AD-A102 774   AEROSPACE CORP EL SEGUNDO CA CHEMISTRY AND PHYSICS LAB   F/G 10/3
TRANSIENT TECHNIQUES FOR BATTERY IMPEDANCE MEASUREMENTS, SMALL --ETC(U)
JUL 81   A H ZIMMERMAN, M C JANECKI   F04701-80-C-0081
UNCLASSIFIED   TR-0081(6970-01)-2   SD-TR-81-46   NL

1 OF 1
AD A
102774

REPORT SD-TR-81-46

Transient Techniques for Battery Impedance Measurements Small-Amplitude Exponential Perturbation Technique

Prepared by

A. H. ZIMMERMAN and M. C. JANECKI
Chemistry and Physics Laboratory
The Aerospace Corporation
El Segundo, Calif. 90245

1 July 1981

Prepared for

SPACE DIVISION
AIR FORCE SYSTEMS COMMAND
Los Angeles Air Force Station
P.O. Box 92960, Worldway Postal Center
Los Angeles, Calif. 90009

[illegible]

This report was submitted by The Aerospace Corporation, El Segundo, CA 90245, under Contract No. F04701-80-C-0081 with the Space Division, Deputy for Technology, P.O. Box 92960, Worldway Postal Center, Los Angeles, CA 90009. It was reviewed and approved for The Aerospace Corporation by S. Siegel, Director, Chemistry and Physics Laboratory. Lieutenant Thomas D. Hebblewaite, SD/YLVS, was the project officer for the Mission Oriented Investigation and Experimentation (MOIE) Program.

This report has been reviewed by the Public Affairs Office (PAS) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication. Publication of this report does not constitute Air Force approval of the report's findings or conclusions. It is published only for the exchange and stimulation of ideas.

Thomas D. Hebblewaite
Thomas D. Hebblewaite, 1st Lt, USAF
Project Officer

Florian P. Meinhardt
Florian P. Meinhardt, Lt Col, USAF
Director of Advanced Space Development

FOR THE COMMANDER

William Goldberg
William Goldberg, Colonel, USAF
Deputy for Technology

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
SD-TR-81-46

2. GOVT ACCESSION NO.
AD-A102 774

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (*and Subtitle*)
TRANSIENT TECHNIQUES FOR BATTERY IMPEDANCE MEASUREMENTS, SMALL AMPLITUDE EXPONENTIAL PERTURBATION TECHNIQUE

5. TYPE OF REPORT & PERIOD COVERED

6. PERFORMING ORG. REPORT NUMBER
TR-0081(6970-01)-2

7. AUTHOR(s)
A. H. Zimmerman and M. C. Janecki

8. CONTRACT OR GRANT NUMBER(s)
F04701-80-C-0081

9. PERFORMING ORGANIZATION NAME AND ADDRESS
The Aerospace Corporation
El Segundo, Calif. 90245

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Space Division
Air Force Systems Command
Los Angeles, Calif. 90009

12. REPORT DATE
1 July 1981

13. NUMBER OF PAGES
24

14. MONITORING AGENCY NAME & ADDRESS (*if different from Controlling Office*)

15. SECURITY CLASS. (*of this report*)
Unclassified

15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (*of this Report*)
Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (*of the abstract entered in Block 20, if different from Report*)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*)
Battery
Impedance
Transforms
Transients

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*)

A perturbation technique is reported for measuring the impedance of battery cells under conditions of controlled potential. The small amplitude exponential perturbation (SAEP) technique is applicable over an extremely wide frequency range and appears to be the method of choice for measuring the impedance of battery cells that contain very little stored electrochemical energy.

DD FORM 1473 (FACSIMILE)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

CONTENTS

APPENDIX

FIGURES

I. INTRODUCTION

The proper operation of battery cells invariably depends on a number of internal physical and chemical reactions occuring at rates that are sufficient to sustain cell performance. These reactions typically involve charge transfer processes at the electrodes, as well as diffusional transport of materials to the active electrode surfaces. Kinetic measurements permit determination of the relative importance of these processes in controlling cell performance. The most general method for making these kinetic measurements is to measure the electrical impedance of the battery cell as a function of frequency. The rates of the various processes that affect the cell voltage are inferred directly from the frequency dispersion of the cell impedance.

A number of techniques have been used to measure the impedance of battery cells. The most commonly used is that of applying a sinusoidally varying ac signal to the battery cell and monitoring the cell response in terms of amplitude and ac phase shift. This ac method is relatively easy to use, but if data are required over a wide frequency range or at very low frequencies, it becomes somewhat cumbersome. Other techniques for impedance measurements of battery cells incorporate perturbing functions other than sinusoidal ac. For example, in the galvanostatic transient technique[1] a step change in the current passing through the cell is applied, and the response of the cell to the current change is measured. The relationship between the change in cell current and the voltage response gives the cell impedance. This technique is particularly useful when the cell contains appreciable stored capacity, since in this case controlling cell current is much easier than controlling cell voltage. However, when the cell contains very little stored capacity, any measurement attempted under conditions of constant current may change the cell voltage by a large amount and thereby appreciably alter the chemical state of the cell. In this situation, it is desirable to employ a potentiostatic

---

[1] A. H. Zimmerman and M. R. Martinelli, Transient Techniques for Low Frequency Impedance Measurements, TR-0079(4970-10)-1, The Aerospace Corporation, El Segundo, Calif (6 October 1978).

technique that involves the application of a controlled perturbation to the cell potential.

We have developed and applied such a technique to battery cells. This technique is called small amplitude exponential perturbation (SAEP) and involves perturbing the cell voltage with a small amplitude (<5 mV) exponential signal while measuring the current response of the cell. Again, the cell impedance is obtained from the relationship between voltage and current. This technique can be used to measure the impedance of battery cells at any voltage or state of charge that is accessible to them, although very large currents (and power supplies) may be involved when the cell has appreciable active electrochemical capacity.

