Radiation Project Progress Report Number 11

Program WAVETRAN

by

D. C. DEPACKH AND J. E. ULRICH

Electron Beams Branch
Plasma Physics Division

May 23, 1969

NAVAL RESEARCH LABORATORY
Washington, D.C.

PROGRAM WAVETRAN

by

D. C. DEPACKH and J. T. ULRICH

Program WAVETRAN is designed to follow the progress of a TEM electromagnetic wave through an arbitrary coaxial structure. There is no restriction on the conductor shape in these systems nor on the configuration of the dielectrics, which may have a variety of dielectric constants. The present version is not concerned with either series or shunt losses, however, and the dielectric properties are assumed constant with frequency.

The transmission line is represented as a number of cascaded incremental regions. These regions are bounded by the inner and outer conductors and by the electric field surfaces which extend between them (figure 1). Field lines along these surfaces in general undergo discontinuous changes in direction at dielectric boundaries. In arranging the problem for the program one must first have a solution of the Laplace equation for the whole region, and it is assumed here that such a solution is at hand. It is convenient to lay out equidistant points (rings) on one of the conductors, usually the inner one. From the Laplace equation solution the appropriate incremental regions are traced out by following an E-line from each of these points to the outer conductor. The program then considers the characteristic impedance and propagation velocity of each of these regions. The resulting incremental discontinuities in the elementary sections contribute to forward and backward moving waves in the line and allow the whole time history of the wave to be followed as a function of time and distance. The characteristic impedance Z of each section, as well as the time delay t_d referred to distance along the reference conductor, can be determined from a knowledge of the incremental inductance and capacitance of each section. If these are respectively

ΔL and ΔC, then

$$Z = (\Delta L/\Delta C)^{\frac{1}{2}}, \; t_d = (\Delta L \Delta C)^{\frac{1}{2}}.$$

Inductance and capacitance are calculated from the definition of inductance as total magnetic flux divided by the circuit current and of capacitance as total electric flux divided by the total voltage. The presence of dielectrics is of no importance to the inductance calculation, and neither is the shape of the conductors except as the integration limits are affected. The fundamental definition in practical units gives

$$L = 4\pi \times 10^{-9} \int \underline{H} \cdot \underline{dS} / \oint \underline{H} \cdot \underline{d\ell}$$

where $\underline{dS}$ is an element of area traversed by the flux and $\underline{d\ell}$ is an element of length of a line encircling the inner conductor. For coaxial geometry $H = 2i/R$, $d\ell = R d\phi$, so that

$$\Delta L = 2 \times 10^{-9} \int dS/R,$$

integration being over the elementary section in the R-Z plane (figure 1).

If one supposed that the entire elementary area were filled with a homogeneous dielectric of constant K, then the capacity would be

$$\Delta C = (K/4\pi)(1.1 \times 10^{-12}) \int \underline{E} \cdot \underline{dS'} / \int \underline{E} \cdot \underline{d\ell'},$$

where $\underline{dS'}$ is an element of surface traversed by the electric flux (and conveniently taken as an element of the inner conductor surface), and $\underline{d\ell'}$ is a line element between inner and outer conductors; the denominator is of course just the potential drop V. If the dielectric is composite, the solution of the Laplace equation provides information about the potential drop across each dielectric in the elementary sandwich. Thus the capacity of each layer of the sandwich can be calculated, and the total value of ΔC is $1/\Sigma\, C_i^{-1}$, where C_i are the layer capacities. So

$$\Delta C = (1/36\pi) \times 10^{-11} \int \underline{E} \cdot \underline{dS'} (\Sigma\, V_i/K_i)^{-1}.$$

The WAVETRAN program requires as input information about the characteristic impedance and propagation velocity of each line section. This information can be obtained from calculations as outlined above or from some other source. The procedure in the program follows from the diagram of figure 2, which represents a single line section. The quantities F_i, F_{i+1}, B_i, B_{i+1} are forward and backward voltage wave amplitudes at the beginning and end of the section. The operator P represents a time delay; it is defined by

$$Pf(t) = f(t-\Delta t),$$

where Δt is a program parameter. Δk_i and Δk_{i+1} are elementary reflection coefficients. From the diagram we can write

$$F(x_{i+1}, t+\Delta t) = (1+\Delta k_{i+1})\left[F(x_i, t) - \Delta k_i(1-\Delta k_i)^{-1} B(x_i, t)\right],$$

$$B(x_i, t+\Delta t) = (1-\Delta k_i)\left[B(x_{i+1}, t) + \Delta k_{i+1}(1+\Delta k_{i+1})^{-1} F(x_{i+1}, t)\right],$$

where $x_{i+1} \equiv x_i + v_i \Delta t$, x is the position as measured along the reference conductor, and v_i is the propagation velocity in the section under consideration. It is seen that the program is tied to a fixed time increment. In the original data the impedance and velocity are determined by laying out fixed length increments along the reference conductor; since the velocity is variable, these do not in general correspond to fixed time increments. This problem is handled by presenting the impedance and velocity data as a continuous curve; the program then interpolates from these in order to obtain information suitable for the appropriate time step. The reflection coefficients are given by

$$\Delta k_i = (Z_i - Z_{i-1})/(Z_i + Z_{i-1}),$$

so that Z_i and v_i and impedance and velocity just beyond x_i; in the interpolation from original data the curves are assumed linear between x_i and x_{i+1}. The mean values of Z and v are used; the length of the interval and the mean velocity over it are determined iteratively until the change in the interval length is <.1 mm. Accordingly

$$F(x_{i+1}, t+\Delta t) = 2Z_{i+1}(Z_i + Z_{i+1})^{-1}\left[F(x_i, t) - (Z_i - Z_{i-1})(2Z_{i-1})^{-1} B(x_i, t)\right],$$

$$B(x_i, t+\Delta t) = 2Z_i(Z_i + Z_{i-1})^{-1}\left[B(x_{i-1}, t) + (Z_{i+1} - Z_i)(2Z_{i+1})^{-1} F(x_{i+1}, t)\right].$$

There is of course nothing in the program which prevents any parameters from varying with time or amplitude.

In carrying out the program points are laid out in accordance with the scheme of figure 3, the x-separations being, as previously stated, such as to provide a constant propagation time interval between points. The input voltage is fed into point 1 of the forward wave array. Point N+2 of the backward wave array, representing the line termination, is permanently set at zero. Reflections due to a discontinuous impedance change between points N+1 (last line point) and N+2 are thus generated at the boundary.

An application to a sample region appears in figure 4, for which impedance and velocity have been determined from an approximate solution of the Laplace equation. Figure 5 shows this data reduced to equal propagation time intervals. Figure 6 shows the time propagation of a unit step through the system of figure 4; figure 7 is the output voltage as a function of time for this input.

