NASA Technical Memorandum 100691

The Focal Plane Reception Pattern Calculation for a Paraboloidal Antenna with a Nearby Fence

Richard F. Schmidt, Hwai-Soon Cheng
and Michael W. Kao

DECEMBER 1987

(NASA-TM-100691) THE FOCAL PLANE RECEPTION PATTERN CALCULATION FOR A PARABOLOIDAL ANTENNA WITH A NEARBY FENCE (NASA) 56 p CSCL 20N

N88-30039

Unclas
G3/32 0160870

NASA

NASA Technical Memorandum 100691

The Focal Plane Reception Pattern Calculation for a Paraboloidal Antenna with a Nearby Fence

Richard F. Schmidt
NASA/Goddard Space Flight Center
Greenbelt, Maryland 20771

Hwai-Soon Cheng and Michael W. Kao
Science Applications Research
Lanham, Maryland 20736

NASA
National Aeronautics and Space Administration

Goddard Space Flight Center

1987

TABLE OF CONTENTS

ORIGINAL PAGE IS
OF POOR QUALITY

1. INTRODUCTION

This document describes a computer simulation program which is used to estimate the effects of a proximate diffracting fence on the performance of paraboloid antennas.

The computer program is written in FORTRAN language for running on an IBM 3081 computer system at Goddard Space Flight Center, Greenbelt, MD.

The physical problem, mathematical formulation, and coordinate references are described in the general description section.

The main control structure of the program and the contents of the individual subroutines are discussed in the program description section.

The Job Control Language set-up and program instruction are provided in the user's instruction section to help users to execute the present program.

A sample problem with an appropriate output listing is made available as an illustration of the usage of the program.

Finally, a summary and comprehensive results are presented in the last section.

2. GENERAL DESCRIPTION

The computer program is written to compute the effects of a nearby fence on the antenna receiving characteristics. Specifically, a purpose of this investigation is to find out how the mainlobe and the first sidelobe of the paraboloidal antenna's reception pattern change due to the fields diffracted from the edge of a fence. This diffraction field could be described by a Sommerfeld's half-plane solution if the fence is not very far from the antenna and the angle from the antenna center to the both ends of the fence is large. (For details, see NASA TM-84996, R.F. Schmidt, "A Radio-Frequency Analysis of Paraboloidal Antenna Located Near Diffracting Fences.") In order to estimate the effect of diffracting fields on the performance of an antenna, a focal plane reception pattern of the incoming field which includes an incident plane wave and a diffracted field from a fence is needed. This pattern shows how the mainlobe and sidelobes are distorted due to the introduction of the diffracting field. The pattern of a normally incident plane wave (without diffracting field) is well-known by studying the distrotion of the pattern of a Sommerfeld half-plane solution, which includes an incident plane wave and a diffracting field, the effect of the diffracted field from a fence on the performance of an antenna is estimated.

The focal plane reception pattern are obtained as followed. An aperture field distribution is obtained by calculation of a Sommerfeld's half-plane solution at the aperture region in the antenna reference frame. The Fourier transformation of this aperture field distribution is the focal plane reception pattern. (Details, see appendix A of this documentation.)

2.1 COORDINATE SYSTEM

Figure 1 illustrates an inertial reference frame $(x,y,z)_I$, antenna reference frame $(x,y,z)_A$, and fence reference frame $(x,y,z)_F$. The orientation of the antenna reference frame relative to the inertial reference frame is represented by three Eulerian angles $(\alpha,\beta,\gamma)_A$ and a translation vector T_A. Similarly, the orientation of the fence reference frame relative to the inertial reference frame is represented by three Eulerian angles $(\alpha,\beta,\gamma)_F$ and a translation vector T_F.

Using these transformations, the desired quantities can be easily transformed among these three reference frames. The notation $R^{I\to F}$ denotes the Eulerian rotational transformation from inertial reference frame to fence reference frame. Similarly, the notation $R^{I\to A}$ means the Eulerian rotational transformation from inertial reference frame to antenna reference frame, etc. Some useful relations among these transformations are cited below.

The cascade transformation: $R^{A\to F} = R^{I\to F} R^{A\to I}$. The inverse rotational transformation: $R^{I\to F} = (R^{F\to I})$ transpose. The basic rotation transformation:

$$R^{I\to F} = \begin{bmatrix} \cos\gamma \cos\alpha - \cos\beta \sin\alpha \sin\gamma & \cos\gamma \sin\alpha + \cos\beta \cos\alpha \sin\gamma & \sin\gamma \sin\beta \\ -\sin\gamma \cos\alpha - \cos\beta \sin\alpha \cos\gamma & -\sin\gamma \sin\alpha + \cos\beta \cos\alpha \cos\gamma & \cos\gamma \sin\beta \\ \sin\beta \sin\alpha & -\sin\beta \cos\alpha & \cos\beta \end{bmatrix}$$

Where (α,β,γ) are three Eulerian rotational angles.

The inertial reference frame used in this document is an earth-fixed coordinate system.

2.2 PLANE-WAVE

The incident wave could be E-polarized, H-polarized or mixture of both polarization.