II. THEORY OF SAEP

Any potentiostatic transient technique for measuring impedance employs a transient potential function V(t). This potential function is applied as a perturbation to a battery cell that has the initial potential V_o. The cell current is initially $I_o + I_N(t)$, where I_o is the steady-state current at V_o, and $I_N(t)$ is any change in current resulting from depletion of the stored electrochemical capacity of the cell at the initial voltage. After the perturbation V(t) is applied, the current is $I_o + I_N(t) + I(t)$. For this analysis to be correct, the amplitude of V(t) must be sufficiently small that $I_N(t)$ does not change appreciably in response to V(t). This means that typically V(t) should be less than 5 mV in amplitude. In addition, the time constant associated with $I_N(t)$ must be much greater than that associated with I(t) so that they can be separated in time.

The impedance as a function of time is then directly given by Ohm's law

$$Z(t) = \frac{V(t)}{I(t)} \qquad (1)$$

However, the cell impedance is more conveniently analyzed in the frequency domain. Laplace transformation of V(t) and I(t) permits us to obtain the impedance as a function of frequency.

$$Z(\omega) = \frac{V(\omega)}{I(\omega)} \qquad (2)$$

where $V(\omega)$ and $I(\omega)$ are the Laplace transforms of voltage and current, respectively.

The digital Laplace transforms required are calculated from the voltage and current data,

$$F(\omega) = \int_0^\infty f(t) \exp(-j\omega t)\, dt = \sum_i \int_{t_i}^{t_{i+1}} f_i(t) \exp(-j\omega t) dt \tag{3}$$

which are digitized by computer into arrays having i data points, each corresponding to a given time. The function $f_i(t)$ fits the data for $f(t)$ in the interval t_i to t_{i+1} and may be any convenient function that fits the data. Functions used for $f(t)$ include linear, quadratic, and exponential forms as follows.

1. Linear: $f_i(t) = A_i t + B_i$

$$A_i = \frac{f(t_i) - B_i}{t_i}$$

$$B_i = \frac{\lfloor f(t_{i+1}) - t_{i+1}/t_i\, f(t_i) \rfloor}{1 - \frac{t_{i+1}}{t_i}} \tag{4}$$

2. Quadratic: $f_i(t) = L_i t^2 + M_i t + N_i$

$$M_i = \left\lfloor \frac{\Delta f_{12}}{\Delta t_{12}^2} \left(\frac{t_1^2}{t_{i+2}} - t_{i+2} \right) - \frac{\Delta f_{13}}{t_{i+2}} \right\rfloor \left\lfloor 1 - (t_{i+2}) \frac{\Delta t_{12}}{\Delta t_{12}^2} + \frac{t_i}{t_{i+2}} \left(\frac{t_i \Delta t_{12}}{\Delta t_{12}^2} - 1 \right) \right\rfloor^{-1}$$

$$L_i = \frac{\Delta f_{12}}{\Delta t_{12}^2} - M_i \left(\frac{\Delta t_{12}}{\Delta t_{12}^2} \right) \tag{5}$$

$$N_i = f(t_i) - L_i t_i^2 - M_i t_i$$

where

$$\Delta t_{12} = t_i - t_{i+1}$$

$$\Delta t_{12}^2 = t_i^2 - t_{i+1}^2$$

$$\Delta f_{12} = f(t_i) - f(t_{i+1})$$

$$\Delta f_{13} = f(t_i) - f(t_{i+2})$$

3. Exponential:

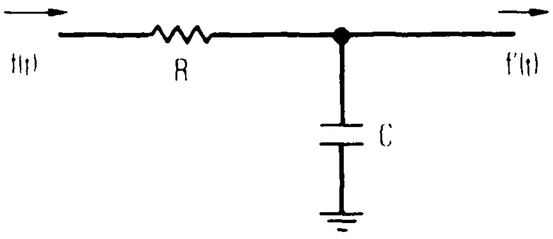
$$f_i(t) = C_i \exp(-a_i t) \tag{6}$$

$$a_i = \ln \left[\frac{f(t_{i+1})/f(t_i)}{t_i - t_{i+1}}\right]$$

$$C_i = f(t_i) \exp(a_i t_i)$$

The exponential function of Eq. (6) gave the best results for all data that were not near a point where f(t) crossed zero. A listing of a Fortran program for doing the transformations that give the impedance is provided in the Appendix.

Actual experimental data contain noise; in particular, 60-Hz noise may pose a problem when small changes in voltage or current are being monitored. The simplest way to eliminate this kind of noise is with an RC filter.



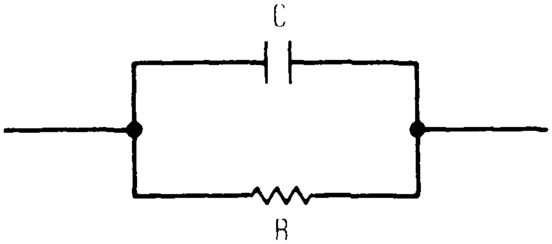
However, the transfer function of this filter must then be deconvoluted from the data. This is easily done in the frequency domain simply by multiplying the transformed function by the inverse filter function transform

$$f(\omega) = f'(\omega)\left(1 + j\omega\tau_F\right) \tag{7}$$

where τ_F = RC is the filter time constant.

III. RESULTS

The impedance of battery cells below 1 kHz is generally capacitive in nature, behaving as an equivalent parallel RC circuit where the values of R and C may have a complicated dependence on frequency. The results of an SAEP experiment on a simple dummy cell consisting of the RC circuit



where C = 1 F and R = 10 Ω are examined first. For simplicity, let us assume that $V_o + V_N(t)$, the initial cell voltage, is *zero*. An increasing exponential perturbation having amplitude α and time constant τ is applied to the cell

$$V(t) = \alpha\left(1 - e^{-t/\tau}\right) \qquad (8)$$

V(t) and the current response of the dummy cell I(t) are indicated in Fig. 1 for τ = 2s, R = 10 Ω, and C = 1 F. I(t) is given by the relationship

$$I(t) = \frac{\alpha}{R}\left[1 - \left(1 - \frac{RC}{\tau}\right)e^{-t/\tau}\right] \qquad (9)$$

Note that the values of R and C do not influence the time constant for current decay, but only control the amplitude of the current transient. From the time-dependent voltage and current functions, the impedance is calculated, with the results shown in Fig. 2 in the complex plane. The results in Fig. 2 agree with the theoretical result for the impedance

