j<2; j++) LastFit[j] = NLastFit[j];
ii = My_AL_ADC_VALUE(nchan, 1, &jl); Chek(ii);
if (Arch) { printf("Now archiving\n");
if ((fp = fopen("PAF","a") ==NULL) { printf("Open error\007\n"); exit(99); }
if (nchan > -2) {
  if (i == 0) { dsnum++; fprintf(fp,"\n%d \(\%s\) \{\n",dsnum,y[0];
  fprintf(fp,"%6.3f 1 %ld %6.3f %lx %6.3f %lx \%f\n",
    Theta,fmtcd,P,PT,A,AT,jl/1000.0);
  fprintf("%6.3f 1 %ld %6.3f %lx %6.3f %lx \%f\n",
    Theta,fmtcd,P,PT,A,AT,LastDANA = jl/1000.0);
  else { LastDANA = 0.0;
  if (i == 0) { dsnum++; fprintf(fp,"\n%d \(\%s\) \{\n",dsnum++,y[0];
  fprintf(fp,"%6.3f 1 %ld %6.3f %lx %6.3f %lx \%f\n",
    Theta,fmtcd,P,PT,A,AT);
  fprintf("%6.3f 1 %ld %6.3f %lx %6.3f %lx \%f\n",Theta,fmtcd,P,PT,A,AT);}}}
Appendix D Ellipsometry Control Programs

```c
fclose(fp); } } Running = 1; printf("Computing Stats\n");
Stats(nrep, 2.0, &lastAavg, &lastAstd, AAr);
Stats(nrep, 2.0, &lastPavg, &lastPstd, PAR);
printf("Stats and RepBal1Z completed\n");
if (StopBal) { printf("007\007\007You requested that I pause after\n"
"balancing in this zone\n"); MPseO; } }

void NBal2Z(char x[]) { double scr;
   RepBal1Z(5, 0, x);
   Aprime = lastAavg;
   Pprime = lastPavg;
   DFlipO;
   RepBal1Z(5, 0, x);
   Athree = lastAavg;
   Pthree = lastPavg;
   // see DS p92
   if (Zone(Pprime, Aprime) != 1) {
      scr = Aprime; Aprime = Athree; Athree = scr;
      scr = Pprime; Pprime = Pthree; Pthree = scr; } }

#define MaxV 10.0
#define MinV (-10.0)

int VtoDac(float x) { float y; /* assume 12 bits, +10V */
if ((x<MinV) | |(x>MaxV)) {printf("Error, setting x=%5.3f\n", x=MinV; }
   y = ((1 « 12) - 1)*(x - MinV)/(MaxV - MinV);
   printf("Over range %5.3f - %5.3f, estimate %5.3fV as %5.3f bits\n", MinV,MaxV,x,y); return((int)y); }

void DAC(float V) { int i; i = VtoDac(V); AL_DAC_VALUE(0 /* DACO */, &i); }

float ADC(int chan) { float x; int j; long il; x = 0;
for (j=0; j<100; j++) { My_AL_ADC_VALUE(chan, 1, &il);
   x += il/40960.0; } return(x); }

void DkpFlgs(void) {
   printf("Dkplay mode k %d
Number of readings/point k %d (grp)
Number of steps/point k %d (ssz)
MovIt flag k %d
Ramp flag is %d
Predictive balance flag k %d
Number of points per balance is %d\n", Dkplay, grp,
   ssz, MovIt, Ramp, Predictive, numdat); }

void SetFlgs(void) { DkpFlgsO;
   printf("Enter: Dkpky, grp, ssz, Movlt, Ramp: ");
   scanf("%d %d %d %d %d", &Dkpky,&grp,&ssz,&MovIt,&Ramp);
   printf("What angle of incidence (presendy %6.3f): ",Theta);
   scanf("%f",&Theta);
   printf("Use predictive balancing (0 or 1, presently %d): ",Predictive);
   scanf("%d",&Predictive);
   printf("Write GRAPHER files ? (0/1, presently %d): ",Graph);
   scanf("%d",&Graph);
   printf("Set Numdat "); scanf("%d",&numdat);
```

