CR 160105 21

ANTENNA EVALUATION STUDY

FOR THE SHUTTLE MULTISPECTRAL RADAR: PHASE III

FINAL REPORT

(NASA-CR-160105) ANTENNA EVALUATION STUDY FOR THE SHUTTLE MULTISPECTRAL RADAR, PHASE 3 Final Report (New Mexico State Univ.) 120 p HC A06/MF A01 CSCL 20N

N79-17070

Unclas
G3/32 14031

by

Edgar L. Coffey, III
Keith R. Carver

prepared for

NASA-Johnson Space Center
Houston, Texas

Contract No.
NAS9-95479

December 1978

NEW MEXICO STATE UNIVERSITY

Physical Science Laboratory

BOX 3-PSL, LAS CRUCES, NEW MEXICO 88003
AREA (505) 522-4400 TWX 910-983-0541

ANTENNA EVALUATION STUDY

FOR THE SHUTTLE MULTISPECTRAL RADAR: PHASE III

FINAL REPORT

by

Edgar L. Coffey, III
Keith R. Carver

prepared for

NASA Johnson Space Center
Houston, Texas

Contract No.
NAS9-95479

December 1978

TABLE OF CONTENTS

1.0 INTRODUCTION: SCOPE OF WORK AND ORGANIZATION OF REPORT

The central objective of Phases I, II, and III of this study has been to develop mathematically-based computer models which simulate the performance of electrically large space-borne imaging radar antennas. Such an approach provides the systems designer with a powerful tool for use in the analysis of tolerance and environmental effects on the performance of the system, and thus greatly reduces developmental and testing costs that would be associated with trial-and-error methods of assessing these effects.

This final report presents the results of the third phase of the Antenna Evaluation Study for the Shuttle Imaging Radar (SIR). The principal objectives of Phase III were (1) to continue simulations of the effects of excitation errors in the performance of a planar array antenna, (2) to apply this model specifically, to the SIR-A antenna developed by Ball Brothers, (3) to support measurements of the SIR-A antenna at JPL by quick-turnaround analyses of the SIR-A antenna under a variety of conditions encountered at the JPL West Mesa Antenna Range, and (4) to document fully all software developed for NASA Johnson Space Center (JSC) during Phases I, II, and III of this contract. In addition, support was provided to JSC in the following areas:

1. Presentation at JSC of significant results concerning other possible antenna configurations, potential problems with SIR-A, potential problems at X-band, etc. (January 2-3, 1978). A demonstration of all software developed for the Tektronix 4051 graphics terminal was made at the same time.

2. Participation in Preliminary Design Reviews (PDRs) and Critical Design Reviews (CDRs) of the SIR-A antenna and interface.

3. Investigation of the effects of reflection of radiated energy from the SIR-A antenna in a building similar to the KSC O&C building.

Some of the tasks originally scheduled for the Phase III effort were either performed in Phases I and II (such as the computer model development) or were shared with the companion JSC contract for Analysis and Support of SIR-A development and test.

Because of the large amounts of data that can be generated by the computer programs developed in all three phases, it seems more logical and efficient to present samples of the data that illustrate the potential and versatility of the software rather than provide an encyclopedic report. Ample references are made to other reports so that the interested reader may study the actual data generated and conclusions drawn. This final report is divided into the following sections:

Section 2.0	The Shuttle Imaging Radar Antenna General Footprint Simulation Program
Section 3.0	The Shuttle Imaging Radar Antenna Specialized Far-Field Pattern Program
Section 4.0	Ray Tracing Program for the SIR-A Antenna and Payload Bay Inside an Enclosed Building
Section 5.0	Summary

The examples used to illustrate the use of the computer software are actual cases simulated for JSC use. Complete listings of each program may be found in the Appendix. All TEKTRONIX software developed in the course of this contract is available on DC-300A cassette tape compatible with the TEKTRONIX 4051.

2.0 THE SHUTTLE IMAGING RADAR ANTENNA GENERAL FOOTPRINT SIMULATION PROGRAM

The mathematical model and the computer program for antenna simulations were developed and tested during Phase I; the majority of the simulation scenarios were carried out during Phase II and reported in the Phase II Final Report [2, pp. 16-60] and elsewhere [3, 4, 5]. Traditional analyses of the antenna footprint on the earth's surface required assumptions of uniform antenna illumination, planar geometry, etc., that were too restrictive for the information needed in this study. Hence a more general computer model was developed by PSL to gain insight into the effect of antenna surface flatness on:

1. Beam pointing error
2. Gain degradation
3. Main beam spreading and breakup
4. Sidelobe level degradation

No changes in the mathematics of the model have been made since Phase I [1, pp. 18-30], but some modifications to the computer code have been implemented since Phase II [1, pp. 31-37]. These changes will be summarized in this section, and detailed information regarding data input and output will be documented. One sample program output will be included for completeness and continuity.

2.1 Modifications to the Footprint Program

The only modifications to the program which are visible to the user are in the auxiliary I/O routines of Figure 1. These routines (Tape1, Tape2, DISK, and STATUS) were not implemented at PSL because there was no subsequent demonstration of need. These routines should be deleted from Figure 1 (and Figure 8 of Reference 1). However, if the need arises, these routines may be added. The only potential problem is that of compatibility between computer tapes and disk drives. Therefore, before these I/O routines are written, users should agree on media types, formats, etc.

2.2 User's Guide to the Footprint Program

2.2.1 Program Description

The function of each routine listed in Figure 1 is described.

MAINPGM - Performs mostly housekeeping chores and calls other routines according to user input commands. It calculates the "YAW-TILT-TWIST" array, predicted beam center location, plot normalization factor (pattern value at predicted beam center location), increments beam pointing angle to cover entire "footprint" region and initiates profiles along lines of constant latitude and longitude intersecting the predicted beam center.

ANTENNA - Inputs and calculates appropriate antenna parameters: number of elements, spacing, phase shift, number of subarrays, polarization, and element type. After calculations have been performed, a summary is printed.

MECH - Calculates mechanical deformation data based on the inputs from STRESS, THERML, and MISC. For each subarray, MECH calculates the average displacement ZAVG, the tilt angles, ALPHAX and ALPHAY, and the error coefficient of the bilinear approximations. A summary is printed after execution.

MISC - Inputs any other deforamtion data not covered by STRESS and THERML.

ELEC - Inputs the amplitude, phase, and linear phase gradients of each subarray. This information is passed to PAT via common block ELCTRC, and a summary is printed.

ORBIT - Inputs orbital parameters and antenna YAW, TILT, and TWIST, then prints a summary.

PAT - Is the basic pattern subprogram. Given latitude-longitude coordinates on the earth's surface, it performs all translations and rotations to determine the beam pointing angle (u,v). It then calls AF to find the subarray factor and finally computes the sum of all subarrays.

AF - This function calculates the array factors of a subpanel by "table look-up" with linear interpolation if $|u| < \text{UMAX}$ and $|v| < \text{VMAX}$. Otherwise, the AF is computed in the usual way.

GENER - This routine generates the table used by AF. Table length is 1001 words: UMAX and VMAX are chosen to include the main beam plus the first three side lobes.

OUTPUT - Translates user commands for printer and pattern output into logical switches for the program.

PROFIL - Is the printer profile plot routine. It will print a one-dimensional plot down the page for up to 501 data points. The exact values of both the abscissa and ordinate are printed with each data point, the ordinate also being printed in dB.

PATCON - Is chiefly a "bookkeeping" subroutine for CONTUR. It calls subroutine CONTUR three times to generate a complete 151 x 151 two-dimensional contour map.

CONTUR - Prints a 51 x 151 contour plot of the footprint. To obtain a complete 151 x 151 plot, this routine is called three times by subroutine PATCON. Then the three separate plots are pasted together to make the composite.

PLOT1 - Is the Calcomp one-dimensional plot routine. It is called twice by the main program, and it is used to generate the profile plots of the principle planes through the main beam.

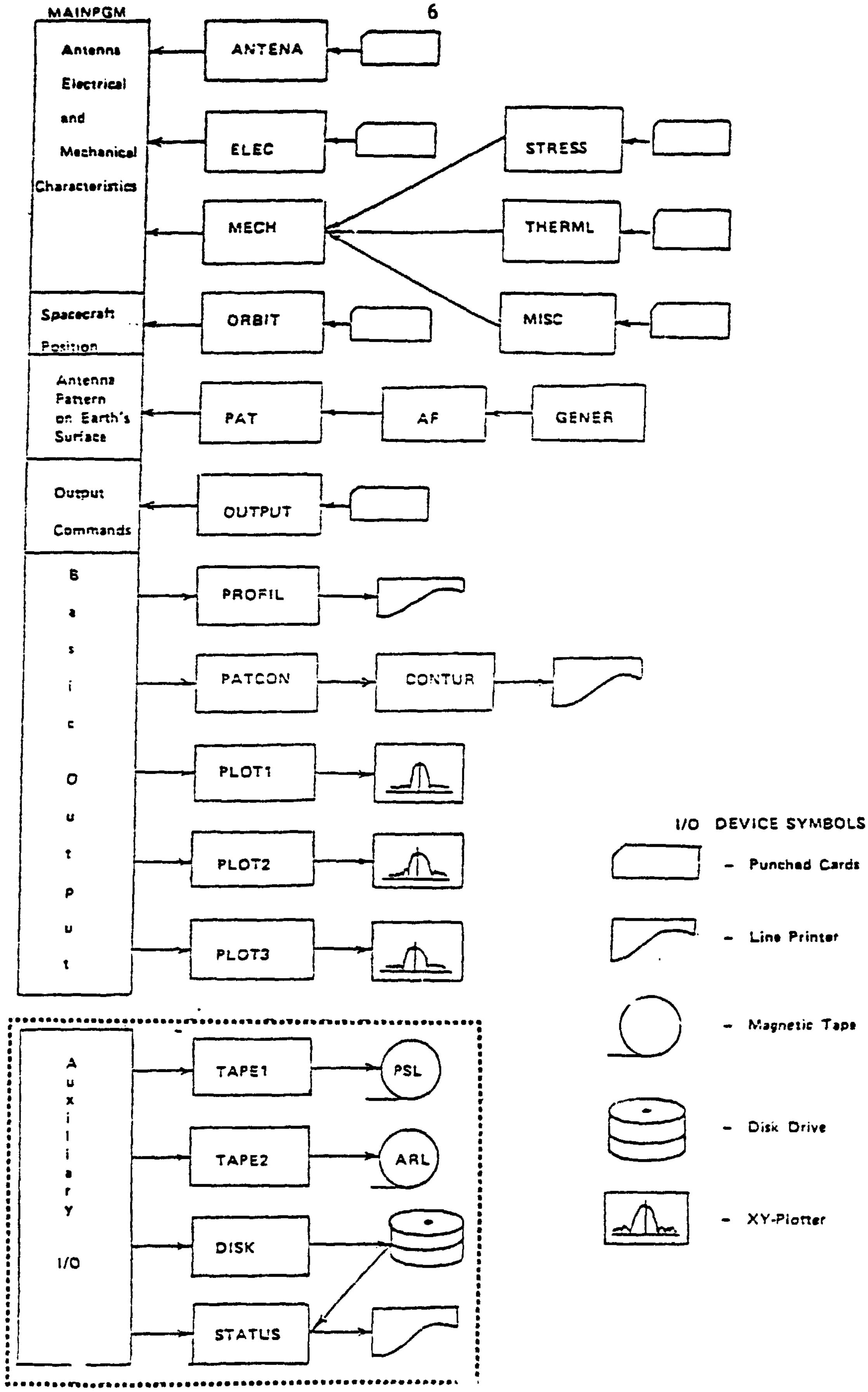


Figure 1. Organization of the final Footprint Simulation Program. (Deleted routines are enclosed in a dotted line.)

PLOT2 - Is the Calcomp coutour map routine. From the data used to generate the printer contour map, it draws a continuous contour plot of the antenna ground footprint over the rectangular region specified by the user.

PLOT3 - Is the Calcomp three-dimensional plot routine. It plots the magnitude (in dB) of the footprint pattern over rectangular region specified by the user.

2.2.2 Input/Output Commands and Data Cards

The program input data has been structured so that user knowledge of computer fundamentals is not required. Consequently, the program can be used by anyone desiring information concerning the footprint of a particular antenna configuration.

The input has been divided into six modules. These are:

A. Simulation Information
B. Antenna Position and Orientation
C. Antenna Configuration
D. Antenna Mechanical Parameters
E. Antenna Electrical Parameters
F. Output Commands and Parameters

By including input parameters in these six categories, it is possible for the user to simulate a wide range of electrical, mechanical, and physical scenarios. Furthermore, a change in the simulation does not require that the complete input deck be repunched. When related scenarios are being studied, only one of the six modules need be changed.

Simulation Information

Card 1: Simulation number (I5)
Date (5A4)

Card 2: Narative description of simulation (80A1)
End narrative with ampersand (&)

Antenna Position and Orientation

Card 1: Frequency in GHz (F10)
Card 2: Orbit altitude (km) (F10)
Card 3: Yaw, tilt, twist angles in degrees (3F10)

Antenna Configuration (Subroutine ANTENA)

Card 1: Number of subarray sections in aximuth and elevation (2I5)
Card 2: Number of elements in azimuth and elevation (2I5)
Card 3: Element spaceing (cm) in azimuth and elevation (2F10)
Card 4: Interelement phase shift (degrees) in azimuth and elevation (2F10)
Card 5: Polarization (0=isotropic, 1=horizontal, 2=vertical) (I5)

Antenna Mechanical Information (Subroutines MECH and MISC)

Cards 1 Warp array in cm (8F10). The displacement of each corner of the subarrays is entered. For example, suppose a systematic tilt as shown below is to be simulated. Then the data entered would be:

```
0.0  1.0  2.0  3.0  4.0  5.0  6.0  1.0  (Card 1)
2.0  3.0  4.0  5.0  6.0  7.0  2.0  3.0  (Card 2)
4.0  5.0  6.0  7.0  8.0                 (Card 3)

     0.0  1.0  2.0  3.0  4.0  5.0  6.0
     1.0  2.0  3.0  4.0  5.0  6.0  7.0
     2.0  3.0  4.0  5.0  6.0  7.0  8.0
```

Antenna Electrical Parameters (Subroutine ELEC)

Card 1 One card for each subarray section to include:
Phase shift in azimuth (degrees)
Phase shift in elevation (degrees)
Magnitude of excitation (volts)
Phase of excitation (degrees)
(4F10)

Output Commands and Parameters (Subroutine OUTPUT)

Card 1: Printer output commands (2I1)
PR1=1: Printer profile plots
PR2=1: Printer contour map of footprint

Card 2: Plotter output commands (3I1)
PL1=1: Plotter profile plots
PL2=1: Plotter contour map of footprint
PL3=1: Plotter three-dimensional relief map of footprint

[Cards 3 - 8: Printer/Plotter output parameters]

Card 3: Limits on X-axis (degrees longitude) (2F10.0)

Card 4: Limits on Y-axis (degrees latitude) (2F10.0)

Card 5: Number of contours to be plotted
PL2=1: Use Card 5
PL2=0: Omit Card 5

Card 6: Contour levels to be plotted (dB) (8F10.0)
PL2=1: Use Card 6
PL2=0: Omit Card 6

Card 7: Limits on printer contour map (dB) (2F10.0)
PR2=1: Use Card 7
PR2=0: Omit Card 7

Card 8: Number of points along each axis for the printer contour map (2I5), maximum of 151 x 151 points
PR2+PL2+PL3 $\geq$ 1: Use Card 8
PR2+PL2+PL3 = 0: Omit Card 8

2.3 Example of Input/Output with the Footprint Program

Many simulations were run for the seven-panel SIR-A array antennas. The characteristics of this antenna are listed below:

Frequency = 1.275 GHz
Number of Panels = 7
Element Spacing = 16.75 x 13.00 cm
Excitation = 1/8, 1/8, 1/6, 1/6, 1/6, 1/8, 1/8

Suppose it is desired to simulate this antenna in earth orbit (altitude = 200 km) and a tilt angle of 47° with a parabolic bow deformation of one cm at the edges. The data card deck would be composed of:

simulation information
```
00001 16DEC78
EXAMPLE 1:  BBRC 7-PANEL ARRAY WITH NATURAL TAPER
AND PARABOLIC BOW OF ONE CM &
```

antenna position
```
1.275
200.0
0.0       47.0      0.0
```

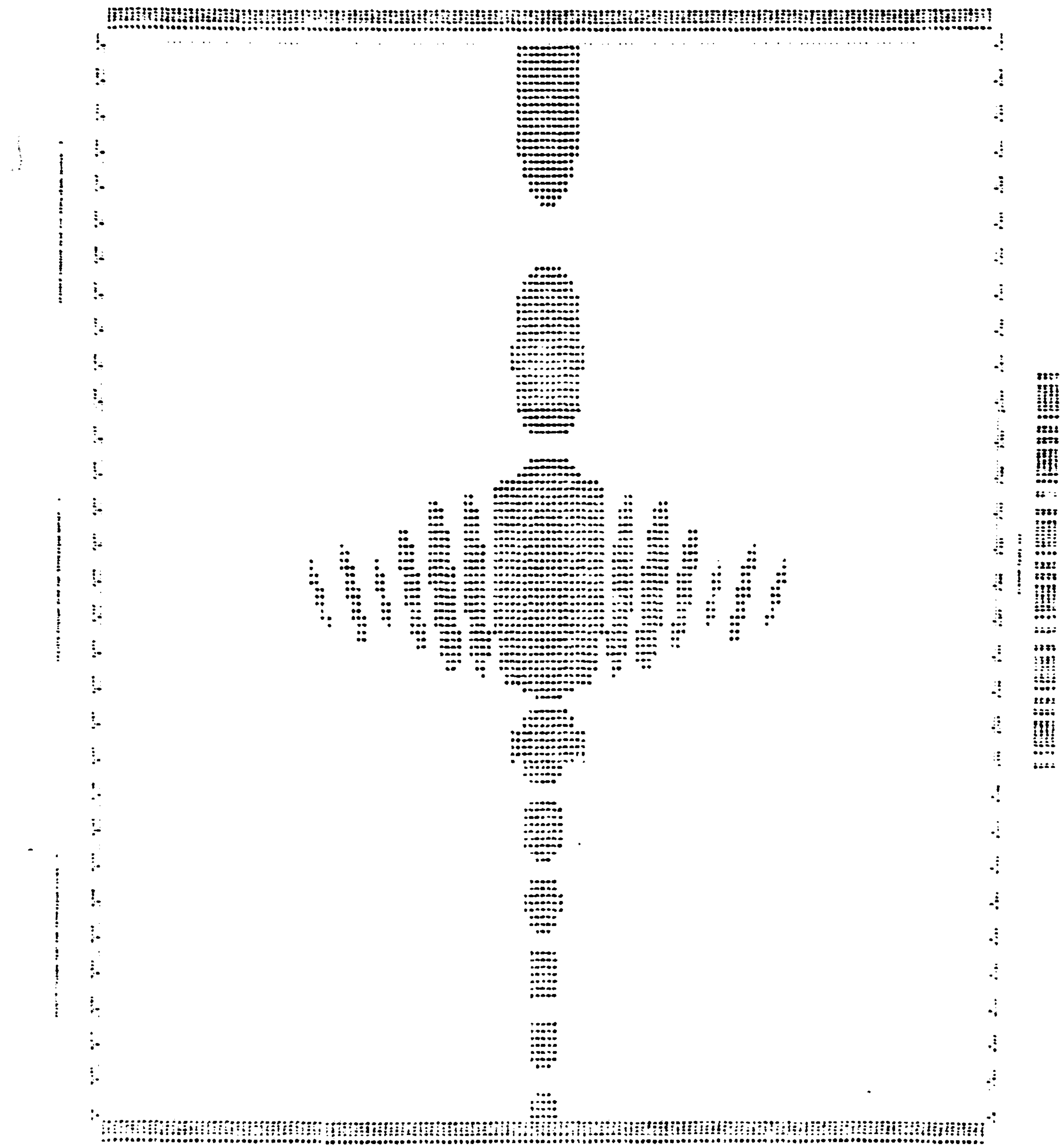
antenna configuration
```
0000700001
0005600016
16.75     13.00
0.0       0.0
00001
```

mechanical information
```
1.0  0.5  0.166  0.0  0.0  0.166  0.5  1.0
1.0  0.5  0.166  0.0  0.0  0.166  0.5  1.0
[NOTE:  Some blanks have been deleted in the
above two lines to save space . . . use 8F10
format]
```

excitation
```
0.0       0.0       0.875       0.0
0.0       0.0       0.875       0.0
0.0       0.0       1.1667      0.0
0.0       0.0       1.1667      0.0
0.0       0.0       1.1667      0.0
0.0       0.0       0.875       0.0
0.0       0.0       0.875       0.0
```

output
```
11
111
-1.0                1.0
0.0                 4.0
00005
0.0       3.0       -10.0     -20.0     -27.0
0.0       -27.0
0015100151
```

The complete output from this simulation is shown on the following pages, with the exception that the printer contour plot pages have been pasted together and photo-reduced for clarity. The plotter form output has not been included.



X-AXIS PROFILE PLOT ALONG 1.966 DEGREES LATITUDE.
PATTERN NUMBER 1

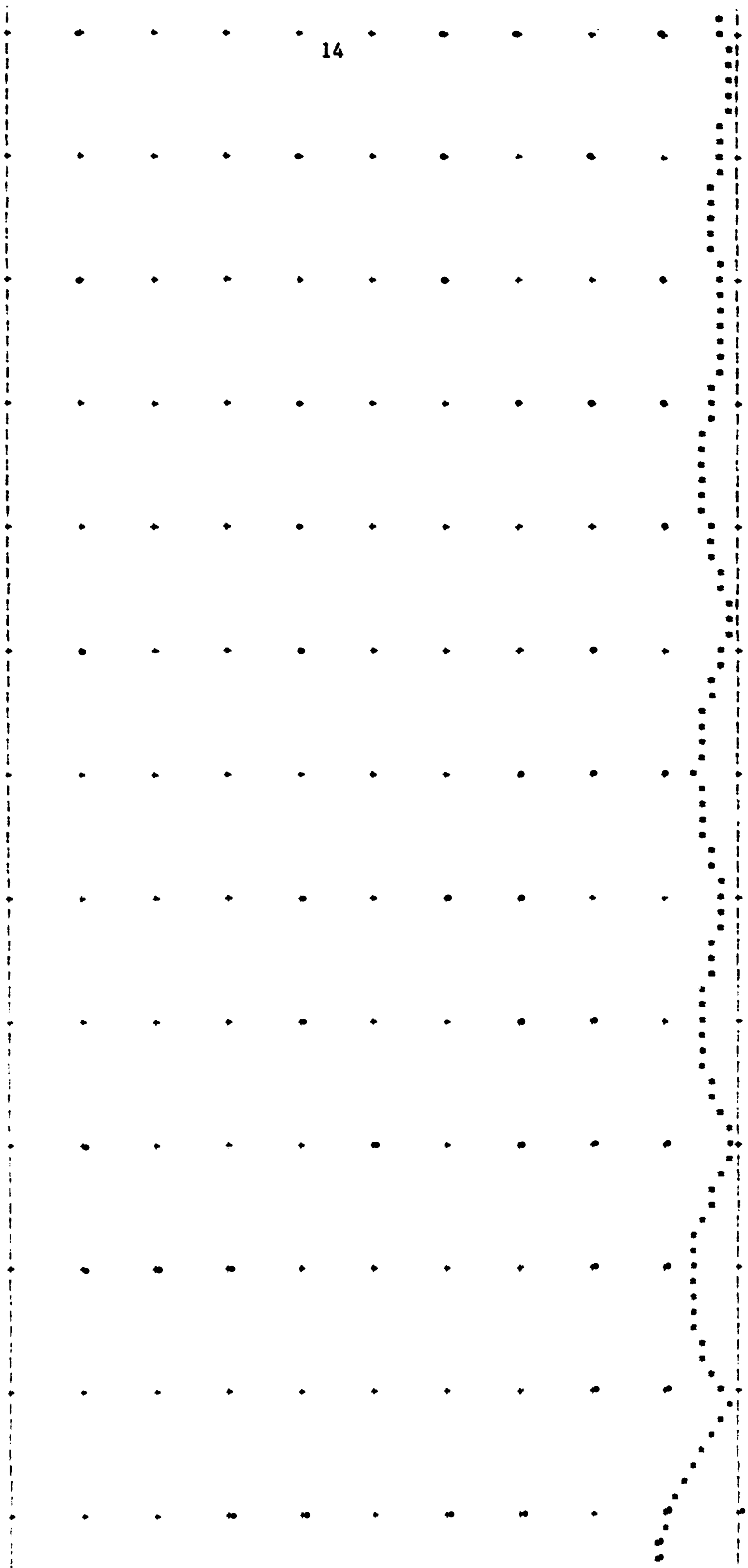
SCALE FACTOR IS 10**-2

99.9999 80.0027 60.0055 40.0082 20.0110 0.0138

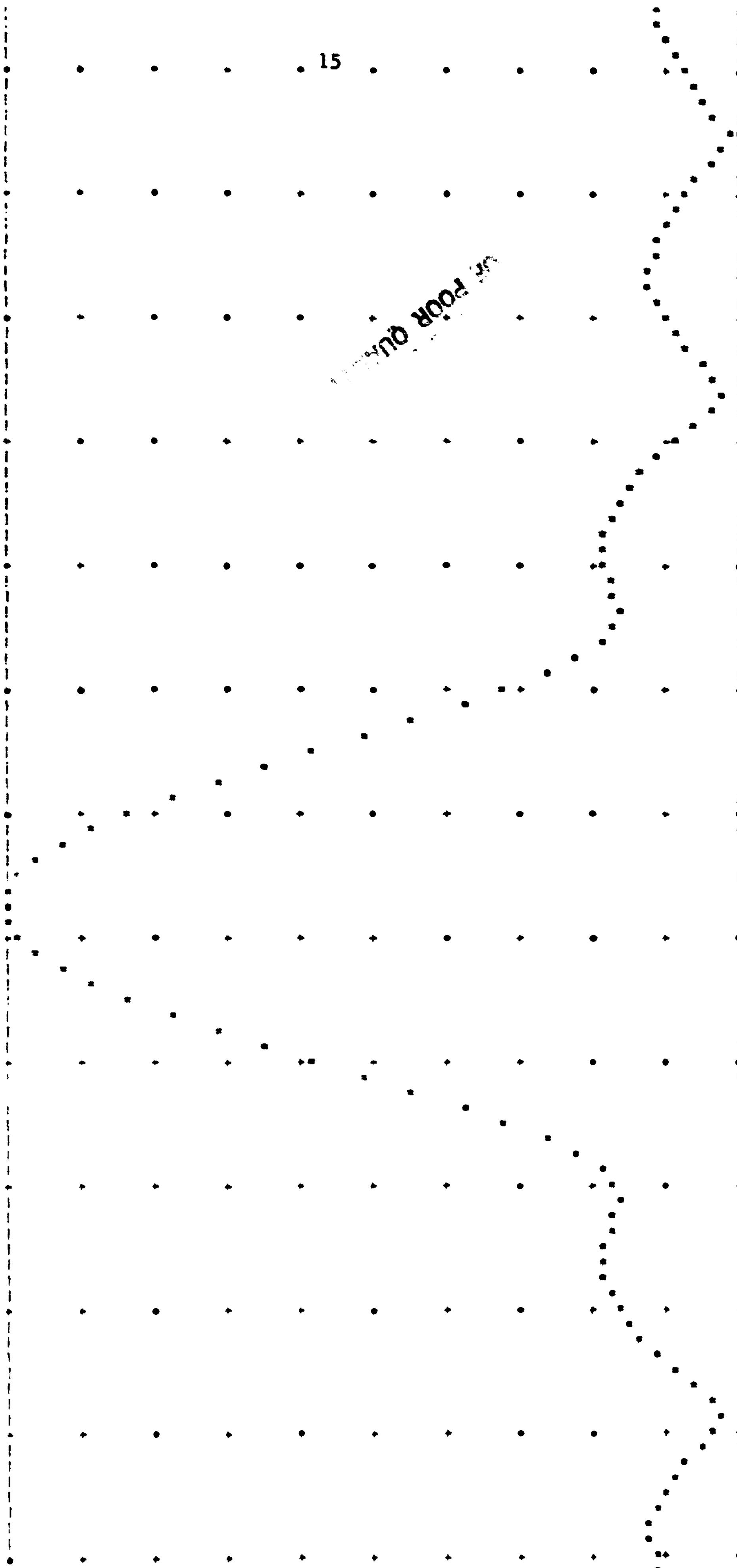
DB.	REAL	
-41.34	0.857	1.0000
-38.79	1.151	0.9960
-36.93	1.425	0.9920
-35.54	1.670	0.9880
-34.50	1.883	0.9840
-33.73	2.059	0.9800
-33.18	2.193	0.9760
-32.83	2.283	0.9720
-32.66	2.327	0.9680
-32.67	2.325	0.9640
-32.85	2.279	0.9600
-33.20	2.137	0.9560
-33.73	2.057	0.9520
-34.46	1.893	0.9480
-35.38	1.732	0.9440
-36.51	1.495	0.9400
-37.81	1.287	0.9360
-39.19	1.099	0.9320
-40.33	0.963	0.9280
-40.81	0.911	0.9240
-40.38	0.957	0.9200
-39.35	1.078	0.9160
-38.14	1.235	0.9120
-37.04	1.407	0.9080
-36.13	1.562	0.9040
-35.44	1.690	0.9000
-34.99	1.781	0.8960
-34.75	1.830	0.8920
-34.75	1.831	0.8880
-34.99	1.783	0.8840
-35.47	1.685	0.8800
-36.26	1.533	0.8760
-37.42	1.346	0.8720
-39.09	1.141	0.8680
-41.51	0.840	0.8640
-45.26	0.539	0.8600
-53.23	0.217	0.8560
-58.25	0.122	0.8520
-46.53	0.463	0.8480
-41.93	0.801	0.8440
-39.98	1.125	0.8400
-36.92	1.426	0.8360
-35.41	1.696	0.8320
-34.31	1.925	0.8280
-33.53	2.107	0.8240
-33.01	2.236	0.8200
-32.76	2.326	0.8160
-32.71	2.314	0.8120
-32.92	2.253	0.8080
-33.40	2.139	0.8040
-34.17	1.957	0.8000
-35.31	1.716	0.7960
-36.94	1.423	0.7920
-39.29	1.085	0.7880
-42.87	0.719	0.7840
-44.44	0.579	0.7800
-48.77	0.364	0.7760
-42.77	0.727	0.7720
-38.74	1.156	0.7680
-35.99	1.537	0.7640
-33.99	1.999	0.7600
-32.42	2.373	0.7560
-31.33	2.714	0.7520
-30.47	2.995	0.7480
-29.85	3.215	0.7440
-29.46	3.365	0.7400
-29.27	3.443	0.7360
-29.27	3.439	0.7320
-29.47	3.359	0.7280
-29.89	3.232	0.7240
-30.56	2.971	0.7200
-31.47	2.671	0.7160
-32.73	2.310	0.7120
-34.44	1.897	0.7080
-36.81	1.443	0.7040
-40.32	0.964	0.7000
-46.22	0.489	0.6960
-51.84	0.255	0.6920
-43.91	0.638	0.6880
-39.19	1.098	0.6840
-36.27	1.535	0.6800
-34.29	1.929	0.6760
-32.80	2.266	0.6720
-31.92	2.534	0.6680
-31.28	2.729	0.6640
-30.94	2.839	0.6600
-30.85	2.867	0.6560
-31.03	2.809	0.6520
-31.43	2.656	0.6480
-32.24	2.444	0.6440