E-polarization

$$\overline{E} = (-\cos\alpha \sin\beta,\ -\sin\alpha \sin\beta,\ \cos\beta)\ e^{-iKS} \tag{1}$$

$$\overline{H} = (-\sin\alpha,\ \cos\alpha,\ 0)\ e^{-iKS} \tag{2}$$

H-polarization

$$\overline{E} = (\sin\alpha,\ -\cos\alpha,\ 0)\ e^{-iKS} \tag{3}$$

$$\overline{H} = (-\cos\alpha \sin\beta,\ -\sin\alpha \sin\beta,\ \cos\beta)\ e^{-iKS} \tag{4}$$

The phase factor of the plane wave is

$$e^{-iKS} = e^{-i\overline{K} \cdot (\hat{x} \cos\alpha \cos\beta + \hat{y} \sin\alpha \cos\beta + \hat{z} \sin\beta)} \tag{5}$$

In our program, we use the incident angle $(\alpha, \beta)_I$, and the polarization angle δ_I in the inertial reference frame. The incident plane wave is plane wave = $\cos\delta_I$ (E−pol) + $\sin\delta_I$ (H−pol) depending on the polarization angle δ_I, the plane wave could be E-polarized, H-polarized, or other linearly polarized plane wave.

2.3 DIFFRACTION FIELD

Sommerfeld's solution for a three dimensional diffraction of a plane-wave by a half-plane is given in the Gaussian system unit as

(E−plane polarization)

$$E_x = -H_y \sin\beta \tag{6}$$

$$E_y = H_x \sin\beta \tag{7}$$

$$E_z = \frac{e^{-\frac{\pi i}{4}}}{\sqrt{\pi}} \cos\beta \, e^{iK(r\cos\beta - z\sin\beta)} [G(p) - G(q)] \tag{8}$$

$$H_x = \frac{e^{-\frac{\pi i}{4}}}{\sqrt{\pi}} e^{iK(r\cos\beta - z\sin\beta)} \cdot \left\{ \sin\alpha \, [G(p) + G(q)] + i\sqrt{\frac{2}{K\, r\cos\beta}} \sin\frac{\alpha}{2} \cos\frac{\theta}{2} \right\} \tag{9}$$

$$H_y = \frac{e^{-\frac{\pi i}{4}}}{\sqrt{\pi}} e^{iK(r\cos\beta - z\sin\beta)} \cdot \left\{ \cos\alpha \, [G(p) - G(q)] + i\sqrt{\frac{2}{K\, r\cos\beta}} \sin\frac{\alpha}{2} \sin\frac{\theta}{2} \right\} \tag{10}$$

$$H_z = 0 \tag{11}$$

The companion expressions for the H-plane polarization are given below as

(H − plane polarization)

$$E_x = \frac{e^{-\frac{\pi i}{4}}}{\sqrt{\pi}} e^{iK(r\cos\beta - z\sin\beta)} \cdot \left\{ \sin\alpha \, [G(p) - G(q)] + i\sqrt{\frac{2}{K\, r\cos\beta}} \cos\frac{\alpha}{2} \sin\frac{\theta}{2} \right\} \tag{12}$$

$$E_y = \frac{-e^{-\frac{\pi i}{4}}}{\sqrt{\pi}} e^{iK(r\cos\beta - z\sin\beta)} \cdot \left\{ \cos\alpha \, [G(p) + G(q)] + i\sqrt{\frac{2}{K\, r\cos\beta}} \cos\frac{\alpha}{2} \cos\frac{\theta}{2} \right\} \tag{13}$$

$$E_z = 0 \tag{14}$$

$$H_x = E_y \sin\beta \tag{15}$$

$$H_y = -E_x \sin\beta \tag{16}$$

$$H_z = \frac{e^{-\frac{\pi i}{4}}}{\sqrt{\pi}} \cos\beta \, e^{iK(r\cos\beta - z\sin\beta)} [G(p) + G(q)] \tag{17}$$

Where $G(a) = e^{-ia^2}\left[\sqrt{\pi}\, e^{i\pi/4}\, U(-a)\right] + \mathrm{sgn}(a) \int_{|a|}^{\infty} e^{iu^2}\, du$

retains the Fresnel integral without assuming $kr \gg 1$

$$
\begin{aligned}
U(a) &= 1, \text{ when } a \geq 0 \\
&= 0, \text{ when } a < 0
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{sgn}(a) &= 1, \text{ when } a \geq 0 \\
&= 1, \text{ when } a < 0
\end{aligned}
$$

and

$$q = -(2Kr\cos\beta)^{1/2} \cos\frac{\theta + \alpha}{2}$$

$$p = -(2Kr\cos\beta)^{1/2} \cos\frac{\theta - \alpha}{2}$$

The incident angles (α, β) and the cylindrical coordinates (r, θ, z) used in the above equations are in fence reference frame.