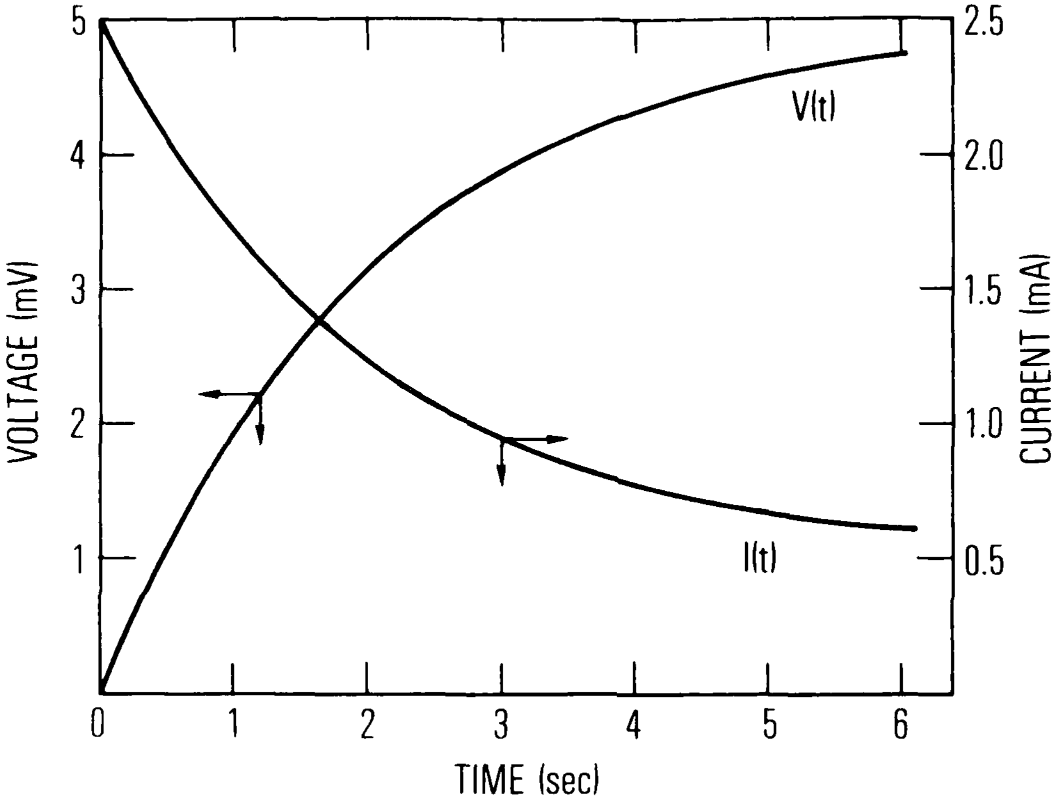


Fig. 1. Voltage Perturbation and Response Current for Dummy Cell

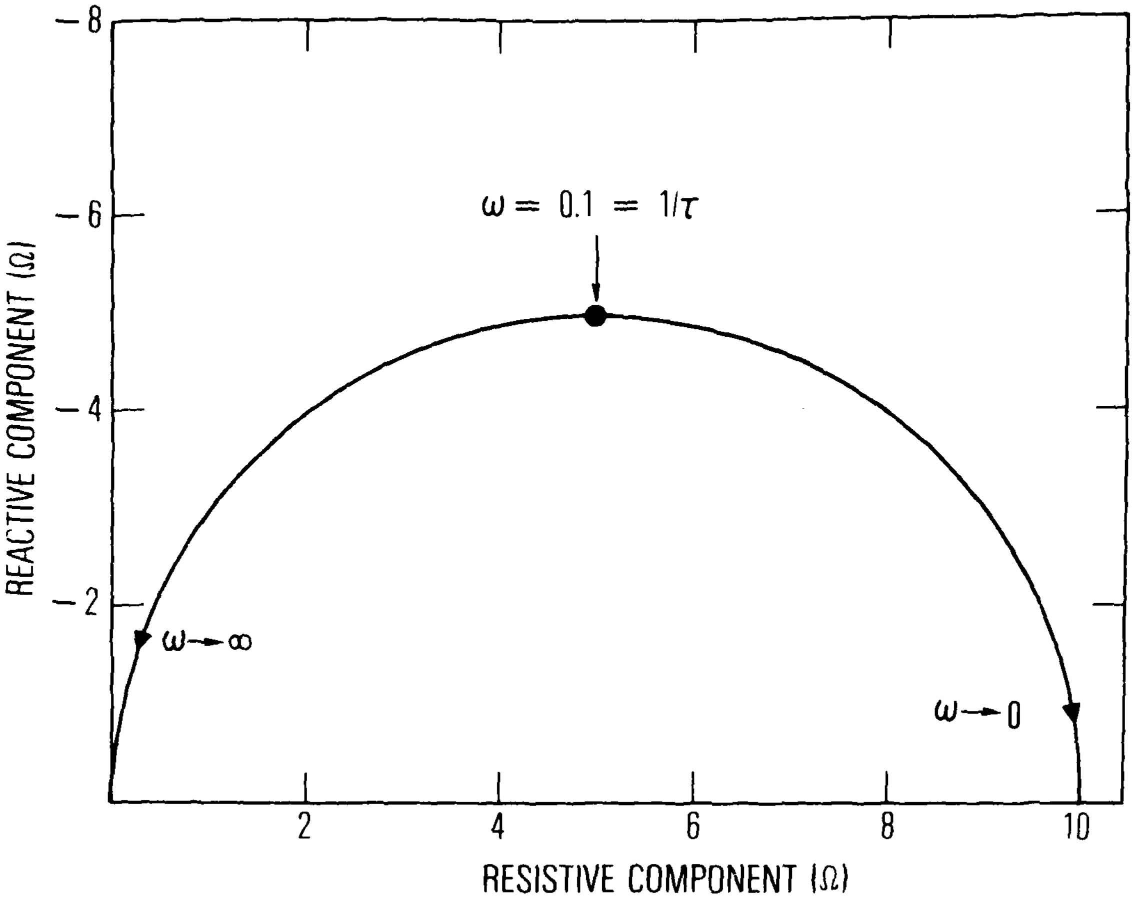


Fig. 2. Impedance of Dummy Cell from Data of Fig. 1

$$Z(\omega) = \frac{R}{1 + j\omega RC} \tag{10}$$

This simple example illustrates that the expected current response for a battery cell consists of a rapid rise to a maximum, followed by a decay to a steady-state current that is different from the initial current by the amount α/R. The magnitude of the peak current is controlled by the relative time constants of the exponential perturbation and the cell, and is given by $\alpha C/\tau$ in the preceding example. Thus, the experimental maximum transient current can be controlled simply by controlling the perturbation time constant.