if (numdat > (MaxD - 1))
    printf("\007Too large, set to %d\n",numdat = MaxD-1); }

void SavDat(char fnam[], int np, float fsam, float y[]) {
    char *ptr, fulfnam[80]; time_t t; FILE *fp; int i;
    sprintf(fulfnam,"%s.dat",fnam);
    if ((fp = fopen(fulfnam,"w")) == NULL) {
        printf("FOPEN Error\007\n"); exit(88); }
    for (i=0; i<np; i++) fprintf(fp,"%6.4f %6.4f\n",i/fsam,y[i]);
    fclose(fp);
    ptr = strchr(fnam,':'); if (ptr == NULL) ptr = fnam; else ptr += 1;
    sprintf(fulfram,\"c:\chris\ndeldat\%s.dat\",ptr);
    if ((fp = fopen(fulfnam,"w")) == NULL) {
        printf("FOPEN Error\007\n"); exit(88); }
    for (i=0; i<np; i++) fprintf(fp,"%6.4f %6.4f\n",i/fsam,y[i]);
    fclose(fp);
    if ((fp = fopen("c:\chris\ndeldat\log","a")) == NULL) {
        printf("FOPEN Error\007\a"); exit(88); }
    time(&t); fprintf(fp,"%s %s %6.5f %6.5f %s\n",mam,samnam,Il j2jl,ctime(&t)); fclose(fp);
    if ((fp = fopen("log","a")) == NULL) {
        printf("FOPEN Error\007\n"); exit(88); }
    time(&t); fprintf(fp,"%s %s %6.5f %6.5f %s\n",fnam,samnam,Il j2jl,ctime(&t)); fclose(fp);}

int HaveSt(FILE *fp) { int i = 1;
    while ( ( ((char)fgetc(fp)) != '{') && (!feof(fp)) )
        if (feof(fp)) i = 0; return(i); }

void GetCal(int LimOpt) { float Fthet; int Flag3, Flag4, Flag7, Flag8, Opt;
    time_t t; FILE *fp;
    Flag3 = Flag4 = Flag7 = Flag8 = 0;
    if ((fp = fopen("pabals\TALLYPA.DAT","r")) == NULL) {
        printf("FOPEN Error\n"); exit(99); }
    time(&t);
    while (!feof(fp)) {
        while (HaveSt(fp)) {
            fscanf(fp,"%d",&Opt);
        }
        switch (Opt) {
            case 3: fscanf(fp, "%f",&Fthet); Flag3 = 1;
                if (fabs(Fthet - Theta) < 0.1) {
                    fscanf(fp, "%f %f",&Aint,&Pint);
                    printf("Lookup (%5.3f): Aint = %f Pint = %f\n",Fthet,Aint,Pint); }
                break;
            case 4: if (LimOpt > 4) {
                    fscanf(fp,"%f %f", &DC, &DA); Flag4 = 1;
                    printf("Lookup (MAI): DC = %5.3f DA = %5.3f\n",DC,DA); }
                break;
            case 7: if (LimOpt > 7) { fscanf(fp,"%f",&Fthet);
                if (fabs(Fthet - Theta) < 0.1) {
                    fscanf(fp,"%f %f %f", &t1c, &t2c); Flag7 = 1;
                    printf("Lookup (%5.3f): t1c = %5.3f t2c = %5.3f\n",Fthet,t1c,t2c); }
                break;
            case 8: if (LimOpt > 8) { fscanf(fp,"%f %f", &t2p, &npc); Flag8 = 1;
                    printf("Lookup (MAI): t2p = %5.3f npc = %5.3f\n",t2p,npc); }
                break;
        }
    }
}
Appendix D Ellipsometry Control Programs

break;
    default: printf("Warning - case not known\007\n"); exit(9); break; })
if ( (Flag3 && (LimOpt > 3)) || (Flag4 && (LimOpt > 4)) ||
    (Flag7 && (LimOpt > 7)) || (Flag8 && (LimOpt > 8)) ) {
    printf("Warning - incomplete calibration\007\n");
    printf("Status: 3,4,7,8 : %d %d %d %d\n",Flag3,Flag4,Flag7,Flag8);
fclose(fp); MPseO; }