Once the magnetic field components

$$(H_x, H_y, H_z)_F = \cos\delta_F\, H_{E-pol} + \sin\delta_F\, H_{H-pol} \tag{18}$$

have been determined, the transformation of

$$(H_x, H_y, H_z)_A = R^{F\to A} (H_x, H_y, H_z)_F \tag{19}$$

is used to obtain the magnetic field of the antenna aperture in the antenna reference frame, where δ_F is the polarization angle in the fence reference frame.

2.4 TRANSFORMATION AMONG REFERENCE FRAMES

In order to use Sommerfeld's half-plane solution, the plane-wave incident angles and polarization angle should transform from the inertial reference frame to the fence reference frame.

The incident angle pair (α, β) corresponds to a unit vector as

$$(x, y, z)_I = (\cos\alpha \cos\beta,\ \sin\alpha \cos\beta,\ \sin\beta)_I \tag{20}$$

The angles of arrival of the plane-wave in the fence reference frame are obtained by the rotation transformation

$$(x, y, z)_F = R^{I\to F} (x, y, z)_I \tag{21}$$

The translation transformation is ignored here since only angles are of concern. The arrival angles of the plane-wave in the fence reference frame are found by solving the equation below.

$$(x, y, z)_F = (\cos\alpha \cos\beta,\ \sin\alpha \cos\beta,\ \sin\beta)_F \tag{22}$$

For the transformation of the polarization angle δ_I from the inertial reference frame to the fence reference frame, linearly-polarized magnetic field with polarization angle δ_I in the inertial reference frame has components

$$
\begin{aligned}
(H_x, H_y, H_z)_I = {} & \cos\delta_I\, (-\sin\alpha,\ \cos\alpha,\ 0)_I\ + \\
& \sin\delta_I\, (-\cos\alpha \sin\beta,\ -\sin\alpha \sin\beta,\ \cos\beta)_I
\end{aligned} \tag{23}
$$

These magnetic field components transform into fence reference frame by

$$(H_x, H_y, H_z)_F = R^{I\to F}\ (H_x, H_y, H_z)_I \tag{24}$$

The polarization angle δ_F, in fence reference frame, are found by solving the equation

$$
\begin{aligned}
(H_x, H_y, H_z)_F = {} & \cos\delta_F\, (-\sin\alpha,\ \cos\alpha,\ 0)_F\ + \\
& \sin\delta_F\, (-\cos\alpha \sin\beta,\ -\sin\alpha \sin\beta,\ \cos\beta)_F
\end{aligned} \tag{25}
$$

ORIGINAL PAGE IS
OF POOR QUALITY

2.5 FOCAL PLANE RECEPTION PATTERN

The antenna analysis program calculates the focal plane reception pattern of the incoming diffracted electromagnetic field by Fourier transformation of a Sommerfeld half-plane solution (which is in antenna reference frame) at the antenna aperture region. The pattern given in decibels (dB), is obtained by

$$P = 20 \log \left| \int_{\text{aperture}} H_p(\theta)\, H_s\, da \right| \qquad (25)$$

Where $H_p(\theta)$ is a plane wave function and H_s is the aperture field distribution which is approximated by a Sommerfeld half-plane solution.

3. PROGRAM DESCRIPTION

This section describes a FORTRAN program which is used to calculate a focal plane reception pattern for a paraboloidal antenna and a nearby fence.

The program hierarchy chart is shown in Figure 1. This chart shows the flow of the program. The COMMON block cross-reference matrix is shown in Figure 2. This matrix shows the COMMON blocks used in each subroutine.

The subroutine functional descriptions and input parameters list are also included in this section. The program listing is provided in appendix B.

3.1 SUBROUTINE FUNCTIONAL DESCRIPTION

This section describes the functions performed by each subroutine.

MAIN The main routine controls the flow of the program. MAIN routine first calls subroutine ATIN to read the input parameters. MAIN routine then calls subroutine ATDEF to define the constant values in the program. The third subroutine called by MAIN is subroutine ATEXPL, which is used to print out the significant parameters used in the program. The actual calculation is performed after these calls.

ATIN This subroutine provides the input values for the program.

ATDEF This subroutine defines constant values and converts physical units.

ATEXPL This subroutine prints out some input parameters for the user's information and record.

ATAPER This subroutine subdivides the reflector aperture into small differential areas and evaluates the coordinates, unit normal vector, and differential area for each small area at the aperture surface.

ATROP This subroutine provides the rotation operators for the use in the rotational transformations among inertial, antenna, and fence reference frames.

ANGROP This subroutine defines. Eulerian rotation operation A for the rotational transformation

Where

$$A = \begin{bmatrix} (\cos\gamma \cos\alpha - \cos\beta \cos\alpha \sin\gamma) & (\cos\gamma \sin\alpha + \cos\beta \cos\alpha \sin\gamma) & (\sin\gamma \sin\beta) \\ (-\sin\gamma \cos\alpha - \cos\beta \sin\alpha \cos\gamma) & (-\sin\gamma \sin\alpha + \cos\beta \cos\alpha \cos\gamma) & (\cos\gamma \sin\beta) \\ (\sin\beta \cos\alpha) & (-\sin\beta \cos\alpha) & (\cos\beta) \end{bmatrix}$$

INPUT: Eulerian angles α, β, γ.
OUTPUT: Rotational Operator martrix A.