14

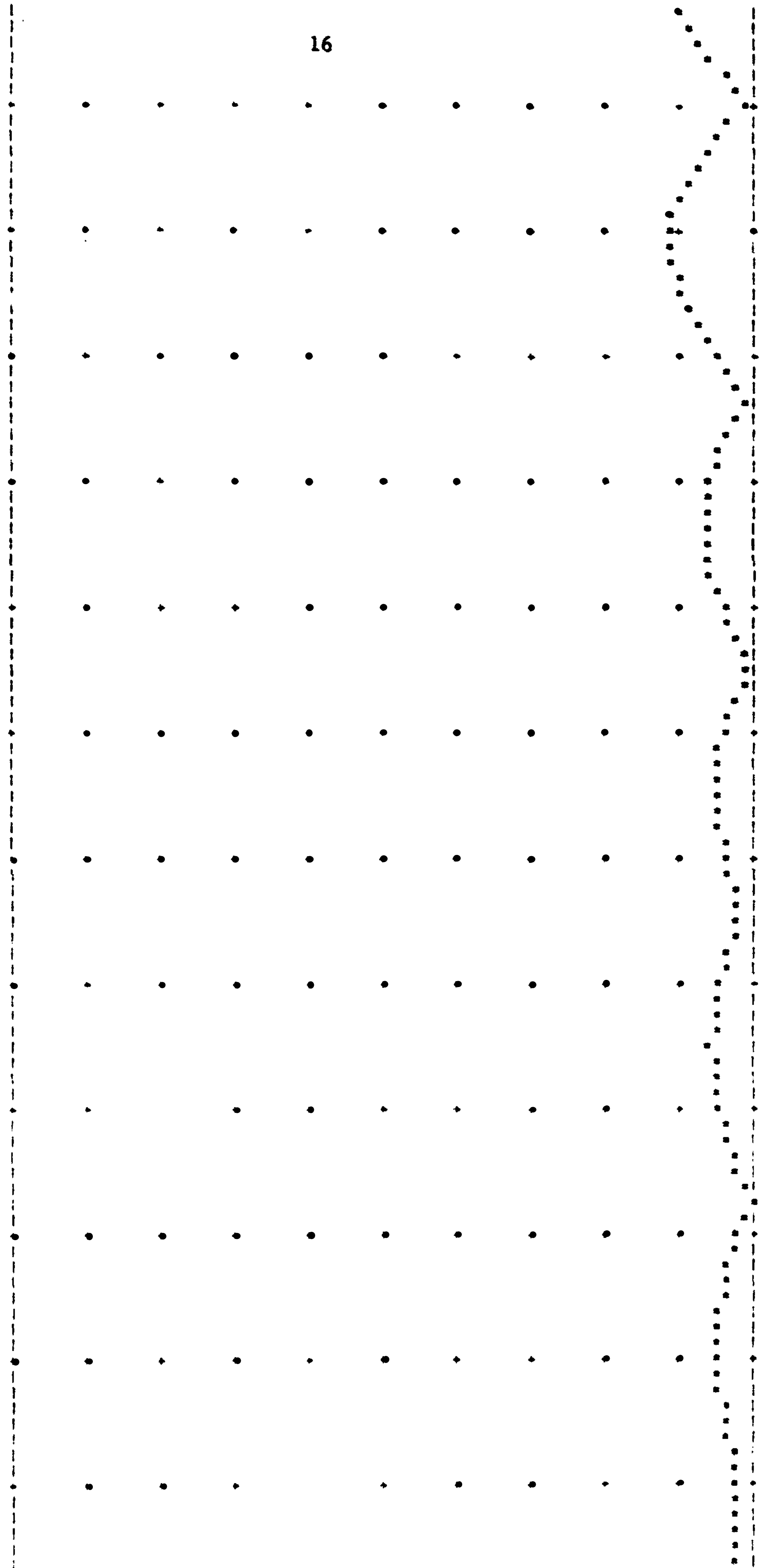
-34.75	1.759	0.6360
-37.23	1.376	0.6320
-40.79	0.922	0.6280
-44.39	0.647	0.6240
-57.02	0.141	0.6200
-44.78	0.577	0.6160
-39.59	1.049	0.6120
-36.52	1.493	0.6080
-34.47	1.891	0.6040
-33.03	2.231	0.6000
-32.03	2.502	0.5960
-31.39	2.695	0.5920
-31.34	2.804	0.5880
-30.97	2.823	0.5840
-31.15	2.763	0.5800
-31.60	2.631	0.5760
-32.29	2.429	0.5720
-33.21	2.186	0.5680
-34.26	1.936	0.5640
-35.20	1.738	0.5600
-35.57	1.664	0.5560
-35.06	1.756	0.5520
-33.84	2.031	0.5480
-32.39	2.400	0.5440
-31.02	2.813	0.5400
-29.33	3.223	0.5360
-28.39	3.598	0.5320
-28.15	3.912	0.5280
-27.65	4.145	0.5240
-27.36	4.294	0.5200
-27.30	4.317	0.5160
-27.46	4.239	0.5120
-27.86	4.045	0.5080
-28.55	3.737	0.5040
-29.57	3.321	0.5000
-31.04	2.835	0.4960
-33.15	2.200	0.4920
-36.35	1.523	0.4880
-42.05	0.790	0.4840
-73.26	0.022	0.4800
-42.38	0.751	0.4760
-36.29	1.535	0.4720
-32.85	2.279	0.4680
-30.55	2.970	0.4640
-28.70	3.588	0.4600
-27.71	4.125	0.4560
-26.37	4.535	0.4520
-26.31	4.835	0.4480
-25.31	5.006	0.4440
-25.94	5.044	0.4400
-26.11	4.949	0.4360
-26.31	4.728	0.4320
-27.15	4.393	0.4280
-28.04	3.962	0.4240
-29.21	3.465	0.4200
-30.61	2.946	0.4160
-32.13	2.475	0.4120
-33.34	2.133	0.4080
-33.59	2.072	0.4040
-32.73	2.309	0.4000
-31.35	2.705	0.3960
-29.99	3.166	0.3920
-28.86	3.607	0.3880
-28.01	3.975	0.3840
-27.46	4.235	0.3800
-27.20	4.364	0.3760
-27.24	4.346	0.3720
-27.59	4.175	0.3680
-28.29	3.849	0.3640
-29.44	3.372	0.3600
-31.19	2.753	0.3560
-33.89	2.021	0.3520
-38.54	1.183	0.3480
-51.25	0.274	0.3440
-43.21	0.691	0.3400
-35.53	1.563	0.3360
-31.65	2.615	0.3320
-29.09	3.515	0.3280
-27.29	4.326	0.3240
-25.99	5.019	0.3200
-25.09	5.564	0.3160
-24.53	5.936	0.3120
-24.27	6.116	0.3080
-24.31	6.099	0.3040
-24.65	5.848	0.3000
-25.36	5.392	0.2960
-26.50	4.730	0.2920
-28.22	3.891	0.2880
-30.91	2.880	0.2840
-34.34	1.312	0.2800
-39.39	1.075	0.2760
-35.57	1.666	0.2720
-30.64	2.933	0.2680
-27.22	4.325	0.2640
-24.33	5.698	0.2600
-23.11	6.991	0.2560
-21.77	8.152	0.2520
-20.73	9.136	0.2480
-20.03	9.905	0.2440
-19.56	10.425	0.2400
-19.44	10.672	0.2360

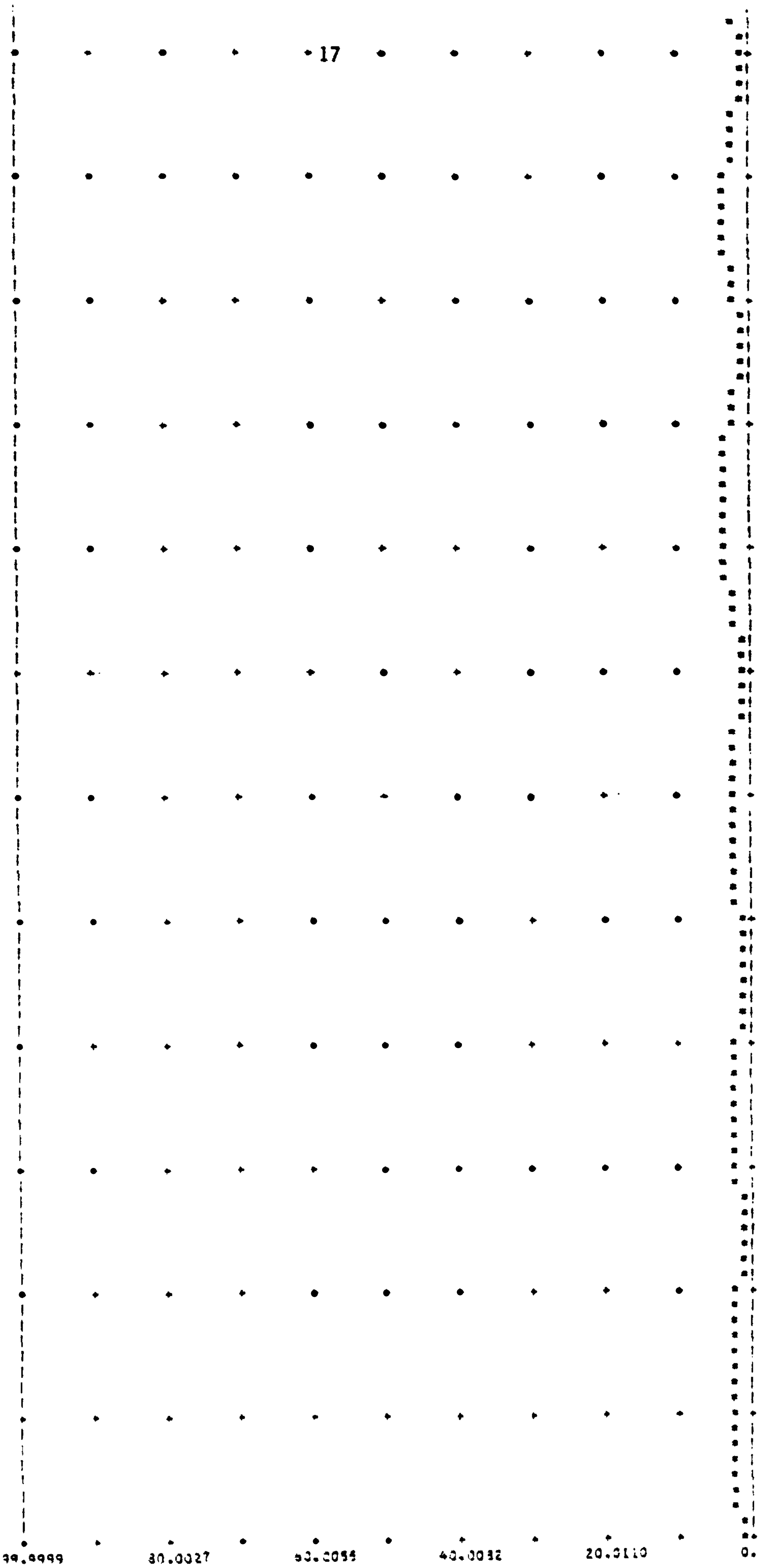


[illegible]	[illegible]	0.2320
-19.76	10.293	0.2280
-20.32	9.443	0.2240
-21.20	8.708	0.2200
-22.49	7.303	0.2160
-24.34	6.068	0.2120
-27.07	4.429	0.2080
-31.56	2.642	0.2040
-41.75	0.817	0.2000
-37.61	1.317	0.1960
-29.85	3.216	0.1920
-25.70	5.071	0.1880
-23.36	6.793	0.1840
-21.63	8.319	0.1800
-20.36	9.593	0.1760
-19.52	10.565	0.1720
-19.02	11.194	0.1680
-18.83	11.446	0.1640
-18.96	11.302	0.1600
-19.37	10.753	0.1560
-20.17	9.805	0.1520
-21.43	8.478	0.1480
-23.34	6.810	0.1440
-26.26	4.866	0.1400
-31.04	2.807	0.1360
-36.06	1.575	0.1320
-33.32	2.155	0.1280
-25.35	5.592	0.1240
-21.33	8.099	0.1200
-19.57	10.505	0.1160
-17.92	12.700	0.1120
-16.72	14.595	0.1080
-15.85	16.117	0.1040
-15.29	17.202	0.1000
-14.98	17.817	0.0960
-14.91	17.959	0.0920
-15.05	17.675	0.0880
-15.35	17.089	0.0840
-15.68	16.440	0.0800
-15.85	16.119	0.0760
-15.58	16.629	0.0720
-14.70	18.399	0.0680
-13.33	21.558	0.0640
-11.72	25.943	0.0600
-10.09	31.304	0.0560
-8.55	37.355	0.0520
-7.16	43.877	0.0480
-5.90	50.672	0.0440
-4.30	57.551	0.0400
-3.43	64.355	0.0360
-2.98	70.922	0.0320
-2.26	77.101	0.0280
-1.64	82.763	0.0240
-1.13	87.780	0.0200
-0.72	92.044	0.0160
-0.40	95.462	0.0120
-0.15	97.960	0.0080
-0.04	99.484	0.0040
-0.00	100.000	0.0000
-0.04	99.497	-0.0040
-0.18	97.987	-0.0080
-0.40	95.503	-0.0120
-0.72	92.097	-0.0160
-1.13	87.844	-0.0200
-1.64	82.340	-0.0240
-2.25	[illegible]	-0.0280
-2.97	71.009	-0.0320
-3.82	64.447	-0.0360
-4.73	57.649	-0.0400
-5.89	50.760	-0.0440
-7.14	43.972	-0.0480
-8.53	37.442	-0.0520
-10.07	[illegible]	-0.0560
-11.69	[illegible]	-0.0600
-13.31	[illegible]	-0.0640
-14.69	[illegible]	-0.0680
-15.57	16.645	-0.0720
-15.85	16.119	-0.0760
-15.51	16.432	-0.0800
-15.35	17.090	-0.0840
-15.05	17.659	-0.0880
-14.91	17.958	-0.0920
-14.93	17.823	-0.0960
-15.23	17.214	-0.1000
-15.84	16.134	-0.1040
-16.70	14.620	-0.1080
-17.90	12.730	-0.1120
-19.34	10.508	-0.1160
-21.79	8.135	-0.1200
-25.00	5.626	-0.1240
-29.93	3.187	-0.1280
-36.04	1.578	-0.1320
-31.12	2.778	-0.1360
-26.31	4.333	-0.1400
-23.27	6.734	-0.1440
-21.46	8.457	-0.1480
-20.19	9.789	-0.1520
-19.34	10.743	-0.1560
-18.94	11.297	-0.1600
-18.83	11.447	-0.1640
-19.02	11.200	-0.1680
-19.51	10.574	-0.1720



-20.35	9.609	-0.1760
-21.38	8.338	-0.1800
-23.33	6.615	-0.1840
-25.36	5.096	-0.1880
-29.79	3.243	-0.1920
-37.44	1.342	-0.1960
-42.00	0.795	-0.2000
-31.64	2.617	-0.2040
-27.12	4.404	-0.2080
-24.37	6.047	-0.2120
-22.51	7.489	-0.2160
-21.22	8.693	-0.2200
-20.33	9.629	-0.2240
-19.76	10.276	-0.2280
-19.47	10.625	-0.2320
-19.43	10.673	-0.2360
-19.63	10.431	-0.2400
-20.08	9.914	-0.2440
-20.77	9.148	-0.2480
-21.76	8.156	-0.2520
-23.09	7.008	-0.2560
-24.86	5.716	-0.2600
-27.24	4.344	-0.2640
-30.59	2.956	-0.2680
-35.49	1.681	-0.2720
-39.39	1.072	-0.2760
-34.90	1.799	-0.2800
-30.85	2.266	-0.2840
-28.25	3.370	-0.2880
-26.32	4.721	-0.2920
-25.38	5.385	-0.2960
-24.67	5.843	-0.3000
-24.31	6.087	-0.3040
-24.27	6.117	-0.3080
-24.52	5.940	-0.3120
-25.08	5.570	-0.3160
-25.97	5.027	-0.3200
-27.26	4.336	-0.3240
-29.05	3.525	-0.3280
-31.61	2.627	-0.3320
-35.52	1.675	-0.3360
-43.05	0.704	-0.3400
-51.63	0.262	-0.3440
-38.62	1.172	-0.3480
-33.93	2.011	-0.3520
-31.21	2.750	-0.3560
-29.46	3.356	-0.3600
-28.30	3.844	-0.3640
-27.59	4.172	-0.3680
-27.24	4.345	-0.3720
-27.20	4.364	-0.3760
-27.46	4.237	-0.3800
-28.01	3.973	-0.3840
-28.35	3.612	-0.3880
-29.97	3.172	-0.3920
-31.34	2.711	-0.3960
-32.72	2.312	-0.4000
-33.39	2.093	-0.4040
-33.35	2.151	-0.4080
-32.15	2.470	-0.4120
-30.63	2.941	-0.4160
-29.22	3.459	-0.4200
-28.05	3.957	-0.4240
-27.15	4.339	-0.4280
-26.51	4.725	-0.4320
-26.11	4.948	-0.4360
-25.95	5.044	-0.4400
-26.01	5.007	-0.4440
-26.31	4.837	-0.4480
-26.86	4.539	-0.4520
-27.70	4.121	-0.4560
-28.89	3.595	-0.4600
-30.52	2.977	-0.4640
-32.32	2.295	-0.4680
-36.23	1.543	-0.4720
-42.23	0.759	-0.4760
-77.22	0.014	-0.4800
-42.13	0.782	-0.4840
-36.39	1.515	-0.4880
-33.13	2.194	-0.4920
-31.06	2.799	-0.4960
-29.59	3.316	-0.5000
-28.56	3.724	-0.5040
-27.87	4.042	-0.5080
-27.46	4.237	-0.5120
-27.30	4.317	-0.5160
-27.36	4.285	-0.5200
-27.65	4.147	-0.5240
-28.15	3.914	-0.5280
-28.87	3.601	-0.5320
-29.82	3.227	-0.5360
-31.01	2.817	-0.5400
-32.33	2.404	-0.5440
-33.83	2.035	-0.5480
-35.25	1.755	-0.5520
-35.57	1.665	-0.5560
-35.20	1.737	-0.5600
-34.27	1.934	-0.5640
-33.22	2.183	-0.5680
-32.30	2.427	-0.5720
-31.60	2.629	-0.5760





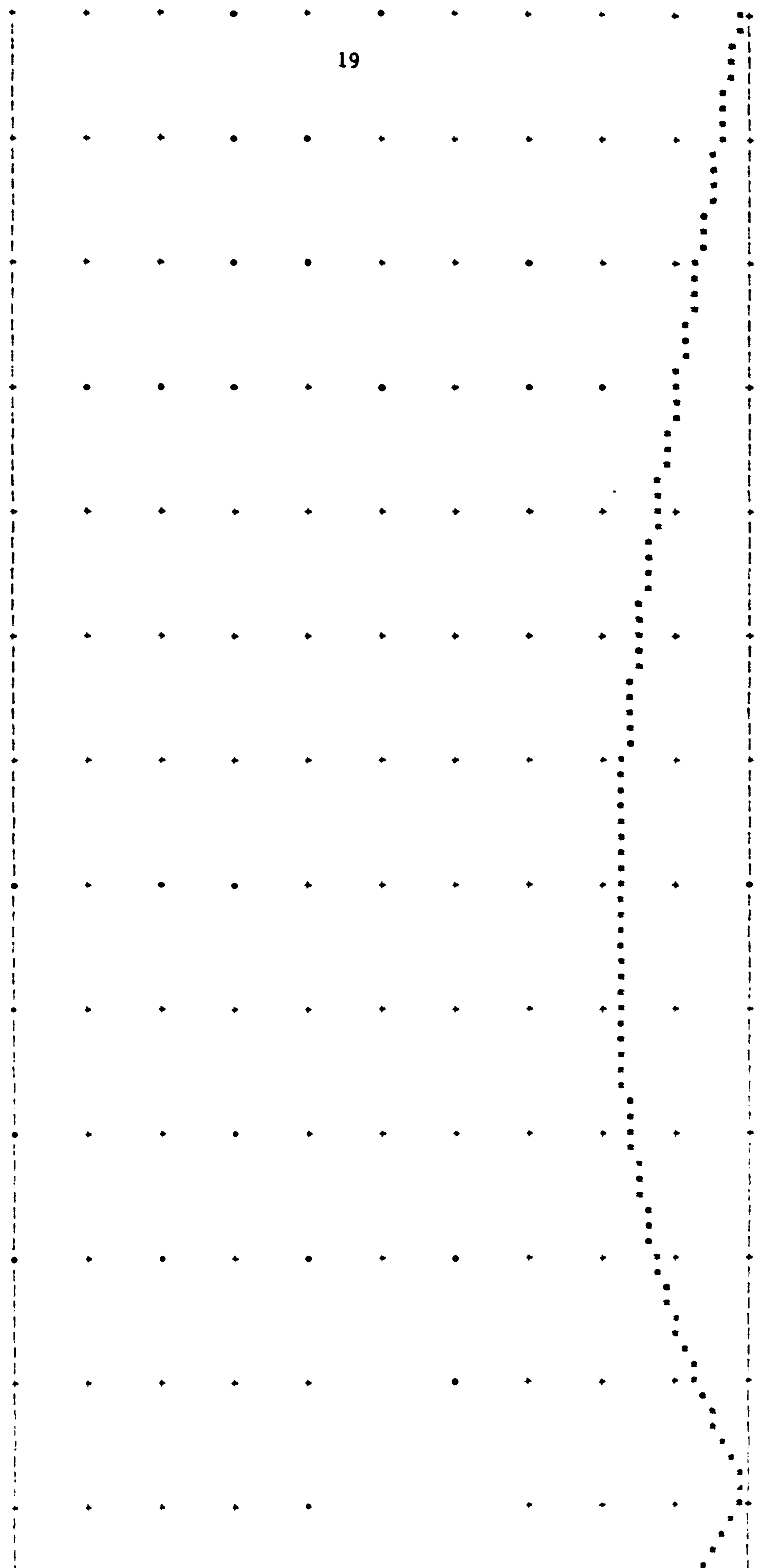
[illegible]	[illegible]	[illegible]
-34.50	1.496	-0.6080
-37.35	1.053	-0.6120
-44.72	0.581	-0.6160
-56.89	0.143	-0.6200
-47.07	0.443	-0.6240
-40.74	0.913	-0.6280
-37.25	1.372	-0.6320
-34.76	1.736	-0.6360
-33.37	2.146	-0.6400
-32.26	2.442	-0.6440
-31.49	2.665	-0.6480
-31.03	2.808	-0.6520
-30.85	2.867	-0.6560
-30.93	2.840	-0.6600
-31.23	2.729	-0.6640
-31.92	2.536	-0.6680
-32.89	2.268	-0.6720
-34.28	1.932	-0.6760
-36.26	1.539	-0.6800
-39.16	1.131	-0.6840
-43.86	0.641	-0.6880
-51.80	0.257	-0.6920
-46.27	0.486	-0.6960
-40.35	0.960	-0.7000
-36.83	1.440	-0.7040
-34.45	1.894	-0.7080
-32.74	2.307	-0.7120
-31.47	2.669	-0.7160
-30.55	2.969	-0.7200
-29.89	3.201	-0.7240
-29.48	3.358	-0.7280
-29.27	3.439	-0.7320
-29.27	3.441	-0.7360
-29.46	3.365	-0.7400
-29.85	3.216	-0.7440
-30.47	2.997	-0.7480
-31.32	2.716	-0.7520
-32.47	2.380	-0.7560
-33.97	2.001	-0.7600
-35.98	1.589	-0.7640
-38.72	1.159	-0.7680
-42.73	0.730	-0.7720
-48.73	0.366	-0.7760
-48.48	0.377	-0.7800
-42.39	0.717	-0.7840
-39.31	1.093	-0.7880
-36.95	1.421	-0.7920
-35.32	1.715	-0.7960
-34.17	1.956	-0.8000
-33.40	2.138	-0.8040
-32.93	2.258	-0.8080
-32.71	2.314	-0.8120
-32.74	2.306	-0.8160
-33.01	2.236	-0.8200
-33.52	2.108	-0.8240
-34.31	1.926	-0.8280
-35.41	1.697	-0.8320
-36.91	1.429	-0.8360
-38.96	1.127	-0.8400
-41.91	0.803	-0.8440
-46.65	0.465	-0.8480
-58.13	0.124	-0.8520
-53.35	0.215	-0.8560
-45.39	0.538	-0.8600
-41.53	0.839	-0.8640
-39.10	1.110	-0.8680
-37.43	1.345	-0.8720
-36.27	1.537	-0.8760
-35.47	1.684	-0.8800
-34.93	1.732	-0.8840
-34.75	1.821	-0.8880
-34.75	1.830	-0.8920
-34.98	1.792	-0.8960
-35.44	1.690	-0.9000
-36.12	1.562	-0.9040
-37.03	1.407	-0.9080
-38.14	1.239	-0.9120
-39.34	1.079	-0.9160
-40.35	0.957	-0.9200
-40.81	0.911	-0.9240
-40.33	0.963	-0.9280
-39.13	1.099	-0.9320
-37.81	1.286	-0.9360
-36.51	1.494	-0.9400
-35.38	1.701	-0.9440
-34.46	1.892	-0.9480
-33.76	2.057	-0.9520
-33.20	2.187	-0.9560
-32.85	2.278	-0.9600
-32.67	2.325	-0.9640
-32.66	2.327	-0.9680
-32.83	2.283	-0.9720
-33.18	2.193	-0.9760
-33.73	2.059	-0.9800
-34.30	1.894	-0.9840
-35.54	1.671	-0.9880
-36.92	1.426	-0.9920
-38.77	1.152	-0.9960
-41.33	0.858	-1.0000
DB.	[illegible]	