The results obtained when an exponential perturbation is applied to a nickel cadmium cell are shown in Fig. 3. The nickel cadmium cell used was a 10-Ah prismatic cell, and the initial cell voltage was 0.5 V. The impedance is indicated in the complex plane in Fig. 4. In making these measurements, it was found that signal to noise became relatively poor unless the time constant of the applied perturbation was the same order of magnitude as the relaxation time for the battery cell. With this general requirement satisfied, the SAEP technique provides a convenient method for making impedance measurements on battery cells over an extremely wide range of frequencies, under conditions of battery operation for which potential control is acceptable.

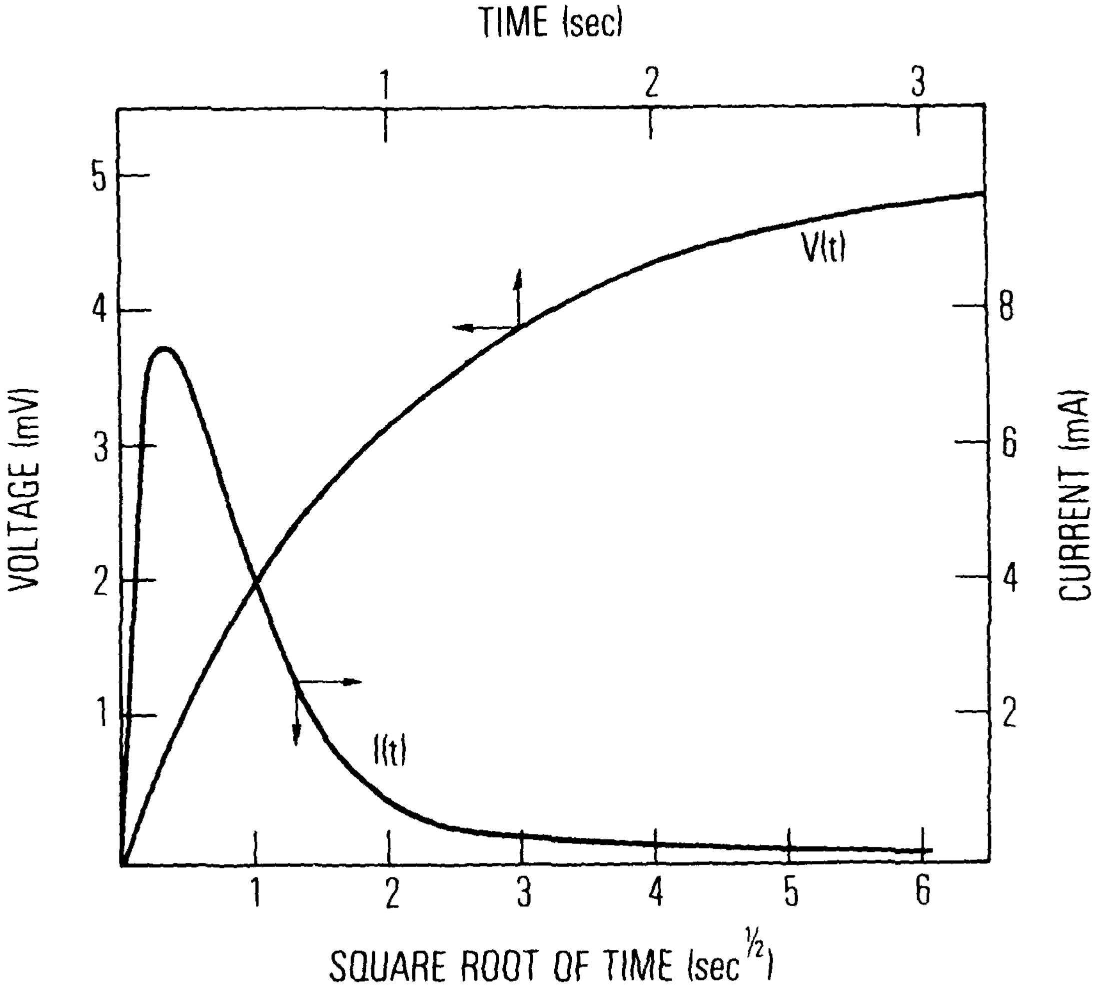


Fig. 3. Voltage Perturbation and Current Response for Nickel Cadmium Cell at 0.5 V