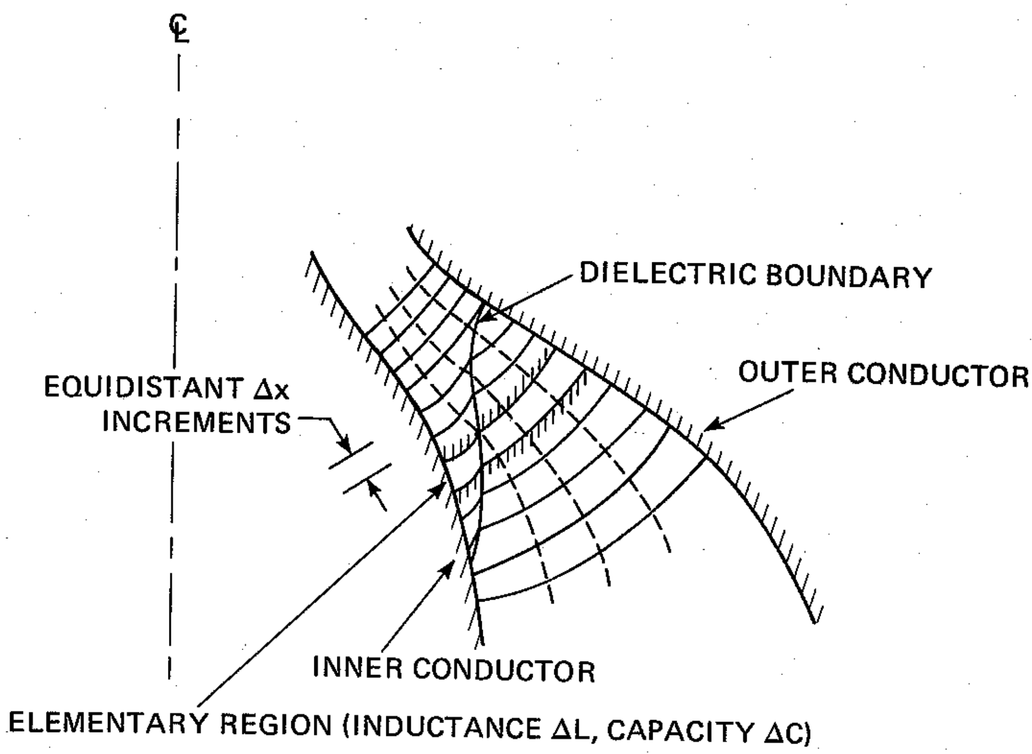


Figure 1 - Schematic of region treated by WAVETRAN.

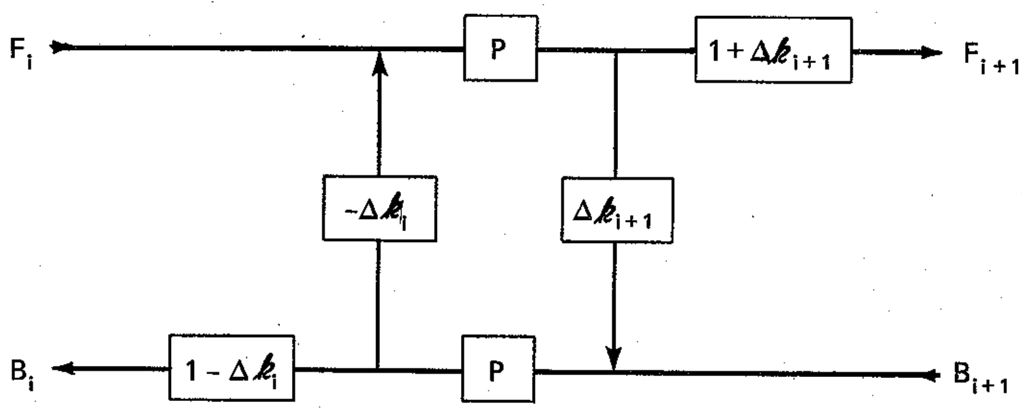


Figure 2 - Model of an elementary line section.

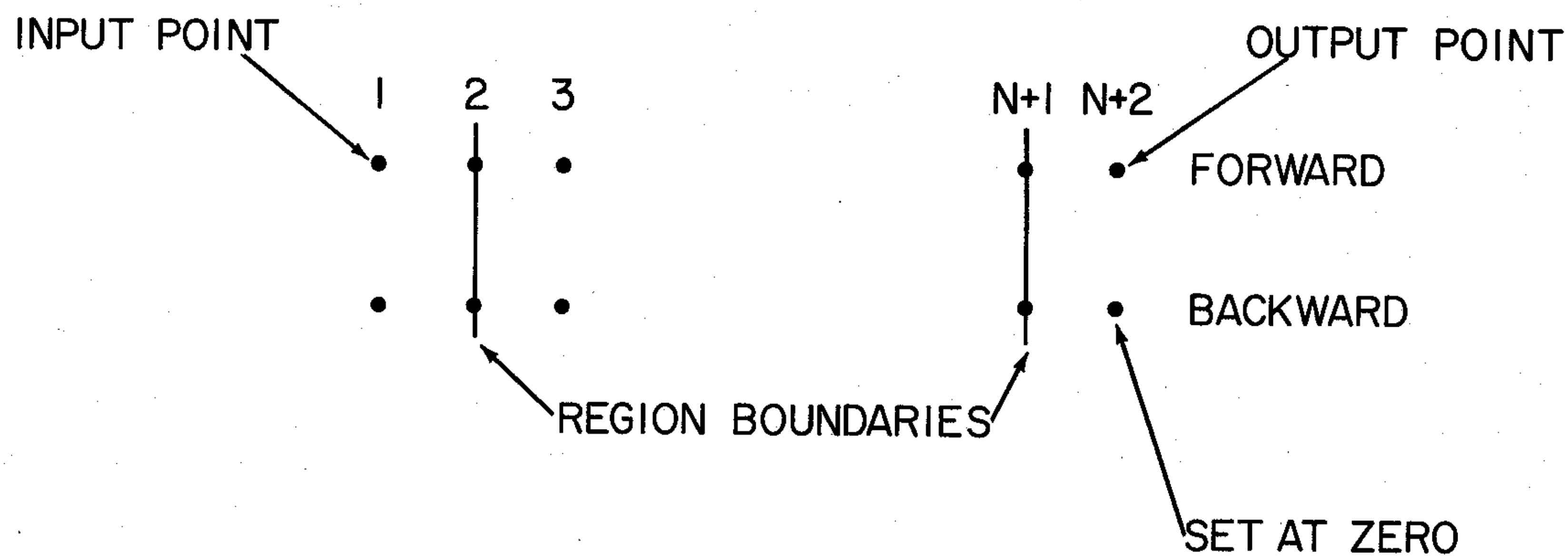


Figure 3 - X-point labeling in WAVETRAN.

Section	Impedance (Ω)	Velocity $\left(\frac{cm}{ns}\right)$
1	1.72	3.45
2	1.71	3.12
3	1.62	2.78
4	1.66	2.04
5	1.73	1.96
6	2.04	2.70
7	5.07	5.88
8	5.20	10.64
9	5.19	9.26
10	5.20	9.26
11	4.60	10.75
12	3.70	20.0
for $15 < x < 25$	56/(27-x)	20.0