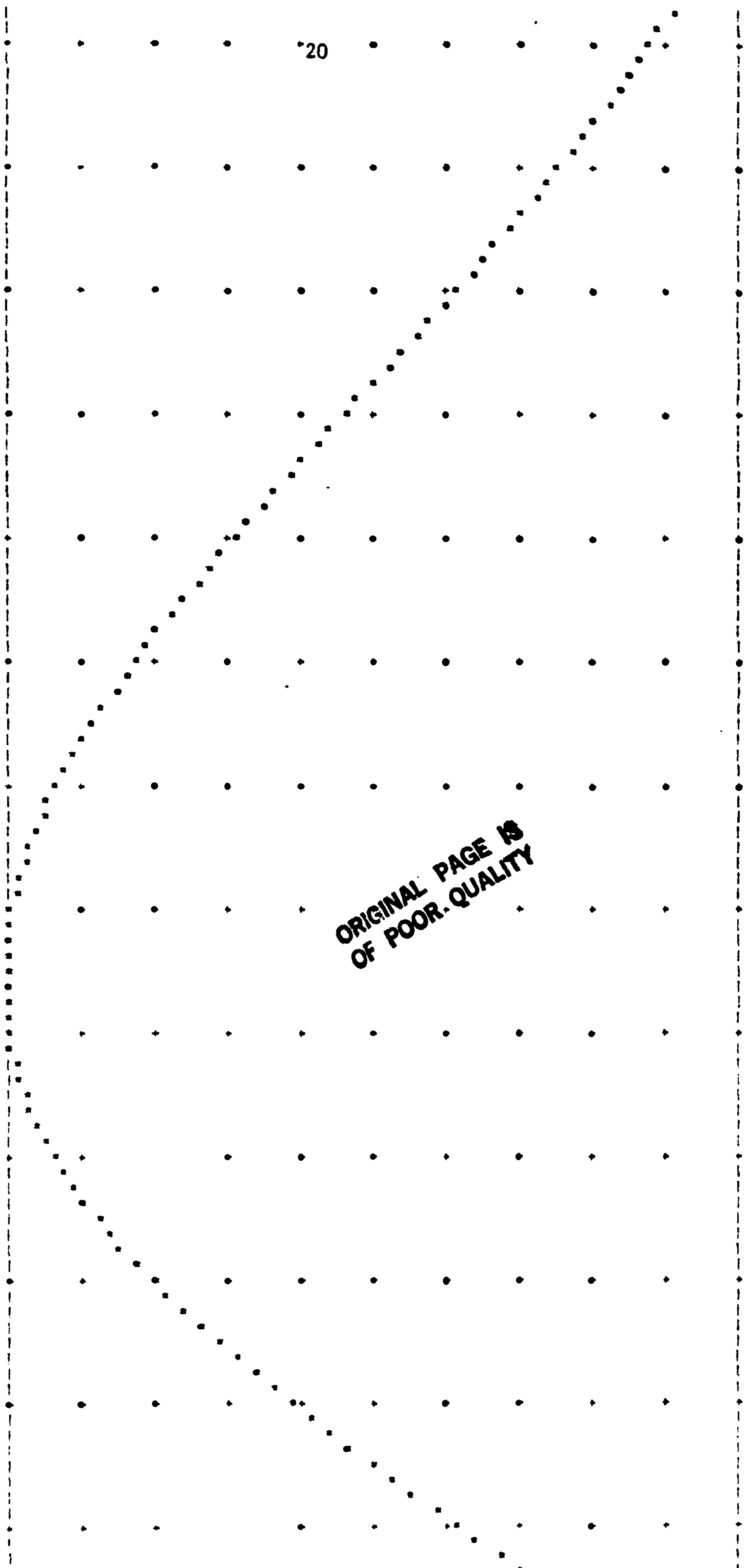
Y-AXIS PROFILE PLOT ALONG 0.0 LONGITUDE.
PATTERN NUMBER 1

Plot scale: 1.0024 0.8022 0.6020 0.4017 0.2015 0.0013

DB.	REAL	
-22.69	0.073	3.9998
-22.61	0.074	3.9918
-22.53	0.075	3.9839
-22.45	0.075	3.9759
-22.38	0.076	3.9679
-22.31	0.077	3.9599
-22.24	0.077	3.9519
-22.17	0.078	3.9439
-22.11	0.079	3.9359
-22.05	0.079	3.9279
-21.99	0.079	3.9199
-21.94	0.080	3.9119
-21.89	0.080	3.9039
-21.84	0.081	3.8959
-21.79	0.081	3.8879
-21.75	0.082	3.8799
-21.71	0.082	3.8719
-21.67	0.082	3.8639
-21.64	0.083	3.8559
-21.61	0.083	3.8479
-21.58	0.083	3.8399
-21.55	0.084	3.8319
-21.53	0.084	3.8239
-21.51	0.084	3.8159
-21.50	0.084	3.8079
-21.48	0.084	3.7999
-21.47	0.084	3.7919
-21.47	0.084	3.7839
-21.47	0.084	3.7759
-21.47	0.084	3.7679
-21.47	0.084	3.7599
-21.48	0.084	3.7519
-21.50	0.084	3.7439
-21.51	0.084	3.7359
-21.53	0.084	3.7279
-21.56	0.084	3.7199
-21.59	0.083	3.7119
-21.62	0.083	3.7039
-21.66	0.083	3.6959
-21.70	0.082	3.6879
-21.75	0.082	3.6799
-21.80	0.081	3.6719
-21.85	0.081	3.6639
-21.91	0.080	3.6559
-21.98	0.080	3.6479
-22.05	0.079	3.6399
-22.13	0.078	3.6319
-22.22	0.077	3.6239
-22.31	0.077	3.6159
-22.40	0.076	3.6079
-22.51	0.075	3.5999
-22.62	0.074	3.5919
-22.74	0.073	3.5839
-22.86	0.072	3.5759
-22.99	0.071	3.5679
-23.14	0.070	3.5599
-23.29	0.068	3.5519
-23.45	0.067	3.5439
-23.62	0.066	3.5359
-23.80	0.065	3.5279
-23.99	0.063	3.5199
-24.19	0.062	3.5119
-24.41	0.060	3.5039
-24.64	0.059	3.4959
-24.89	0.057	3.4879
-25.14	0.055	3.4799
-25.42	0.054	3.4719
-25.72	0.052	3.4639
-26.03	0.050	3.4559
-26.37	0.048	3.4479
-26.73	0.046	3.4399
-27.11	0.044	3.4319
-27.53	0.042	3.4239
-27.99	0.040	3.4159
-28.47	0.038	3.4079
-29.00	0.035	3.3999
-29.58	0.033	3.3919
-30.21	0.031	3.3839
-30.91	0.028	3.3759
-31.69	0.026	3.3679
-32.57	0.024	3.3599
-33.57	0.021	3.3519
-34.72	0.018	3.3439
-36.09	0.016	3.3359
-37.72	0.013	3.3279
-39.79	0.010	3.3199
-42.58	0.007	3.3119
-46.30	0.005	3.3039
-55.56	0.002	3.2959
-57.82	0.001	3.2879
-47.36	0.004	3.2799
-42.73	0.007	3.2719

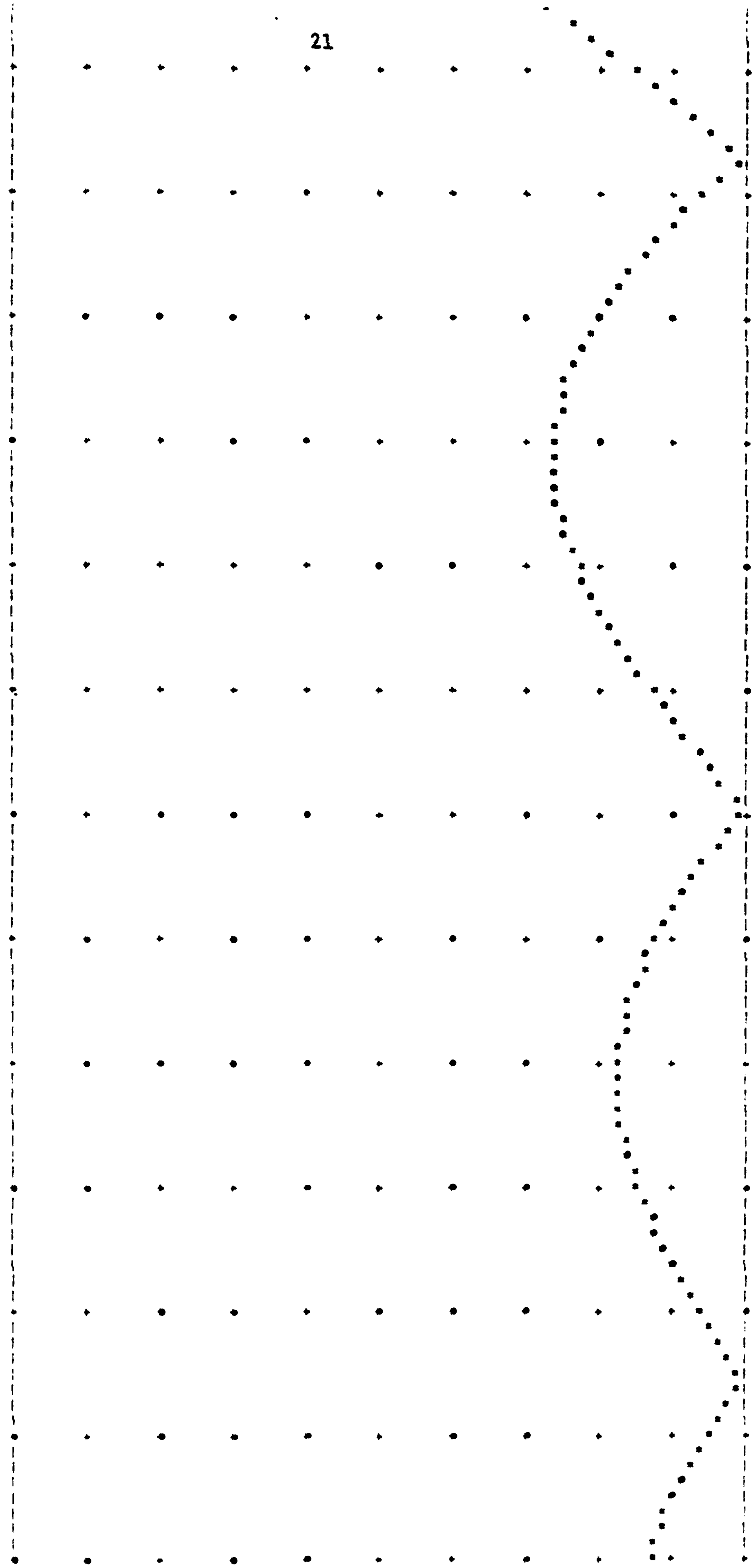
-39.64	0.010	3.2639
-37.36	0.014	3.2559
-35.53	0.017	3.2479
-34.01	0.020	3.2399
-32.70	0.023	3.2319
-31.55	0.026	3.2239
-30.52	0.030	3.2159
-29.59	0.033	3.2079
-28.75	0.037	3.1999
-27.97	0.040	3.1919
-27.25	0.043	3.1839
-26.58	0.047	3.1759
-25.96	0.050	3.1679
-25.37	0.054	3.1599
-24.82	0.057	3.1519
-24.30	0.061	3.1439
-23.81	0.065	3.1359
-23.34	0.068	3.1279
-22.89	0.072	3.1199
-22.47	0.075	3.1119
-22.07	0.079	3.1039
-21.68	0.082	3.0959
-21.31	0.086	3.0879
-20.96	0.090	3.0800
-20.62	0.093	3.0719
-20.29	0.097	3.0640
-19.99	0.100	3.0560
-19.68	0.104	3.0480
-19.39	0.107	3.0400
-19.12	0.111	3.0320
-18.86	0.114	3.0240
-18.60	0.117	3.0160
-18.36	0.121	3.0080
-18.13	0.124	3.0000
-17.90	0.127	2.9920
-17.69	0.131	2.9840
-17.48	0.134	2.9760
-17.29	0.137	2.9680
-17.10	0.140	2.9600
-16.92	0.143	2.9520
-16.75	0.145	2.9440
-16.59	0.148	2.9360
-16.43	0.151	2.9280
-16.29	0.153	2.9200
-16.15	0.156	2.9120
-16.02	0.158	2.9040
-15.90	0.160	2.8960
-15.79	0.162	2.8880
-15.68	0.164	2.8800
-15.58	0.166	2.8720
-15.50	0.168	2.8640
-15.42	0.170	2.8560
-15.34	0.171	2.8480
-15.28	0.172	2.8400
-15.23	0.173	2.8320
-15.18	0.174	2.8240
-15.14	0.175	2.8160
-15.12	0.175	2.8080
-15.10	0.175	2.8000
-15.09	0.176	2.7920
-15.09	0.176	2.7840
-15.11	0.176	2.7760
-15.13	0.175	2.7680
-15.16	0.175	2.7600
-15.21	0.174	2.7520
-15.27	0.172	2.7440
-15.34	0.171	2.7360
-15.42	0.169	2.7280
-15.52	0.167	2.7200
-15.64	0.165	2.7120
-15.76	0.163	2.7040
-15.91	0.160	2.6960
-16.08	0.157	2.6880
-16.26	0.154	2.6800
-16.47	0.150	2.6720
-16.69	0.146	2.6640
-16.95	0.142	2.6560
-17.23	0.138	2.6480
-17.54	0.133	2.6400
-17.88	0.128	2.6320
-18.26	0.122	2.6240
-18.69	0.116	2.6160
-19.16	0.110	2.6080
-19.69	0.104	2.6000
-20.28	0.097	2.5920
-20.95	0.090	2.5840
-21.71	0.082	2.5760
-22.59	0.074	2.5680
-23.61	0.066	2.5600
-24.83	0.057	2.5520
-26.31	0.048	2.5440
-28.13	0.039	2.5360
-30.67	0.029	2.5280
-34.34	0.019	2.5200
-41.19	0.009	2.5120
-53.53	0.002	2.5040
-37.51	0.013	2.4960
-32.08	0.025	2.4880
-29.53	0.037	2.4800
-26.17	0.049	2.4720
-24.18	0.062	2.4640
[illegible]	[illegible]	[illegible]

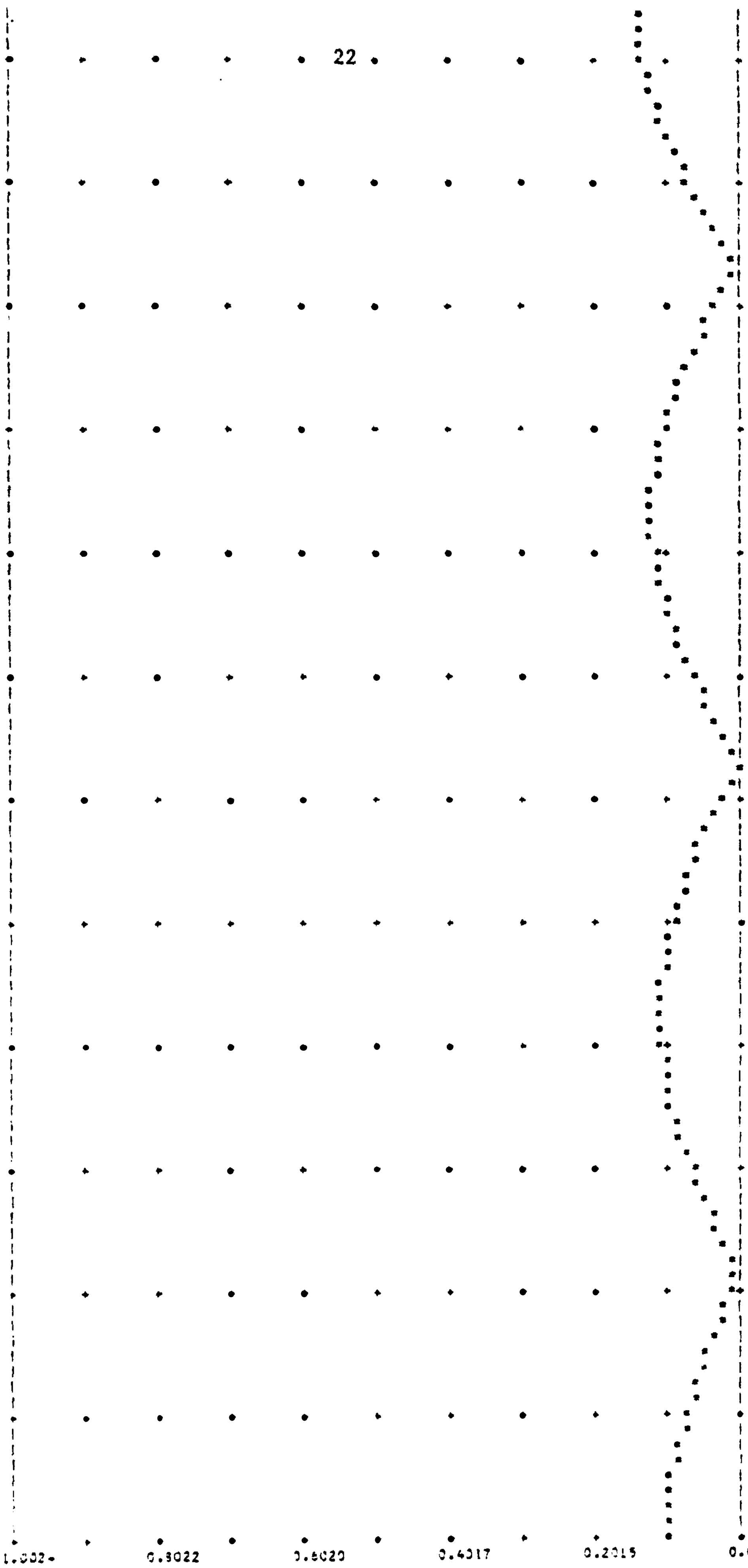




-21.39	0.088	2.4480
-19.33	0.102	2.4400
-18.71	0.116	2.4320
-17.69	0.130	2.4240
-16.75	0.145	2.4160
-15.90	0.160	2.4080
-15.10	0.175	2.4000
-14.36	0.191	2.3920
-13.66	0.207	2.3840
-13.00	0.224	2.3760
-12.38	0.240	2.3680
-11.79	0.257	2.3600
-11.23	0.274	2.3520
-10.70	0.292	2.3440
-10.19	0.309	2.3360
-9.70	0.327	2.3280
-9.23	0.345	2.3200
-8.79	0.364	2.3120
-8.36	0.382	2.3040
-7.94	0.401	2.2960
-7.55	0.419	2.2880
-7.16	0.438	2.2800
-6.80	0.457	2.2720
-6.44	0.476	2.2640
-6.10	0.495	2.2560
-5.77	0.515	2.2480
-5.45	0.534	2.2400
-5.15	0.553	2.2320
-4.85	0.572	2.2240
-4.57	0.591	2.2160
-4.29	0.610	2.2080
-4.03	0.629	2.2000
-3.77	0.648	2.1920
-3.52	0.666	2.1840
-3.29	0.685	2.1760
-3.06	0.703	2.1680
-2.84	0.721	2.1600
-2.63	0.739	2.1520
-2.43	0.756	2.1440
-2.23	0.773	2.1360
-2.05	0.790	2.1280
-1.87	0.806	2.1200
-1.70	0.822	2.1120
-1.54	0.837	2.1040
-1.39	0.852	2.0960
-1.24	0.867	2.0880
-1.11	0.880	2.0800
-0.98	0.894	2.0720
-0.85	0.906	2.0640
-0.74	0.919	2.0560
-0.63	0.930	2.0480
-0.53	0.940	2.0400
-0.44	0.950	2.0320
-0.36	0.959	2.0240
-0.29	0.968	2.0160
-0.22	0.975	2.0080
-0.16	0.982	2.0000
-0.11	0.987	1.9920
-0.07	0.992	1.9840
-0.03	0.996	1.9760
-0.01	0.999	1.9680
0.01	1.001	1.9600
0.02	1.002	1.9520
0.02	1.002	1.9440
0.01	1.001	1.9360
-0.00	0.999	1.9280
-0.03	0.996	1.9200
-0.07	0.992	1.9120
-0.11	0.987	1.9040
-0.17	0.981	1.8960
-0.23	0.973	1.8880
-0.31	0.965	1.8800
-0.40	0.955	1.8720
-0.49	0.945	1.8640
-0.60	0.933	1.8560
-0.72	0.920	1.8480
-0.85	0.907	1.8400
-1.00	0.892	1.8320
-1.15	0.876	1.8240
-1.32	0.859	1.8160
-1.51	0.841	1.8080
-1.70	0.822	1.8000
-1.92	0.802	1.7920
-2.15	0.781	1.7840
-2.39	0.759	1.7760
-2.65	0.737	1.7680
-2.93	0.713	1.7600
-3.23	0.689	1.7520
-3.55	0.664	1.7440
-3.90	0.638	1.7360
-4.26	0.612	1.7280
-4.65	0.585	1.7200
-5.07	0.558	1.7120
-5.52	0.530	1.7040
-6.00	0.501	1.6960
-6.51	0.473	1.6880
-7.06	0.444	1.6800
-7.65	0.414	1.6720
-8.30	0.385	1.6640
-8.99	0.355	1.6560
-9.75	0.325	1.6480
[illegible]	[illegible]	[illegible]

[illegible]	[illegible]	[illegible]
-12.52	0.237	1.5240
-13.66	0.203	1.6160
-14.96	0.179	1.6080
-16.47	0.150	1.6000
-18.27	0.122	1.5920
-20.49	0.095	1.5840
-23.40	0.068	1.5760
-27.67	0.041	1.5680
-36.02	0.016	1.5600
-40.99	0.009	1.5520
-29.63	0.033	1.5440
-25.08	0.056	1.5360
-22.19	0.078	1.5280
-20.12	0.099	1.5200
-19.53	0.113	1.5120
-17.25	0.137	1.5040
-16.22	0.154	1.4960
-15.36	0.171	1.4880
-14.63	0.186	1.4800
-14.02	0.199	1.4720
-13.50	0.211	1.4640
-13.07	0.222	1.4560
-12.71	0.232	1.4480
-12.41	0.240	1.4400
-12.19	0.246	1.4320
-12.00	0.251	1.4240
-11.37	0.255	1.4160
-11.80	0.257	1.4080
-11.77	0.258	1.4000
-11.79	0.257	1.3920
-11.86	0.255	1.3840
-11.97	0.252	1.3760
-12.13	0.247	1.3680
-12.34	0.242	1.3600
-12.60	0.234	1.3520
-12.92	0.226	1.3440
-13.29	0.217	1.3360
-13.72	0.206	1.3280
-14.21	0.195	1.3200
-14.79	0.182	1.3120
-15.44	0.169	1.3040
-16.19	0.155	1.2960
-17.05	0.140	1.2880
-18.04	0.125	1.2800
-19.20	0.110	1.2720
-20.59	0.094	1.2640
-22.25	0.077	1.2560
-24.35	0.061	1.2480
-27.14	0.044	1.2400
-31.23	0.027	1.2320
-39.37	0.011	1.2240
-45.07	0.006	1.2160
-33.31	0.022	1.2080
-28.58	0.037	1.2000
-25.62	0.052	1.1920
-23.49	0.067	1.1840
-21.35	0.081	1.1760
-20.54	0.094	1.1680
-19.48	0.106	1.1600
-18.59	0.118	1.1520
-17.86	0.128	1.1440
-17.24	0.137	1.1360
-16.73	0.146	1.1280
-16.32	0.153	1.1200
-15.98	0.159	1.1120
-15.73	0.164	1.1040
-15.54	0.167	1.0960
-15.42	0.169	1.0880
-15.37	0.170	1.0800
-15.38	0.170	1.0720
-15.46	0.169	1.0640
-15.60	0.166	1.0560
-15.81	0.162	1.0480
-16.08	0.157	1.0400
-16.43	0.151	1.0320
-16.86	0.144	1.0240
-17.37	0.135	1.0160
-17.98	0.126	1.0080
-18.70	0.115	1.0000
-19.55	0.105	0.9920
-20.56	0.094	0.9840
-21.76	0.082	0.9760
-23.22	0.069	0.9680
-25.04	0.056	0.9600
-27.41	0.043	0.9520
-30.73	0.029	0.9440
-36.22	0.015	0.9360
-54.62	0.002	0.9280
-35.70	0.012	0.9200
-32.09	0.025	0.9120
-28.46	0.038	0.9040
-25.99	0.050	0.8960
-24.14	0.062	0.8880
-22.49	0.073	0.8800
-21.53	0.084	0.8720
-20.37	0.094	0.8640
-19.79	0.102	0.8560
-19.15	0.110	0.8480
-18.62	0.117	0.8400
-18.21	0.123	0.8320
[illegible]	[illegible]	0.8240





DB.	REAL	
-17.42	0.135	0.7920
-17.50	0.133	0.7840
-17.65	0.131	0.7760
-17.89	0.128	0.7680
-18.21	0.123	0.7600
-18.62	0.117	0.7520
-19.13	0.111	0.7440
-19.75	0.103	0.7360
-20.49	0.094	0.7280
-21.39	0.085	0.7200
-22.46	0.075	0.7120
-23.78	0.065	0.7040
-25.40	0.054	0.6960
-27.48	0.042	0.6880
-30.31	0.031	0.6800
-34.61	0.019	0.6720
-43.62	0.007	0.6640
-45.40	0.005	0.6560
-35.29	0.017	0.6480
-30.82	0.029	0.6400
-27.97	0.040	0.6320
-25.90	0.051	0.6240
-24.31	0.061	0.6160
-23.04	0.070	0.6080
-22.02	0.079	0.6000
-21.18	0.087	0.5920
-20.50	0.094	0.5840
-19.95	0.101	0.5760
-19.51	0.106	0.5680
-19.17	0.110	0.5600
-18.93	0.113	0.5520
-18.78	0.115	0.5440
-18.71	0.116	0.5360
-18.72	0.116	0.5280
-18.82	0.115	0.5200
-19.00	0.112	0.5120
-19.26	0.109	0.5040
-19.62	0.104	0.4960
-20.08	0.099	0.4880
-20.64	0.093	0.4800
-21.33	0.086	0.4720
-22.16	0.078	0.4640
-23.16	0.070	0.4560
-24.37	0.060	0.4480
-25.87	0.051	0.4400
-27.77	0.041	0.4320
-30.30	0.031	0.4240
-33.97	0.020	0.4160
-40.55	0.009	0.4080
-57.98	0.001	0.4000
-38.54	0.012	0.3920
-33.07	0.022	0.3840
-29.81	0.032	0.3760
-27.53	0.042	0.3680
-25.79	0.051	0.3600
-24.43	0.060	0.3520
-23.32	0.068	0.3440
-22.42	0.076	0.3360
-21.69	0.082	0.3230
-21.09	0.088	0.3200
-20.60	0.093	0.3120
-20.22	0.098	0.3040
-19.93	0.101	0.2960
-19.73	0.103	0.2880
-19.61	0.105	0.2900
-19.57	0.105	0.2720
-19.61	0.105	0.2640
-19.72	0.103	0.2560
-19.92	0.101	0.2480
-20.19	0.099	0.2400
-20.55	0.094	0.2320
-21.01	0.089	0.2240
-21.56	0.084	0.2160
-22.23	0.077	0.2080
-23.03	0.071	0.2000
-23.98	0.063	0.1920
-25.13	0.055	0.1840
-26.53	0.047	0.1760
-28.27	0.039	0.1680
-30.53	0.030	0.1600
-33.67	0.021	0.1520
-39.71	0.012	0.1440
-52.23	0.002	0.1360
-43.53	0.007	0.1280
-36.11	0.016	0.1200
-32.24	0.024	0.1120
-29.63	0.033	0.1040
-27.70	0.041	0.0960
-26.19	0.049	0.0880
-24.97	0.056	0.0800
-23.97	0.063	0.0720
-23.14	0.070	0.0640
-22.44	0.075	0.0560
-21.87	0.081	0.0480
-21.39	0.085	0.0400
-21.00	0.089	0.0320
-20.70	0.092	0.0240
-20.46	0.095	0.0160
-20.30	0.097	0.0080
-20.20	0.098	0.0

3.0 THE SHUTTLE IMAGING RADAR ANTENNA SPECIALIZED FAR-FIELD PATTERN PROGRAM

The footprint program discussed in Section 2.0 is quite general and could be used to generate far-field pattern data. However, the computer time consumed by the program is rather large for the far-field results which would be produced. For this reason, a separate program was written to quickly compute far-field pattern profiles from a nominally planar array antenna which has been mechanically warped or with electrical excitation errors. Two programs were written during the course of this study. The first program was written in FORTRAN IV to be run on the IBM 370 computer system at PSL. While execution time for this program was minimal (less than one minute per profile generated), the support time (punching cards, waiting for output, etc.) was excessive. Hence a second program was written in BASIC for the TEKTRONIX 4051 Graphics System. While the BASIC program takes about ten minutes per profile, there is no support time involved.

The value of having such a facility as the TEK 4051 can be illustrated by two examples.

1. In March 1978, there was a disagreement between Ball Brothers and JPL as to the nominal side lobe level of the SIR-A antenna with a natural taper. Using the TEK 4051 and this second program it was possible to resolve the question (in favor of Ball) in less than fifteen minutes.
2. In October/November, 1978, the SIR-A antenna was measured at JPL. The preliminary pattern data taken did not agree with previously calculated results. The question arose, "Is the measurement in error, or does the antenna not work?" PSL ran several simulations independent of Ball Brothers and JPL, and suggested that a pseudo near field probe of the individual antenna panel radiation be made. Using this data PSL (along with JPL) verified that the problems lay in the antenna as the data measured at JPL and the simulations run at PSL agreed to within 0.2dB at the -20 dB level.

The data needed to run either of the two far-field programs is essentially the same as the data needed for the footprint program. The second program has prompted data entry, so a separate section on data I/O need not be included in this report. However, a complete example is given below.

3.1 Example Using the Footprint Program

The sample program used here is the same as is used in Section 2.3. All data entries are prompted by the computer, and user input has been underlined. The next three pages are copies of computer output from the program.