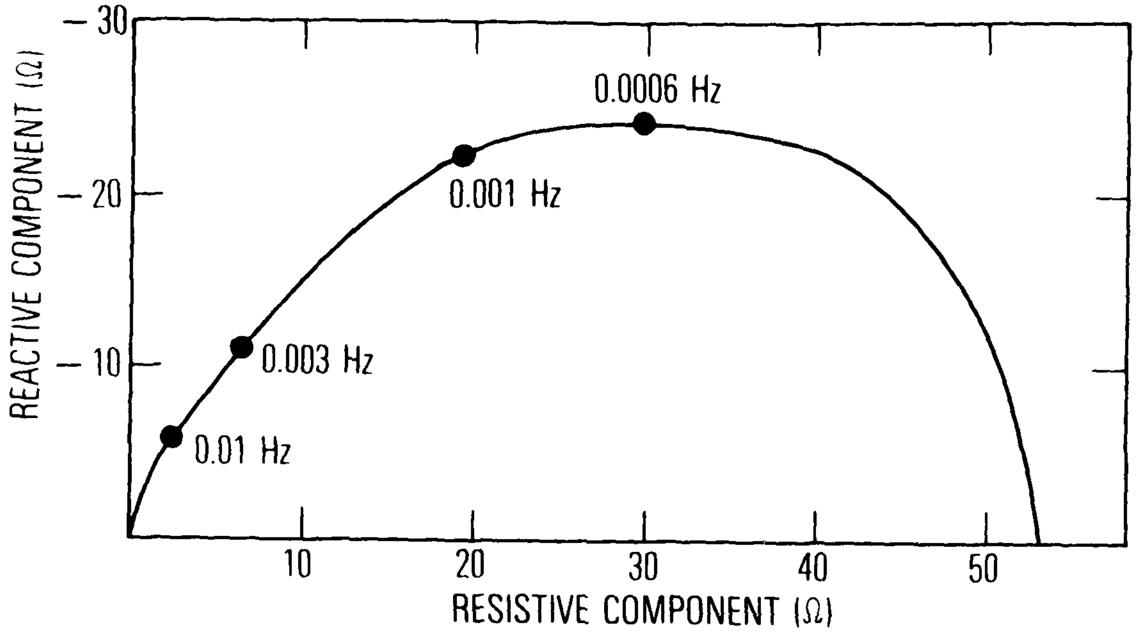


Fig. 4. Impedance of Nickel Cadmium Cell from Data of Fig. 3

IV. CONCLUSIONS

The SAEP technique has been developed and applied to measuring the impedance of battery cells under conditions of controlled potential. This appears to be the optimum method for measuring the impedance of battery cells that contain little stored electrochemical capacity.