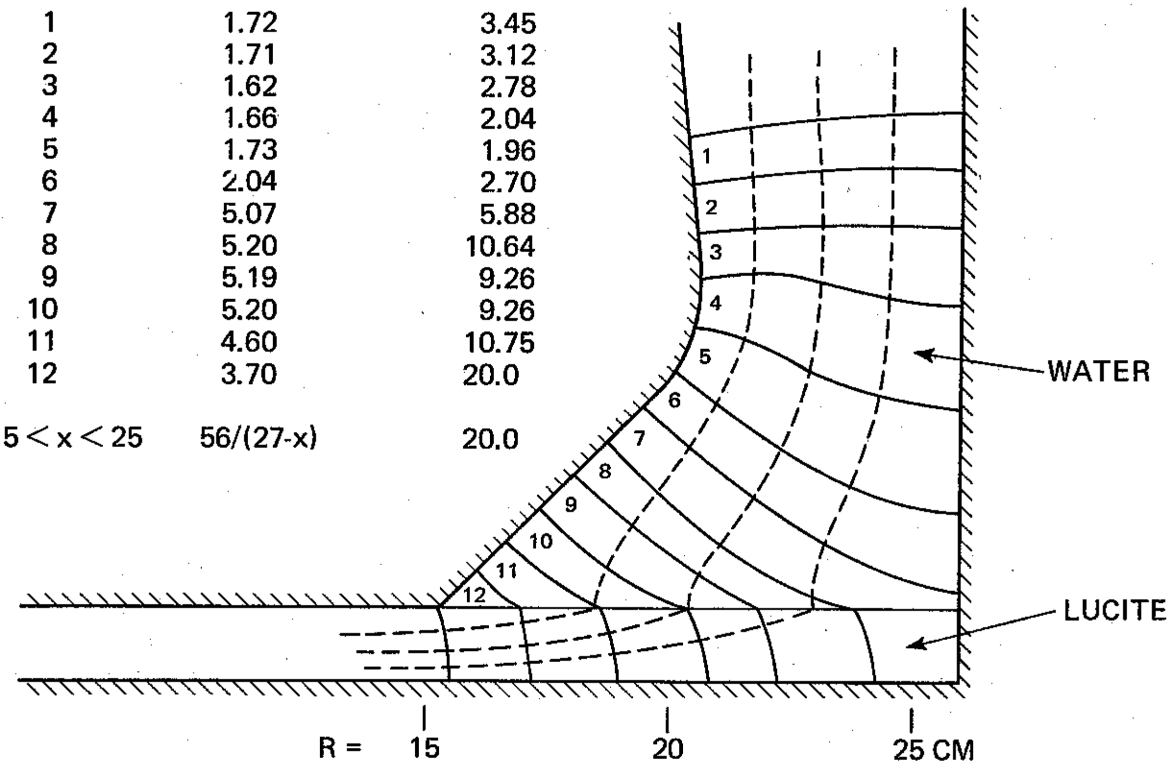
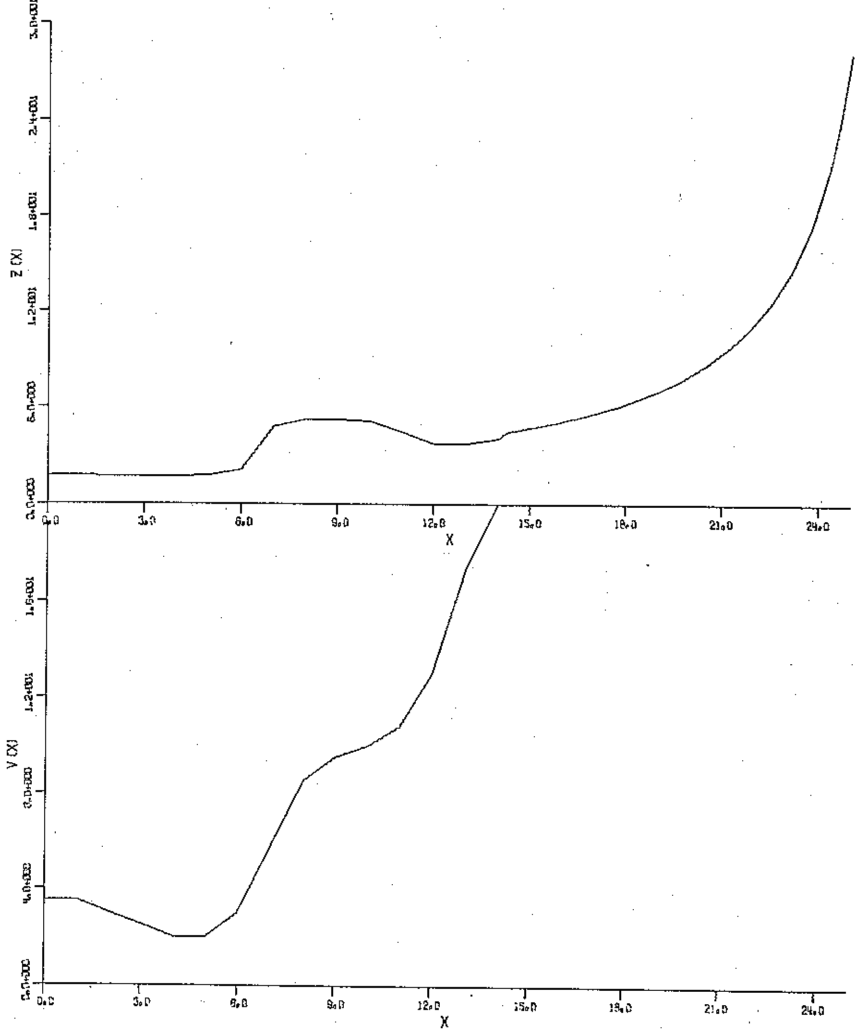


Figure 4 - Geometry used in WAVETRAN example. Material assumed to be lucite of the same thickness down to R = 2 cm. Numbers in the elementary regions are the x-values used in the program to determine impedance and velocity.



IX	X	Z(X)	V(X)
0	0.00	1.70000+000	3.50000+000
1	0.00	1.70000+000	3.50000+000
2	0.00	1.70284+000	3.52844+000
3	0.35	1.70544+000	3.55436+000
4	0.71	1.70171+000	3.51838+000
5	1.06	1.68460+000	3.38319+000
6	1.40	1.66636+000	3.21826+000
7	1.72	1.65399+000	3.06137+000
8	2.03	1.65000+000	2.91213+000
9	2.32	1.65000+000	2.77016+000
10	2.59	1.65000+000	2.63512+000
11	2.86	1.64886+000	2.50094+000
12	3.11	1.64190+000	2.34482+000
13	3.34	1.63824+000	2.21304+000
14	3.56	1.63956+000	2.11187+000
15	3.78	1.64520+000	2.03669+000
16	3.98	1.64974+000	1.96735+000
17	4.18	1.65341+000	1.90864+000
18	4.37	1.66609+000	1.88022+000
19	4.55	1.68739+000	1.89036+000
20	4.74	1.71805+000	1.93681+000
21	4.94	1.71741+000	2.00531+000
22	5.14	1.63206+000	2.08458+000
23	5.35	1.63088+000	2.24033+000
24	5.57	1.76284+000	2.51871+000
25	5.82	2.11386+000	2.97105+000
26	6.12	3.13929+000	3.87511+000
27	6.51	4.28886+000	5.13059+000
28	7.02	4.96772+000	6.87544+000
29	7.71	5.19108+000	8.60022+000
30	8.57	5.20921+000	9.47451+000
31	9.51	5.03781+000	9.99233+000
32	10.51	4.33543+000	1.09733+001
33	11.61	3.72565+000	1.45037+001
34	13.06	4.02594+000	1.93961+001
35	15.00	4.74754+000	2.00000+001
36	17.00	5.71727+000	2.00000+001
37	19.00	7.18517+000	2.00000+001
38	21.00	9.66840+000	2.00000+001
39	23.00	1.47818+001	2.00000+001
40	25.00	1.70000+000	2.00000+001

Figure 5 - Impedance and velocity values for geometry of Figure 4 at propagation time interval of 0.1 for G1 Diode.