TRANSP This subroutine transposes the rotation operator to obtain an inverse matrix for the inverse rotation transformation.

This subroutine computes the inverse matrix

$$\begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}^T = \begin{bmatrix} M_{11} & M_{21} & M_{31} \\ M_{12} & M_{22} & M_{32} \\ M_{13} & M_{23} & M_{33} \end{bmatrix}$$

CROSS This subroutine performs matrix multiplication.

$$C\,(3,3) = A\,(3,3)\;B\,(3,3).$$

Where

$$C\,(i,j) = \sum_{k=1,3} A\,(i,k)\;B\,(k,j)$$

ROT This subroutine performs rotational transformation from one reference frame to another reference frame.

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = [A] \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

where A is Eulerian rotation operator.

ATXPW This subroutine transforms plane-wave incident angles of arrival and polarization angle from inertial reference frame to fence reference frame.

ATRAT This subroutine performs the necessary transformation to transform the cartesian coordinate point in antenna reference frame to the cylindrical coordinate point in the fence reference frame.

This subroutine computes

$$(x,\,y,\,z)_I = \left[R^{A\to I}\right]\,(x,\,y,\,z)_A + T_A \tag{1}$$

$$(x,\,y,\,z)_F = \left[R^{I\to F}\right]\,(x,\,y,\,z)_I - T_F \tag{2}$$

and

$$(r,\,\theta,\,z)_F = \left(\sqrt{x_F^2 + x_F^2}\,,\ \tan^{-1}\frac{y_F}{x_F}\,,\ z_F\right) \tag{3}$$

where T_A, T_F are translation vectors of the antenna reference frame and the fence reference frame to the inertial reference frame, respectively. The first equation transforms a cartesian coordinate point from the antenna reference frame to the inertial reference frame. The second equation transforms that same point from the inertial reference frame to the fence reference frame. The third equation transforms that point from the cartesian coordinate system to the cylindrical coordinate system. (Sommerfeld solution equations are described by a cylindrical coordinate system in the fence reference frame.)

ORIGINAL PAGE IS
OF POOR QUALITY

ATPL This subroutine computes G(p) and G(q) values in the Sommerfeld solution equations.

This subroutine evaluates

$$G(a) = e^{-ia^2}\left[\sqrt{\pi}\, e^{\frac{\pi i}{4}}\, U(-a) + \mathrm{sgn}(a)\int_{|a|}^{\infty} e^{i\mu^2}\, d\mu\right]$$

where

$$U(x) = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

$$\mathrm{sgn}(x) = \begin{cases} 1, & x \geq 0 \\ -1, & x < 0 \end{cases}$$

and

$$q = -(2Kr\cos\beta)^{1/2}\cos\frac{\theta+\alpha}{2}$$

$$p = -(2Kr\cos\beta)^{1/2}\cos\frac{\theta-\alpha}{2}$$

$$\int_{|a|}^{\infty} e^{i\mu^2}d\mu = \frac{\sqrt{2\pi}}{4}(1+i) - \sqrt{\frac{\pi}{2}}\int_0^{a^2}\frac{e^{it}}{\sqrt{2\pi t}}\,dt$$

The Fresnel integral $\int_0^x \frac{e^{it}}{\sqrt{2\pi t}}\,dt$ is evaluated at the subroutine ATCS.

ATCS This subroutine evaluates Fresnel integrals.

C(X) = INTEGRAL (COS(T)/SQRT (2 * PI * T) SUMMED OVER T FROM 0 TO X)

S(X) = INTEGRAL (SIN(T)/SQRT (2 * PI * T) SUMMED OVER T FROM 0 TO X)

ATCY This subroutine evaluates H-plane Sommerfeld solution equations.

ATRFA This subroutine transforms the H-plane electromagnetic field components of the Sommerfeld solution from fence reference frame to antenna reference frame.

This subroutine evaluates

$$(H_x, H_y, H_z)_A = \left[R^{F\to A}\right](H_x, H_y, H_z)_F$$

ATPWH This subroutine evaluates the plane-wave functions H_p at every point on the reflector aperture

$$H_p = (-\sin\alpha, \cos\alpha, 0)\, e^{-iKS} \qquad (1)$$

$$e^{-iKS} = e^{-iK\cdot(x\cos\alpha\cos\beta + y\sin\alpha\cos\beta + z\sin\beta)} \qquad (2)$$

angles α, β describe the direction of the plane wave.