Data entry for the FORTRAN program is essentially the same as for the footprint program, with the exception that orbit information (altitude, yaw, tilt, twist) is not needed for far-field plots. Many examples using both these programs were given in the Phase II Final Report [2, Ch3], at the review meeting held at JSC on January 2-3, 1978, and in the SARTC conference proceedings [3].

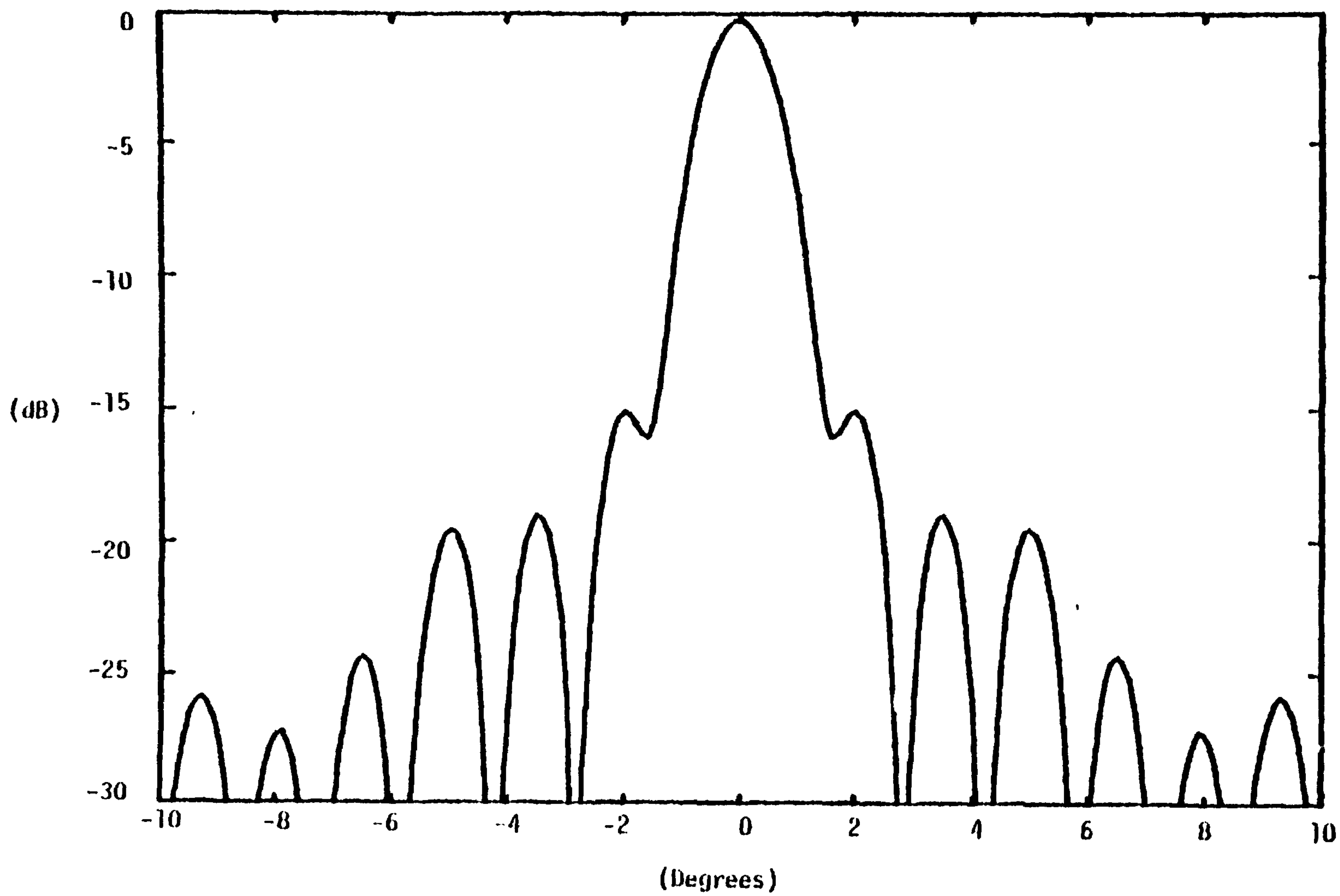
```
***PSL DEFORMED ANTENNA FAR-FIELD PATTERN PROGRAM***

ENTER FREQUENCY IN GHZ: 1.275

ENTER TOTAL NUMBER OF ELEMENTS (NX,NY): 56,16

ENTER INTERELEMENT SPACING (SX,SY) IN CM: 16.75,13.00

ENTER INTERELEMENT PHASE SHIFT (PXD,PYD) IN DEGREES: 0.0,0.0

ENTER NUMBER OF SUBARRAY DIVISIONS (MSECT,NSECT): 7,1

ENTER EXCITATION COEFFICIENTS FOR EACH SUBARRAY
(AMAG,APHS,PHSX,PHSY):
( 1, 1):  0.875,0.0,0.0
( 2, 1):  0.875,0.0,0.0
( 3, 1):  1.1667,0.0,0.0
( 4, 1):  1.1667,0.0,0.0
( 5, 1):  1.1667,0.0,0.0
( 6, 1):  0.875,0.0,0.0
( 7, 1):  0.875,0.0,0.0

ENTER WARP ARRAY:
1.0  0.5  0.166  0.0  0.0  0.166  0.5  1.0

1.0  0.5  0.166  0.0  0.0  0.166  0.5  1.0

ENTER TITLE: EXAMPLE 1: BBRC 7-PANEL ARRAY WITH NATURAL TAPER AND PARABO
LIC BOW

PRELIMINARIES COMPLETED

BEAMWIDTH IS NOMINALLY  2.875 BY 12.964 DEGREES

ENTER PHI = CONSTANT(DEGREES): 0
```

SAMPLE OF

DATA OUTPUT LISTING

ANGLE (degrees)	REL. GAIN (dB)	
-3.300	-19.726	49.82
-3.200	-20.841	51.80
-3.100	-22.686	54.94
-3.000	-25.692	60.90
-2.900	-30.973	77.44
-2.800	-35.932	152.85
-2.700	-28.442	156.84
-2.600	-23.551	144.97
-2.500	-20.473	139.36
-2.400	-18.376	135.44
-2.300	-16.920	131.97
-2.200	-15.945	128.38
-2.100	-15.367	124.27
-2.000	-15.137	119.19
-1.900	-15.213	112.55
-1.800	-15.528	103.52
-1.700	-15.922	91.07
-1.600	-16.064	74.62
-1.500	-15.525	55.63
-1.400	-14.177	37.71
-1.300	-12.350	23.55
-1.200	-10.432	13.30
-1.100	-8.633	6.01
-1.000	-7.025	0.77
-0.900	-5.621	3.08
-0.800	-4.409	5.96
-0.700	-3.377	8.15
-0.600	-2.510	9.83
-0.500	-1.795	11.10
-0.400	-1.222	12.07
-0.300	-0.785	12.77
-0.200	-0.476	13.25
-0.100	-0.292	13.53
0.000	-0.231	13.62
0.100	-0.292	13.53

EXAMPLE 1: BBRC 7-PANEL ARRAY WITH NATURAL TAPER AND PARABOLIC BOW



4.0 RAY TRACING FOR THE SIR-A ANTENNA AND PAYLOAD BAY INSIDE AN ENCLOSED BUILDING

In April 1978, it was suggested that an end-to-end system test of the SIR-A radar and antenna be conducted prior to launch to assess and, if necessary correct, system problems arising from electromagnetic interference (EMI) of the radar with other shuttle experiments. Of necessity, this test would have had to be conducted inside a closed building (perhaps the O & C building at Kennedy Space Center). PSL conducted a preliminary study, using approximate methods, to assess the feasibility of such a test. The study results were communicated to NASA/JSC by technical memo on 28 April 1978.

To determine whether additional high level radiation would be reflected by the enclosing structure back into the critical payload area, a ray tracing program was written for the Tektronix 4051 interactive graphics system. The following assumptions were made:

- The SIR-A antenna pattern would not be perturbed by the surrounding environment.
- The energy emitted from the antenna traveled in ray-like paths (geometrical optics approximation) whose directions did not change unless incident on a surface (Snell's Law).
- Diffraction and other higher-order effects were ignored.
- Reflection coefficients were assumed unity (no energy absorbed by the structures).
- Power density in the ray direction varied as the inverse square of the distance from the antenna (Far-field approximation).
- Only two dimensions, corresponding to the pallet cross section were simulated. Since the SIR-A beam is narrow in the third dimension (depth or x-axis), this approximation is fairly good.
- No external objects (GSE, overhead crane, personnel, etc.) were included in the simulations.

It is not possible to provide accurate figures for reflected radiation from an analysis as approximate as this one is; however, it is possible to state that significant amounts of energy will be coupled from the SIR-A antenna to other shuttle experiments via reflections from surrounding objects. In other words, the test results would be meaningless because a small amount of reflected energy would completely mask out the EMI effects of radiation coming directly from the antenna.

4.1 Use of the Ray Tracing Program

The program is written for the TEK 4051; hence, data entry is prompted by the computer. Data may be entered in one (or both) of two ways:

1. By specifying the number of tape file on which data has been previously stored

2. By entering data directly from the keyboard

After all data has been entered, it may be stored on a tape file by pressing User Definable Key #4. Object shapes are entered by specifying the (x,y)-coordinates of the endpoints of the line segments making up the object. (Several objects may be entered at once by concatenating their respective tape files.) In addition, the line segment number representing the antenna and the (x,y)-coordinates of the ray source must be entered. For the SIR-A antenna mounted in the pallet, this information is given in TABLE I. This data has been stored on tape file #2.

4.2 An Example using the Ray Tracing Program

Figures 2 and 3 illustrate the problem to be solved. The SIR-A antenna plus pallet are placed inside a 50' x 100' rectangle. Data are entered as described in Section 4.1, and the ray tracing begins. To stop the trace, user key #1 is pressed and the following results are printed:

R - total length of the ray (in the same units as data input)
ANGLE - "take off angle of ray (degrees) from horizontal
XO,YO - (x,y) - coordinates where the ray terminates.

To restart the program changing only the take-off angle, press user key #3.

The ray trace program is not limited to this type of problem only. For example, it has been used to study the effects of ground surface profiles on reflected signal levels for antenna range work.

TABLE 1. Boundary surface computer description for EMI hazard study.

	Describing Coordinates		
Surface	(x_o, y_o)	(x_1, y_1)	Description
1	00.000,00.000	100.000,000000	Floor
2	00.000,50.000	100.000,50.000	Ceiling
3	00.000,00.000	00.000,50.000	Left Wall
4	100.000,00.000	100.000,50.000	Right Wall
5	46.162,04.781	53.838,4.781	Pallet
6	53.838,04.781	56.506,07.747	Pallet
7	56.506,07.747	56.974,11.650	Pallet
8	56.974,11.650	57.156,12.041	Pallet
9	57.156,12.041	57.937,12.041	Pallet
10	57.937,12.041	57.937,12.301	Pallet
11	57.937,12.301	56.375,12.236	Pallet
12	56.375,12.236	54.944,08.658	Pallet
13	54.944,08.658	52.862.6.446	Pallet
14	52.862,06.446	47.138,6.446	Pallet
15	47.138,06.446	45.056,08.558	Pallet
16	45.056,08.658	43.624,12.236	Pallet
17	43.624,12.236	42.063,12.301	Pallet
18	42.063,12.301	42.063,12.041	Pallet
19	42.063,12.041	42.844,12.041	Pallet
20	42.844,12.041	43.026,11.650	Pallet
21	43.026,11.650	43.494,07.747	Pallet
22	43.494,07.747	46.162,04.781	Pallet
23	56.636,12.691	51.720,17.276	SIR-A Antenna

NOTE: Additional surfaces may be added in order to provide more detail of the O & C Building.

Antenna source location: 54.178,14.9835.

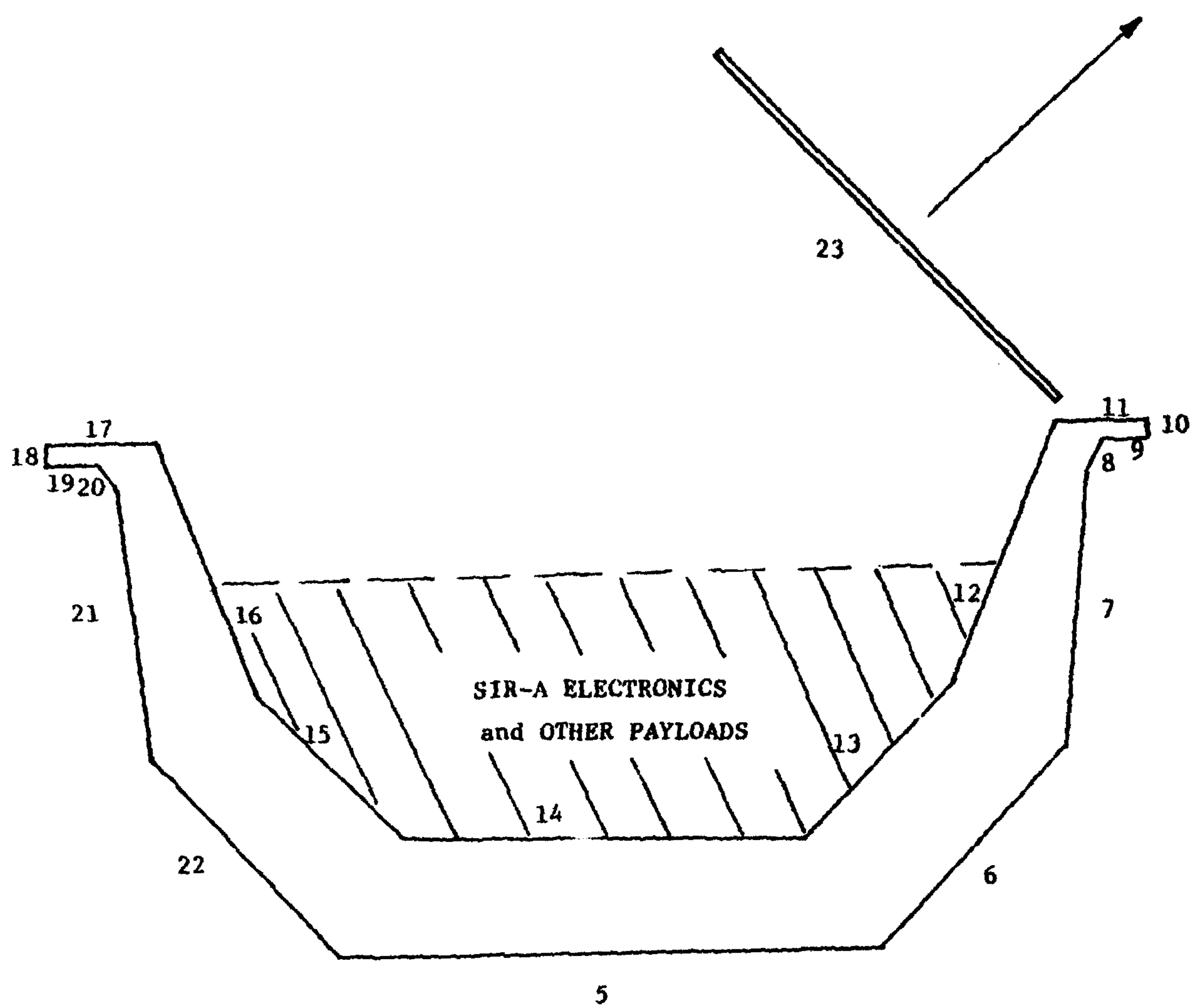


Figure 2. Reflecting surfaces of the pallet and SIR-A antenna. Surfaces 1 through 4 describe the *four* surrounding walls of the O&C Building.

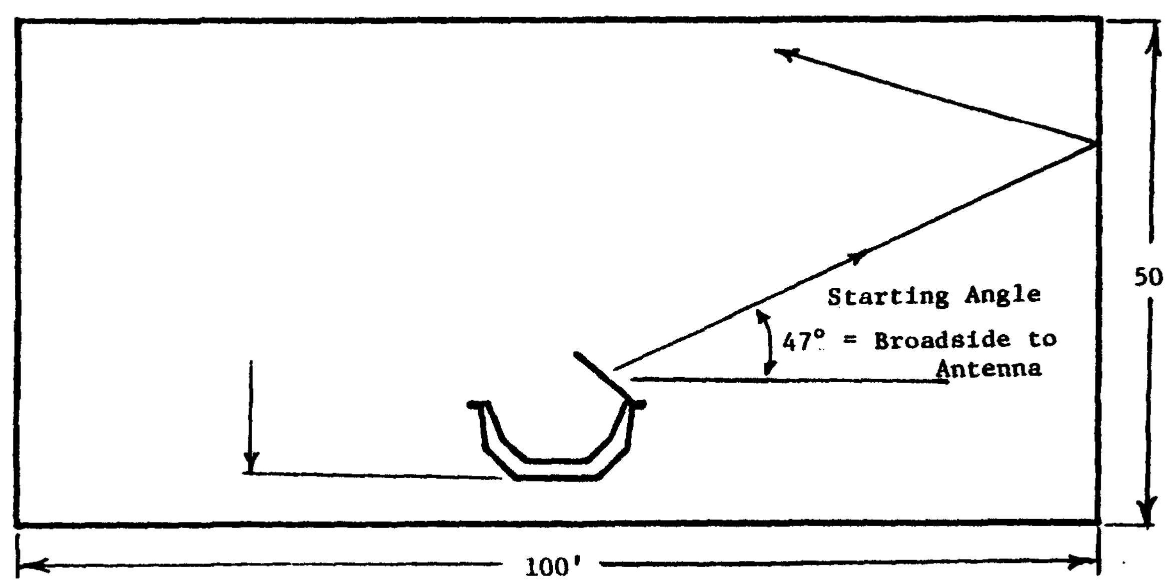
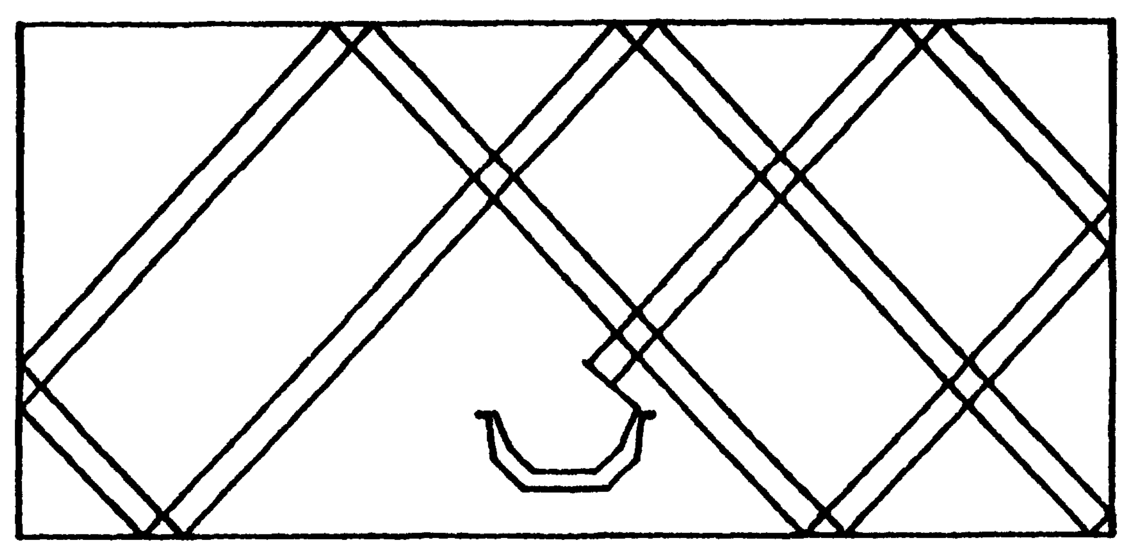
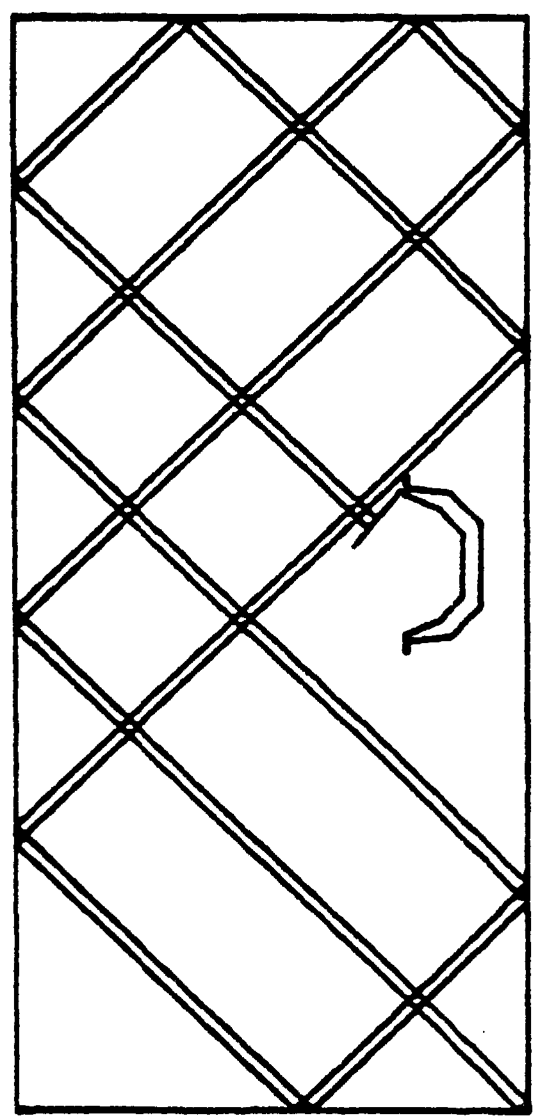


Figure 3. SIR-A Antenna and pallet inside a rectangular building.

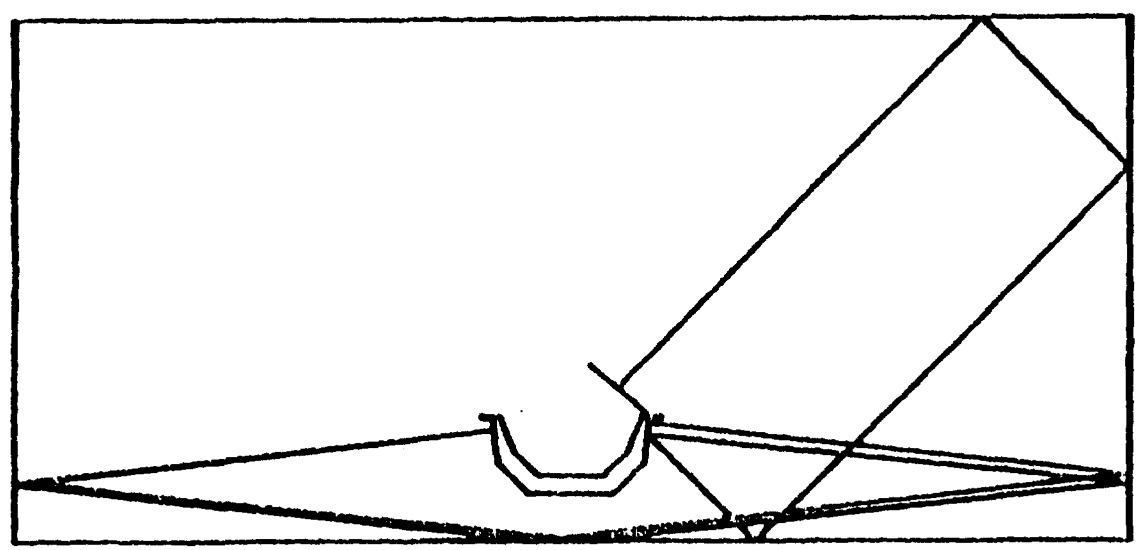
R = 752.654403475 ANGLE = 49 X0,Y0: 52.0362828909,16.9810111007



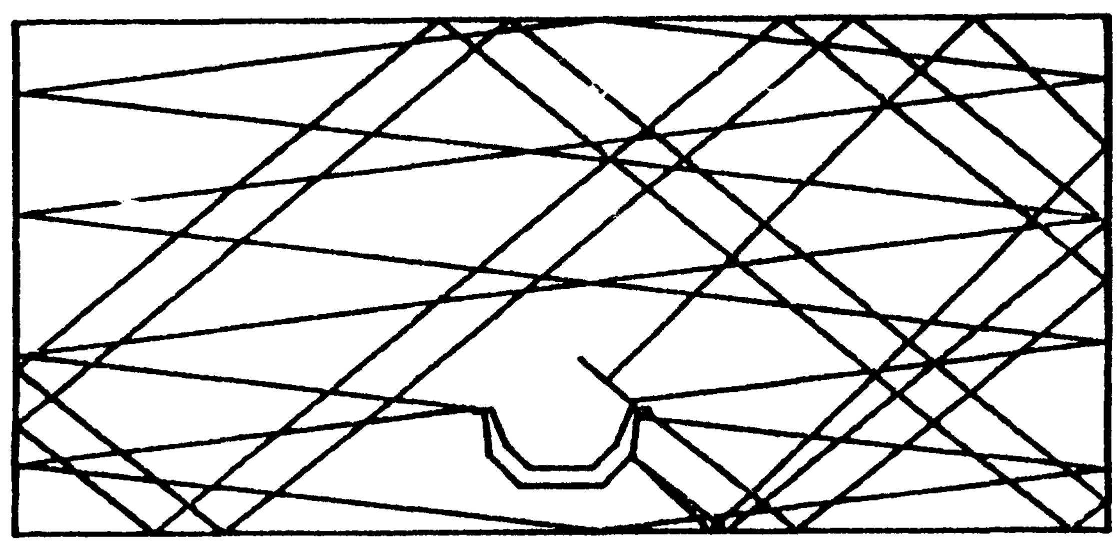


R = 1035.02409538 ANGLE = 40 X0,Y0: 53.2556994643,15.8436992762

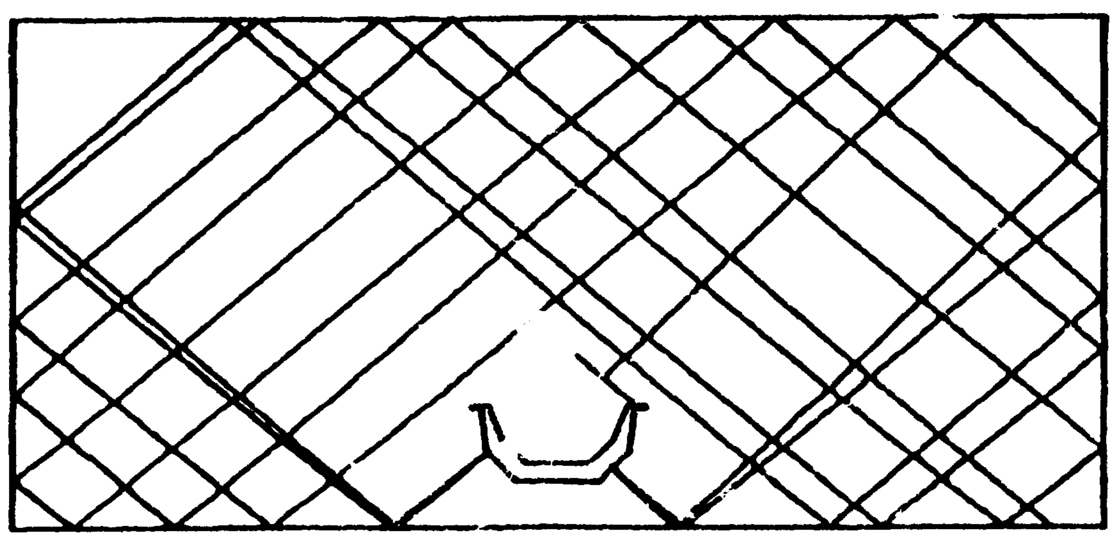
R = 602.506813876 ANGLE = 47 X0,Y0: 46.7223791347,0



R = 1548.79918851 ANGLE = 46 X0,Y0: 56.4102602589,12.9015387617



R = 1367.63482061 ANGLE = 45 X0,Y0: 95.3936475681,0



5.0 SUMMARY

This report has outlined the work performed during Phase III of the Antenna Evaluation Study for the Shuttle Imaging Radar. Greatest emphasis has been given in the report to areas which have not been reported elsewhere, namely, the documentation of all computer software written during all three phases of this effort. All three programs which are described in this volume have applicability to other projects within NASA and elsewhere; hence the user-oriented format of the report. Technical information regarding algorithms used, mathematics, programming, etc., may be found in the Final Reports for Phases I and II, or in the literature cited at the end of this report.

If the SIR-A mission is successful, future synthetic aperture radars will undoubtedly be launched into space either as free-flyers or as Shuttle experiments. Potential users of SAR data will be requiring multiple frequencies, multiple swath widths and incidence angles, and dual polarization. For example, it has been shown that soil moisture determination is best carried out at about 4.5 GHz and for incidence angles in the 7°-12° range, whereas crop classification is optimally accomplished at about 14 GHz and with a 40°-50° incident angle range. When such an experiment is finally implemented, the software that has been developed under the present effort should be very helpful in systems planning and design.

LIST OF REFERENCES

1. E. L. Coffey and K. R. Carver, Final Report on Antenna Evaluation Study for the Shuttle Multispectral Radar: Phase I (NASA Contract NAS9-95469), Physical Science Laboratory Report No. PA00874, New Mexico State University, Las Cruces, New Mexico, December 1976.