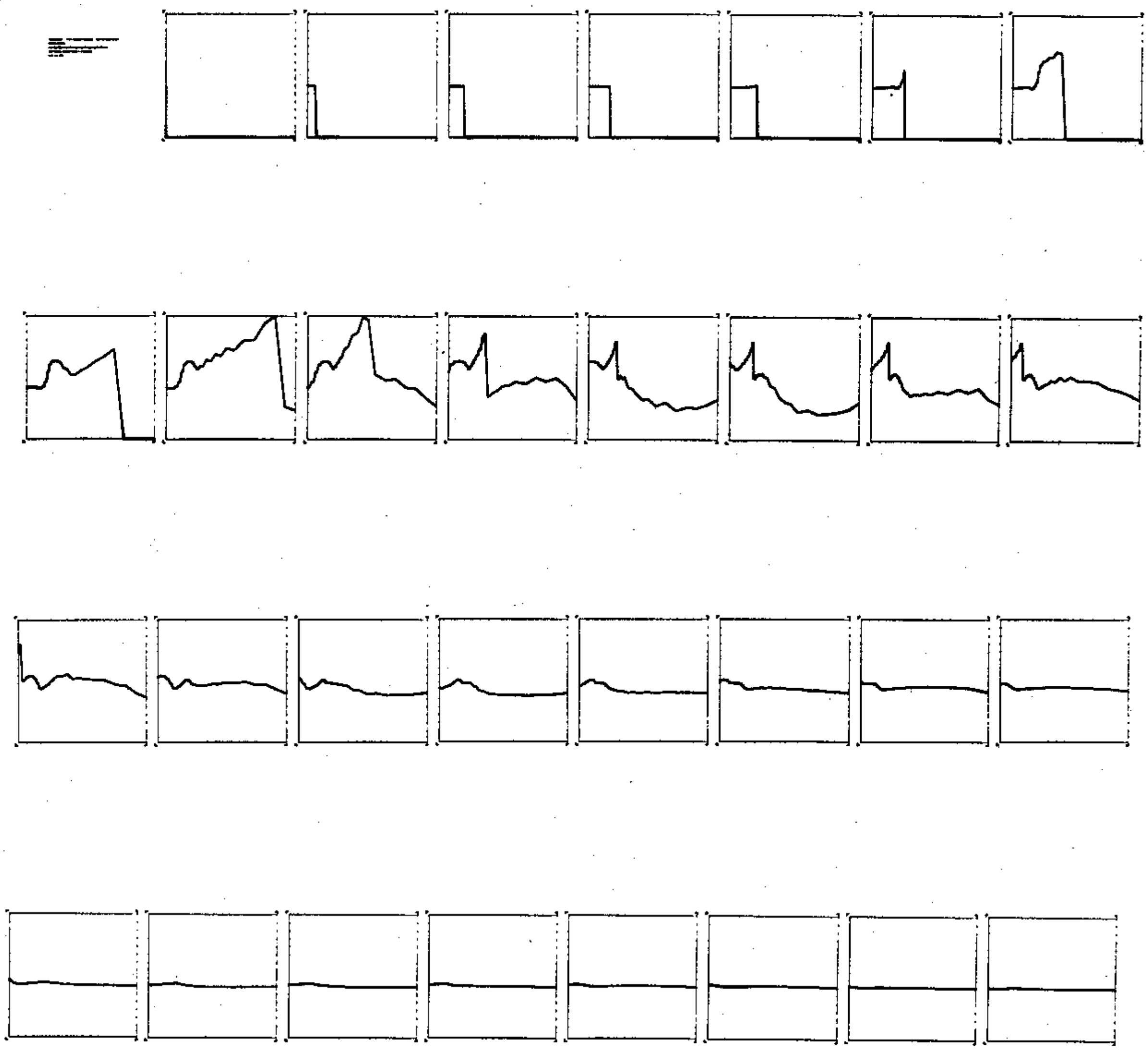


Figure 6 - Propagation of a unit step through the geometry of Figure 4. Time interval between frames is 0.5 nsec. Quantity shown is the sum of forward and backward waves.

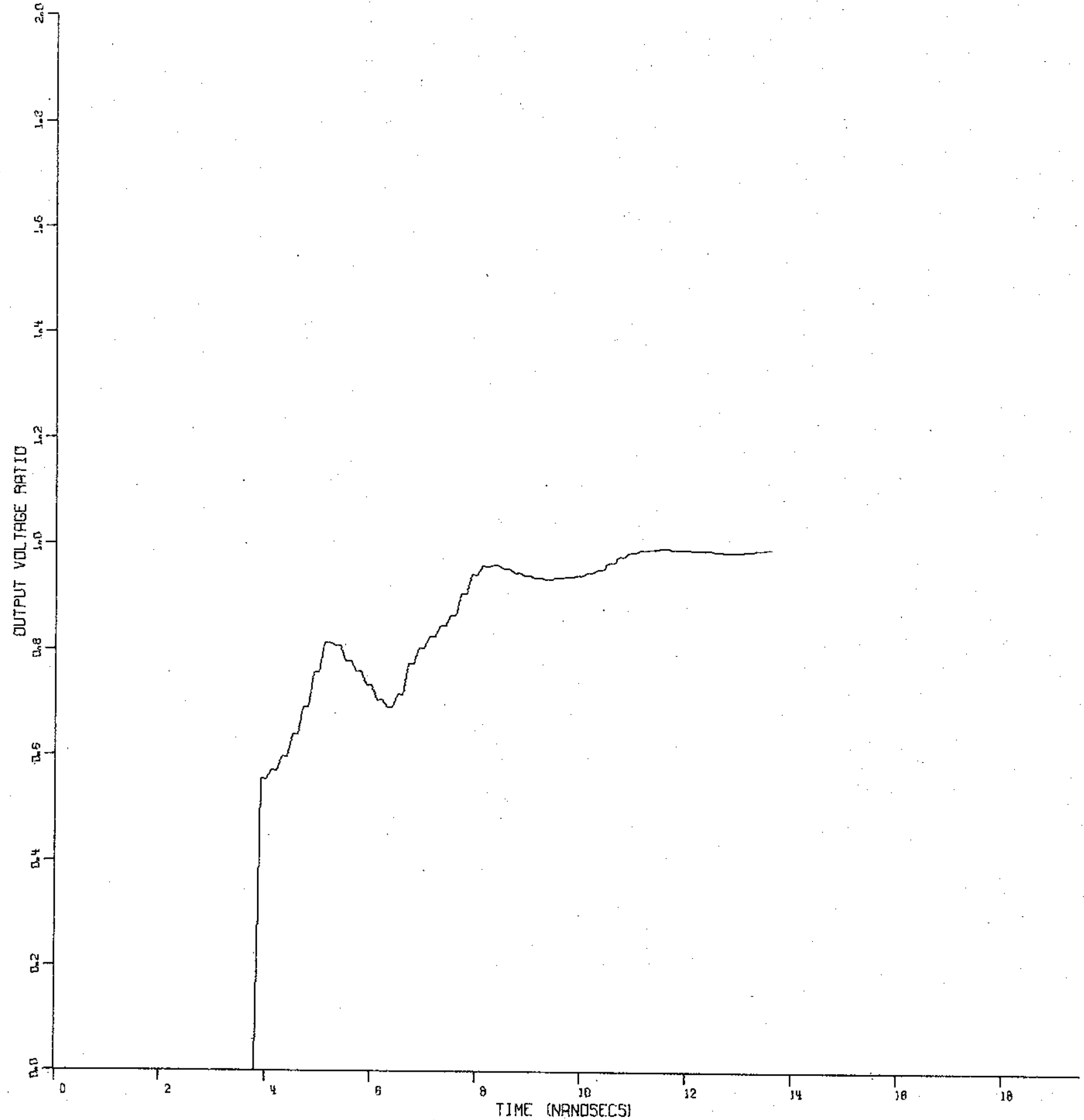


Figure 7 - Output voltage corresponding to a unit voltage step applied to the system of Figure 4.