ATCORR This subroutine calculates focal plane reception pattern by the following equation

$$\text{Pattern} = 20\log\left|\int_{\text{aperture}} H_s \cdot H_p\, ds\right|$$

where H_S is Sommerfeld solution and H_p is plane wave function

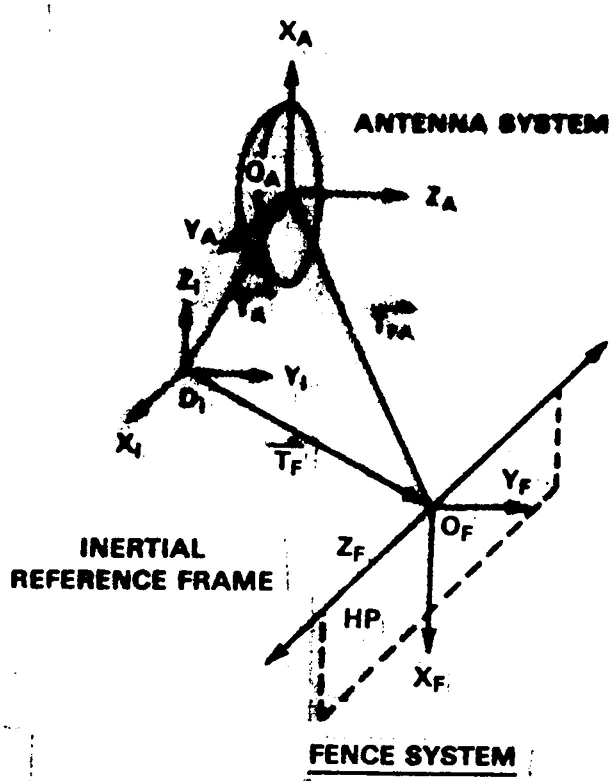

X_A
ANTENNA SYSTEM
O_A
Z_A
Y_A
Z_I
Y_I
O_I
X_I
T_F
Y_F
O_F
INERTIAL
REFERENCE FRAME
Z_F
HP
X_F
FENCE SYSTEM

Figure 1

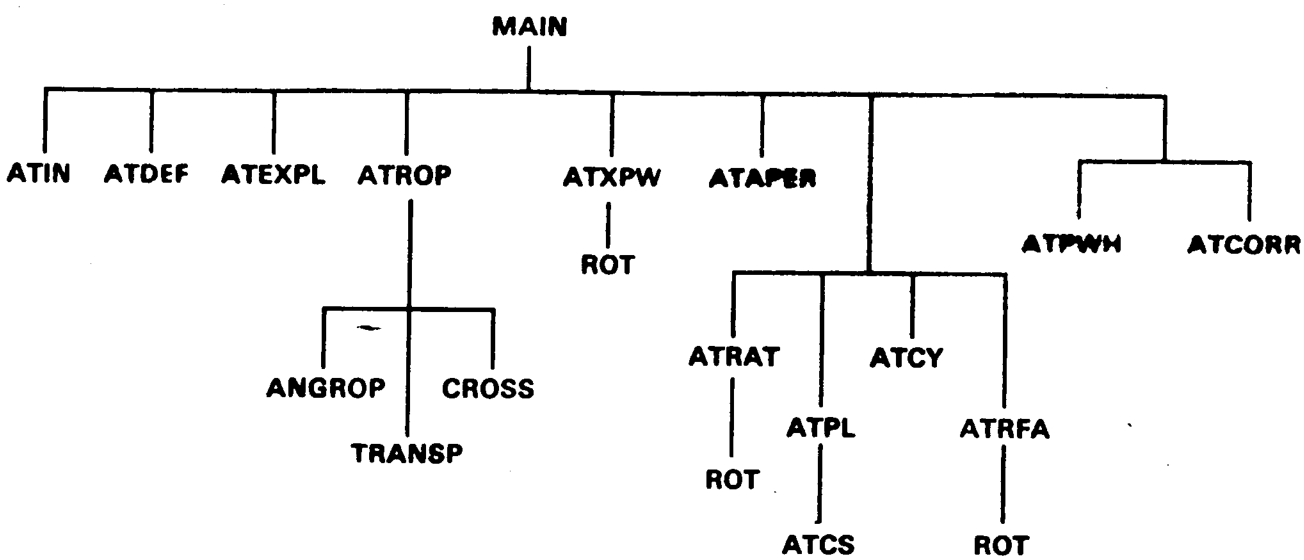


Figure 2. Hierarchy Chart

Table 1. Cross Reference Matrix

	Subroutine / Common Block	ATIH	ATBKDT	ATDEF	ATEXPL	ATROP	ATXPW	ATAPER	ATRAT	ATPL	ATCY	ATRFA	ATPWH	ATCORR
1	INPUT1	X	X	X	X			X						
2	INPUT2	X	X	X	X	X	X							
3	INPUT3	X	X	X	X	X			X					
4	INPUTX	X	X	X	X						X			X
5	ANTΦ			X	X		X	X		X	X		X	X
6	ANTROP					X	X		X			X		
7	ANTXP1						X							
8	ANTXP2						X							
9	ANTXP3						X			X	X			
10	ANTXP4						X				X			
11	ANTPA1							X	X				X	X
12	ANTPA2							X					X	X
13	ANTRAT								X	X	X			
14	ANTPL									X	X			
15	ANTCY										X	X		
16	ANTRFA											X		
17	ANTRF1											X		X
18	ANTRF2											X		X
19	ANTPW1												X	X
20	ANTPW2												X	X
21	ANTLP1								X			X		