2. E. L. Coffey and K. R. Carver, Final Report on Antenna Evaluation Study for the Shuttle Imaging Radar: Phase II (NASA Contract NAS9-95471), Physical Science Laboratory Report No. PA00881, New Mexico State University, Las Cruces, New Mexico, May 1977.

3. R. W. Campbell, K. E. Melick, and E. L. Coffey, "Mathenatical Modeling and Simulation of the Space Shuttle Imaging Radar Antennas," Proceedings of the Synthetic Aperture Radar Technology Conference, New Mexico State University, Las Cruces, New Mexico, March 8, 9, 10, 1978.

4. R. W. Campbell and E. L. Coffey, "Computer Simulation of the Space Shuttle Imaging Radar Antennas," Proceedings of the 1978 Int'l. Antennas and Propagation Society Conference, Washington, D.C., pp. 30-33.

5. E. L. Coffey, "Practical Mathematical Modeling of Large Non-Planar Array Antennas," Proceedings of the 1978 Int'l. Antennas and Propagation Society Conference, Washington, D.C., pp. 136-139.

6. K. R. Carver and A. C. Newell, "SAR Antenna Calibration Techniques," Proceedings of the Synthetic Aperture Radar Technology Conference, New Mexico State University, Las Cruces, New Mexico, March 8, 9, 10, 1978.

7. K. R. Carver, Antenna Pattern Measurements of the Four-Panel EMU Array, (NASA Contract NAS9-95482,) Physical Science Laboratory, Preliminary Report, New Mexico State University, Las Cruces, New Mexico, September, 1978.

APPENDIX A

The Shuttle Imaging Radar Antenna
General Footprint Program

APPENDIX B

The Shuttle Imaging Radar Antenna
Specialized Far-Field Program

APPENDIX C

Ray Tracing Program

APPENDIX A

THE SHUTTLE IMAGING RADAR ANTENNA
GENERAL FOOTPRINT PROGRAM

```
            C*****************************************************************************
            C
            C         MAIN PROGRAM FOR SPACE SHUTTLE EARTH RESOURCES IMAGING RADAR ANTENNA
            C
            C         WRITTEN BY: E. L. COFFEY, SENIOR ENGINEER
            C                     PHYSICAL SCIENCE LABORATORY
            C                     ELECTROMAGNETICS SECTION
            C                     NEW MEXICO STATE UNIVERSITY
            C                     LAS CRUCES, NEW MEXICO  88001
            C                     PHONE: (505) 524 - 2851
            C
            C               DATE: 2 AUGUST 1976
            C            REVISED: 23 AUGUST 1976
            C
            C            VERSION 2.3
            C
            C*****************************************************************************
            C
0001                REAL APLOT(151,151),CON(10),K,MU,XPRINT(501,2),YPRINT(501,2),
                   $XPLOT(501),YPLOT(501),ZAVG(24,12),ALPHAX(24,12),ALPHAY(24,12),
                   $ANGLE(3,3)
0002                REAL KX,KY
0003                REAL PHSX(24,12),AMAG(24,12),APHS(24,12),PHSY(24,12)
            C
0004                COMPLEX AF,AF1,PAT,CEXP,J
            C
0005                INTEGER NAR(80),DELIM,PR1,PR2,PR3,PL1,PL2,PL3,TP1,TP2
0006                INTEGER DATE(5)
0007                EQUIVALENCE (XPRINT(1,1),YPRINT(1,1)),(XPLOT(1),YPLOT(1))
            C
0008                COMMON /ANG/ YAW,TILT,TWIST,CTW,STW,CT,ST,CYAW,SYAW
0009                COMMON /IO/ PR1,PR2,PL1,PL2,PL3,TP1,TP2,NCON,CON,STARTX,STOPX,
                   $DELTAX,STARTY,STOPY,DELTAY,CONLOW,CONMAX,NPTSX,NPTSY,FLOOR,DASH
0010                COMMON /ARRAY/ APLOT
0011                COMMON /SYS/ REARTH,RE3,ALT,ALT3,ANGLE
0012                COMMON /ANT/ MSECT,NSECT,MEL,NEL,SX,SY,PX,PY,IHV,K,OMU4PI,ZAVG,
                   $ALPHAX,ALPHAY,KX,KY,PXSECT,PYSECT
0013                COMMON /ELCTRC/ PHSX,PHSY,AMAG,APHS
```

```
            C
0014              DATA J/(0.,1.)/
0015              DATA PI,MU,RADDEG/3.14159265E0,12.56637E-7,0.01745329F0/
0016              DATA DELIM /'&'/
0017              FLOOR=-30.0
            C
            C........INPUT USER PARAMETERS
0018         1000 READ(1,1,END = 9999) NUMPAT,DATE
0019            1 FORMAT(I5,1X,5A4)
0020              WRITE(3,2) NUMPAT,DATE
0021            2 FORMAT(1H1,12X,'PATTERN ',I3,15X,'SPACE SHUTTLE IMAGING RADAR ANTE
                 $NNA SIMULATION PROGRAM',14X,5A4////)
0022            9 IYES=0
0023              READ(1,3) NAR
0024            3 FORMAT(80A1)
            C........CHECK FOR DELIMITER
0025              DO 10 I=1,80
0026              IF(NAR(I).EQ.DELIM) GO TO 11
0027           10 CONTINUE
0028              IYES=1
0029              I=81
0030           11 I=I-1
0031              WRITE(3,4) (NAR(JJ),JJ=1,I)
0032            4 FORMAT(20X,80A1)
0033              IF(IYES.EQ.1) GO TO 9
0034              WRITE(3,5)
0035            5 FORMAT(////)
            C........END OF NARRATIVE
0036            6 FORMAT(8F10.0)
0037            7 FORMAT(2I5)
0038            8 FORMAT(5I1)
            C
            C         CALL SUBROUTINES FOR FURTHER OUTPUT CALCULATIONS
            C
0039              CALL ORBIT(NUMPAT,FREQ)
0040              CALL ANTENA(NUMPAT)
0041              CALL MECH(NUMPAT)
0042              CALL ELEC(MSECT,NSECT,NUMPAT)
```

```
0043              CALL OUTPUT(NUMPAT,IDIREC)
           C
           C........CALCULATE RUN PARAMETERS
           C
0044              K=2.*PI*FREQ/30.0
0045              OMEGA=2.*PI*FREQ*1E9
0046              OMU4PI=OMEGA*MU/(4.*PI)
0047              DXD=DELTAX/RADDEG
0048              DYD=DELTAY/RADDEG
           C
0049              WRITE(3,200) NUMPAT
0050          200 FORMAT(20X,'SUBARRAY DATA SUMMARY FOR PATTERN ',I3,':'/
                 $1X,'AREA',8X,'XCENT',9X,'YCENT',9X,'ZAVG',8X,'ALPHAX',7X,
                 $'ALPHAY',8X,'AMAG',9X,'APHS',9X,'PHSX',9X,'PHSY')
0051              IAREA=0
0052              DO 201 N=1,NSECT
0053              YCENT=SY*NEL*(N-FLOAT(NSECT+1)*0.5)
0054              DO 201 M=1,MSECT
0055              XCENT=SX*MEL*(M-FLOAT(MSECT+1)*0.5)
0056              IAREA=IAREA+1
0057              P1=ZAVG(M,N)
0058              P2=ALPHAX(M,N)*57.295
0059              P3=ALPHAY(M,N)*57.295
0060              P4=AMAG(M,N)
0061              P5=APHS(M,N)*57.295
0062              P6=PHSX(M,N)*57.295
0063              P7=PHSY(M,N)*57.295
0064              WRITE(3,202) IAREA,XCENT,YCENT,P1,P2,P3,P4,P5,P6,P7
0065          202 FORMAT(2X,I3,6X,F8.2,6X,F8.2,6X,F7.2,6X,F7.3,6X,F7.3,6X,F7.4,
                 $6X,F7.2,6X,F7.2,6X,F7.2)
0066          201 CONTINUE
           C
           C........CALCULATE BEAM CENTER
           C
0067              KX=K*SX
0068              KY=K*SY
0069              CALL GENER
0070              UP=-PX/KX
```

```
DOS FORTRAN IV 360N-FO-479 3-9          MAINPGM            DATE    12/21/78      TIME     13.37.46

0071                VP=-PY/KY
0072                IF(UP.EQ.0. .AND. VP.EQ.0.) GO TO 20
0073                PHIP=ATAN2(VP,UP)
0074                GO TO 21
0075             20 PHIP=0.
0076             21 CONTINUF
0077                WP=SQRT(1.0-(UP*UP+VP*VP))
0078                THETAP=ARCOS(WP)
            C
            C........UNTWIST
            C
0079                PHIP=PHIP+TWIST
0080                XP=SIN(THETAP)*COS(PHIP)
0081                YP=SIN(THETAP)*SIN(PHIP)
0082                ZP=COS(THETAP)
            C
            C........"UNTILT"
            C
0083                X=XP
0084                Y=CT*YP+ST*ZP
0085                Z=-ST*YP+CT*ZP
            C
            C........"UNYAW"
            C
0086                XP=CYAW*X-SYAW*Y
0087                YP=SYAW*X+CYAW*Y
0088                ZP=Z
            C
            C........CALCULATE ANGLES
            C
0089                RP=SQRT(XP*XP+YP*YP+ZP*ZP)
0090                THETAP=ARCOS(ZP/RP)
0091                IF(XP.EQ.0. .AND. YP.EQ.0.) GO TO 22
0092                PHIP=ATAN2(YP,XP)
0093                GO TO 23
0094             22 PHIP=0.0
0095             23 CONTINUE
            C
```

```
C........PREDICT LATITUDE AND LONGITUDE ON EARTH ASSUMING SUBSATELLITE
C         POINT IS AT (0,0).
C
      ALPHAR=ARSIN(SIN(THETAP)/REARTH*(REARTH+ALT))-THETAP
      PHIR=PHIP-PI/2.0
      RANGE=REARTH*ALPHAR
C
      FLONG=ATAN(SIN(PHIR)*TAN(ALPHAR))
      FLAT=ARCOS(COS(ALPHAR)/COS(FLONG))
      IF(FLAT.GT.PI/2.) FLAT=FLAT-PI
      FLATD=FLAT/RADDEG
      FLONGD=FLONG/RADDEG
      PHIRD=PHIR/RADDEG
      WRITE(3,102) FLATD,FLONGD,RANGE,PHIRD
  102 FORMAT(///20X,'PREDICTED BEAM CENTER:'/25X,'LATITUDE = ',F8.4/25X,
     $'LONGITUDE = ',F9.4/25X,'RANGE = ',F7.3/25X,'HEADING = ',F8.3///)
C
C........NOW BEGIN POINT-BY-POINT CALCULATION OF PATTERN.
C         FIRST CALCULATE AF AT BEAM CENTER.
C
      AF=PAT(FLONG,FLAT)
      BIG1=CABS(AF)
      IF(BIG1.LT.1E-10) BIG1=1E-10
      BIG=20.0*ALOG10(BIG1)
C
C
      WRITE(3,113) BIG
  113 FORMAT(///20X,'PLOT NORMALIZATION FACTOR = ',F8.3,' DB.'////)
C
C
      IF(PR2+PL2+PL3.EQ.0) GO TO 50
      YR=STARTY-DELTAY
      DO 40 NN=1,NPTSY
      YR=YR+DELTAY
      XR=STARTX-DELTAX
      DO 40 MM=1,NPTSX
      XR=XR+DELTAX
      AF=PAT(XR,YR)
```

```
            IF(CABS(AF).LT.1E-10) AF=(1E-10,0.)
            APLOT(MM,NN)=20.*ALOG10(CABS(AF))-BIG
         40 CONTINUE
      C
         41 IF(PL2.EQ.1) CALL PLOT2(NPTSX,NPTSY,CON,NCON,NUMPAT,-1000.)
            IF(PR2.EQ.1) CALL PATCON(APLOT,CONLOW,CONMAX,STARTX/RADDEG,
           $STOPX/RADDEG,STARTY/RADDEG,STOPY/RADDEG,NUMPAT)
            IF(PL3.NE.1) GO TO 50
            DO 45 N=1,NPTSY
            DO 45 M=1,NPTSX
            IF(APLOT(M,N).LT.FLOOR) APLOT(M,N)=FLOOR
         45 CONTINUE
            CALL PLOT3(NPTSX,NPTSY,NUMPAT)
      C
      C........ONE DIMENSIONAL PLOTS
      C
         50 CONTINUE
            IF(PR1+PL1.EQ.0) GO TO 51
      C
      C        CALCULATE PROFILE ALONG X-AXIS AT Y=YC
      C
            DX=(STOPX-STARTX)/500.0
            YR=FLAT
            XR=STARTX-DX
            DO 60 MM=1,501
            XR=XR+DX
            AF=PAT(XR,YR)
            XPRINT(MM,1)=XR/RADDEG
            XPRINT(MM,2)=CABS(AF)/BIG1
            IF(XPRINT(MM,2).LT.1E-10) XPRINT(MM,2)=1E-10
         60 XPLOT(MM)=20.*ALOG10(XPRINT(MM,2))
            IF(PR1.NE.1) GO TO 63
            WRITE(3,114) FLATD
        114 FORMAT(1H1,25X,'X-AXIS PROFILE PLOT ALONG ',F8.3,' DEGREES LATITUD
           $E.')
            CALL PROFIL(XPRINT,501,NUMPAT)
         63 CONTINUE
            IF(PL1.NE.1) GO TO 64
```

```
DOS FORTRAN IV 360N-FO-479 3-9          MAINPGM          DATE   12/21/78     TIME    13.37.46

0150              CALL PLOT1(STARTX/RADDEG,STOPX/RADDEG,500,1,FLATD,NUMPAT,XPLOT)
0151           64 CONTINUE
            C
            C........CALCULATE PROFILE ALONG Y-AXIS AT X=FLAT
0152              DY=(STOPY-STARTY)/500.0
0153              XR=FLONG
0154              YR=STARTY-DY
0155              DO 70 MM=1,501
0156              YR=YR+DY
0157              AF=PAT(XR,YR)
0158              YPRINT(MM,1)=YR/RADDEG
0159              YPRINT(MM,2)=CABS(AF)/BIG1
0160              IF(YPRINT(MM,2).LT.1E-10) YPRINT(MM,2)=1E-10
0161           70 YPLOT(MM)=20.*ALOG10(YPRINT(MM,2))
            C
            C........OUTPUT SEQUENCE
0162              IF(PR1.NE.1) GO TO 80
0163              WRITE(3,115) FLONGD
0164          115 FORMAT(1H1,25X,'Y-AXIS PROFILE PLOT ALONG',F8.3,'LONGITUDE.')
0165              CALL PROFIL(YPRINT,501,NUMPAT)
            C........PLOTTER PLOTS
0166           80 CONTINUE
0167              IF(PL1.NE.1) GO TO 90
0168              CALL PLOT1(STARTY/RADDEG,STOPY/RADDEG,500,2,FLONGD,NUMPAT,YPLOT)
0169           90 CONTINUE
0170           51 CONTINUE
0171           95 CONTINUE
            C........NEW SEARCH CODE
0172              CALL PLOT(0.,0.,-3)
0173              GO TO 1000
            C
            C
            C
0174         9999 CONTINUE
0175              WRITE(3,117)
0176          117 FORMAT(20X,'NORMAL TERMINATION')
0177              CALL PLOT(0.,0.,999)
0178              STOP
```