APPENDIX

FORTRAN PROGRAM FOR SAEP IMPEDANCE CALCULATION

```
              PROGRAM SAEP(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)                  G          2
C             THIS PROGRAM CALCULATES IMPEDANCE AS A FUNCTION OF FREQUENCY FROM    G          3
C             AN EXPONENTTIAL VOLTAGE PERTURBATION AND ITS CURRENT RESPONSE USING  G          4
C             THE LAPLACE TRANSFORMATION TECHNIQUE.                                G          5
C             A FILTER FUNCTION MAY BE INCLUDED IN THE CURRENT RESPONSE, AND IS    G          6
C             DECONVOLUTED FROM THE TRANSFORM.                                     G          7
C                                                                                  G          8
C             THE PROGRAM THE FITS THE DATA TO A MAXIMUM OF TWO DIFFUSION          G          9
C             PROCESSES. IT REQUIRES INITIAL GUESSES FOR THE FITTING PARAMETERS    G         10
C             WHICH ARE TIME CONSTANTS A1 AND A2, WARBURG COEFFICIENTS S1 AND S2   G         11
C             AND CAPACITANCES C1 AND C2 AS WELL AS ELECTROLYTE RESISTANCE RZ.     G         12
C                                                                                  G         13
C             DATA INPUTS ARE                                                      G         14
C             MPROB= EXPERIMENT IDENTIFICATION NUMBER                              G         15
C                  NDATV=NUMBER OF VOLTAGE DATA POINTS                             G         16
C                  NDATI=NUMBER OF CURRENT DATA POINTS                             G         17
C                  TSAEP=NOMINAL PERTURBATION TIME CONSTANT IN SECONDS             G         18
C               AMPV=AMPLITUDE OF EXPONENTIAL PERTURBATION IN MV                   G         19
C                AMPI=AMPLITUDE OF CURRENT CHANGE AT INFINITE TIME IN MA           G         20
C               FILTC=TIME CONSTANT OF FILTER USED FOR CURRENT/TIME DATA (SEC)     G         21
C               FILTV=TIME CONSTANT OF FILTER USED FOR VOLTAGE/TIME DATA (SEC)     G         22
C               VV(I),TSV(I) ARE ORDERED PAIRS CORRESPONDING TO TIME (SEC) AND     G         23
C                  VOLTAGE DATA (MV), AND SHOULD BE A NOMINALLY INCREASING         G         24
C                 EXPONENTIAL, ZERO AT ZERO TIME, AND AMPV AT INFINITE TIME.       G         25
C                XI(I), TSI(I) ORDERED PAIRS CONSISTING OF THE SQUARE ROOT OF      G         26
C                 TIME AND CURRENT DATA (MA), AND SHOULD BE ZERO AT ZERO TIME      G         27
C                 AND XIC AT INFINITE TIME.                                        G         28
              COMMON N,X(200),OBS(400),YMD(800),XS(4000),F(200),RZ                 A0         2
              COMMON/A/A1,S1,C1,A2,S2,C2,VV(200),TSV(200),XI(200),TSI(200)         A1         2
              COMMON/2/NPROB,PEPS(7)                                               A2         2
              COMPLEX ZM(200),VW,XIW,P,XFF,VFF                                     G         32
              EQUIVALENCE (OBS,ZM)                                                 G         33
              DATA WINIT,WINC1,WBRK,WINC2,WFIN/0.1,0.4,11.0,1.0,40.0/              G         34
C                SWITCH=0 GIVES IMPEDANCE CALCULATION ONLY                         G         35
C                SWITCH=1 GIVES IMPEDANCE CALCULATION AND FIT TO MODEL             G         36
              SWTCH=0                                                              G         37
     3   300  CONTINUE                                                             G         38
     3        READ(5,2001) MPROB,NDATV,NDATI,TSAEP,AMPV,AMPI,FILTC,FILTV           G         39
    27        IF(EOF,5) 1000,301                                                   G         40
    32   301  CONTINUE                                                             G         41
    32        WRITE(6,6000)                                                        G         42
    36        WRITE(6,6001) MPROB                                                  G         43
    44        WRITE(6,7000)                                                        G         44
    50        WRITE(6,6002) TSAEP,FILTC                                            G         45
    60        WRITE(6,6003) AMPV,AMPI                                              G         46
    70        READ(5,2003)(TSV(I),VV(I),I=1,NDATV)                                 G         47
   107        READ(5,2003)(TSI(I),XI(I),I=1,NDATI)                                 G         48
   126        WRITE(6,7000)                                                        G         49
   132        WRITE(6,6004)                                                        G         50
   136        DO 302 J=1,NDATV                                                     G         51
   140        WRITE(6,6005) J,VV(J),TSV(J)                                         G         52
   153        VV(J)=AMPV-VV(J)                                                     G         53
   156   302  TSV(J)=TSV(J)**2                                                     G         54
   161        WRITE(6,7000)                                                        G         55
   165        WRITE(6,6006)                                                        G         56
   171        DO 303 J=1,NDATI                                                     G         57
   173        WRITE(6,6007) J,XI(J),TSI(J)                                         G         58
```

```
206              TSI(J)=TSI(J)**2
210       300    XI(J)=XI(J)-AMPI*(1-EXP(-TSI(J)/TSAEP))
223              WRITE(6,7000)
227              N=0
230       320    N=N+1
232              X(N)=WINIT+(N-1)*WINC1
237              IF(X(N).LE.WBRK)  GO TO 360
241              NN=N-1
242              X(N)=X(NN)+WINC2
246       360    CONTINUE
250              W=1.0/(X(N)**2)
252              F(N)=W/(2*3.141592654)
254              P=CMPLX(0.,W)
257              CALL  LPXFRM(NDATV,TSV,VV,W,VW)
263              CALL  LPXFRM(NDATI,TSI,XI,W,XIW)
          C             ADD ZERO FREQUENCY COMPONENT TO TRANSFORM
270              VW=AMPV/P-VW
301              XIW=AMPI/(P*(1+P*TSAEP))+XIW
          C                COMPUTE INVERSE FILTER FUNCTIONS
326              XFF=CMPLX(1.,W*FILTC)
331              VFF=CMPLX(1.,W*FILTV)
          C             COMPUTE INPEDANCE
334              ZM(N)=(VW*VFF)/(XIW*XFF)
361              IF(X(N).GE.WFIN-WINC2)  GO TO 401
365              GO TO 320
366       401    CONTINUE
366              WRITE(6,6008)
372              WRITE(6,7000)
376              WRITE(6,6009)
402              DO 420  I=1,N
404              I1=2*I-1
405              I2=I1+1
407              WRITE(6,6010)  I,X(I),OBS(I1),OBS(I2),F(I)
430       420    CONTINUE
433              IF(SWTCH.EQ.0.)  GO TO 300
434       430    CONTINUE
434              READ(5,2002)  A1,S1,C1,A2,S2,C2,RZ
456              N=2*N
457              CALL  GAULLL
460              NPROB=1
461              CALL  GAUHHH
462              GO TO 300
463      1000    CONTINUE
463              STOP
         2001    FORMAT(3I5,5F10.5)
         2002    FORMAT(7F10.5)
         2003    FORMAT(8F10.5)
         6000    FORMAT(1H1)
         6001    FORMAT(1X,[EXPERIMENT NUMBER  [,I5)
         6002    FORMAT(5X,[SAEP TIME CONSTANT =[,2X,F10.5,20X,[FILTER TIME CONSTAN
                *T =[,F10.5,[SECONDS[)
         6003    FORMAT(5X,[VOLTAGE CHANGE IS [,F10.5,20X,[CURRENT CHANGE AT LONG T
                *IMES IS [,F10.5)
         6004    FORMAT(5X,[INPUT DATA  VOLTAGE[,7X,[SQRT TIME[)
         6005    FORMAT(10X,I5,2X,F12.6,2X,F12.6)
         6006    FORMAT(5X,[INPUT DATA  CURRENT[,7X,[SQRT TIME[)
         6007    FORMAT(10X,I5,2X,F12.6,2X,F12.6)
```

```
            6008 FORMAT(1X,(IMPEDANCE DATA())                                         G        116
            6009 FORMAT(8X,(DATA POINT(,1X,(1/SQRT(OMEGA)(,7X,(REAL  Z(,13X,(IMAGINA G        117
                *RY Z(,8X,(FREQUENCY(HZ)())                                           G        118
            6010 FORMAT(10X,I5,4(4X,F14.6))                                           G        119
            7000 FORMAT(1X)                                                           G        120
      465        END                                                                  G        121


PROGRAM LENGTH INCLUDING I/O BUFFERS
000745

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
300     -  000004    301     -  000033    320     -  000231    360     -  000247
401     -  000357    430     -  000435    1000    -  000464    2001    -  000506
2002    -  000511    2003    -  000513    6000    -  000515    6001    -  000517
6002    -  000524    6003    -  000536    6004    -  000547    6005    -  000555
6006    -  000561    6007    -  000567    6008    -  000573    6009    -  000577
6010    -  000611    7000    -  000615

BLOCK NAMES AND LENGTHS
SAEP    -  000745            -  012742    A       -  001446    2       -  000010

VARIABLE ASSIGNMENTS
AMPI    -  000670    AMPV    -  000667    A1      -  000000S02 A2      -  000003S02