Appendix A

FTN5.4A   04/14/69

```
      PROGRAM WAVETRAN
C     DATA CARDS ARE READ IN AS FOLLOWS
C        1) XMAX, HT, ITMAX, ITSUB, ISCOPE, IPRINT (F5.3,E10.2,4I5)
C           XMAX = DISTANCE ALONG SURFACE OF THE INNER CONDUCTOR, IN CMS
C           HT = SIZE OF TIME STEP, IN NANOSECS
C           ITMAX = NUMBER OF TIME STEPS
C           ITSUB = NUMBER OF SUBDIVISIONS OF THE TIME STEP, HT, TO USED IN THE
C              CALCULATIONS AND THE VOLTAGE PLOT, BUT OMITTED FROM THE PRINTOUT
C              AND THE MOVIE OUTPUT
C           ISCOPE = 1 FOR MOVIE OUTPUT,  = 0 TO OMIT MOVIE
C           IPRINT = 1 FOR PRINTOUT,  = 0 TO OMIT PRINTOUT
C        2,0) N, HX  (I5,F10.1)
C           N = NUMBER OF CARDS IN THE FOLLOWING TABLE
C           HX = THE X INTERVAL BETWEEN EACH OF THE FOLLOWING PAIRS OF V AND Z
C              VALUES, FROM 0 TO (N-1)HX ... (CASE 1)
C                 OR
C           HX = 0 INDICATES THAT THE X VALUE IS GIVEN ON EACH CARD ...(CASE 2)
C        2,1-2,N) V, Z  (2F10.1) ...(CASE 1), OR
C        2,1-2,N) V, Z, X  (3F10.1) ... (CASE 2)
C           V = VELOCITY OF THE WAVE AT EACH X VALUE
C           Z = IMPEDANCE AT EACH X VALUE
C        3) MOVIE LABEL CARDS (10A8) -- UP TO 20 CARDS MAY BE PLACED HERE
C           TO BE REPRODUCED ON THE 1ST FRAME OF THE MOVIE PRODUCED BY THIS RUN.

      DIMENSION F(2,50), B(50), Z(50), XA(50), PARRAY(1000),YP(50,500),
     1   XP(50), V(50), IXP(50), BCDLOC(10)
      EQUIVALENCE (XP,IXP)

  100 FORMAT (F5.2, E10.2, 4I5)
  101 FORMAT (9H1 IT     T 6X15HF(IT,IX), IX=1, I2/15X15HB(IT,IX), IX=1,
     1   I2)
  102 FORMAT (/2XI3, F8.3, 10E10.2/(13X10E10.2))
  103 FORMAT (22H WAVETRAN INPUT, XMAX=F6.1, 5H, HT=E10.2, 8H, ITMAX=I4,
     1   9H, ITSUB = I4/)
  104 FORMAT (1XI5,F7.2, 2E12.5)
  105 FORMAT (32H1    IX     X      Z(X)          V(X)/)
  106 FORMAT (/7H FCON = E12.5)
  107 FORMAT (13X10E10.2)
  108 FORMAT (113X E10.2)
  109 FORMAT (10A8)
      REAL K
      IXBOUND = 50
      IXMAX = IXBOUND - 1
      XFACTOR = .1
      FFACTOR = 5.
      DF = 1./FFACTOR
      TFACTOR = .5
      DT = 1./TFACTOR
      CONST = 377./6.2831852
      READ 100, XMAX, HT, ITMAX, ITSUB, ISCOPE, IPRINT
      PRINT 103, XMAX, HT, ITMAX, ITSUB
      HTSUB = HT/ITSUB
      HTINCH = HTSUB*TFACTOR
      TINCH = 0.0
      CALL PLOTS (PARRAY, 1000, 1)
```

```
      CALL AXIS (0,0, 20HOUTPUT VOLTAGE RATIO,20,10.,90.,1.,0,DF,4HF3.1)
      CALL PLOT (0.,0.,3)
C     CALCULATE Z TABLE AND INITIALIZE F AND B ARRAYS
      PRINT 105
      CALL XVZSET
      CALL XVZ (0., 0., V0, Z0)
      V(1) = V0
      Z(1) = Z0
      PRINT 104, 0, 0.0, Z0, V0
      PRINT 104, 1, 0.0, Z(1), V(1)
      XA = 0.0
      XP(1) = 0.0
      YP(1) = 1.0
      DO 1 IX=2,IXBOUND
      XP(IX) = XFACTOR*XA
      YP(IX,1) = 0.0
      F(1,IX) = B(IX) = 0.0
      CALL XVZ (XA, XA, VA, ZA)
      XLAST = XB = XA + VA*HTSUB
      IF (XB.GT.XMAX) GO TO 2
C     LOOP CALCULATION OF NEXT X UNTIL DIFFERENCE ≤ .01
      DO 4 LOOP=1,10
      CALL XVZ (XA, XB, VA, ZA)
      XB = XA + VA*HTSUB
      IF (ABSF(XB-XLAST)-.01) 5,5,4
    4 XLAST = XB
    5 V(IX) = VA
      Z(IX) = ZA
      XA = XB
    1 CONTINUE
      IX = IX - 1
    2 IXBOUND = IX
      IXMAX = IXBOUND - 1
      Z(IXBOUND) = Z(1)
      V(IXBOUND) = VA
      DO 3 IX=2,IXBOUND
      X = XP(IX)/XFACTOR
    3 PRINT 104, IX, X, Z(IX), V(IX)
      F(1,IXBOUND) = B(IXBOUND) = 0.0
      F(1,1) = 1.0
      B(1) = 0.0
      PRINT 101, IXBOUND, IXBOUND
      PRINT 102, 1, 0.0, (F(1,IX),IX=1,IXBOUND)
      PRINT 108, 0.0
      PRINT 107, (B(IX),IX=1,IXBOUND)
C     BEGIN TIME CYCLE
C     FIRST DETERMINE PAGING
      TRUNCLNS = IXBOUND/10
      FLOATLNS = FLOATF(IXBOUND)/10.0
      IF (FLOATLNS.GT.TRUNCLNS) TRUNCLNS = TRUNCLNS + 1.0
```

```
      LINESPER = 2.*TRUNCLNS + 2.0
      NPAGE = 58/LINESPER
      IFIRST = 2
      ILAST = NPAGE
    8 DO 10 IT=IFIRST,ILAST
      DO 9 ISKIP =1,ITSUB
C     FORWARD WAVE
      B(1) = B(2)
      DO 11 IX=2,IXMAX
      F(2,IX+1) = 2.*Z(IX+1)/(Z(IX) +Z(IX+1))*(F(1,IX) -(Z(IX) -Z(IX-1))
     1   *.5/Z(IX-1)*B(IX))
C     BACKWARD WAVE
      B(IX) = 2.*Z(IX-1)/(Z(IX) + Z(IX-1))*(B(IX+1) +(Z(IX+1) -Z(IX))
     1   *.5/Z(IX+1)*F(1,IX+1))
   11 CONTINUE
C     BOUNDARY CONDITIONS
      F(2,1) = F(1,1)
      F(2,2) = F(1,1)
      B(IXBOUND) = 0.0
      DO 12 IX=1,IXBOUND
      F(1,IX) = F(2,IX)
   12 CONTINUE
      FINCH = F(1,IXBOUND)*FFACTOR
      TINCH = TINCH + HTINCH
      IF (FINCH.GT.10.1) GO TO 99
      CALL PLOT (TINCH,FINCH,2)
    9 CONTINUE
C     SUM FORWARD AND BACKWARD WAVES FOR MOVIE OUTPUT
      DO 13 IX=1,IXBOUND
      YP(IX,IT) = F(1,IX) + B(IX)
   13 CONTINUE
C     OUTPUT
      FOUT=F(1,IXBOUND)
      IF (IPRINT.LE.0) GO TO 10
      TIME = (IT-1)*HT
      PRINT 102, IT, TIME, (F(2,IX),IX=1,IXBOUND)
      PRINT 108, FOUT
      PRINT 107, (B(IX), IX=1,IXBOUND)
   10 CONTINUE
      IF (ILAST.EQ.ITMAX) GO TO 99
      IF (IPRINT.EQ.1) PRINT 101, IXBOUND, IXBOUND
      IFIRST = ILAST + 1
      ILAST = ILAST + NPAGE
      IF (ILAST.GT.ITMAX) ILAST=ITMAX