int Stupidity(void) { int i, j, k; long int P, A, P2; k = 0;
    SelGain(0); i = PhotoSig0; SelGain(7); j = PhotoSig0;
    if (j < 10) { printf("Check Detector - may have no power\007\n"); k=1; } else
    if (i > 10) { printf("Check Detector - may be on manual\007\n"); k =1; }
P = RawRead(POL); A = RawRead(ANA);
    if (P == A) {
        printf("Check Shaft Encoder Controller - probably on manual (or off)\007\n");
        k = 1; }
    else {
        Step(POL, CCW); P2 = RawRead(POL);
        if (P2 == P) {
            printf("Check Step Motor Power - May be off\007\n"); k = 1; }
        Step(POL, CCW);
    }
    return(k); }
Appendix E

Parameters from the Cole-Cole Plots

<table>
<thead>
<tr>
<th>T(K)</th>
<th>J (mA/cm²)</th>
<th>A/qm</th>
<th>(A+B)/qm</th>
<th>τₘ (sec×2π)</th>
<th>θ</th>
</tr>
</thead>
<tbody>
<tr>
<td>273.8</td>
<td>0.030</td>
<td>4.99</td>
<td>13.80</td>
<td>18.2</td>
<td>0.21</td>
</tr>
<tr>
<td>273.8</td>
<td>0.088</td>
<td>4.29</td>
<td>13.73</td>
<td>6.14</td>
<td>0.31</td>
</tr>
<tr>
<td>273.8</td>
<td>0.26</td>
<td>4.17</td>
<td>12.77</td>
<td>1.76</td>
<td>0.29</td>
</tr>
<tr>
<td>273.8</td>
<td>0.79</td>
<td>3.92</td>
<td>12.19</td>
<td>0.78</td>
<td>0.31</td>
</tr>
<tr>
<td>298.0</td>
<td>0.029</td>
<td>5.70</td>
<td>15.82</td>
<td>23.28</td>
<td>0.21</td>
</tr>
<tr>
<td>298.0</td>
<td>0.088</td>
<td>4.58</td>
<td>15.56</td>
<td>6.63</td>
<td>0.33</td>
</tr>
<tr>
<td>298.0</td>
<td>0.26</td>
<td>4.54</td>
<td>14.04</td>
<td>1.86</td>
<td>0.28</td>
</tr>
<tr>
<td>298.0</td>
<td>0.79</td>
<td>3.41</td>
<td>13.47</td>
<td>0.74</td>
<td>0.40</td>
</tr>
<tr>
<td>323.0</td>
<td>0.029</td>
<td>5.55</td>
<td>17.99</td>
<td>26.31</td>
<td>0.31</td>
</tr>
<tr>
<td>323.0</td>
<td>0.088</td>
<td>5.12</td>
<td>17.39</td>
<td>8.57</td>
<td>0.35</td>
</tr>
<tr>
<td>323.0</td>
<td>0.27</td>
<td>4.94</td>
<td>15.34</td>
<td>2.45</td>
<td>0.30</td>
</tr>
<tr>
<td>323.0</td>
<td>0.79</td>
<td>4.83</td>
<td>15.41</td>
<td>0.89</td>
<td>0.33</td>
</tr>
<tr>
<td>348.0</td>
<td>0.03</td>
<td>6.14</td>
<td>19.88</td>
<td>31.9</td>
<td>0.30</td>
</tr>
<tr>
<td>348.0</td>
<td>0.088</td>
<td>5.70</td>
<td>19.33</td>
<td>11.0</td>
<td>0.34</td>
</tr>
<tr>
<td>348.0</td>
<td>0.27</td>
<td>5.14</td>
<td>17.61</td>
<td>2.95</td>
<td>0.38</td>
</tr>
<tr>
<td>348.0</td>
<td>0.80</td>
<td>5.00</td>
<td>17.58</td>
<td>1.12</td>
<td>0.39</td>
</tr>
</tbody>
</table>

Table E1 The fitted parameters from Cole-Cole plot. T is the temperature, J is the steady state current density. A, B, τₘ, and θ correspond to the parameters shown in Eq. 4.4.1.1, q is proton charge, and m is meter.