C1      -  000002S02 C2      -  000005S02 F       -  012431S01 FILTC   -  000571
FILTV   -  000672    I       -  000673    I1      -  000677    I2      -  000703
J       -  000674    MPROB   -  000663    N       -  000000S01 NDATI   -  000665
NDATV   -  000664    NN      -  000675    NPROB   -  000000S03 OBS     -  000311S01
P       -  000647    PEPS    -  000001S03 RZ      -  012741S01 SWTCH   -  000662
S1      -  000001S02 S2      -  000004S02 TSAEP   -  000666    TSI     -  001135S02
TSV     -  000316S02 VFF     -  000653    VV      -  000006S02 VW      -  000543
W       -  000676    WBRK    -  000657    WFIN    -  000661    WINC1   -  000555
WINC2   -  000650    WINIT   -  000655    X       -  000001S01 XFF     -  000551
XI      -  000626S02 XIW     -  000645    XS      -  002571S01 YMD     -  001131S01
ZM      -  000311S01

START OF CONSTANTS
000470

START OF TEMPORARIES
000617

START OF INDIRECTS
000635

UNUSED COMPILER SPACE
007400
```

```
                SUBROUTINE GAUHHH                                                     G          122
                EXTERNAL MODEL                                                        G          123
                COMMON N,X(200),OBS(400),YMD(800),XS(4000),F(200),RZ                  A0           2
                COMMON/A/A1,S1,C1,A2,S2,C2,VV(200),TSV(200),XI(200),TSI(20))          A1           2
                COMMON/2/NPROB,PEPS(7)                                                A2           2
                DIMENSION P(7),BNDUPR(7),BNDLWR(7)                                    G          127
                IS=0                                                                  G          128
     2      100 CONTINUE                                                              G          129
     2          NP=7                                                                  G          130
     3          P(1)=A1                                                               G          131
     4          P(2)=S1                                                               G          132
     6          P(3)=C1                                                               G          133
     7          P(4)=A2                                                               G          134
    11          P(5)=S2                                                               G          135
    12          P(6)=C2                                                               G          136
    14          P(7)=RZ                                                               G          137
    15          IF(IS.NE.0) GO TO 110                                                 G          138
    17          DO 105 I=1,NP                                                         G          139
    26          BNDUPR(I)=P(I)*1000.                                                  G          140
    27          BNDLWR(I)=P(I)*.001                                                   G          141
    27          PEPS(I)=P(I)*.01                                                      G          142
    30      105 CONTINUE                                                              G          143
    31          EPS1=1.E-6                                                            G          144
    32          EPS2=EPS3=0.                                                          G          145
    35          MAXIT=100                                                             G          146
    35          GAMMA=.01                                                             G          147
    37          FNU=10.                                                               G          148
    41          CALL GAUSAUS(NPROB,MODEL,N,OBS,NP,P,BNDUPR,BNDLWR,EPS1,EPS2,EPS3,     G          149
               *MAXIT,GAMMA,FNU,XS)                                                   G          150
    57      110 CONTINUE                                                              G          151
    57          CALL MODEL(NPROB,P,YMD,N,NP)                                          G          152
    63          U=0.                                                                  G          153
    64          S=0.                                                                  G          154
    65          WRITE(6,6000)                                                         G          155
    70          DO 120 I=1,N                                                          G          156
    73          ERR=YMD(I)-OBS(I)                                                     G          157
    75          ERR2=ERR**2                                                           G          158
    75          U=U+ERR                                                               G          159
   100          S=S+ERR2                                                              G          160
   101          J=(I+1)/2                                                             G          161
   104          WRITE(6,6001) I,X(J),OBS(I),YMD(I),ERR,ERR2                           G          162
   126      120 CONTINUE                                                              G          163
   131          U=U/N                                                                 G          164
   132          S=SQRT(S/N)                                                           G          165
   137          WRITE(6,6001)                                                         G          166
   142          WRITE(6,6003) N,U,S                                                   G          167
   154          WRITE(6,6001)                                                         G          168
   160          WRITE(6,6005) (I,P(I),I=1,NP)                                         G          169
   176          RETURN                                                                G          170
   177          ENTRY GAULLL                                                          G          171
   204          IS=1                                                                  G          172
   205          GO TO 100                                                             G          173
           6000 FORMAT(1H1)                                                           G          174
           6001 FORMAT(1X,I10,3F15.4,2(1PE20.3))                                      G          175
           6003 FORMAT(1X,I10,45X,2(1PE20.3))                                         G          176
           6005 FORMAT(1X,I10,F15.6)                                                  G          177
   206          END                                                                   G          178
```

```
                    SUBROUTINE MODEL(NPROB,Q,G,NG,NQ)                                      G         179
                    COMMON N,X(200),OBS(400),YMD(800),XS(4000),F(200),RZ                   A0          2
                    COMMON/A/A1,S1,C1,A2,S2,C2,VV(200),TSV(200),XI(200),TSI(200)           A1          2
                    DIMENSION G(1),Q(1)                                                    G         182
                    COMPLEX ZW1,ZW2,ZC1,ZC2,Z1,Z2,G                                        G         183
                    COMPLEX UR,UI                                                          G         184
                    COMPLEX CTANH                                                          G         185
                    DATA UR,UI/1.,0.,0.,1./                                                G         186
                    A1=Q(1)                                                                G         187
      7             S1=Q(2)                                                                G         188
     11             C1=Q(3)                                                                G         189
     12             A2=Q(4)                                                                G         190
     14             S2=Q(5)                                                                G         191
     15             C2=Q(6)                                                                G         192
     17             RZ=Q(7)                                                                G         193
     20             NG2=NG/2                                                               G         194
     22             DO 120 I=1,NG2                                                         G         195
     25             XX=X(I)                                                                G         196
     26             ZC1=-UI*XX**2/C1                                                       G         197
     35             ZW1=S1*XX*(1.-UI)*CTANH(A1*CSQRT(UI)/XX)                               G         198
     70             Z1=ZC1*ZW1/(ZC1+ZW1)                                                   G         199
    107             Z2=0.                                                                  G         200
    110             IF(A2.EQ.0.) GO TO 110                                                 G         201