      GO TO 8
   99 CALL AXIS (0.,0.,15HTIME (NANOSECS),-15,TINCH,0.,1.,0.,DT,4HF4.0)
      XNEW = TINCH + 2.
      CALL PLOT (XNEW, 0., -3)
```

```
      XMAX = XP(IXBOUND)/XFACTOR
      CALL PLOTXVZ (XMAX, V(IXBOUND), Z(IXBOUND))
      CALL STOPPLOT
      IF (ISCOPE.LE.0) GO TO 90

C     SCOPE PLOTTING SECTION

      ITMAX = IT-1
      REWIND 16
      SECS = TIMELEFT (IDUM)
      PRINT 111, SECS
  111 FORMAT (25H TIMELEFT BEFORE ENTFLM = F7.1, 5H SECS)
      CALL ENTFLM
      XL = XP(1)
      XR = XP(IXBOUND)
      CALL XSCALV (XL, XR, 24, 0)

      YB = YT = YP(1,1)
      DO 94 IX=1,IXBOUND
      IXP(IX) = NXV(XP(IX))
      DO 94 IT=1,ITMAX
      Y = YP(IX,IT)
      IF (YB-Y) 92,94,91
   91 YB = Y
      GO TO 94
   92 IF (YT - Y) 93,94,94
   93 YT = Y
   94 CONTINUE

      CALL YSCALV (YB, YT, 24, 24)
      CALL BRITEV
      IY = 832
      DO 97 JCARD=1,20
      IY = IY - 32
      READ 109, BCDLOC
      IF (EOF,60) 98,97
   97 CALL PRINTV (80, BCDLOC, 200, IY)

   98 DO 96 IT=1,ITMAX
      CALL FRAMEV (2)
      CALL LINEV (  24,  24,  24, 999)
      CALL LINEV (  24, 999,1023, 999)
      CALL LINEV (1023, 999,1023,  24)
      CALL LINEV (1023,  24,  24,  24)
      IYA = NYV(YP(1,IT))
      DO 96 IDARKEN=1,4
      DO 95 IX=2,IXBOUND
      IYB = NYV (YP(IX,IT))
      CALL LINEV (IXP(IX-1), IYA, IXP(IX), IYB)
   95 IYA = IYB
      IYA = NYV(YP(1,IT))
   96 CONTINUE
      SECS = TIMELEFT (IDUM)
      PRINT 116, SECS
  116 FORMAT (25H TIMELEFT BEFORE EXTFLM = F7.1, 5H SECS)
      CALL EXTFLM
```

```
      PRINT 120, ITMAX
  120 FORMAT (/1XI5, 46H SC4020 SCOPE PLOTS HAVE BEEN WRITTEN ON LU 16 )
   90 STOP
      END
```

```
      SUBROUTINE XVZ (XA, XB, VARG, ZARG)
C     XVZSET WILL READ IN DATA CARDS WITH X, V, AND Z VALUES AS FOLLOWS
C        1) N, HX
C        2-N+1) V, Z
C           OR IF HX = 0
C        2-N+1) V, Z, X
C     SUBSEQUENT XVZ (XA, XB, V(X), Z(X)) CALLS WILL PROVIDE THE AVERAGE VALUES
C     OF V(X) AND Z(X) FOR THE INTERVAL XA≤X≤XB, BY INTERPOLATION.
      DIMENSION X(100), V(100), Z(100)
  100 FORMAT (3F10.1)
  101 FORMAT (I5, F10.1)
      IAB = 1
      XARG = XA
      IF (XARG.GT.XLAST) GO TO 6
      I = 1
      IF (XARG.LE.X(1)) GO TO 5
      DO 4 I=2,N
      IF (XARG - X(I)) 7,5,4
    4 CONTINUE
    5 VARG = V(I)
      ZARG = Z(I)
      I = I+1
      GO TO 14
C     FORMULA FOR VALUES BEYOND XLAST
    6 VARG = V(N)
      ZARG = 56./(27. - XARG)
      I = N+1
      GO TO 14
C     INTERPOLATE BETWEEN I-1 AND I
    7 GO TO IGATE,(8,9)
C     LINEAR INTERPOLATION
    8 U = (XARG - X(I-1))/(X(I) - X(I-1))
      VARG = V(I-1) + (V(I) - V(I-1))*U
      ZARG = Z(I-1) + (Z(I) - Z(I-1))*U
      GO TO 14
C     PARABOLIC INTERPOLATION -- NEWTONS FORMULA
    9 IF (I.EQ.N) GO TO 10
      U = (XARG - X(I-1))/HX
      VARG = V(I-1)+U*(V(I)-V(I-1))+U*(U-1.)*.5*(V(I+1)-2.*V(I)+V(I-1))
      ZARG = Z(I-1)+U*(Z(I)-Z(I-1))+U*(U-1.)*.5*(Z(I+1)-2.*Z(I)+Z(I-1))
      GO TO 14
   10 U = (XARG - XLAST)/HX
      VARG = V(N)+U*(V(N)-V(N-1))+U*(U+1)*.5*(V(N)-2.*V(N-1)+V(N-2))
      ZARG = Z(N)+U*(Z(N)-Z(N-1))+U*(U+1)*.5*(Z(N)-2.*Z(N-1)+Z(N-2))
      GO TO 14
```

```
   14  GO TO (15,18), IAB
   15  IF (XB.LE.XA) RETURN
       IAB = 2
       IN = I
       VSUM = ZSUM = XSUM = 0.0
       V1 = VARG
       Z1 = ZARG
       X1 = XA
       XARG = XB

       DO 17 I=IN,N
       IF (XARG - X(I)) 7,5,16
   16  X2 = X(I)
       V2 = V(I)
       Z2 = Z(I)
       DELX = X2 - X1
       VSUM = VSUM + (V1+V2)*.5*DELX
       ZSUM = ZSUM + (Z1+Z2)*.5*DELX
       XSUM = XSUM + DELX
       X1 = X2
       V1 = V2
       Z1 = Z2
   17  CONTINUE

C      XB IS BEYOND X(N) -- USE EXTENSION FORMULA FOR 5 EXTENDED STEPS
       XSTEP = (XB - X1)/5,
       DO 19 II=1,5
       VSUM = VSUM + V(N)*XSTEP
       ZSUM = ZSUM + (Z1 + 56./(27.-(X1+XSTEP)))*.5*XSTEP
   19  XSUM = XSUM + XSTEP
       GO TO 20

C      XB LOCATED BEFORE X(N)
   18  DELX = XARG - X1
       VSUM = VSUM + (V1 + VARG)*.5*DELX
       ZSUM = ZSUM + (Z1 + ZARG)*.5*DELX
       XSUM = XSUM + DELX

   20  VARG = VSUM/XSUM
       ZARG = ZSUM/XSUM
       RETURN

C      INPUT SECTION

       ENTRY XVZSET

       READ 101, N, HX
       IF (HX.GT.0) GO TO 2

C      VZ INPUT WITH X VALUES
       ASSIGN 8 TO IGATE
       DO 1 I=1,N
    1  READ 100, V(I),Z(I),X(I)
       GO TO 11
```

```
C     VZ INPUT AT EQUAL X INTERVALS
    2 ASSIGN 9 TO IGATE
      DO 3 I=1,N
      READ 100, V(I), Z(I)
    3 X(I) = (I-1)*HX

   11 XLAST = X(N)
      RETURN

      ENTRY PLOTXVZ
C     THIS ENTRY RESCALES THE ARRAYS TO INCHES AND SHOULD NOT BE CALLED
C     UNTIL THE DATA IS NO LONGER NEEDED.
C     FOR THIS ENTRY, THE 1ST ARGUMENT, XA, SHOULD CONTAIN THE MAXIMUM VALUE OF
C     X USED BY THE CALLING PROGRAM.  OTHER ARGUMENTS ARE NOT USED.

      I = N + 1
      NXTRA = 50 - N
      IF (NXTRA.GT.0) M=N+NXTRA
      XXTRA = XA - XLAST
      DELX = XXTRA/NXTRA
      IF (XXTRA.LT.2.0) M=N
      DO 12 I=I,M
      X(I) = X(I-1) + DELX
      V(I) = V(N)
   12 Z(I) = 56./(27. - X(I))
      CALL ZEROSCAL (X, M, 10., XMIN, DX, 1)