ORIGINAL PAGE IS
OF POOR QUALITY

3.2 INPUT PARAMETERS

The names and descriptions of the program input variables are listed below

Table 2 Names and Descriptions of the Program Variables

Common Block	Parameter Name	Parameter Type	Description
INPUT1	SIGMAM	R*8	Antenna reflector radius
	SIGMAO	R*8	Original value of sigma
	XLJ	R*8	Integration control constant
	XLM	R*8	Wavelength
	FR	R*8	Frequency in GHZ units
	F	R*8	Focal length of ideal paraboloid
INPUT2	ALPHAI	R*8	Plane-wave azimuthal angle in inertial system
	BETAI	R*8	Plane-wave polar angle in inertial system
	DELTAI	R*8	Plane-wave polarization angle in inertial system
	ALF1	R*8	Eulerian angle in antenna system
	BET1	R*8	Eulerian angle in antenna system
	GAM1	R*8	Eulerian angle in antenna system
INPUT3	ALF2	R*8	Eulerian angle in fence system
	BET2	R*8	Eulerian angle in fence system
	GAM2	R*8	Eulerian angle in fence system
	XTA	R*8	Translation vector component for inertial-antenna correspondence
	YTA	R*8	Translation vector component for inertial-antenna correspondence
	ZTA	R*8	Translation vector component for inertial-antenna correspondence
	XTF	R*8	Translation vector component for inertial-fence correspondence
	YTF	R*8	Translation vector component for inertial-fence correspondence
	ZTF	R*8	Translation vector component for inertial-fence correspondence

Table 3

INPUT	ANGA	I*4	angle o of plane wave H_p
	ANGB	R*8	angle β of plane wave H_p
	DB	R*8	increment of angle β
	NANG	I*4	number of angles to be evaluated in the pattern
	XDF	R*8	If XDF = 1, the Sommerfeld solution equations are fully calculated If XDF ≠, the cylindrical diffraction part of Sommerfeld solution is discarded

ORIGINAL PAGE IS
OF POOR QUALITY

4. USER INSTRUCTION

This section comprises the user's guide for the antenna analysis program.

Presently, the program exists in the form of non-executable load modules in library XRHSC.ANTENNA.LOAD for all but the MAIN and BLOCK DATA subprogram. Since these cannot be linked in automatically, only object modules were generated for them. These are located in library XRHSC.LIB.OBJ, which also includes the object modules generated in the process of creating the members of the nonexecutable load library.

Figure A shows the JCL to be used to recompile the program when no nonexecutable load library module is to be created, Figure B shows the JCL to be used when a member is to be created and added to the non-executable load library. Cataloged procedures in library XRHSC.MVS790X.CNTL eliminate the need for detailed copying of this JCL (members FORTVCOM and FORTVADD)

For executing the program, PAT, a link-edit and go procedure, has been created. PAT is located in library XRHSC.LIB.CNTL. Using PAT is equivalent to using the JCL in Figure C.

5. SAMPLE PROBLEM

This section provides a sample problem to simulate the radiation pattern of a 9-meter paraboloidal reflector with F/D ratio of 0.44 with feed at focal point. The incident electric field is assumed to be linearly polarized in the Z-direction in the initial coordinate system. Operating frequency is 2.0 GHz with integration control constant XLI set to 0.5. The Eulerian angles in antenna coordinate system are (180,90,90) and in the fence coordinate system are (90,90,270). The translation vector components from inertial to aperture origin is (0,1000,0) and from inertial to fence origin is (0,2200,−700) in cm units. The radiation pattern is computed at aperture transmitted region with angular coordinate Q varying from 93.0 to 87.0 at 0.1 decrements.

All the default input values are set up in the BLOCK DATA of the program ANTENNA.FORT(PAT). The user may use the namelisted input in the data file ANTENNA.DATA(CASEP01) to override the default input values.

Computer print-out listings are shown in Figures D-G. The focal plane reception pattern is plotted in Figure H.

6. SUMMARY

A comprehensive test of the program has been performed. Figure 1 shows that the focal plane reception pattern with a fence oriented orthogonally to the paraboloid axis and located directly in front of the antenna 16 meters away with the antenna lower half blocked by the fence. Figures 2 - 10 show a series of focal plane reception patterns with the fence lower by 30 cm, 100 cm, 200 cm, 300 cm, 400 cm, 450 cm, 600 cm, 700 cm and 800 cm, respectively. It is clearly illustrated that the focal plane reception patterns are distorted for all the cases that the antenna are blocked by the fence. The radius of the antenna is 450 cm. The distortion depends on how much area is blocked. For example, Figure 6 shows slight distortion of the reception pattern for a slightly blocked antenna, and Figure 1 shows great distortion of reception pattern for a half-blocked antenna. It is also shown that all the unblocked antenna cases, the reception pattern are undistorted as illustrated in Figures 7 - 10. The first sidelobe level is seen to be at −17.6 dB with respect to the main beam peak due to the fact that the effect of space divergence between a focal point feed and the parabolic reflector, and feed directivity, were not included in the program initially.