    114             ZC2=-UI*XX**2/C2                                                       G         202
    124             ZW2=S2*XX*(1.-UI)*CTANH(A2*CSQRT(UI)/XX)                               G         203
    156             Z2=ZC2*ZW2/(ZC2+ZW2)                                                   G         204
    200       110   CONTINUE                                                               G         205
    200             G(I)=Z1+Z2+RZ                                                          G         206
    207       120   CONTINUE                                                               G         207
    211             RETURN                                                                 G         208
    212             END                                                                    G         209


SUBPROGRAM LENGTH
000303

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
110     -   000201

BLOCK NAMES AND LENGTHS
MODEL   -   000303            -  012742    A        -  001446

VARIABLE ASSIGNMENTS
A1      -   000000S02  A2     -  000003S02 CTANH    -  000276     C1      -  000002S02
C2      -   000005S02  F      -  012431S01 I        -  000301     NG2     -  000300
OBS     -   000311S01  RZ     -  012741S01 S1       -  000001S02  S2      -  000004S02
TSI     -   001136S02  TSV    -  000316S02 UI       -  000274     UR      -  000272
VV      -   000006S02  X      -  000001S01 XI       -  000626S02  XS      -  002571S01
XX      -   000302     YMD    -  001131S01 ZC1      -  000262     ZC2     -  000264
ZW1     -   000256     ZW2    -  000260    Z1       -  000266     Z2      -  000270

START OF CONSTANTS
000215
```

```
                 SUBROUTINE PARTIAL(Q,DGDQ,NG,NQ)                            G          210
                 COMMON N,X(200),OBS(400),YMD(800),XS(4000),F(200),RZ        A1           2
                 COMMON/2/NPROB,PEPS(7)                                      A2           2
                 DIMENSION Q(1),DGDQ(NG,NQ)                                  G          213
                 DO 120 I=1,NQ                                               G          214
     10          PEPS12=.5/PEPS(I)                                           G          215
     11          QSAVE=Q(I)                                                  G          216
     12          Q(I)=QSAVE+PEPS(I)                                          G          217
     14          CALL MODEL(NPROB,Q,DGDQ(1,I),NG,NQ)                         G          218
     26          Q(I)=QSAVE-PEPS(I)                                          G          219
     31          CALL MODEL(NPROB,Q,YMD(1),NG,NQ)                            G          220
     37          Q(I)=QSAVE                                                  G          221
     44          DO 120 J=1,NG                                               G          222
     54          DGDQ(J,I)=(DGDQ(J,I)-YMD(J))*PEPS12                         G          223
     56      120 CONTINUE                                                    G          224
     64          RETURN                                                      G          225
     64          END                                                         G          226


SUBPROGRAM LENGTH
000105

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS

BLOCK NAMES AND LENGTHS
PARTIAL-  000105           - 012742    2       - 000010

VARIABLE ASSIGNMENTS
F      -  012431S01 I      - 000101    J       - 000104    NPROB  - 000101S02
OBS    -  000311S01 PEPS   - 000001S02 PEPS12  - 000102    QSAVE  - 000103
X      -  000001S01 XS     - 002571S01 YMD     - 001131S01

START OF CONSTANTS
000067

START OF TEMPORARIES
000071

START OF INDIRECTS
000075

UNUSED COMPILER SPACE
011200
```

```
                COMPLEX FUNCTION CTANH(Z)                                              G          227
                COMPLEX UI,7                                                           G          228
                DATA UI/0.,1./                                                         G          229
                IF(CABS(Z).GT.1.) GO T) 1                                              G          230
  7             CTANH=-UI*CSIN(UI*Z)/C)OS(UI*Z)                                        G          231
 44             RETURN                                                                 G          232
 46    1        IF(CABS(Z).GT.39.) GO TO 2                                             G          233
 53             CTANH=(1.-CEXP(-2.*Z))/(1.+CEXP(-2.*Z))                                G          234
103             RETURN                                                                 G          235
105    2        CONTINUE                                                               G          236
105             CTANH=CMPLX(1.,0.)                                                     G          237
110             RETURN                                                                 G          238
112             END                                                                    G          239


SUBPROGRAM LENGTH
000143

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
1     -  000047     2        -  000106

BLOCK NAMES AND LENGTHS
CTANH  -  000143

VARIABLE ASSIGNMENTS
CTANH  -  000137    UI       -  000141

START OF CONSTANTS
000115

START OF TEMPORARIES
000123

START OF INDIRECTS
000137

UNUSED COMPILER SPACE
011200
```

```
                  SUBROUTINE LPXFRM(NDAT,T,VT,W,VW)                              G        240
           C      THIS SUBROUTINE COMPUTES THE LAPLACE TRANSFORM AT ANGULAR FREQUENC G        241
           C      W OF THE SET OF ORDERED PAIRS T AND VT CORRESPONDING TO TIME AND   G        242
           C      FUNCTION VALUE AT THAT TIME.                                   G        243
           C      VT MUST APPROACH ZERO AT INFINITE TIME AND CANNOT CROSS ZERO.  G        244
                  DIMENSION T(200),VT(200)                                       G        245
                  COMPLEX VW,CA,CI,CIP,CA1,CI1,CIP1                              G        246
                  VW=CMPLX(0.,0.)                                                G        247
      11          IM=NDAT-1                                                      G        248
      13          DO 101 I=1,IM                                                  G        249
      14          IP=I+1                                                         G        250
      15          IF(VT(I).EQ.0.0) VT(I)=0.00001                                 G        251
      21          IF(VT(IP).EQ.0.0) VT(IP)=0.00001                               G        252
      25          IF(VT(I).LT.0.0) GO TO 150                                     G        253
      27          IF(VT(IP).LT.0.0) GO TO 150                                    G        254
      31          A=ALOG(VT(IP)/VT(I))/(T(I)-T(IP))                              G        255
      45          CA=CMPLX(A,W)                                                  G        256
      47          CI=CMPLX(0.,-W*T(I))                                           G        257
      53          CIP=CMPLX(0.,-W*T(IP))                                         G        258
      57          VW=VW+(VT(I)*CEXP(CI)-VT(IP)*CEXP(CIP))/CA                     G        259
     120          GO TO 100                                                      G        260
     121   150    IF(T(I).EQ.0.0) T(I)=0.00001                                   G        261
     126          DT=T(IP)/T(I)                                                  G        262
     130          B=(VT(IP)-DT*VT(I))/(1-DT)                                     G        263
     134          A=(VT(I)-B)/T(I)                                               G        264
     136          CA1=CMPLX(0.,W)                                                G        265
     141          CI1=CEXP(-CA1*T(I))/CA1                                        G        266
     163          CIP1=CEXP(-CA1*T(IP))/CA1                                      G        267
     205          VW=VW+A*(CI1*(T(I)+1./CA1)-CIP1*(T(IP)+1./CA1))+B*(CI1-CIP1)   G        268
     264   100    CONTINUE                                                       G        269
     267          VW=VW+VT(IP)*CEXP(CIP)/CA                                      G        270
     311   101    CONTINUE                                                       G        271
     311          RETURN                                                         G        272
     312          END                                                            G        273


SUBPROGRAM LENGTH
000420

FUNCTION ASSIGNMENTS

STATEMENT ASSIGNMENTS
100     -   000255     101     -   000312     150     -   000122

BLOCK NAMES AND LENGTHS
LPXFRM  -   000420

VARIABLE ASSIGNMENTS
A       -   000415     B       -   000417     CA      -   000376     CA1     -   000404
CI      -   000400     CIP     -   000402     CIP1    -   000410     CI1     -   000406
DT      -   000416     I       -   000413     IM      -   000412     IP      -   000414

START OF CONSTANTS
000315
```

DATE
ILMED

-8