      CALL ZEROSCAL (V, M,  5., VMIN, DV, 1)
      CALL ZEROSCAL (Z, M,  5., ZMIN, DZ, 1)
      CALL LINE (X, V, M, 1, -1, 0, 0)
      CALL AXIS (0., 0., 1HX, -1, X(M), 0., 1., XMIN, DX, 4HF4.1)
      CALL AXIS (0., 0., 4HV(X), 4, 4.8, 90., 1., VMIN, DV, 4HE8.1)
      CALL PLOT (0., 5., -3)
      CALL AXIS (0., 0., 1HX, -1, X(M), 0., 1., XMIN, DX, 4HF4.1)
      CALL AXIS (0., 0., 4HZ(X), 4, 5., 90., 1., ZMIN, DZ, 4HE8.1)
      CALL LINE (X, Z, M, 1, -1, 0, 0)
      CALL PLOT (X(M), 0., -3)
      RETURN
      END
```

```
      SUBROUTINE SCALE (Y,N,HEIGHT,YMIN,DY,K)

C     SCALE WILL CONVERT TO INCHES, THE N VALUES IN ARRAY Y, (STORED IN
C     EVERY KTH LOCATION OF ARRAY Y), TO FIT BETWEEN 0 AND HEIGHT INCHES
C     YMIN IS THE VALUE AT 0 INCHES, AND DY = Y-UNITS/INCH VALUE.
C     SCALE SELECTS THE ROUNDEST VALUES FOR DY AND YMIN THAT WILL BEST ADJUST
C     THE DATA TO THE GIVEN HEIGHT.
C     SCALE PRODUCES A DY = 1,2,3,4,5,6,7,8,OR 9 (TIMES SCALE FACTOR)
C     AND A YMIN = 0 MOD DY

      DIMENSION Y(1)
      INTEGER DYGATE

C     FIRST FIND YMIN AND YMAX VALUES

      ASSIGN 101 TO DYGATE
      YMIN = YMAX = Y(1)
      GO TO 100

C     ENTRY ZEROSCAL INCLUDES ZERO AMONG THE VALUES TO BE SCALED.

      ENTRY ZEROSCAL
      ASSIGN 101 TO DYGATE
      YMIN = YMAX = 0
      GO TO 100

C     ENTRY SCALETEN RESTRICTS THE VALUES OF DY TO 1,2,OR 5 (TIMES SCALE
C     FACTOR), FOR PLOTTING ON 01 (10 DIVISIONS/INCH) GRAPH PAPER, WHERE
C     IT IS DESIREABLE THAT 10=0 MOD DY

      ENTRY SCALETEN
      ASSIGN 102 TO DYGATE
      YMIN = YMAX = Y(1)
      GO TO 100

C     ENTRY ZEROTEN ADDS THE VALUE ZERO TO THOSE VALUES TO BE SCALED BY SCALETEN

      ENTRY ZEROTEN
      ASSIGN 102 TO DYGATE
      YMIN = YMAX = 0

  100 M = K*(N-1) + 1
      DO 4 I=1,M,K
      IF (Y(I) - YMIN) 1,4,2
    1 YMIN = Y(I)
      GO TO 4
    2 IF (YMAX - Y(I)) 3,4,4
    3 YMAX = Y(I)
    4 CONTINUE

C     CALCULATE APPROPRIATE DY VALUE AND ADJUST YMIN

    8 RANGE = YMAX - YMIN
      IF (RANGE) 24,24,25

C     SCALE A STRAIGHT LINE WHEN YMIN = YMAX
```

```
   24 DY = 1.0
      IF (YMIN.EQ.0.0) RETURN
      YMIN = 0
      DY = RØUNDEXP(YMAX)*10./HEIGHT
      GØ TØ 20

   25 DY = RANGE/HEIGHT
      IF (YMIN) 11,9,12
   11 IF (-YMIN -DY/10.0) 10,10,14
   12 IF (YMIN - DY) 10,14,14

C     WHEN YMIN CLØSE TØ 0 (BETWEEN +1 AND -.1 INCH), SET YMIN=0 AND RESCALE

   10 YMIN = 0
      GØ TØ 21

C     WHEN YMIN IS NØT NEAR 0, TRUNCATE TØ NEAREST RØUND NUMBER

   14 YMIN = RØUNDØWN(YMIN)

C     RECALULATE RANGE WITH NEW YMIN VALUE

   21 RANGE = YMAX - YMIN
      DY = RANGE/HEIGHT
    9 GØ TØ DYGATE,(101,102)

C     ADJUST DY VALUE TØ NEXT BIGGEST INTEGER (TIMES SCALE FACTØR)

  101 DY = RØUNDUP(DY)
      GØ TØ 109

C     ADJUST DY VALUE TØ 1,2, ØR 5 (TIMES SCALE FACTØR)

  102 CALL NØRMAL(DY,IEXP)
      IF (DY -5.0) 105,104,103
  103 DY = 10.**(IEXP+1)
      GØ TØ 109
  104 DY = 5.*10.**IEXP
      GØ TØ 109
  105 IF (DY-2.0) 107,106,104
  106 DY = 2.*10.**IEXP
      GØ TØ 109
  107 IF (DY - 1.0) 108,108,106
  108 DY = 10.**IEXP

  109 CØNTINUE

C     TEST YMIN = 0 MØD DY

      AMØD = ABSF(YMIN/DY)
      IF (AMØD-INTF(AMØD) -.01) 20,20,15
   15 IF (-AMØD +INTF(AMØD) +.99) 20,20,16
   16 IF (YMIN) 17,20,18
   17 YINTS = INTF(YMIN/DY -.95)
      GØ TØ 19
   18 YINTS = INTF(YMIN/DY)
```

```
   19 YMIN = YINTS*DY
      GO TO 21
C     SCALE ARRAY TO DY UNITS/INCH
C
   20 DO 7 I=1,M,K
    7 Y(I) = (Y(I) - YMIN)/DY
      RETURN
      END
```

```
      FUNCTION ROUND (X)

C     FUNCTION ROUND GIVES THE ROUNDED VALUE OF ANY + OR - REAL NUMBER,
C     THE RESULT IS AN INTEGER TIMES SOME POWER OF TEN.

C     EXAMPLES OF THE THREE ROUND FUNCTIONS

C         X            ROUND(X)   ROUNDUP    ROUNDOWN
C     .........       .........  .........  .........
C         534.0        500.0      600.0      500.0
C         0.6666       0.7        0.7        0.6
C         -1.5         -2.0       -1.0       -2.0
C         -.0333       -.03       -.03       -.04

      ARG = X
      CALL NORMAL (ARG,IEXP)
      IF (ARG) 3,2,1
    1 TARG = INTF(ARG+.5)
      GO TO 4
    3 TARG = INTF(ARG-.5)
    4 ROUND = TARG*10.**IEXP
    2 RETURN

C     ROUNDOWN GIVES THE NEXT LOWEST ROUND NUMBER

      ENTRY ROUNDOWN
      ARG = X
      CALL NORMAL (ARG,IEXP)
      TARG = INTF(ARG)
      IF (TARG.GT.ARG) TARG = TARG - 1.0
      ROUND = TARG*10.**IEXP
      RETURN

C     ROUNDUP GIVES THE NEXT HIGHEST ROUND NUMBER

      ENTRY ROUNDUP
      ARG = X
      CALL NORMAL (ARG,IEXP)
      TARG = INTF(ARG)
      IF (TARG.LT.ARG) TARG=TARG+1.0
      ROUND = TARG*10.**IEXP
      RETURN

C     ROUNDEXP(X) GIVES THE SCALE FACTOR FOR A GIVEN X, WHEN REDUCED TO THE
C     NORMALIZED FORM, BETWEEN 1.000 AND 9.999 TIMES THE SCALE FACTOR.
C     WHEN X=0, ROUNDEXP GIVES THE SCALE FACTOR FOR THE PREVIOUSLY ROUNDED X.

      ENTRY ROUNDEXP
      IF (X.EQ.0) GO TO 5
      ARG = X
      CALL NORMAL (ARG,IEXP)
      GO TO 6
    5 IF (ABSF(TARG).EQ.10.) IEXP = IEXP+1
    6 ROUND = 10.**IEXP
      RETURN
      END
```