ORIGINAL PAGE IS
OF POOR QUALITY

For the following two cases, although the antennas are not blocked by the fence geometrically, but the diffraction effects are shown on the sidelobe of reception pattern nevertheless. Figure 11 illustrates that with the fence 127 meters away from the antenna and 5 meters lower from the level of center of antenna, with this orientation the angle of incoming diffraction field is 2.15 degrees and correspondingly the reception pattern has distortion at $\theta = 87.75$ degrees. The same effect is also shown in the second case. In Figure 12, with the fence located at (450 m, −9 m), the angle of incoming diffraction field from the fence is 1.15 degrees. The reception pattern clearly shows distortion at $\theta = 88.85$ degrees.

In this investigation we found that as long as the antenna is not blocked by the fence, the main lobe of the reception pattern will not be distorted. However, the sidelobe will be distorted if the angle of the incoming diffraction field is roughly equal to the inclination angle of that sidelobe. If the angle of the incoming diffraction field is large compared to the inclination angle of the sidelobe concerned, the diffraction effect from the fence is not shown in the focal plane reception pattern.

REFERENCES

1. Born, M. and Wolf, E., "Principles of Optics," 2nd Edition, Pergamon Press, 1964.
2. Schmidt, R.F., "A Radio-Frequency Analysis of Paraboloidal Antenna Located Near Diffracting Fences," TM 84996.
3. Stratton, J.A., "Electromagnetic Theory," McGraw-Hill Book Company, Inc.
4. Silver, S., "Microwave Antenna Theory and Design," McGraw-Hill Book Company, Inc., 1949.
5. Schmidt, R.F., "The Calculation of Electromagnetic Fields by the Kirchhoff-Kottler Method," GSFC-X-525-70-293, May 1970.
6. Schmidt, R.F., "The Calculation of Electromagnetic Fields in the Fresnel and Fraunhofer Regions Using Numerical Integration Methods," GSFC-X-811-71-392, July, 1971.
7. Goldstein, H., "Classical Mechanics," Addison-Wesley Publishing Company, Inc., 1959.
8. Apostol, T.M., "Mathematical Analysis," Addison-Wesley Publishing Company, Inc., 1960.
9. Collin, R.E. and Zucker, F.J., "Antenna Theory," McGraw-Hill Book Company, Inc., 1969.
10. Beckmann, P., "The Depolarization of Electromagnetic Waves," The Golem Press, 1968.
11. Kaplan, W., "Advanced Calculus," Addison-Wesley Publishing Company, Inc., 1959.
12. Jackson, J.D., "Classical Electrodynamics," John Wiley and Sons, Inc., 1962.

APPENDIX A

RELATION OF FOCAL PLANE RECEPTION PATTERN AND FOURIER TRANSFORMATION OF THE FIELD DISTRIBUTION AT ANTENNA APERTURE REGION

The aperture field distribution H_s is approximated by a Sommerfeld half-plane solution described in the antenna reference frame. This aperture field distribution includes the incident plane wave and the incoming diffraction field from the fence. In order to estimate the effect of the diffracting fence on the performance of the antenna, a focal plane reception pattern corresponding to this aperture field distribution is needed to illustrate how the introduction of the diffraction field from the fence changes the focal plane reception pattern.

In Figure A-1, P is an arbitrary point on the focal plane of antenna, A is the vertex of antenna, and F is the focal point. The angle between PA and FA is θ. For a small angle approximation, a plane wave with an incident angle θ from the opposite side will focus on point P after reflection from the paraboloidal antenna. Alternatively, the phase difference function for any point on the aperture plane is $e^{iK_P \cdot R}$ for that corresponding point P on the focal plane.

[Here k_p is the unit wave vector of the plane wave with an incident angle θ and R is a position vector to the aperture plane.] The field at point P for the aperture field distribution H_s can therefore be calculated by the integration of the aperture field distribution H_S multiplied with the phase difference function $e^{iK_P \cdot R}$ over the entire aperture region.

$$\int_{\text{aperture}} H_s \cdot e^{iK_P \cdot R}\, da$$

Similarly, for any other point Q on the focal plane, there is a wave vector k_Q in the opposite side of AF with an incident angle equal to FAQ.

The field at point Q is

$$\int_{\text{aperture}} H_s \cdot e^{iK_Q \cdot R}\, da$$

The focal plane reception pattern for an aperture field distribution H_s is now represented as the Fourier transformation of H_s at the aperture region

$$\int_{\text{aperture}} H_s \cdot e^{iK \cdot R}\, da$$

APPENDIX B

The entire antenna analysis program is listed below.