DOS FORTRAN IV 360N-FO-479 3-9 ORBIT DATE 12/21/78 TIME 13.38.25

```
      SUBROUTINE ORBIT(NUMPAT,FREQ)
      DATA RADDEG /0.01745329E0/
      REAL ANGLE(3,3)
      COMMON /ANG/ YAW,TILT,TWIST,CTW,STW,CT,ST,CYAW,SYAW
      COMMON /SYS/ REARTH,RE3,ALT,ALT3,ANGLE
      REARTH = 6370.0
      RE3=REARTH*1E3
      READ(1,6) FREQ
      READ(1,6) ALT
    6 FORMAT(8F10.0)
      ALT3=ALT*1E3
      READ(1,6) YAWD,TILTD,TWISTD
      TWIST=RADDEG*TWISTD
      TILT=RADDEG*TILTD
      YAW=RADDEG*YAWD
      CTW=COS(TWIST)
      STW=SIN(TWIST)
      CT=COS(TILT)
      ST=SIN(TILT)
      CYAW=COS(YAW)
      SYAW=SIN(YAW)
C
      ANGLE(1,1)=CTW*CYAW-SYAW*CT*STW
      ANGLE(1,2)=SYAW*CTW+CYAW*CT*STW
      ANGLE(1,3)=-ST*STW
      ANGLE(2,1)=-STW*CYAW-SYAW*CT*CTW
      ANGLE(2,2)=CYAW*CT*CTW-SYAW*STW
      ANGLE(2,3)=-ST*CTW
      ANGLE(3,1)=-SYAW*ST
      ANGLE(3,2)=CYAW*ST
      ANGLE(3,3)=CT
C
C........OUTPUT SYSTEM INFORMATION
C
      WRITE(3,100) FREQ
  100 FORMAT(20X,'SYSTEM INFORMATION:'/25X,'FREQUENCY = ',F6.3,' GHZ.')
      WRITE(3,101) YAWD,TILTD,TWISTD,ALT
  101 FORMAT(25X,'YAW = ',F8.3,' DEGREES'/25X,'TILT = ',F8.3,' DEGREES'/
```

```
              $25X,'TWIST = ',F8.3,' DEGREES'/25X,'ALTITUDE = ',F8.3,' KM.'///)
          C
0035            RETURN
0036            END
```

```
DOS FORTRAN IV 360N-FO-479 3-9          ANTENA              DATE   12/21/78     TIME     13.38.37

0001            SUBROUTINE ANTENA(NUMPAT)
          C
          C      THIS ROUTINE INPUTS AND CALCULATES APPROPRIATE ANTENNA PARAMETERS
          C      FOR THE SPACE SHUTTLE SYNTHETIC IMAGING RADAR ANTENNA.
          C
          C      FINALLY, ALL ANTENNA PARAMETERS ARE OUTPUTTED.
          C
          C
          C
          C...............ANTENNA CONFIGURATION -- TWO-DIMENSIONAL ARRAY.................C
          C
0002            REAL ZAVG(24,12),ALPHAX(24,12),ALPHAY(24,12),K
0003            REAL KX,KY
0004            COMMON /ANT/ MSECT,NSECT,MEL,NEL,SX,SY,PX,PY,IHV,K,OMU4PI,ZAVG,
               $ALPHAX,ALPHAY,KX,KY,PXSECT,PYSECT
          C
          C
0005            READ(1,1) MSECT,NSECT
0006            READ(1,1) NX,NY
0007          1 FORMAT(8I5)
0008            MEL=NX/MSECT
0009            NEL=NY/NSECT
0010            READ(1,2) SX,SY
0011          2 FORMAT(8F10.0)
0012            READ(1,2) PXD,PYD
0013            PX=PXD*0.01745329E0
0014            PY=PYD*0.01745329E0
0015            READ(1,1)IHV
0016            WRITE(3,3) NUMPAT
0017          3 FORMAT(20X,'ANTENNA PARAMETERS FOR SIMULATION NUMBER ',I3)
0018            IF(IHV.EQ.0) WRITE(3,100)
0019            IF(IHV.EQ.1) WRITE(3,101)
0020            IF(IHV.EQ.2) WRITE(3,102)
0021        100 FORMAT(25X,'ELEMENT TYPE: ISOTROPIC')
0022        101 FORMAT(25X,'ELEMENT TYPE: HORIZONTAL DIPOLE')
0023        102 FORMAT(25X,'ELEMENT TYPE: VERTICAL DIPOLE')
0024            WRITE(3,4) NX,NY
0025          4 FORMAT(25X,'NUMBER OF ELEMENTS (X,Y) = (',I3,' , ',I3,')')
0026            WRITE(3,5) SX,SY
```

```
    5 FORMAT(25X,'INTERELEMENT SPACING (CM.): ',F7.4,' , ',F7.4)
      WRITE(3,6) PXD,PYD
    6 FORMAT(25X,'INTERELEMENT PHASE SHIFT (DEG.): ',F7.4,' , ',F7.4///)
      RETURN
      END
```

```
DOS FORTRAN IV 360N-FO-479 3-9          MECH               DATF    12/21/78      TIME     13.38.55

              SUBROUTINE MECH(NUMPAT)
              RFAL ZAVG(24,12),ALPHAX(24,12),ALPHAY(24,12),K
              REAL A(3,3)
              REAL KX.KY
              RFAL WARP(201)
              COMMON /SYS/ RE,RE3,ALT,ALT3,A
              COMMON /ANT/ MSECT,NSECT,MEL,NEL,SX,SY,PX,PY,IHV,K,OMU4PI,ZAVG,
             $ALPHAX,ALPHAY,KX,KY,PXSECT,PYSECT
        C
        C
              CALL MISC(MSECT,NSECT,WARP)
              WRITE(3,100) NUMPAT
          100 FORMAT(20X,'DEFORMATION DATA FOR SIMULATION ',I3,':'/)
              NSECT1=NSECT+1
              MSECT1=MSECT+1
              DO 50 I1=1,NSECT1
              I=NSECT1-I1+1
              II=(I-1)*MSECT1+1
              III=I*MSECT1
              WRITE(3,101) (WARP(IJ),IJ=II,III)
          101 FORMAT(10(3X,F7.2))
              WRITE(3,105)
          105 FORMAT(/)
           50 CONTINUF
              WRITE(3,104)
        C
        C
              PXSECT=MEL*PX
              PYSECT=NEL*PY
              XLEN=SX*MEL
              YLEN=SY*NEL
              DX2=0.5/XLEN
              DY2=0.5/YLEN
        C
        C
          104 FORMAT(////)
              IAREA=0
              DO 30 N=1,NSECT
```

```
0032            YCENT=YLEN*(N-FLOAT(NSECT+1)*0.5)
0033            DO 30 M=1,MSECT
0034            XCENT=XLEN*(M-FLOAT(MSECT+1)*0.5)
0035            IAREA=IAREA+1
0036            NUM=(IAREA-1)+N
0037            A0=WARP(NUM)
0038            A1=WARP(NUM+1)
0039            A2=WARP(NUM+1+MSECT)
0040            A3=WARP(NUM+2+MSECT)
         C
0041            B0=(A0+A1+A2+A3)*0.25
0042            B1=DX2*(-A0+A1-A2+A3)
0043            B2=DY2*(-A0-A1+A2+A3)
0044            B3=DX2*DY2*4*(A0-A1-A2+A3)
         C
0045            ZAVG(M,N)=B0
0046            ALPHAX(M,N)=ATAN(B1)
0047            ALPHAY(M,N)=ATAN(B2)
0048         30 CONTINUE
0049            WRITE(3,104)
0050            RETURN
0051            END
```

```
DOS FORTRAN IV 360N-FO-479 3-9          MISC                DATE    12/21/78     TIME     13.39.23

                  SUBROUTINE MISC(MSECT,NSECT,WARP)
                  REAL WARP(201)
                  NWARP=(MSECT+1)*(NSECT+1)
                  READ(1,2)  (WARP(I),I=1,NWARP)
                2 FORMAT(8F10.0)
                  RETURN
                  END
```

```
DOS FORTRAN IV 360N-FO-479 3-9          ELEC               DATE   12/21/78     TIME    13.39.39

            SUBROUTINE ELEC(MSECT,NSECT,NUMPAT)
            REAL PHSX(24,12),AMAG(24,12),APHS(24,12),PHSY(24.12)
            COMMON /ELCTRC/ PHSX,PHSY,AMAG,APHS
            MNSECT=(MSECT+1)*(NSECT+1)
            DO 10 J=1,NSECT
            DO 10 I=1,MSECT
         10 READ(1,11) PHSX(I,J),PHSY(I,J),AMAG(I,J),APHS(I,J)
         11 FORMAT(8F10.0)
      C
      C
            WRITE(3,12) NUMPAT
         12 FORMAT(20X,'ELECTRICAL DATA FOR SIMULATION ',I3,':'/)
            J=NSFCT+1
            DO 20 JJ=1,NSECT
            J=J-1
            WRITE(3,25) (I,J,I=1,MSECT)
            WRITE(3,21) (PHSX(I,J),I=1,MSECT)
            WRITE(3,22) (PHSY(I,J),I=1,MSECT)
            WRITE(3,23) (AMAG(I,J),I=1,MSECT)
            WRITE(3,24) (APHS(I,J),I=1,MSECT)
            WRITE(3,104)
         20 CONTINUE
         21 FORMAT(5X,'PHSX:',10(3X,F7.2))
         22 FORMAT(5X,'PHSY:',10(3X,F7.2))
         23 FORMAT(5X,'AMAG:',10(3X,F7.2))
         24 FORMAT(5X,'APHS:',10(3X,F7.2))
         25 FORMAT(10X,10(3X,'(',I2,',',I2,')'))
            DO 30 J=1,NSECT
            DO 30 I=1,MSECT
            PHSX(I,J)=PHSX(I,J)*0.017453293E0
            PHSY(I,J)=PHSY(I,J)*0.017453293E0
            APHS(I,J)=APHS(I,J)*0.017453293E0
         30 CONTINUE
            WRITE(3,104)
        104 FORMAT(////)
            RETURN
            END
```

```
0001               FUNCTION PAT(XR,YR)
            C
            C         PAT COMPUTES THE COMPLEX-VALUED PATTERN FROM A PIECEWISE
            C         BILINEAR RECTANGULAR ARRAY.
            C
            C        ANTENNA PARAMETERS ARE PASSED BY COMMON BLOCK ANT.
            C         OTHER SYSTEM PARAMETERS ARE PASSED BY COMMON BLOCK SYS.
            C
            C
            C         WRITTEN BY: E. L. COFFEY
            C               DATE:  26 JULY 1976
            C
            C
0002               COMPLEX PAT,CEXP,J
0003               COMPLEX AF2
0004               REAL ZAVG(24,12),ALPHAX(24,12),ALPHAY(24,12),K
0005               REAL KX,KY
0006               REAL PHSX(24,12),AMAG(24,12),APHS(24,12),PHSY(24,12)
0007               REAL A(3,3)
            C
0008               COMMON /SYS/ RE,RE3,ALT,ALT3,A
0009               COMMON /ANT/ MSECT,NSECT,MEL,NEL,SX,SY,PX,PY,IHV,K,OMU4PI,
                  $ZAVG,ALPHAX,ALPHAY,KX,KY,PXSECT,PYSECT
0010               COMMON /ELCTRC/ PHSX,PHSY,AMAG,APHS
            C
            C
0011               DATA J/(0.,1.)/
            C
            C........TRANSLATION FROM EARTH SURFACE TO UNROTATED ANTENNA.
0012               CYR=COS(YR)
0013               XP=-RE3*SIN(XR)*CYR
0014               YP=RE3*SIN(YR)
0015               ZP=RE3*(1.-COS(XR)*CYR)+ALT3
            C
            C........ANTENNA YAW,TILT,TWIST
0016               XPP=A(1,1)*XP+A(1,2)*YP+A(1,3)*ZP
0017               YPP=A(2,1)*XP+A(2,2)*YP+A(2,3)*ZP
0018               ZPP=A(3,1)*XP+A(3,2)*YP+A(3,3)*ZP
```

```
DOS FORTRAN IV 360N-FO-479 3-9          PAT                 DATE   12/21/78      TIME     13.39.51

 0019                RPP=SQRT(XPP*XPP+YPP*YPP+ZPP*ZPP)
 0020                U=XPP/RPP
 0021                V=YPP/RPP
 0022                ALPHA1=ARCOS(U)
 0023                ALPHA2=ARCOS(V)
 0024                AF1=OMU4PI/RPP
           C
           C......TRANSLATE TO CENTER OF SUBARRAY
           C
 0025                DX=SX*MEL
 0026                DY=SY*NEL
 0027                AF2=(0.,0.)
 0028                YCENT=-FLOAT(NSECT+1)/2.0*DY
 0029                DO 10 N=1,NSECT
 0030                YCENT=YCENT+DY
 0031                XCENT=-FLOAT(MSECT+1)/2.0*DX
 0032                DO 10 M=1,MSECT
 0033                XCENT=XCENT+DX
 0034                UP=COS(ALPHA1-ALPHAX(M,N))+PHSX(M,N)/KX
 0035                VP=COS(ALPHA2-ALPHAY(M,N))+PHSY(M,N)/KY
 0036                WP=SQRT(1.-(UP*UP+VP*VP))
 0037                PHASE=K*(XCENT*UP+YCENT*VP+ZAVG(M,N)*WP)+M*PXSECT+N*PYSECT
                    $+APHS(M,N)
 0038                AF2=AF2+AF1*CMPLX(COS(PHASE),SIN(PHASE))*AF(UP,VP)
                    $*AMAG(M,N)
 0039             10 CONTINUE
           C
           C
 0040                PAT=AF2
 0041                RETURN
 0042                END
```

```
DOS FORTRAN IV 360N-FO-479 3-9          GENER               DATE   12/21/78     TIME    13.40.16

0001            SUBROUTINE GENER
0002            REAL ZAVG(24,12),ALPHAX(24,12),ALPHAY(24,12),K
0003            REAL KX,KY
0004            REAL UARRAY(1001),VARRAY(1001)
0005            COMMON /PATTRN/ UARRAY,VARRAY,UMAX,VMAX,DU,DV
0006            COMMON /ANT/ MSECT,NSECT,MEL,NEL,SX,SY,PX,PY,IHV,K,OMU4PI,
               $ZAVG,ALPHAX,ALPHAY,KX,KY,PXSECT,PYSECT
          C
          C........THIS ROUTINE GENERATES PATTERN DATA FOR THE ARRAYS UARRAY AND VARRAY.
          C
0007            NULNUM=3
0008            UMAX=6.28318531E0/K/(MEL*SX)*NULNUM
0009            VMAX=6.28318531E0/K/(NEL*SY)*NULNUM
0010            IF(UMAX.GT.1.0) UMAX=1.0
0011            IF(VMAX.GT.1.0) VMAX=1.0
          C
0012            DU=UMAX*1E-3
0013            DV=VMAX*1E-3
          C
          C
0014            DO 10 I=1,1001
0015            U=(I-1)*DU
0016            PSIX=KX*U+PX
0017            PSIX2=0.5*PSIX
0018            U1=1.0
0019            IF(PSIX.NE.0.) U1=SIN(MEL*PSIX2)/(MEL*SIN(PSIX2))
0020            IF(IHV.EQ.1.AND.ABS(U).NE.1.0)U1=U1*COS(1.570796E0*U)/SQRT(1.-U*U)
0021            UARRAY(I)=U1
          C
0022            V=(I-1)*DV
0023            PSIY=KY*V+PY
0024            PSIY2=0.5*PSIY
0025            V1=1.0
0026            IF(PSIY.NE.0.) V1=SIN(NEL*PSIY2)/(NEL*SIN(PSIY2))
0027            IF(IHV.EQ.2.AND.ABS(V).NE.1.)V1=V1*COS(1.570796E0*V)/SQRT(1.-V*V)
0028            VARRAY(I)=V1
0029         10 CONTINUE
          C
```

```
               C
 0030                RETURN
 0031                END
```

```
DOS FORTRAN IV 360N-FO-479 3-9          AF                  DATE   12/21/78     TIME    13.40.32

0001               FUNCTION AF(U1,V1)
             C
             C........THIS FUNCTION CALCULATES THE ARRAY FACTOR OF A SUBPANEL BY
             C          "TABLE LOOK-UP" IF U < UMAX AND V < VMAX.   OTHERWISE, THE FUNCTION
             C          IS COMPUTED IN THE USUAL WAY.
             C
             C          LINEAR INTERPOLATION IS USED IN THE TABLE LOOK-UP.
             C
0002               REAL ZAVG(24,12),ALPHAX(24,12),ALPHAY(24,12),K
0003               REAL KX,KY
0004               REAL UARRAY(1001),VARRAY(1001)
             C
0005               COMMON /PATTRN/ UARRAY,VARRAY,UMAX,VMAX,DU,DV
0006               COMMON /ANT/ MSECT,NSECT,MEL,NEL,SX,SY,PX,PY,IHV,K,OMU4PI,
                  $ZAVG,ALPHAX,ALPHAY,KX,KY,PXSECT,PYSECT
             C
             C
0007               U=ABS(U1)
0008               V=ABS(V1)
             C
0009               IF(U.GT.UMAX) GO TO 10
             C........U IS IN RANGE
0010               FU=U/DU+1.0
0011               IFU=IFIX(FU)
0012               DIFF=FU-IFU
0013               UPAT=(1.0-DIFF)*UARRAY(IFU)+DIFF*UARRAY(IFU+1)
0014               GO TO 15
0015            10 CONTINUE
0016               PSIX2=(KX*U+PX)*0.5
0017               UPAT=SIN(MEL*PSIX2)/(MEL*SIN(PSIX2))
0018               IF(IHV.EQ.1) UPAT=UPAT*COS(1.570796E0*U)/SQRT(1.-U*U)
0019            15 CONTINUE
0020               IF(V.GT.VMAX) GO TO 20
             C........V IS IN RANGE
0021               FV=V/DV+1.0
0022               IFV=IFIX(FV)
0023               DIFF=FV-IFV
0024               VPAT=(1.0-DIFF)*VARRAY(IFV)+DIFF*VARRAY(IFV+1)
```

```
DOS FORTRAN IV 360N-FO-479 3-9          AF                    DATE    12/21/78       TIME      13.40.32

                  GO TO 25
               20 CONTINUE
                  PSIY2=(KY*V+PY)*0.5
                  VPAT=SIN(NEL*PSIY2)/(NEL*SIN(PSIY2))
                  IF(IHV.EQ.2) VPAT=VPAT*COS(1.570796E0*V)/SQRT(1.-V*V)
               25 CONTINUE
                  AF=UPAT*VPAT/(MSECT*NSECT)
                  RETURN
                  END
```

```
0001                SUBROUTINE OUTPUT(NUMPAT,IDIREC)
            C
            C
0002                INTEGER BUFFER(1000),SYS13
0003                REAL CON(10)
0004                INTEGER PR1,PR2,PL1,PL2,PL3,TP1,TP2
0005                DATA RADDEG/0.01745329E0/
            C
            C
0006                COMMON /IO/ PR1,PR2,PL1,PL2,PL3,TP1,TP2,NCON,CON,STARTX,STOPX,
                   $DELTAX,STARTY,STOPY,DELTAY,CONLOW,CONMAX,NPTSX,NPTSY,FLOOR,DASH
            C
            C
0007                READ(1,8) PR1,PR2
0008                READ(1,8) PL1,PL2,PL3
0009                READ(1,8) TP1,TP2
0010             41 IF(PL1+PL2+PL3.EQ.0) GO TO 10
            C........INITIALIZE PLOT BUFFER
            C
0011                NBUF=1000
0012                SYS13=13
0013                CALL PLOTS(BUFFER,NBUF,SYS13)
            C
0014             10 CONTINUE
0015                READ(1,6) STRXD,STPXD
0016                READ(1,6) STRYD,STPYD
0017                STARTX=STRXD*RADDEG
0018                STOPX=STPXD*RADDEG
0019                STARTY=STRYD*RADDEG
0020                STOPY=STPYD*RADDEG
0021                IF(PL2.EQ.1) READ(1,7) NCON
0022                IF(PL2.EQ.1) READ(1,6) (CON(JJ),JJ=1,NCON)
0023                IF(PR2.EQ.1) READ(1,6) CONLOW,CONMAX
0024                IF(PR2+PL2+PL3.GE.1) READ(1,7) NPTSX,NPTSY
0025                DELTAX=(STOPX-STARTX)/(NPTSX-1)
0026                DELTAY=(STOPY-STARTY)/(NPTSY-1)
0027                DXD=DELTAX*57.295
0028                DYD=DELTAY*57.295
```

```
C
C........OUTPUT PRINT/PLOT INFORMATION
C
          WRITE(3,103)
      103 FORMAT(////20X,'PRINT/PLOT INFORMATION:'/25X,'REQUESTED OUTPUT:')
          IF(PR1.EQ.1) WRITE(3,104)
          IF(PR2.EQ.1) WRITE(3,105)
          IF(PL1.EQ.1) WRITE(3,106)
          IF(PL2.EQ.1) WRITE(3,107)
          IF(PL3.EQ.1) WRITE(3,108)
      104 FORMAT(30X,'PRINTER PROFILE')
      105 FORMAT(30X,'PRINTER CONTOUR')
      106 FORMAT(30X,'PLOTTER PROFILE')
      107 FORMAT(30X,'PLOTTER CONTOUR')
      108 FORMAT(30X,'PLOTTER THREE-D')
C
          IF(PL2.NE.1 ) GO TO 24
          WRITE(3,109) (CON(I),I=1,NCON)
      109 FORMAT(//25X,'CONTOURS TO BE PLOTTED: ',5(F6.2,3X)/49X,5(F6.2,3X))
       24 CONTINUE
          IF(PR2+PL2+PL3.GE.1) WRITE(3,110) NPTSX,NPTSY
      110 FORMAT(25X,'PLOT RESOLUTION: ',I3,' X ',I3,' POINTS'//)
          WRITE(3,111) STRXD,STPXD,DXD
      111 FORMAT(25X,'STARTX = ',F8.3/25X,'STOPX = ',F8.3/25X,'DELTAX = ',
         $F8.3/)
          WRITE(3,112) STRYD,STPYD,DYD
      112 FORMAT(25X,'STARTY = ',F8.3/25X,'STOPY = ',F8.3/25X,'DELTAY = ',
         $F8.3//)
        6 FORMAT(8F10.0)
        7 FORMAT(2I5)
        8 FORMAT(5I1)
       32 FORMAT(10X,'SYSNO =  'I2,5X,'LENGTH =  ',I5)
          RETURN
          END
```

```
0001               SUBROUTINE PROFIL(DATA,NPT,NUMPAT)
            C
            C       DATA = DATA INPUT (DESTROYED)
            C       NPT = NUMBER OF POINTS
            C       NUMPAT = PATTERN NUMBER
            C
0002               INTEGER SF
0003               INTEGER OUTPUT(81)
0004               INTEGER BLANK,PLUS,SLASH,STAR
0005               REAL DATA(501,2),BOUND(81)
0006               DATA BLANK,PLUS,SLASH,STAR /' ','+','|','*' /
            C
            C         FIND THE RANGE OF DEPENDENT DATA AND SCALE IF NECESSARY
            C
0007               IF(NPT.GT.501) GO TO 999
0008               BIG=-1.E10
0009               SMALL = 1.E10
0010               DO 1 J=1,NPT
0011               IF(DATA(J,2).LT.-60.0) DATA(J,2)=-60.0
0012               IF(DATA(J,2).LT.SMALL) SMALL=DATA(J,2)
0013               IF(DATA(J,2).GT.BIG) BIG=DATA(J,2)
0014             1 CONTINUE
0015               DIFF=ABS(BIG-SMALL)
0016               SF = 0
0017               IF(DIFF.LT.1.) GO TO 10
0018               IF(DIFF.LT.100.)GO TO 21
0019               DO 2 J=1,10
0020               IF(DIFF*10.**(-J).GT.100.) GO TO 2
0021               SF=J
0022               GO TO 20
0023             2 CONTINUE
0024           400 WRITE(3,100)
0025           100 FORMAT('0 YOUR DATA IS TOO LARGE FOR THIS PROGRAM.')
0026               RETURN
0027            10 DO 3 J=1,10
0028               K=11-J
0029               IF(DIFF*10**K.GT.100.) GO TO 3
0030               SF=-K
```

```
0031               GO TO 20
0032             3 CONTINUE
0033               GO TO 400
0034            20 DO 4 J=1,NPT
0035             4 DATA(J,2) = DATA(J,2)*10.**(-SF)
          C
          C           CALCULATE BOUNDS
          C
0036            21 SCALE=DIFF/80.
0037               DO 5 J=1,81
0038               K=J-1
0039             5 BOUND(J)=(BIG-K*SCALE)*10.**(-SF)
0040               SFDB=20.*SF
          C
          C           PRINT TITLE
          C
0041               WRITE(3,640) NUMPAT
0042           640 FORMAT(26X,'PATTERN NUMBER ',I5//)
0043               IF (SF.EQ.0) GO TO 200
0044               WRITE(3,4004) SF
0045          4004 FORMAT(53X,'SCALE FACTOR IS 10**',I2/)
0046           200 WRITE(3,650) (BOUND(J),J=1,81,16)
0047           650 FORMAT(3X,'DB.',5X,'REAL',2X,6(F7.4,9X))
0048               DO 6 J1=1,NPT
0049               J=NPT+1-J1
0050               DO 50 K=1,81
0051            50 OUTPUT(K)=BLANK
0052               IF((J-1)/8*8-(J-1)) 62,61,62
0053            61 DO 40 K=1,81,8
0054            40 OUTPUT(K)=PLUS
0055               GO TO 87
0056            62 OUTPUT(1)=SLASH
0057               OUTPUT(81)=SLASH
0058            87 DO 7 K=1,80
0059               IF(DATA(J,2).GT.BOUND(K)) GO TO 7
0060               IF(DATA(J,2).LE.BOUND(K+1)) GO TO 7
0061               OUTPUT(K)=STAR
0062               GO TO 69
```

```
      7 CONTINUF
        OUTPUT(81)=STAR
     69 IF(DATA(J,2).EQ.0.0) DATA(J,2) = 1.0E-6
        DATADB=20.*ALOG10(ABS(DATA(J,2)))+SFDB
    141 WRITE(3,4000) DATADB,DATA(J,2),OUTPUT,DATA(J,1)
   4000 FORMAT(1X,F6.2,2X,F8.3,2X,81A1,1X,F9.4)
        GO TO 6
      6 CONTINUE
        WRITE(3,650) (BOUND(J),J=1,81,16)
        WRITE(3,651)
    651 FORMAT(1H1)
    999 RETURN
        END
```

```
0001            SUBROUTINE PATCON(A    ,CONLOW,CONMAX,STARTX,STOPX,STARTY,STOPY,
               $NUMPAT)
          C
          C       THIS ROUTINE CALLS SUBROUTINE CONTUR THREE TIMES TO GENERATE A
          C       COMPLETE 151 X 151 TWO-DIM. PLOT.
          C
0002            COMMON/AREA1/NUMPT(12),LEVEL(12)
0003            DIMENSION A(151,151)
0004            DO 10 I=1,12
0005         10 NUMPT(I)=0
0006            DY=(STOPY-STARTY)/3.0
0007            CALL CONTUR(A ,1,CONLOW,CONMAX,STARTX,STOPX,STARTY,STARTY+DY,
               $NUMPAT)
0008            CALL CONTUR(A ,2,CONLOW,CONMAX,STARTX,STOPX,STARTY+DY,STOPY-DY,
               $NUMPAT)
0009            CALL CONTUR(A ,3,CONLOW,CONMAX,STARTX,STOPX,STOPY-DY,STOPY,NUMPAT)
0010            WRITE(3,12)
0011         12 FORMAT('1','NUMBER OF POINTS IN EACH CONTOUR LEVEL')
0012            WRITE(3,13) (LEVEL(J),NUMPT(J),J=1,12)
0013         13 FORMAT('0',5X,'NUMPT(',A1,')=',I6)
0014            RETURN
0015            END
```

```
DOS FORTRAN IV 360N-FO-479 3-9          MAINPGM          DATE   12/21/78     TIME     13.45.04

                  BLOCK DATA
                  COMMON/AREA1/NUMPT(12),LEVEL(12)
                  DATA LEVEL/'0','1','2','3','4','5','6','7','8','9',' ','+'/
                  END
```

```
DOS FORTRAN IV 360N-FO-479 3-9          CONTUR              DATE   12/21/78     TIME    13.45.04

0001               SUBROUTINE CONTUR(A,ICODE,CONLOW,CONMAX,STARTX,STOPX,STARTY,STOPY,
                  $NUMPAT)
             C
             C       THIS ROUTINE PRINTS A  51 X 151 CONTOUR PLOT OF THE FOOTPRINT.
             C       TO OBTAIN A COMPLETE 151 X 151 PLOT, THIS ROUTINE IS CALLED THREE
             C       TIMES BY SUBROUTINE PATCON.  THEN THE THREE SEPARATE PLOTS ARE
             C       JOINED TOGETHER TO MAKE THE COMPOSITE.
             C
             C
0002               REAL A(151,151),LOW(12),HIGH(12),AXIS(11)
0003               INTEGER OUTPUT(101),BLANK,DOT
0004               COMMON/AREA1/NUMPT(12),LEVEL(12)
             C
             C
0005               DATA BLANK /' '/
             C
0006               WRITE(3,10) NUMPAT
0007            10 FORMAT(1H1,37X,'PRINTER CONTOUR PLOT FOR SIMULATION NUMBER ',I3//
                  $///)
             C
0008               DELTAX=(STOPX-STARTX)/150.0
0009               DELTAY=(STOPY-STARTY)/10.0
0010               NUMCON=10
0011               CONINT=(CONMAX-CONLOW)/FLOAT(NUMCON-1)
0012               DELCON=CONINT*0.5
0013               DO 71 J=1,NUMCON
0014               LOW(J)=CONLOW+(J-1)*CONINT-DELCON
0015               HIGH(J)=LOW(J)+CONINT+1E-5
0016            71 CONTINUE
0017               LOW(11)=-1E50
0018               HIGH(12)=1E50
0019               HIGH(11)=LOW(1)
0020               LOW(12)=HIGH(NUMCON)
             C
0021               DO 40 I=1,11
0022            40 AXIS(I)=STARTY+(I-1)*DELTAY
0023               WRITE(3,42) (AXIS(I),I=1,11)
0024            42 FORMAT(6X,11(F8.3,2X)/10X,10('|.........'),'|')
```

```
               DO 50 N=1,151
               X=STARTX+(N-1)*DELTAX
               DO 51 K=1,101
            51 OUTPUT(K)=BLANK
      C
               DO 60 M=1,51
               MM=2*M-1
               M1=M+50*(ICODE-1)
               F=A(N,M1)
               IF(F.LE.LOW(1)) GO TO 1001
               IF(F.GT.HIGH(NUMCON)) GO TO 1002
               DO 61 K=1,NUMCON
               IF(F.GT.LOW(K) .AND. F.LE.HIGH(K)) GO TO 62
            61 CONTINUE
          1002 OUTPUT(MM)=LEVEL(12)
               IF (ICODE.EQ.3) GO TO 66
               IF (M.EQ.51) GO TO 60
            66 NUMPT(12) = NUMPT(12)+1
               GO TO 60
          1001 OUTPUT(MM)=LEVEL(11)
               IF (ICODE.EQ.3) GO TO 67
               IF (M.EQ.51) GO TO 60
            67 NUMPT(11) = NUMPT(11)+1
               GO TO 60
            62 OUTPUT(MM)=LEVEL(K)
               IF(ICODE.EQ.3) GO TO 68
               IF(M.EQ.51) GO TO 60
            68 NUMPT(K) = NUMPT(K)+1
            60 CONTINUE
      C
      C........OUTPUT
      C
               IF(ICODE.EQ.1) WRITE(3,1) X,OUTPUT
               IF(ICODE.EQ.2) WRITE(3,2) OUTPUT
               IF(ICODE.EQ.3) WRITE(3,3) OUTPUT,X
             1 FORMAT(1X,F8.4,'.',101A1)
             2 FORMAT(10X,101A1)
             3 FORMAT(10X,101A1,'.',1X,F8.4)
```

DOS FORTRAN IV 360N-FO-479 3-9   CONTUR   DATE   12/21/78   TIME   13.45.04

```
      50 CONTINUE
         WRITE(3,43) (AXIS(I),I=1.11)
      43 FORMAT(10X,10('|.........'),'|'/6X,11,F8.3,2X))
C
C
         IF(ICODE.NE.2) GO TO 80
         WRITE(3,44)
      44 FORMAT(///51X,'CONTOUR LEVEL KEY'//)
         DO 45 I=1,4
      45 WRITE(3,46) (LEVEL(J),LOW(J),HIGH(J),J=I,12,4)
      46 FORMAT(2X,3(A1,': ',E14.7,' TO ',E14.7,3X))
      80 CONTINUE
         RETURN
         END
```

```
0001              SUBROUTINE PLOT1(PSTRT,PEND,IP,CODE,CONST,NUMPAT,PTS)
           C
           C
           C         PROFILE PLOT ROUTINE
           C
           C         WRITTEN BY: E. L. COFFEY
           C               DATE: 22 JUNE 1976
           C
           C         INPUT:
           C      PSTRT= BEGINNING OF PLOT
           C      PEND= END OF PLOT
           C      IP= NUMBER OF POINTS TO BE PLOTTED
           C      CODE= LABELLING CODE -- 1 FOR X-AXIS, 2 FOR Y-AXIS
           C      CONST= "OTHER" AXIS CONSTANT
           C      NUMPAT= PATTERN NUMBER...USER DEFINED
           C
           C
0002              INTEGER CODE
0003              DIMENSION PTS(501)
0004              CALL PLOT(8.,2.,23)
0005              CALL FACTOR(0.7)
0006              FNUM=NUMPAT
           C
           C         DRAW AXES AND LABEL THEM
           C
0007              Y=0.05
0008              DO 11 J=1,11
0009              X=-5.0+(J-1)*1.0
0010              CALL PLOT(X,-Y,3)
0011           11 CALL PLOT(X,Y,2)
0012              CALL PLOT(5.,0.,3)
0013              CALL PLOT(-5.,0.,2)
           C
0014              CALL PLOT(0.,5.5,3)
0015              CALL PLOT(0.,0.,2)
0016              X=0.05
0017              DO 10 J=1,6
0018              Y=0.5+(J-1)*1.0
```

```
DOS FORTRAN IV 360N-FO-479 3-9          PLOT1               DATE   01/08/79     TIME    18.57.30

0019            CALL PLOT(-X,Y,3)
0020         10 CALL PLOT(X,Y,2)
          C
          C
0021            CALL NUMBER(-5.25,-.75,0.125,PSTRT,0.,6)
0022            PMID=(PSTRT+PEND)*0.5
0023            CALL NUMBER(-.25,-.75,0.125,PMID,0.,6)
0024            CALL NUMBER(4.75,-.75,0.125,PEND,0.,6)
0025            CALL SYMBOL(-5.0,-1.5,0.125,7HPATTERN,0.,7)
0026            CALL NUMBER(-3.5,-1.5,0.125,FNUM,0.,-1)
          C
0027            IF(CODE.EQ.1) CALL SYMBOL(-2.0,-1.5,0.125,32HX-AXIS PROFILE ALONG
               $LATITUDE = ,0.,32)
0028            IF(CODE.EQ.2) CALL SYMBOL(-2.0,-1.5,0.125,33HY-AXIS PROFILE ALONG
               $LONGITUDE = ,0.,33)
0029            CALL NUMBER(2.0,-1.5,0.125,CONST,3)
0030            CALL ZIP(1,3,18)
          C
          C        PLOT IT
          C
0031            IF(PTS(1).LT.-50.) PTS(1)=-50.0
0032            FS=PTS(1)/10.+5.5
0033            CALL PLOT(-5.0,FS,3)
0034            DELTA=10./FLOAT(IP-1)
0035            DO 1 IW1=1,IP
0036            X=-5.0+(IW1-1)*DELTA
0037            Y=PTS(IW1)/10.+5.5
0038            IF(Y .LT. 0.5) GO TO 1
0039            CALL PLOT(X,Y,2)
0040          1 CONTINUE
0041            CALL FACTOR(1.0)
0042            CALL PLOT(8.0,-2.0,23)
0043            CALL ZIP(3,3,18)
0044            RETURN
0045            END
```

```
 0001                 SUBROUTINE PLOT2(N,M,CONTUR,NCON,NUMPAT,DASH)
             C                                                                    00000240
             C        A= N BY M MATRIX OF DATA POINTS
             C        CONTUR= ARRAY OF CONTOURS TO BE PLOTTED
             C        NCON= NUMBER OF CONTOURS TO BE PLOTTED
             C        NUMPAT= PATTERN NUMBER SPECIFIED BY USER
             C        DASH= CONTOURS BELOW DASH ARE PLOTTED AS DASHED LINES
             C
             C                                                                    00000310
 0002                 DIMENSION A(151,151),RA(151),RB(151),X(151),Y(151)
 0003                 REAL CONTUR(10)
 0004                 COMMON /ARRAY/ A                                            00000335
 0005                 CALL PLOT(8.,0.,23)
 0006                 CALL FACTOR (0.7)
 0007                 JJ=1
 0008                 MS=M                                                        00000340
 0009                 NS=N                                                        00000350
 0010                 RATIO=MS/NS                                                 00000360
 0011                 SCALE=10.
 0012                 ANM=AMAXO(N-1,M-1)                                          00000380
 0013                 IF(RATIO-1.0)1,1,2                                          00000390
 0014               1 SX=ANM                                                      00000400
 0015                 SY=RATIO*ANM                                                00000410
 0016                 GO TO 3                                                     00000420
 0017               2 SX=1./RATIO*ANM                                             00000430
 0018                 SY=ANM                                                      00000440
 0019               3 SMAX=AMAX1(SX,SY)                                           00000450
 0020                 SS=SX/SMAX                                                  00000460
 0021                 SYS=SY/SMAX                                                 00000470
 0022                 CALL PLOT(2.0,.25,3)
 0023                 DO 300 I=1,10
 0024                 CALL PLOT(FLOAT(I)+2.0,.25,2)
 0025                 CALL PLOT(FLOAT(I)+2.0,.5,2)
 0026                 CALL PLOT(FLOAT(I)+2.0,.25,2)
 0027             300 CONTINUE
 0028                 DO 301 I=1,10
 0029                 YY=FLOAT(I)+.25
 0030                 CALL PLOT(12.,YY,2)
```

```
      CALL PLOT(11.75,YY,2)
      CALL PLOT(12.,YY,2)
  301 CONTINUE
      DO 302 I=1,10
      XX=12.-FLOAT(I)
      CALL PLOT(XX,10.25,2)
      CALL PLOT(XX,10.,2)
      CALL PLOT(XX,10.25,2)
  302 CONTINUE
      DO 303 I=1,10
      YY=10.25-FLOAT(I)
      CALL PLOT(2.,YY,2)
      CALL PLOT(2.25,YY,2)
      CALL PLOT(2.,YY,2)
  303 CONTINUE
      CALL PLOT(1.00,0.25,3)
      CALL PLOT(0.60,0.25,2)
      CALL PLOT(0.60,8.25,2)
      CALL PLOT(1.00,8.25,2)
      CALL PLOT(1.00,0.25,2)
      CALL SYMBOL (0.88,0.45,0.12,10HPATTERN = ,90.,10)
      FNUM=NUMPAT
      CALL NUMBER(0.88,2.075,0.12,FNUM,90.,-1)
      CALL ZIP(1,3,18)
C
C        SORT CONTOURS FROM SMALLEST TO LARGEST
C
      NCON1=NCON-1
      DO 200 I=1,NCON1
      I1=I+1
      DO 201 J=I1,NCON
      IF(CONTUR(I).LE.CONTUR(J)) GO TO 201
      TEMP=CONTUR(I)
      CONTUR(I)=CONTUR(J)
      CONTUR(J)=TEMP
  201 CONTINUE
  200 CONTINUE
C
```

DOS FORTRAN IV 360N-FO-479 3-9   PLOT2   DATE   12/21/78   TIME   13.45.57

```
C
 1125 YCONA=1.0/SMAX
      DELTAX=SX/FLOAT(N-1)
      X(1)=0.0
      Y(1)=0.0
      RB(1) = A(1,1)
      DO 27 J=2,N
       RB(J)=A(J,1)
   27  X(J)=X(J-1)+DELTAX
      DELTAY=SY/FLOAT(M-1)
      DO 28 J=2,M
   28  Y(J)=Y(J-1)+DELTAY
      DO 118  K=2,M
       DO 30 J=1,N
       RA(J)=RB(J)
   30  RB(J)=A(J,K)
      DO 118  J=2,N
   35  ASSIGN 112 TO L
      RR=RA(J)
      XX=X(J)
      YY=Y(K-1)
   37 RL=RR
      XL=XX
      YL=YY
   39 IF(RL-RA(J-1))   41,40 ,40
   40 IF(RL-RB(J))42,50 ,50
   41 RL=RA(J-1)
      XL=X (J-1)
      YL= Y(K-1)
      GO TO 40
   42 RL=RB(J)
      XL=X (J)
      YL=Y(K)
      GO TO 50
   50 RS=RR
      XS=XX
      YS=YY
      IF(RS-RA(J-1)) 52, 52,53
```