FTN5.4A 12/31/68

```
      SUBROUTINE NORMAL (ARG,IEXP)
C     NORMAL TAKES ANY NUMBER, ARG, AND NORMALIZES IT, IE, CONVERTS IT
C     TO THE FORM, ARG*10**IEXP,  WHERE 1.LE.ARG.LT.10.
      SIGN = +1.0
      IEXP = 0
      IF (ARG) 6,5,1
    6 SIGN = -1.0
      ARG = -ARG
    1 IF (ARG - 10.0) 2,4,4
    2 IF (ARG - 1.0) 3,5,5
    3 ARG = ARG*10.0
      IEXP = IEXP - 1
      GO TO 1
    4 ARG = ARG/10.0
      IEXP = IEXP + 1
      GO TO 1
    5 ARG = SIGN*ARG
      RETURN
      END
```

APPENDIX B: PROGRAMMING DETAILS

I. Input to WAVETRAN

Card 1

Contains six quantities: XMAX, HT, ITMAX, ITSUM, ISCOPE, IPRINT in format (F5.3, E10.2, 4I5). They are defined as follows. XMAX = The distance along the surface of the inner conductor in cms. HT = The size of the time step in nsecs. ITMAX = The number of time steps. ITSUB = The number of subdivisions of the time step, HT, to be used for the calculations and the voltage plot, but omitted from the printout and the movie output. ISCOPE = 1 for movie output, = 0 to omit the movie. IPRINT = 1 for printout of forward and backward waves, = 0 to omit printout.

Card 2

Contains two numbers: N, HX in format (I5, F101.1) which describe the table of velocity and impedance values to follow. N = The number of cards in the following table. HX = The length of the X-interval between each of the following pairs of V and Z values, from 0 to (N-1)HX, (Case 1) or, if HX = 0, (Case 2) then the X value is given on each card.

Case 1

Cards 2.1 to 2.N - Each contain two numbers: V, Z, in format (2F10.1), or

Case 2

Cards 2.1 to 2.N - Each contain three numbers: V, Z, X, in format (3F10.1), where V = the velocity of the wave at each X value, in cms/nsecs. Z = the impedance of the wave at each X value. X = a distance along the surface of the inner conductor, $0 \leq X \leq X$ max. Card 3 - There may be 0, or as many as 20 cards placed here, terminated by an EOF card. These will be reproduced on the first frame of the movie produced by this run, and should be used to identify the movie, format (10A8).

II. Output from WAVETRAN

After adjusting the table of propagation velocity and impedance values, to a set of mean values over equal time intervals for the given time step, the program lists these values, and draws two line plots of velocity and impedance versus distance, on the on-line Calcomp plotter.

The program prints out the voltage wave amplitude all along the line at each time step. F denotes the foward wave, and B the backward wave. This printout may be omitted by setting IPRINT = 0 on Card 1.

A movie is written on magnetic tape on logical unit 16, showing the sum of the forward and backward wave amplitudes along the transmission line as they develop in time. Each movie frame represents one time step. The tape is later fed into a Stromberg-Carlson SC4020, which films the plots directly from the scope. This movie may be omitted by setting ISCOPE = 0 on Card 1.

The program also produces a graph of the output voltage versus time on the on-line Calcomp plotter.

EXTERNAL SUBROUTINES USED BY WAVETRAN FOR PLOTTING

In order to produce output plots on the Calcomp plotter, and movies on the Stromberg-Carlson SC4020 scope the following subroutines are employed.

Calcomp subroutines - These are in the CDC 3800 system library.

Call PLOTS - initializes the Calcomp plotting routines.
Call PLOT - plots a point.
Call LINE - plots a series of points.
Call AXIS - draws an axis.
Call STOPPLOT - terminates the Calcomp plotting.

SC4020 subroutines - These are on an axuiliary library tape. (A tape containing source versions of all the SC4020 subroutines is available on request.)

Call ENTFLM - initializes the SC4020 plotting.
Call XSCALV - computer scale factors for the x-direction.
Call YSCALV - computer scale factors for the y-direction.
Call NXV - converts x-values to raster coordinates.
Call NYV - converts y-values.
Call PRINTV - Projects hollerith characters on the scope.
Call FRAMEV - advances the film to the next frame.
Call LINEV - draws a line between two points.
Call BRITEV - turns on the bright intensity, for better visibility.
Call EXTFLM - terminates the SC 4020 plotting.

Description of these subroutines are available from the authors. These can be used to facilitate substitution of similar or equivalent routines whicy may be available at other computing facilities.

The Calcomp plotting output is assigned to logical unit 1. If this is not plotted on-line, a magnetic tape must be mounted on this unit to receive the plots.