DOS FORTRAN IV 360N-FO-479 3-9   PLOT2   DATE   12/21/78   TIME   13.45.57

```
      52 IF(RS-RB(J)) 60,60,54                                       00001090
      53 RS=RA(J-1)                                                  00001100
         XS=X (J-1)                                                  00001110
         YS =Y(K-1)                                                  00001120
         GO TO 52                                                    00001130
      54 RS=RB(J)                                                    00001140
         XS=X (J)                                                    00001150
         YS=Y (K)                                                    00001160
         GO TO 60                                                    00001170
      60 RM=RR                                                       00001180
         XM=XX                                                       00001190
         YM=YY                                                       00001200
         IF(RM-RS) 62, 62,61                                         00001210
      61 IF(RM-RL)70 ,62 ,62                                         00001220
      62 RM=RA(J-1)                                                  00001230
         XM=X (J-1)                                                  00001240
         YM=Y (K-1)                                                  00001250
         IF(RM-RS) 64,64,63                                          00001260
      63 IF(RM-RL) 70,64,64                                          00001270
      64 RM = RB(J)                                                  00001280
         XM=X (J)                                                    00001290
         YM=Y (K)                                                    00001300
      70 YCS=YS*YCONA                                                00001310
         YCM=YM*YCONA                                                00001320
         YCL=YL*YCONA                                                00001330
      71 YS=YS-SY                                                    00001340
         YM=YM-SY                                                    00001350
         YL=YL-SY                                                    00001360
      72  XCS=XS/SMAX                                                00001370
          XCM=XM/SMAX                                                00001380
          XCL=XL/SMAX                                                00001390
         JJ=1
      80 IF(JJ .GT. NCON) GO TO 110
         RC=CONTUR(JJ)
         IF ( RC .NE. RM ) GO TO 91                                  00001420
      81 IF ( RM .NE. RS ) GO TO 91                                  00001430
      82 IF ( RL .EQ. RM ) GO TO 100                                 00001440
      91 IF(RC-RS)100,95,92                                          00001450
```

```
   92 IF(RC-RM)96,93,94
   93 XPA=XCM
      YPA=YCM
      GO TO 99
   94 IF(RC-RL)106,103,110
   95 Q=0.0
      GO TO 97
   96 Q = (RC-RS)/(RM-RS)
   97 XPA = XCS-Q*(XCS-XCM)
      YPA = YCS-Q*(YCS-YCM)
   99 Q = (RC-RS)/(RL-RS)
      XPB = XCS-Q*(XCS-XCL)
      YPB = YCS-Q*(YCS-YCL)
      IF(RC-DASH) 10115,10115,10116
10115 XPB1=0.5*(XPA+XPB)
      YPB1=0.5*(YPA+YPB)
      IF(ABS (XPA-XPB1)-.001)5001,5002,5002
 5001 IF(ABS (YPA-YPB1)-.001)100,5002,5002
 5002 CALL PLOT(SCALE*XPA+2.,SCALE*YPA+0.25,3)
      CALL PLOT(SCALE*XPB1+2.,SCALE*YPB1+0.25,2)
      GO TO 100
10116 IF(ABS (XPA-XPB)-.001)5003,5004,5004
 5003 IF(ABS (YPA-YPB)-.001)100,5004,5004
 5004 CALL PLOT(SCALE*XPA+2.,SCALE*YPA+0.25,3)
      CALL PLOT(SCALE*XPB+2.,SCALE*YPB+0.25,2)
  100 JJ=JJ+1
      GO TO 80
  103 XPA = XCL
      YPA = YCL
      GO TO 99
  106 Q=(RC-RM)/(RL-RM)
      XPA=XCM-Q*(XCM-XCL)
      YPA=YCM-Q*(YCM-YCL)
      GO TO 99
  110  GO TO L,(112,118)
  112  ASSIGN 118 TO L
      RR   =RB(J-1)
      XX   =X (J-1)
```

DOS FORTRAN IV 360N-FO-479 3-9   PLOT2   DATE   12/21/78   TIME   13.45.57

```
      YY    =Y (K)                                        00001820
      GO TO 37                                            00001830
  118 CONTINUE                                            00001840
      CALL FACTOR(1.0)
      CALL PLOT(SCALE+6.,0.,23)
      CALL ZIP(3,3,18)
       RETURN                                             00001860
       END                                                00001870
```

```
C          SUBROUTINE PLOT3
C
C          PURPOSE:  TO DRAW A PERSPECTIVE VIEW OF A CONTOURED SURFACE.
C
C
C          DESCRIPTION OF PARAMETERS AND IMPORTANT VARIABLES:
C
C          N        - NUMBER OF DATA POINTS ALONG FIRST AXIS.
C
C          M        - NUMBER OF DATA POINTS ALONG THE SECOND AXIS.
C
C          NUMPAT   - PATTERN NUMBER (FOR LABELLING).
C
C          K        - CODE THAT TELLS WHETHER TO DRAW THE GRID LINES:
C                           K=1: ALONG THE N-DIMENSION ONLY.
C                           K=2: ALONG THE M-DIMENSION ONLY.
C                           K=3: ALONG BOTH DIMENSIONS.
C
C          DISTS    - DISTANCE FROM SURFACE TO EYE WHEN PERSPECTIVE IS
C                     CALCULATED -- SDISTS > 6 USUALLY WONT'T SHOW ANY
C                     DISTORTION DUE TO PARALLAX.
C
C          YAW      - (IN DEGREES) HOW FAR THE OBJECT IS TURNED AWAY FROM
C                     THE VIEWER.
C
C          PITCH    - (IN DEGREES) HOW THE SURFACE IS LOWERED OR RAISED AT
C                     THE FRONT EDGE.  (POSITIVE PITCH TENDS TO EXPOSE THE
C                     TOP OF THE FIGURE).
C
C          SIZE     - (IN INCHES) THE SIZE OF THE CUBE THAT ENCLOSES THE
C                     FIGURE.
C
C          KODE     - "HIDDEN LINE" SWITCH.  IF KODE=0 DO NOT DRAW HIDDEN
C                     LINES...IF KODE=1, ALL HIDDEN LINES ARE PLOTTED.
C
C          MGN      - WHETHER TO DRAW THE OUTLINE OF THE CUBE TO HELP ORIENT
C                     THE VIEWER.  MGN=0: DO NOT DRAW ANY OUTLINE OF THE
C                     CUBE.  MGN=1:  DRAW THE OUTLINE OF THE CUBE SEPARATE
```

```
C                    FROM THE FIGURE.  MGN=2:  DRAW THE OUTLINE OF THE
C                    CUBE SUPERIMPOSED ON THE SURFACE PLOT.  MGN=3:  DRAW
C                    ONLY THE THREE EDGES OF THE CUBE THAT MEET AT THE
C                    ORIGIN, SUPERIMPOSED ON THE SURFACE PLOT.
C
C        SCALF   - HOW TALL TO MAKE THE SURFACE RELATIVE TO THE HEIGHT
C                    OF THE CUBE.  SCALE=0: DO NOT SCALE THE DATA AT ALL
C                    BUT TRUST THE USER THAT THE DATA IS NOT SO HIGH THAT
C                    IT RUNS OFF THE PAPER.  SCALE=1: SCALE THE DATA SO
C                    THE TOP OF THE DATA JUST TOUCHES THE TOP OF THE CUBE.
C                    SCALE=0.3: SCALE THE DATA SO THE TOP OF THE SURFACE IS
C                    THREE TENTHS AS HIGH AS THE CUBE.
C
C
C
C        REMARKS.
C
C      I. IT IS VERY EXPENSIVE TO DRAW OPAQUE SURFACES, BECAUSE THE
C         PROGRAM HAS TO DETERMINE THE VISIBILITY OF EVERY POINT, THE
C         COMPUTER TIME DOUBLES OR TRIPLES...DEPENDING ON HOW MANY LINE
C         SEGMENTS ARE PARTIALLY VISIBLE.
C
C     II. THE CONTENTS OF ARRAY A ARE DESTROYED IN COMPUTATION.
C
C
C        COMMON BLOCKS REQUIRED:
C
C             COMMON /ARRAY/ A
C             COMMON /THREE6/ ANGA,ANGB,HV,D,SH,SV
C             COMMON /THREE7/ SL,SM,SN,CX,CY,CZ,QX,QY,QZ,SD
C
C
C        SUBROUTINE AND FUNCTION SUBPROGRAMS REQUIRED:
C             THREE2
C             THREE3
C             THREE4
C             THREE5
C             PLOT
```

```
DOS FORTRAN IV 360N-FO-479 3-9        MAINPGM          DATE   12/21/78     TIME    13.50.52

             C              FACTOR
             C              SYMBOL
             C              NUMBER
             C
             C
             C
             C          REFERENCE:HOWARD JESPERSON, IOWA STATE UNIVERSITY.
             C
             C          MODIFIED FOR USE AT VPI BY:  ROBERT D. KEPHART.
             C                                       S. R. KAUFFMAN
             C                                       W. L. STUTZMAN
             C                                       E. L. COFFEY
             C
             C
             C
 0001               SUBROUTINE PLOT3(N,M,NUMPAT)                                    00000320
             C                                                                      00000330
             C                                                                      00000370
 0002               COMMON /ARRAY/ A                                                00000389
 0003               COMMON /THREE6/ ANGA,ANGB,HV,D,SH,SV                            00000390
 0004               COMMON /THREE7/ SL,SM,SN,CX,CY,CZ,QX,QY,QZ,SD                   00000400
 0005               DIMENSION H(10),V(10),X(2),Y(2),Z(2),XP(8),A(151,151)           00000410
 0006               K=3                                                             00000420
 0007               SDISTS=6.0                                                      00000430
 0008               PITCH-30.                                                       00000440
 0009               YAW=45.                                                         00000450
 0010               SIZE=10.
 0011               KODE=0                                                          00000470
 0012               MGN=0                                                           00000480
 0013               SCALE=1.0
 0014               CALL FACTOR(1.0)
 0015               CALL PLOT(8.,.2,23)
 0016               CALL PLOT(.4,0.,2)                                              00000510
 0017               CALL PLOT(.4,8.,2)                                              00000520
 0018               CALL PLOT(0.,8.,2)                                              00000530
 0019               CALL PLOT(0.,0.,2)                                              00000540
 0020               CALL ZIP(3,3,18)
 0021               CALL SYMBOL(0.3,1.0  ,0.12,10HPATTERN = ,90.,10)
```

DOS FORTRAN IV 360N-FO-479 3-9   PLOT3   DATE   12/21/78   TIME   13.50.52

```
0022            FNUM=FLOAT(NUMPAT)
0023            CALL NUMBER(0.3,2.130 ,0.12,FNUM,90.,-1)
0024            CALL ZIP(1,3,18)
0025            CALL PLOT(1.5,-.2,23)
          C***** ********
0026            ANGA = (YAW+270.) * .0174532
0027            ANGB = PITCH * .0174532
0028            HV = SIZE
          C  DIRECTION COMPONENTS TO THE EYE.
0029             SL = -COS( ANGA ) * COS( ANGB )
0030             SM = -SIN( ANGA ) * COS( ANGB )
0031             SN = -SIN ( ANGB )
0032                 IF ( ABS( SN ) .NE. 1.0 ) GO TO 10
0033               WRITE(  6     ,    20  )
0034         20   FORMAT (  '1' , 20X,  20('*') , / '0', 'YOU ARE ATTEMPTING TO LO
              :K  STRAIGHT DOWN ( OR UP ) AT THE SURFACE ' )
0035                   GO TO 2150
0036         10  CONTINUE
0037            SD = 1.0 / SQRT( 1.0 - SN ** 2  )
0038             X(1) = L
0039             X(2) = N
0040             Y(1) = L
0041             Y(2) = M
0042            T=MAXO(M,N)
          C  FIND THE DIAGONAL OF THE "CUBE".
0043             D = M ** 2  +  N ** 2  +  T ** 2
0044             D  =  SQRT ( D )
0045             SCL = SDISTS * D
          C  COORDINATES OF YOUR EYE.
0046             CX = -SL * SCL
0047             CY = -SM * SCL
0048             CZ = -SN * SCL
          C  COORDINATES OF THE PROJECTION PLANE.
0049             QX = CX + D * SL
0050             QY = CY + D * SM
0051             QZ = CZ + D * SN
          C      PPPPPPPPPPPPPPPPPPPPPPPPPP
0052             GO TO 2060
```

```
            C      WRITE(6,100) CX,CY,CZ                                           00000910
            C      WRITE(6,100) QX,QY,QZ                                           00000920
 0053       100    FORMAT(1X,3F15.3)                                               00000930
 0054        2060 Z(2)=A(1,1)
 0055             Z(1)=A(1,1)
 0056             DO 1000 J=1,N
 0057             DO 1000 K=1,M
 0058             Z(1)=AMIN1(Z(1),A(J,K))
 0059             Z(2)=AMAX1(Z(2),A(J,K))
 0060        1000 CONTINUE
 0061             RANGE= (Z(2)-Z(1))                                               00000950
 0062              DOL=1.0                                                         00000960
 0063              IF(SCALE.NE.0)  DOL=T/RANGE*SCALE                               00000970
            C   SCALE THE SURFACE TO MAKE A "CUBE".                                00000980
 0064              DO 30 J = 1 , M                                                 00001000
 0065              DO 30 I = 1 , N                                                 00000990
 0066              A ( I , J ) = ( A ( I , J ) - Z ( 1 ) ) * DOL                  00001010
 0067          30 CONTINUE                                                         00001020
 0068              Z(1) = 0.0                                                      00001030
 0069              Z(2) = T                                                        00001040
 0070        2080 CALL THREE2  ( X, Y, Z, XP , H , V ,KODE)                       00001050
 0071              DO 2130 I = 1 , 8                                               00001060
 0072              H( I ) = ( (XP(I) - QX ) * SM - (  H(I) - QY ) * SL ) * SD     00001070
 0073              V ( I ) = (  V( I ) - QZ ) * SD                                 00001080
 0074        2130 CONTINUE                                                         00001090
 0075        2100 H(10)=H(1)
 0076             H(9)=H(1)
 0077             DO 1001 J=1,8
 0078             H(9)=AMIN1(H(9),H(J))
 0079             H(10)=AMAX1(H(10),H(J))
 0080        1001 CONTINUE
 0081        2120 V(9)=V(1)
 0082             V(10)=V(1)
 0083             DO 1002 J=1,8
 0084             V(9)=AMIN1(V(9),V(J))
 0085             V(10)=AMAX1(V(10),V(J))
 0086        1002 CONTINUE
 0087             IF( MGN .EQ. 0) GO TO 2140                                       00001120
```

```
      S=HV
      IF(MGN .EQ. 1) S=1.5
      SH = S/ (H(10)-H(9) )
      SV = S/ (V(10)-V(9) )
      SH = SIGN( AMIN1(SH,SV),SH )
      SV = SIGN(SH,SV)
      IF(MGN .EQ. 1) CALL PLOT (0.,2.,-3)
      CALL SYMBOL((H(1)-H(9))*SH,(V(1)-V(9))*SV,.14,'O',0.,1)
      CALL SYMBOL((H(3)-H(9))*SH,(V(3)-V(9))*SV,.14,'M',0.,1)
      CALL SYMBOL((H(2)-H(9))*SH,(V(2)-V(9))*SV,.14,'Z',0.,1)
      CALL SYMBOL((H(5)-H(9))*SH,(V(5)-V(9))*SV,.14,'N',0.,1)
        CALL PLOT(.03,.05,-3)
      CALL PLOT ( (H(1)-H(9))*SH, (V(1)-V(9))*SV,3)
      CALL PLOT ( (H(2)-H(9))*SH, (V(2)-V(9))*SV,2)
      CALL PLOT ( (H(1)-H(9))*SH, (V(1)-V(9))*SV,2)
      CALL PLOT ( (H(3)-H(9))*SH, (V(3)-V(9))*SV,2)
      CALL PLOT ( (H(1)-H(9))*SH, (V(1)-V(9))*SV,2)
      CALL PLOT ( (H(5)-H(9))*SH, (V(5)-V(9))*SV,2)
      IF( MGN .EQ. 3) GO TO 2139
      CALL PLOT ( (H(6)-H(9))*SH, (V(6)-V(9))*SV,2)
      CALL PLOT ( (H(2)-H(9))*SH, (V(2)-V(9))*SV,2)
      CALL PLOT ( (H(4)-H(9))*SH, (V(4)-V(9))*SV,2)
      CALL PLOT ( (H(3)-H(9))*SH, (V(3)-V(9))*SV,2)
      CALL PLOT ( (H(7)-H(9))*SH, (V(7)-V(9))*SV,2)
      CALL PLOT ( (H(5)-H(9))*SH, (V(5)-V(9))*SV,2)
      CALL PLOT ( (H(6)-H(9))*SH, (V(6)-V(9))*SV,2)
      CALL PLOT ( (H(8)-H(9))*SH, (V(8)-V(9))*SV,2)
      CALL PLOT ( (H(4)-H(9))*SH, (V(4)-V(9))*SV,2)
      CALL PLOT ( (H(8)-H(9))*SH, (V(8)-V(9))*SV,2)
      CALL PLOT ( (H(7)-H(9))*SH, (V(7)-V(9))*SV,2)
 2139  IF(MGN .NE. 1) GO TO 2140
      CALL PLOT (AINT((H(10)-H(9))*SH+2.),-2.05,23)
 2140 CALL THREE3(X,Y,N,M,H,V,K,KODE)
 2150 CONTINUE
      CALL PLOT(16.,0.,23)
      CALL ZIP(3,3,18)
         RETURN
      END
```

```
0001               SUBROUTINE THREE2 ( X, Y, Z, XP , H , V ,KODE)                    00001670
             C FIND THE CORNERS OF THE ROTATED CUBE.                                 00001680
             C                                                                       00001690
0002                DIMENSION X(2),Y(2),Z(2),H(10),V(10),XP(8)                       00001700
             C                                                                       00001710
0003           050  L = 0                                                            00001720
0004           070  DO 180 I = 1, 2                                                  00001730
             C                                                                       00001740
0005           090   DO 170 J = 1, 2                                                 00001750
             C                                                                       00001760
0006           110    DO 160 K = 1, 2                                                00001770
             C                                                                       00001780
0007           130     L = L + 1                                                     00001790
0008           140      CALL THREE4 ( X(I), Y(J), Z(K), XP( L ),                     00001800
                  1      H(L) , V( L ),KODE )                                        00001810
0009           160 CONTINUE                                                          00001820
0010           170 CONTINUE                                                          00001830
0011           180 CONTINUE                                                          00001840
0012           190 RETURN                                                            00001850
0013                END                                                              00001860
```

```
DOS FORTRAN IV 360N-FO-479 3-9          THREE:             DATE   12/21/78     TIME    13.53.14

0001           SUBROUTINE THREE4 ( X, Y, Z, XP , YP , ZP ,KODE)                     00001870
         C  FIND THE LOCATION OF A POINT IN THE ROTATED CUBE.                       00001880
0002            COMMON /THREE6/ ANGA , ANGB , HV , D, SH,SV                         00001890
0003            COMMON /THREE7/SL , SM , SN , CX , CY , CZ , QX , QY , QZ , SD      00001900
0004            SK = D / (  (X - CX ) * SL + ( Y - CY ) * SM + ( Z - CZ ) *SN)      00001910
0005            XP = CX + SK * ( X - CX )                                           00001920
0006            YP = CY + SK * ( Y - CY )                                           00001930
0007            ZP = CZ + SK * ( Z - CZ )                                           00001940
0008                  RETURN                                                        00001950
0009                           END                                                  00001960
```

```
DOS FORTRAN IV 360N-FO-479 3-9          THREE3            DATE   12/21/78      TIME     13.53.33

0001              SUBROUTINE THREE3 (X,Y,N,M,H,V,K,KODE)                          00001970
            C  DRAW THE FIGURE.                                                   00001980
0002               COMMON /THREE6/ ANGA , ANGB , HV , D, SH,SV                    00001990
0003              COMMON /THREE7/SL,SM,SN,CX,CY,CZ,QX,QY,QZ,SD                    00002000
            C                                                                     00002010
0004              DIMENSION X(2),Y(2),H(10),V(10),A(151,151)                      00002020
0005              COMMON /ARRAY/ A                                                00002025
0006              INTEGER UP , DOWN , PEN , P , Q                                 00002030
0007                INTEGER P1 , PO                                               00002040
            C                                                                     00002050
            C                                                                     00002060
            C                                                                     00002070
0008                   END = 1.0/8.0
            C  CAN USE 1 / 32 OR  1 / 64 FOR FINER INTERPOLATION                  00002090
            C                                                                     00002100
            C                                                                     00002110
0009              UP   = 3                                                        00002120
0010              DOWN = 2                                                        00002130
0011              SH =  HV / ( H ( 10 ) - H ( 9 ) )                               00002140
0012              SV = HV / ( V ( 10 ) - V ( 9 ) )                                00002150
0013              SH = SIGN(AMIN1(SH,SV),SH)                                      00002160
0014              SV = SIGN(SH,SV)                                                00002170
0015               MM = M                                                         00002180
0016               NN = N                                                         00002190
            C 080 IF(K-1) 100,120,100
            C                                                                     00002210
            C 100 IF(K-3) 1110,120,1110
            C                                                                     00002230
            C  DRAW LINES ALONG THE Y-AXIS                                         00002240
0017          120 CONTINUE                                                        00002250
0018              L = 0                                                           00002260
0019              LD = 1                                                          00002270
0020                DD = 0.5 * LD                                                 00002280
            C                                                                     00002290
0021          140  DO 1060 J = 1, M                                               00002300
0022              Q    = 0                                                        00002310
0023              YJ = J                                                          00002320
0024          160  DO 1030 I = 1, NN                                              00002330
```

```
C
            L = L + LD
          XI = L
         CALL THREE5 (     XI ,YJ , N , M , P ,KODE)
          PEN = UP
              IF ( P ) 510 , 520 , 530
     510 CONTINUE
             IF ( Q ) 540 , 550 , 540
     520 CONTINUE
             IF ( Q ) 610 , 1020 , 610
     530  CONTINUE
             IF ( Q ) 540 , 550 , 540
     540  CONTINUE
            PEN = DOWN
          GO TO 170
     550  CONTINUE
              IF ( I .EQ. 1 ) GO TO 170
             DI = DD
         TO = L - LD
            T = TO + DI
            P1 = Q
     560  IF (  ABS( DI )  .LT.  END  )  GO TO 570
         CALL THREE5 (T,YJ,N,M,PO,KODE)
            DI = DI * 0.5
          IF ( PO  .EQ.  0  )  GO TO 565
             TO = T
            P1 = PO
            T =  T - DI
            GO TO 560
     565  T = T + DI
            GO TO 560
     570  CONTINUE
          T = TO
         IF ( P1 * P  )  170 , 170 , 580
     580 CONTINUE
     590 CONTINUE
         ZP = A(L-LD,J)+(T-L+LD)*(A(L,J)-A(L-LD,J))/LD
         CALL THREE4(T,YJ,ZP,XP,MH,VV,KODE)
```

```
0062            HH = ( ( XP-QX)*SM- (HH - QY )*SL ) * SD                  00002720
0063            VV =  ( VV - QZ ) * SD                                    00002730
0064             HH = ( HH - H(9) ) * SH                                  00002740
0065             VV = ( VV - V(9) ) * SV                                  00002750
0066             CALL PLOT ( HH , VV , PEN )                              00002760
0067       600  PEN = 5 - PEN                                             00002770
0068            GO TO 170                                                 00002780
0069       610 CONTINUE                                                   00002790
0070              PEN = DOWN                                              00002800
0071               DI = DD                                                00002810
0072             TO = L - LD                                              00002820
0073            T = TO + DI                                               00002830
0074               P1 = Q                                                 00002840
0075       620  IF ( ABS( DI )  .LT.  END  )  GO TO 630                   00002850
0076           CALL THREE5 (T,YJ,N,M,PO,KODE)
0077            DI = DI * 0.5                                             00002870
0078             IF ( PO  .EQ.  0  )  GO TO 625                           00002880
0079             TO = T                                                   00002890
0080            P1 = PO                                                   00002900
0081            T = T + DI                                                00002910
0082             GO TO 620                                                00002920
0083       625   T = T - DI                                               00002930
0084               GO TO 620                                              00002940
0085       630   CONTINUE                                                 00002950
0086             T = TO                                                   00002960
0087               IF (  P1 * Q )  600 , 600 , 590                        00002970
0088       170    CALL THREE4 ( XI , YJ , A( L, J ), XP , HH ,VV ,KODE)   00002980
0089            VV =  ( VV - QZ ) * SD                                    00002990
0090            HH = ( ( XP-QX)*SM- (HH - QY )*SL ) * SD                  00003000
0091       190    HH = ( HH - H(9) ) * SH                                 00003010
0092       200    VV = ( VV - V(9) ) * SV                                 00003020
0093             ALL PLOT (  HH , VV , PEN )                              00003030
0094      1020  W = P                                                     00003040
0095      1030  CONTINUE                                                  00003050
         C                                                                00003060
         C                                                                00003070
0096            L = L + LD                                                00003080
0097            LD = -LD                                                  00003090
```

```
0098                DD =-DD                                        00003100
             C                                                     00003110
0099          1060 CONTINUE                                        00003120
             C                                                     00003130
             C                                                     00003140
             C1090 IF(K-3) 2060,1110,2060
             C                                                     00003160
             C  DRAW LINES ALONG THE X-AXIS.                       00003170
0100          1110 CONTINUE                                        00003180
             C                                                     00003190
0101               L = 0                                           00003200
0102               LD = 1                                          00003210
0103                 DD = 0.5 * LD                                 00003220
0104          1140 DO 2040 I = 1 , N                               00003230
0105                   XI = I                                      00003240
0106                Q = 0                                          00003250
0107          1160    DO 2020 J = 1 , MM                           00003260
0108                       L = L + LD                              00003270
0109                     YJ = L                                    00003280
0110               CALL THREE5 (XI,YJ,N,M,P,KODE)
0111                PEN = UP                                       00003300
0112                      IF ( P ) 1510 , 1520 , 1530              00003310
0113          1510 CONTINUE                                        00003320
0114                   IF ( Q ) 1540 , 1550 , 1540                 00003330
0115          1520   CONTINUE                                      00003340
0116                  IF ( Q )  1610 , 2010 , 1610                 00003350
0117         1530   CONTINUE                                       00003360
0118                    IF ( Q ) 1540 , 1550 , 1540                00003370
0119          1540   CONTINUE                                      00003380
0120                  PEN = DOWN                                   00003390
0121                 GO TO 1170                                    00003400
0122          1550        CONTINUE                                 00003410
0123                 IF ( J .EQ.  1  )  GO TO 1170                 00003420
0124                  DI = DD                                      00003430
0125               TO=L-LD
0126                T = TO + DI                                    00003450
0127                P1 = Q                                         00003460
0128          1560  IF (  ABS( DI )  .LT.  END  )  GO TO 1570      00003470
```

```
           CALL THREE5 (XI,T,N,M,PO,KODE)
             DI = DI * 0.5                                  00003490
           IF ( PO  .EQ.  0  )  GO TO 1565                  00003500
              TO = T                                        00003510
            P1 = PO                                         00003520
            T =  T - DI                                     00003530
              GO TO 1560                                    00003540
      1565 T = T + DI                                       00003550
               GO TO 1560                                   00003560
      1570   CONTINUE                                       00003570
            T = TO                                          00003580
           IF ( P1 * P  )  1170 , 1170 , 1580               00003590
      1580 CONTINUE                                         00003600
      1590 CONTINUE                                         00003610
           ZP=A(I,L-LD) + (T-L+LD) * (A(I,L) - A(I,L-LD));/LD  00003620
              CALL THREE4 ( XI , T , ZP , XP,HH,VV ,KODE)   00003630
           HH = ( ( XP-OX)*SM- (HH - QY )*SL ) * SD         00003640
           VV =   ( VV - QZ ) * SD                          00003650
             HH = ( HH - H(9) ) * SH                        00003660
             VV = ( VV - V(9) ) * SV                        00003670
            CALL PLOT ( HH , VV , PEN )                     00003680
      1600 PEN = 5 - PEN                                    00003690
             GO TO 1170                                     00003700
      1610  CONTINUE                                        00003710
              PEN = DOWN                                    00003720
               DI = DD                                      00003730
             TO = L - LD                                    00003740
            T = TO + DI                                     00003750
                P1 = Q                                      00003760
      1620  IF ( ABS( DI )  .LT.  END  )  GO TO 1630        00003770
           CALL THREE5 (XI,T,N,M,PO,KODE)
              DI = DI * 0.5                                 00003790
             IF ( PO .EQ. 0 )  GO TO 1625                   00003800
              TO = T                                        00003810
            P1 = PO                                         00003820
             T = T + DI                                     00003830
             GO TO 1620                                     00003840
      1625   T = T - DI                                     00003850
```

```
0167               GO TO 1620                                       00003860
0168        1630   CONTINUE                                         00003870
0169              T = TO                                            00003880
0170              IF ( PI * Q ) 1600 , 1600 , 1590                  00003890
0171        1170  CALL THREE4 ( XI, YJ, A( I, L ), XP , HH ,VV ,KODE)  00003900
0172             HH = ( ( XP-QX)*SM- (HH - QY )*SL ) * SD           00003910
0173             VV =  ( VV - QZ ) * SD                             00003920
0174        1180     HH = ( HH - H(9) ) * SH                        00003930
0175        1190     VV = ( VV - V(9) ) * SV                        00003940
0176               CALL PLOT (  HH , VV ,  PEN  )                   00003950
0177        2010 Q = P                                              00003960
0178        2020  CONTINUE                                          00003970
          C                                                         00003980
0179             L = L + LD                                         00003990
0180             LD = - LD                                          00004000
0181                DD =-DD                                         00004010
0182        2040 CONTINUE                                           00004020
          C                                                         00004030
0183        2060 CONTINUE                                           00004040
          C                                                         00004050
0184        2130 RETURN                                             00004060
0185              END                                               00004070
```

```
C
C
C        SUBROUTINE THREES
C
C
C        PURPOSE: TO SEE IF A POINT ON THE PROJECTED THREE DIMENSIONAL
C        FIGURE IS VISIBLE.
C
C
C        USAGE:
C
C        CALL THREES(XI,YJ,M,N,P,KODE)
C
C
C        DESCRIPTION OF PARAMETERS:
C
C
C        XI       - ABSCISSA OF THE PROJECTED POINT
C
C        YJ       - ORDINATE OF THE PROJECTED POINT
C
C        M        - NUMBER OF HORIZONTAL POINTS
C
C        N        - NUMBER OF VERTICAL POINTS
C
C        P        - PLOT CODE:  IF P = -1  INVISIBLE TO VISIBLE
C                                       1  VISIBLE TO INVISIBLE
C                                       0  VISIBLE TO VISIBLE OR
C                                          INVISIBLE TO INVISIBLE.
C
C        KODE     - HIDDEN LINE CODE  (SEE SUBROUTINE PLOT3)
C
C
C
C        COMMON BLOCKS REQUIRED:
C
C        COMMON /ARRAY/ A
C        COMMON /THREE6/ ANGA,ANGB,HV,D,SH,SV
```

```
C         COMMON /THREE7/ SL,SM,SN,CX,CY,CZ,QX,QY,QZ,SD
C
C
C         SUBROUTINE AND FUNCTION SUBPROGRAMS REQUIRED:NONE.
C
C
C
            SUBROUTINE THREE5 (XI,YJ,M,N,P,KODE)
            DIMENSION Z(151,151)
             COMMON /THREE6/ ANGA , ANGB , HV , D, SH,SV                   00004100
            COMMON /THREE7/SL,SM,SN,CX,CY,CZ,QX,QY,QZ,SD                    00004110
            COMMON /ARRAY/ Z
             INTEGER  CUM , CNT , P                                         00004120
               REAL I , J , II , JJ                                         00004130
            IF( KODE .EQ. 1) GO TO 78                                       00004150
            IR = XI                                                         00004160
            JC = YJ                                                         00004170
              ZB = Z ( IR , JC )                                            00004180
            IF (  XI  .EQ.  IR  )  GO TO 2                                  00004190
             ZB = Z(IR ,JC ) + ( XI - IR) * (Z(IR + 1 , JC ) - Z( IR ,JC))  00004200
                 GO TO 4                                                    00004210
         2   IF ( YJ .EQ.  JC )       GO TO 4                               00004220
             ZB = Z(IR , JC) + (YJ-JC)*(Z(IR,JC+1) - Z(IR,JC ))             00004230
         4   CONTINUE                                                       00004240
             XEND = 0.0                                                     00004250
             DX = 0.0                                                       00004260
             YMULT = 0.0                                                    00004270
             ZMULT = 0.0                                                    00004280
               IF (XI   .EQ.  CX  ) GO TO 10                                00004290
            YMULT = (YJ - CY ) /  (XI - CX )                                00004300
               ZMULT = ( ZB - CZ ) / ( XI - CX )                            00004310
            DX = 1.0                                                        00004320
              XEND = M + 1                                                  00004330
              IF ( XI  .LT.  CX)  GO TO 10                                  00004340
              DX = -1.0                                                     00004350
             XEND = 0.0                                                     00004360
         10 CONTINUE                                                        00004370
            YEND = 0.0                                                      00004380
```

```
      DY = 0.0
       XMULT = 0.0
        IF ( YJ  .EQ.  CY  )  GO TO 20
      XMULT = ( XI - CX ) / (YJ - CY )
        IF (  ZMULT  .EQ.  0.0  )  ZMULT=(ZB - CZ ) / ( YJ - CY )
      DY = 1.0
      YEND = N + 1
      IF ( YJ  .LT.  CY )  GO TO 20
      DY = -1.0
      YEND = 0.0
   20 CONTINUE
     CUM = 0
     CNT = 0
     P = 0
      XB = XI
     YB = YJ
   30 CONTINUE
     II = AINT( XB )
     JJ = AINT( YB )
     XSTEP = DX
     YSTEP = DY
       IF ( XB  .EQ.  II )  GO TO 40
       IF ( DX  .LT.  0.0 )  XSTEP = 0.0
     GO TO 45
   40 IF ( YB  .EQ.  JJ )  GO TO 45
       IF ( DY  .LT.  0.0 )  YSTEP = 0.0
   45 CONTINUE
       I = II + XSTEP
       J = JJ + YSTEP
       IF ( I  .EQ.  XEND )  GO TO 80
       IF ( J  .EQ.  YEND )  GO TO 80
     XB = CX + XMULT * ( J - CY )
     YB = CY + YMULT * ( I - CX )
       IF ( DX  .LT.  0.0 )  GO TO 55
       IF ( XB  .LT.  I  )  GO TO 60
   50 XB = I
      GO TO 65
   55 IF ( XB  .LT.  I )  GO TO 50
```

```
      60 YB = J
      65  CONTINUE
         ZB = CZ + ZMULT * ( XB - CX )
         IR = I
         JC = J
           IF ( YB  .NE.  J )  GO TO 70
          IDX = I - DX
         ZS = Z( IR, JC ) - DX * ( XB - I ) * (Z(IDX,JC) - Z(IR,JC))
         GO TO 75
   70    JDY=J-DY
         ZS = Z( IR,JC ) - DY * ( YB-J ) * (Z( IR,JDY ) - Z( IR,JC ) )
      75 CONTINUE
         SGN = 1
         IF (  ZB  .LT.  ZS  )  SGN = -1
         CUM = CUM + SGN
         CNT = CNT + 1
         IF (  IABS ( CUM)  .EQ.  CNT  )  GO TO 30
          GO TO 90
    78   P=1
         GO TO 95
      80 CONTINUE
         P = 1
             IF ( CUM ) 84 , 86 , 90
      84 P = -1
                GO TO 90
      86  CONTINUE
                 IF ( ZB  .LE.  CZ  )  GO TO 90
          P = -1
      90  CONTINUE
    95     RETURN
         END
```

APPENDIX B

THE SHUTTLE IMAGING RADAR ANTENNA SPECIALIZED FAR-FIELD PROGRAM

```
C          DEFORMED ANTENNA FAR-FIELD PATTERN
C
C
           COMMON /ARRAY/ MEL,NEL,PX,PY,K1,K2,MSECT,NSECT
           COMPLEX AF
           REAL K0,K1,K2
           DIMENSION IOA(1000)
           INTEGER TITLE(20)
           REAL A1(20,10),A2(20,10),P3(20,10),P4(20,10),WARP(231),
          $Z0(20,10),A9(20,10),A0(20,10)
           DATA EOF / '$EOF'/
         1 CONTINUE
           READ(1,10,END=999) (TITLE(J),J=2,19)
        10 FORMAT(20A4)
           WRITE(3,200) (TITLE(J),J=2,19)
       200 FORMAT(1H1,20A4)
           WRITE(2,201) (TITLE(J),J=2,19)
       201 FORMAT(20A4)
           TITLE(1)=76*65536+16448
           CALL DKWRIT(TITLE,2030,ICODE,&900)
           ICHAR=0
           READ(1,11) FREQ
        11 FORMAT(8F10.0)
           K0=2.*3.14159265E0*FREQ/30.0
           READ(1,12) NX,NY
        12 FORMAT(8I5)
           READ(1,11) SX,SY
           K1=K0*SX
           K2=K0*SY
           READ(1,11) PXD,PYD
           PX=PXD*0.01745329E0
           PY=PYD*0.01745329E0
           READ(1,12) MSECT,NSECT
           DO 20 J=1,NSECT
           DO 20 I=1,MSECT
        20 READ(1,11) A1(I,J),A2(I,J),P3(I,J),P4(I,J)
           NWARP=(MSECT+1)*(NSECT+1)
           READ(1,11) (WARP(N),N=1,NWARP)
```

```
0035          MEL=NX/MSECT
0036          NEL=NY/NSECT
0037          P5=MEL*PX
0038          P6=NEL*PY
0039          DX=SX*MEL
0040          DY=SY*NEL
0041          DX1=0.5/DX
0042          DY1=0.5/DY
        C........DETERMINE WARP PARAMETERS
0043          I=0
0044          DO 40 N=1,NSECT
0045          DO 40 M=1,MSECT
0046          I=I+1
0047          NO=I-1+N
0048          F0=WARP(NO)
0049          F1=WARP(NO+1)
0050          F2=WARP(NO+1+MSECT)
0051          F3=WARP(NO+2+MSECT)
0052          G0=0.25*(F0+F1+F2+F3)
0053          G1=DX1*(-F0+F1-F2+F3)
0054          G2=DY1*(-F0-F1+F2+F3)
0055          Z0(M,N)=G0
0056          A9(M,N)=ATAN(G1)
0057          A0(M,N)=ATAN(G2)
0058       40 CONTINUE
0059          READ(1,11) PHID
0060          PHI=PHID*0.01745329E0
0061          READ(1,11)STARTD,STOPD,DELTAD
0062          D6=SX*MEL
0063          D7=SY*NEL
0064          NSTEP=(STOPD-STARTD)/DELTAD+1.5
0065          DO 50 I=1,NSTEP
0066          THETAD=STARTD+(I-1)*DELTAD
0067          THETA=THETAD*0.01745329E0
0068          U1=ARCOS(SIN(THETA)*COS(PHI))
0069          V1=ARCOS(SIN(THETA)*SIN(PHI))
0070          AF=(0.,0.)
0071          Y0=-FLOAT(NSECT+1)/2.0*D7
```

```
0072              DO 60 N=1,NSECT
0073              YO=YO+D7
0074              XO=-FLOAT(MSECT+1)/2.0*D6
0075              DO 60 M=1,MSECT
0076              XO=XO+D6
0077              U=COS(U1-A9(M,N))+P3(M,N)/K1
0078              V=COS(V1-A0(M,N))+P4(M,N)/K2
0079              UV=U*U+V*V
0080              IF(UV.GE.1.0) GO TO 60
0081              W=SQRT(1.-UV)
0082              P=KO*(XO*U+YO*V+ZO(M,N)*W)+M*P5+N*P6+A2(M,N)
0083              AF=AF+A1(M,N)*PAT(U,V)*CMPLX(COS(P),SIN(P))
0084           60 CONTINUE
0085              DAF=CABS(AF)
0086              IF(DAF.LT.1E-5) DAF=1E-5
0087              DAF=20.*ALOG10(DAF)
0088              IDAF=IFIX(ABS(DAF*10.0))
         C        WRITE(3,100) PHID,THETAD,IDAF
0089          100 FORMAT(5X,F8.3,5X,F8.3,5X,I3)
0090              ICHAR=ICHAR+4
0091              CALL I4TOA4(IDAF,IOA(ICHAR/4+1))
0092              IF(ICHAR.LT.2000) GO TO 50
0093              IOA(1)=(ICHAR+4)*65536+16448
0094              CALL DKWRIT(IOA,2030,ICODE,&900)
0095              ICH=ICHAR/4+1
0096              WRITE(3,500) (IOA(J),J=1,ICH)
0097          500 FORMAT(2X,Z8,2X,100(20(A4,1X)/12X))
0098              WRITE(2,501) (IOA(J),J=2,ICH)
0099          501 FORMAT(20A4)
0100              WRITE(3,300) ICODE,ICHAR
0101          300 FORMAT(3X,'*** ICODE = ',I3,'   ICHAR = ',I5,' ***')
0102              ICHAR=0
0103           50 CONTINUE
0104              IF(ICHAR.EQ.0) GO TO 1
0105              IOA(1)=(ICHAR+4)*65536+16448
0106              CALL DKWRIT(IOA,2030,ICODE,&900)
0107              ICH=ICHAR/4+1
0108              WRITE(3,500) (IOA(J),J=1,ICH)
```

```
DOS FORTRAN IV 360N-FO-479 3-9          MAINPGM          DATE   12/21/78     TIME    16.18.34

            WRITE(2,501) (IOA(J),J=2,ICH)
            WRITE(3,300) ICODE,ICHAR
            GO TO  1
        900 CONTINUE
            WRITE(3,400)
        400 FORMAT('  END OF EXTENT ON TEK I/O FILE 2')
            STOP
        999 CONTINUE
            CALL DKWRIT(EOF,4,ICODE,&900)
            WRITE(3,70)
         70 FORMAT(3X,'*** END OF FILE WRITTEN ON TEK I/O FILE 2 ***')
            STOP
            END
```

```
DOS FORTRAN IV 360N-FO-479 3-9          PAT                 DATE   12/21/78     TIME    16.19.00

           REAL FUNCTION PAT(U,V)
           COMMON /ARRAY/ MEL,NEL,P1,P2,K1,K2,MSECT,NSECT
           REAL K1,K2
           PAT=1.0
           IF(U.EQ.0) GO TO 2050
           P7=(K1*U+P1)/2.0
           PAT=SIN(MEL*P7)/(MEL*SIN(P7))
      2050 IF(V.EQ.0) GO TO 2080
           P7=(K2*V+P2)/2.0
           PAT=PAT*SIN(NEL*P7)/(NEL*SIN(P7))
      2080 PAT=PAT/(MSECT*NSECT)
           RETURN
           END
```

```
  LOC  OBJECT CODE    ADDR1 ADDR2  STMT   SOURCE STATEMENT                              DOS/VS ASSEMBLER REL 33.0 16

000000                                  1 I4TOA4   START 0
                                        2 *
                                        3 *        CONVERT A POSITIVE INTEGER =< 9999 TO A4 FORMAT
                                        4 *
                                        5 *        CALL I4TOA4(NUM,IOA)
                                        6 *             NUM = INPUT INTEGER
                                        7 *             IOA = OUTPUT CHARACTER STRING (A4)
                                        8 *
                                        9 *
                              00000    10          USING *,2
000000 90E5 D00C              0000C    11          STM   14,5,12(13)
000004 182F                            12          LR    2,15
000006 5830 1000              00000    13 BEGIN    L     3,0(0,1)       ADDR OF NUM
00000A 5840 1004              00004    14          L     4,4(0,1)       ADDR OF IOA
00000E 5850 3000              00000    15          L     5,0(0,3)       VALUE OF NUM
000012 4E50 2038              00038    16          CVD   5,DEC          CONVERT TO DECIMAL
000016 F337 2040 2038 00040 00038      17          UNPK  ZONE(4),DEC
00001C D300 2043 2044 00043 00044      18          MVZ   ZONE+3(1),FF   CHANGE SIGN ZONE TO X'F'
000022 D203 4000 2040 00000 00040      19          MVC   0(4,4),ZONE    MOVE OUTPUT STRING TO IOA
                                       20 *
                                       21 *
000028 9825 D01C              0001C    22          LM    2,5,28(13)     RESTORE REGISTERS
00002C 92FF D00C        0000C          23          MVI   12(13),X'FF'
000030 07FE                            24          BCR   15,14
000038                                 25 DEC      DS    1D
000040                                 26 ZONE     DS    1F
000044 F0                              27 FF       DC    X'F0'
                              00000    28          END   I4TOA4
```

```
LIST
100 REM
101 REM ----- DEFORMED ANTENNA FAR-FIELD PATTERN --------------------
102 REM
103 REM       WRITTEN BY: E. L. COFFEY, SENIOR ENGINEER
104 REM                   PHYSICAL SCIENCE LABORATORY
105 REM                   BOX 3548
106 REM                   LAS CRUCES, NEW MEXICO  88003
107 REM                   (505) - 522 - 4400   EXT. 242
108 REM             DATE: 25 JULY 1977
109 REM
110 REM     IN SUPPORT OF: NASA/JSC
111 REM
112 REM ------------------------------------------------------------
120 PRINT "L***PSL DEFORMED ANTENNA FAR-FIELD PATTERN PROGRAM***"
150 PRINT "JENTER FREQUENCY IN GHZ: ";
160 INPUT F
165 K0=2*PI*F/30
170 PRINT "JENTER TOTAL NUMBER OF ELEMENTS (NX,NY): ";
180 INPUT N1,N2
190 PRINT "JENTER INTERELEMENT SPACING (SX,SY) IN CM: ";
200 INPUT S1,S2
203 K1=K0*S1
206 K2=K0*S2
210 PRINT "JENTER INTERELEMENT PHASE SHIFT (PXD,PYD) IN DEGREES: ";
220 INPUT P1,P2
221 P1=P1/57.29577951
222 P2=P2/57.29577951
228 DELETE W,Z0,A9,A0
230 PRINT "JENTER NUMBER OF SUBARRAY DIVISIONS (MSECT,NSECT): ";
240 INPUT M1,M2
241 DELETE A1,A2,P3,P4
242 DIM A1(M1,M2),A2(M1,M2),P3(M1,M2),P4(M1,M2)
250 PRINT "JENTER EXCITATION COEFFICIENTS FOR EACH SUBARRAY"
255 PRINT "(AMAG,APHS,PHSX,PHSY):"
```

```
257 M3=(M1+1)*(M2+1)
259 DIM W(M3),Z0(M1,M2),A9(M1,M2),A0(M1,M2)
260 FOR J=1 TO M2
270 FOR I=1 TO M1
280 PRINT USING 290:I,J
290 IMAGE "(",2D,",",2D,"):  ",S
300 INPUT A1(I,J),A2(I,J),P3(I,J),P4(I,J)
310 NEXT I
320 NEXT J
321 P3=P3/57.29577951
322 P4=P4/57.29577951
323 A2=A2/57.29577951
330 PRINT "JENTER WARP ARRAY:"
335 INPUT W
340 REM--PRELIMINARY CALCULATIONS
350 N3=N1/M1
360 N4=N2/M2
370 P5=N3*P1
380 P6=N4*P2
390 X1=S1*N3
400 Y1=S2*N4
410 D1=0.5/X1
420 D2=0.5/Y1
430 REM--DETERMINE WARP PARAMETERS
440 I=0
450 FOR N=1 TO M2
460 FOR M=1 TO M1
470 I=I+1
480 N0=I-1+N
490 F0=W(N0)
500 F1=W(N0+1)
510 F2=W(N0+1+M1)
520 F3=W(N0+2+M1)
530 G0=0.25*(F0+F1+F2+F3)
540 G1=D1*(-F0+F1-F2+F3)
```

```
550 G2=D2*(-F0-F1+F2+F3)
560 Z0(M,N)=G0
570 A9(M,N)=ATN(G1)
580 A0(M,N)=ATN(G2)
590 NEXT M
600 NEXT N
610 REM--PRELIMINARIES COMPLETED
613 PRINT "JENTER TITLE: ";
614 INPUT A$
620 B1=114.6/(N1*S1)*(30/F)
630 B2=114.6/(N2*S2)*(30/F)
710 PRINT "JPRELIMINARIES COMPLETED"
722 PRINT USING 723:B1,B2
723 IMAGE "JBEAMWIDTH IS NOMINALLY ", 2D.3D, " BY ",2D.3D," DEGREES"
725 PRINT "JENTER PHI = CONSTANT(DEGREES): ";
726 INPUT P8
727 P8=P8/57.29577951
730 PRINT "JENTER START,STOP AND PLOT INCREMENT IN DEGREES: ";
740 INPUT T7,T8,T9
741 T7=T7/57.29577951
742 T8=T8/57.29577951
743 T9=T9/57.29577951
1000 REM-- PATTERN CALCULATION ROUTINE-------------------------------
1001 REM
1002 REM
1003 REM
1004 REM
1005 REM ---------------------------------------------------------
1006 REM
1007 REM
1008 REM
1010 D6=S1*N3
1020 D7=S2*N4
1021 T0=INT((T8-T7)/T9+1.5)
1022 DELETE T1,T3
```

```
1023 DIM T1(T0),T3(T0)
1025 T2=0
1030 FOR T=T7 TO T8 STEP T9
1032 T2=T2+1
1035 U1=ACS(SIN(T)*COS(P8))
1040 V1=ACS(SIN(T)*SIN(P8))
1050 A4=0
1060 A5=0
1070 Y0=-(M2+1)/2*D7
1080 FOR N=1 TO M2
1090 Y0=Y0+D7
1100 X0=-(M1+1)/2*D6
1110 FOR M=1 TO M1
1120 X0=X0+D6
1130 U=COS(U1-A9(M,N))+P3(M,N)/K1
1140 V=COS(V1-A0(M,N))+P4(M,N)/K2
1150 W1=SQR(1-(U↑2+V↑2))
1160 P=K0*(X0*U+Y0*V+Z0(M,N)*W1)+M*P5+N*P6+A2(M,N)
1170 GOSUB 2000
1180 A4=A4+COS(P)*A6*A1(M,N)
1190 A5=A5+SIN(P)*A6*A1(M,N)
1200 NEXT M
1210 NEXT N
1220 A7=SQR(A4↑2+A5↑2)
1222 A8=0
1223 IF ABS(A4/A7)=>1 THEN 1240
1230 A8=ACS(A4/A7)
1240 IF A7>1.0E-5 THEN 1260
1250 A7=1.0E-5
1260 A7=20*LGT(A7)
1262 T1(T2)=T
1264 T3(T2)=A7
1270 PRINT USING 1280:T*57.29577951,A7,A8*57.29577951
1280 IMAGE 3D.3D,2X,4D.3D,2X,4D.2D
1290 NEXT T
```

```
1300 GOSUB 3000
1310 FOR J=1 TO 10
1311 FOR <=1 TO 5
1312 NEXT K
1313 PRINT "G";
1314 NEXT J
1320 END
2000 REM -- ARRAY FACTOR SUBROUTINE A6(U,V)
2010 A6=1
2020 IF U=0 THEN 2050
2030 P7=(K1*U+P1)/2
2040 A6=SIN(N3*P7)/(N3*SIN(P7))
2050 IF V=0 THEN 2080
2060 P7=(K2*V+P2)/2
2070 A6=A6*SIN(N4*P7)/(N4*SIN(P7))
2080 A6=A6/(M1*M2)
2090 RETURN
3000 REM -- PLOT IT --
3001 PRINT "PRESS RETURN TO PLOT FIGURE";
3002 INPUT Q$
3005 PAGE
3010 VIEWPORT 10,120,10,90
3011 W1=(INT(ABS(T7)*57.29577951/10)*10+10)*SGN(T7)
3012 W2=(INT(ABS(T9)*57.29577951/10)*10+10)*SGN(T9)
3020 WINDOW W1,W2,-30,0
3030 AXIS 2,5,W1,-30
3040 AXIS 2,5,W2,0
3045 T1=T1*57.29577951
3050 MOVE T1(1),T3(1)
3060 DRAW T1,T3
3070 HOME
3080 PRINT A$
3085 COPY
3090 RETURN
```

APPENDIX C

RAY TRACING PROGRAM

```
LIST
1 GO TO 100
2 IF J0=1 THEN 2
3 RETURN
4 J0=1
5 HOME
6 PRINT "R = ";R;" ANGLE = ";A1;" X0,Y0: ";X0;",";Y0
7 END
8 J0=0
9 GO TO 2
12 R=0
13 X0=X9
14 Y0=Y9
15 GO TO 130
16 REM -- ROUTINE TO TRANSFER MODEL TO TAPE
17 GOSUB 3500
18 RETURN
100 REM -------------------RAY TRACE PROGRAM--------------------------
101 REM
102 REM        WRITTEN BY: E. L. COFFEY, III
103 REM              DATE: APRIL 25, 1978
104 REM
105 REM ---------------------------------------------------------------
106 REM
110 INIT
111 SET KEY
112 SET DEGREES
113 DIM M2(100),B2(100),L2(100),U2(100),G1(100),G2(100)
114 PAGE
115 R=0
116 N2=0
120 GOSUB 2000
130 WINDOW 0,130,0,100
140 VIEWPORT 13,130,10,100
150 PRINT "LENTER STARTING ANGLE IN DEGREES: ";
```

```
151 INPUT A1
152 D1=COS(A1)
153 D2=SIN(A1)
170 M=D2/D1
180 B=Y0-M*X0
190 PAGE
200 GOSUB 3000
210 GOSUB 1500
219 R=R+R1
220 DRAW X,Y
230 GOSUB 2500
240 GO TO 210
1000 REM -------------------------------------------------------------------
1010 REM -- SUBROUTINE TO CALCULATE WHERE RAY INTERSECTS BOUNDARY ------
1020 REM -------------------------------------------------------------------
1030 E3=0
1110 IF ABS(M1)>100000 THEN 1150
1115 IF M=M1 THEN 1180
1120 X=-(Y0-M*X0-B1)/(M-M1)
1130 Y=M1*X+B1
1140 RETURN
1150 X=B1
1160 Y=M*X+B
1170 RETURN
1180 E3=1
1190 RETURN
1500 REM -------------------------------------------------------------------
1510 REM -- BOUNDARY DETECTOR ----------------------------------------------
1520 REM -------------------------------------------------------------------
1522 R1=1.0E+10
1523 N4=0
1526 FOR N3=1 TO N2
1527 IF R=0 AND N3=N5 THEN 1650
1530 M1=M2(N3)
1540 B1=B2(N3)
```

```
1550 GOSUB 1000
1555 IF E3=1 THEN 1650
1560 IF ABS(M1)>100000 THEN 1590
1570 IF X=>L2(N3) AND X<=U2(N3) THEN 1610
1580 GO TO 1650
1590 IF Y=>L2(N3) AND Y<=U2(N3) THEN 1610
1600 GO TO 1650
1610 REM -- IS POINT IN PROPER DIRECTION OF RAY?
1615 IF SGN(X-X0)<>SGN(D1) OR SGN(Y-Y0)<>SGN(D2) THEN 1650
1619 R2=SQR((X-X0)↑2+(Y-Y0)↑2)
1620 IF R2=>R1 THEN 1650
1622 X2=X
1623 Y2=Y
1630 R1=R2
1640 N4=N3
1650 NEXT N3
1655 X=X2
1656 Y=Y2
1660 IF N4<>0 THEN 1680
1670 PRINT "ERROR -- RAY INTERCEPTS NONE OF THE BOUNDARIES!"
1680 RETURN
2000 REM ---------------------------------------------------------------
2002 REM -- ROUTINE TO ENTER DATA DESCRIBING BOUNDARY LINES ------------
2003 REM ---------------------------------------------------------------
2005 PRINT "ENTER FILE NUMBER ON WHICH MODEL IS STORED (0 FOR KEYBD): ";
2006 INPUT F0
2007 IF F0<=0 THEN 2019
2008 FIND F0
2009 READ @33:N5
2010 FOR I=N2+1 TO N2+N5
2011 READ @33:M2(I),B2(I),L2(I),U2(I),G1(I),G2(I)
2012 NEXT I
2013 N2=N2+N5
2014 GO TO 2005
2019 PRINT "LENTER NUMBER OF BOUNDING SURFACES: ";
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
2020 INPUT N5
2030 IF N5=0 THEN 2110
2040 PRINT "ENTER DESCRIBING COORDINATES (X0,Y0) & (X1,Y1)"
2050 FOR I=N2+1 TO N2+N5
2051 PRINT "LINE ";I;".  ";
2052 INPUT X7,Y7,X8,Y8
2053 IF X7=X8 THEN 2073
2055 M2(I)=(Y8-Y7)/(X8-X7)
2056 B2(I)=Y7-M2(I)*X7
2057 G1(I)=M2(I)
2059 G2(I)=-1
2060 GO TO 2080
2073 G1(I)=1
2074 G2(I)=0
2075 M2(I)=1000000
2076 B2(I)=X7
2080 REM -- COMPUTE LOWER AND UPPER LIMIT OF LINE I
2081 L2(I)=X7 MIN X8
2082 U2(I)=X7 MAX X8
2083 IF X7<>X8 THEN 2100
2084 L2(I)=Y7 MIN Y8
2085 U2(I)=Y7 MAX Y8
2100 NEXT I
2105 N2=N2+N5
2110 PRINT "ENTER LOCATION OF SOURCE (X0,Y0): ";
2120 INPUT X0,Y0
2122 X9=X0
2123 Y9=Y0
2125 PRINT "ENTER SURFACE NUMBER THAT REPRESENTS ANTENNA: ";
2126 INPUT N5
2130 RETURN
2500 REM ---------------------------------------------------------------
2501 REM -- ROUTINE TO CHANGE RAY DIRECTION AT SURFACE INTERFACE -------
2502 REM ---------------------------------------------------------------
2510 REM -- N4=SURFACE NUMBER INTERCEPTED
```

```
2520 REM -- (X,Y) = POINT INTERCEPTED
2530 X0=X
2540 Y0=Y
2560 G=G1(N4)*D1+G2(N4)*D2
2570 D3=D1-2*G1(N4)*G
2580 D4=D2-2*G2(N4)*G
2590 D1=D3
2600 D2=D4
2610 IF D1=0 THEN 2640
2620 M=D2/D1
2625 B=Y0-M*X0
2630 GO TO 2650
2640 M=100000*SGN(D2)
2650 RETURN
3000 REM -----------------------------------------------------------
3001 REM -- ROUTINE TO DRAW BOUNDARIES ON THE SCREEN AND INITIALIZE-----
3002 REM -- TRACE-----------------------------------------------------
3003 REM -----------------------------------------------------------
3010 FOR I=1 TO N2
3020 IF ABS(M2(I))=>100000 THEN 3060
3030 MOVE L2(I),M2(I)*L2(I)+B2(I)
3040 DRAW U2(I),M2(I)*U2(I)+B2(I)
3050 GO TO 3080
3060 MOVE B2(I),L2(I)
3070 DRAW B2(I),U2(I)
3080 NEXT I
3090 MOVE X0,Y0
3100 RETURN
3500 REM -----------------------------------------------------------
3501 REM -- ROUTINE TO TRANSFER MODEL DATA TO TAPE FILE ---------------
3502 REM -----------------------------------------------------------
3510 PRINT "INSERT DATA TAPE AND ENTER FILE NUMBER: ";
3520 INPUT F1
3530 FIND F1
3540 WRITE @33:N2
```

```
3550 FOR I=1 TO N2
3560 WRITE @33:M2(I),B2(I),L2(I),U2(I),G1(I),G2(I)
3570 NEXT I
3580 CLOSE
3590 PRINT "REINSERT PROGRAM TAPE"
3600 RETURN
```