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.LIB.MVS.CLIST                                (FORTVCOM)

PROC 2 SUFFIX MEMBER TERMON MIN(0) SEC(30)
IF &SUFFIX = XXX THEN SET SUFFIX=
ACCESSJC ACCT(SPONS) BOX(BOX)
QED TEMPJOB.CNTL NEW EMODE
IN //&SYSUID.COM JOB (FH011,B22,2),'LIB&SUFFIX.(&MEMBER.)',TIME=(&MIN.,&SEC.),
IN //  MSGCLASS=U,CLASS=0,NOTIFY=&SYSUID
IN /*JOBPARM QUEUE=FETCH
IN /*PROCLIB=XRHSC.MVSPROC.CNTL
IN // EXEC FORTVCOM,OUT='*',SUFFIX='&SUFFIX.',MEMBER=&MEMBER,PREFIX=ANTENNA
IN // EXEC NTSO
&TERMON. SCHEDULE *
END N
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.MVSPROC.CNTL                                 (FORTVCOM)

//FORTVCOM PROC USRID=XRHSC,PREFIX=,OUT='*',BLKSIZE=7265,
// TERMOUT='*',ERRLVL='NOFIPS,FLAG(E)'
//SOURCE EXEC  PGM=FORTVS,REGION=2048K,COND=(4,LT),                    +
//            PARM='LC(80),&ERRLVL,SOURCE,XREF,MAP,NODECK,NOLIST,OPT(3)'
//STEPLIB  DD DSN=SYS1.FORTVS,DISP=SHR
//SYSLIN   DD DSN=&USRID..&PREFIX&SUFFIX..OBJ(&MEMBER),DISP=SHR,
//            UNIT=SYSDA,
//                DCB=(,RECFM=FB,LRECL=80,BLKSIZE=3200)
//SYSPRINT DD SYSOUT=&OUT,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=&BLKSIZE)
//SYSPUNCH DD DUMMY,DCB=BLKSIZE=3440
//SYSIN    DD DSN=&USRID..&PREFIX&SUFFIX..FORT(&MEMBER),DISP=SHR
//* SYSLIB   DD DSN=&USRID..&PREFIX&SUFFIX..COMN.FORT,DISP=SHR
//SYSTERM  DD SYSOUT=&TERMOUT
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.LIB.MVS.CLIST                                   (FORTVADD)

00000010PROC 2 SUFFIX MEMBER TERMON MIN(0) SEC(30)
00000020IF &SUFFIX = XXX THEN SET SUFFIX=
00000030SET CC = CC
00000040IF &TERMON = TERMON THEN SET CC= L
00000050ACCESSJC ACCT(SPONS) BOX(BOX)
00000060QED TEMPJOB.CNTL NEW EMODE
00000070IN //&SYSUID.CAD JOB (&SPONS,B22,2),'LIB&SUFFIX.(&MEMBER.)',TIME=(&MIN.,&SEC.),
00000080IN //  MSGCLASS=U,CLASS=0,NOTIFY=&SYSUID
00000090IN /*JOBPARM QUEUE=FETCH
00000100IN /*PROCLIB=XRHSC.MVSPROC.CNTL
00000110IN // EXEC FORTVADD,NBLK=4,OUT='*',SUFFIX='&SUFFIX.',MEMBER=&MEMBER,
00000120IN //      PREFIX=ANTENNA
00000130IN // EXEC NTSO
00000140V
00000150&CC
00000160&TERMON. SCHEDULE *
00000170END N
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.MVSPROC.CNTL                                    (FORTVADD)

//FORTVADD PROC USRID=XRHSC,PREFIX=,OUT='*',BLKSIZE=7265,                  00000010
// NBLK=40,TERMOUT='*',ERRLVL='NOFIPS,FLAG(E)'                             00000020
//SOURCE EXEC  PGM=FORTVS,REGION=2048K,COND=(4,LT),                      + 00000030
//             PARM='LC(80),&ERRLVL,SOURCE,XREF,MAP,NODECK,NOLIST,OPT(3)'  00000040
//STEPLIB  DD DSN=SYS1.FORTVS,DISP=SHR                                     00000050
//SYSLIN   DD DSN=&USRID..&PREFIX&SUFFIX..OBJ(&MEMBER),DISP=SHR,           00000060
//          UNIT=SYSDA,                                                    00000070
//             DCB=(,RECFM=FB,LRECL=80,BLKSIZE=3200)                       00000080
//SYSPRINT DD SYSOUT=&OUT,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=&BLKSIZE)       00000090
//SYSPUNCH DD DUMMY,DCB=BLKSIZE=3440                                       00000100
//SYSIN    DD DSN=&USRID..&PREFIX&SUFFIX..FORT(&MEMBER),DISP=SHR           00000110
//SYSTERM  DD SYSOUT=&TERMOUT                                              00000130
//LIBNAME  EXEC PGM=LIBRYGN2,COND=(4,LT),REGION=175K,PARM='$'              00000140
//SYSLIB   DD DSNAME=SYS1.DUMMY,DISP=SHR                                   00000150
//SYSOUT   DD DSNAME=&&LIBMOD,DISP=(NEW,PASS),UNIT=3350,                   00000160
// SPACE=(3200,(&NBLK,80),,,ROUND),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)    00000170
//SYSPRINT DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=81,BLKSIZE=7290)            00000180
//SYSPUNCH DD DUMMY,DCB=(RECFM=FB,LRECL=80,BLKSIZE=7280)                   00000190
//SYSIN    DD DSN=&USRID..&PREFIX&SUFFIX..OBJ(&MEMBER),DISP=SHR            00000200
//LINK     EXEC PGM=LINKEDIT,COND=(4,LT),REGION=150K,                      00000210
// PARM='LIST,MAP,NCAL,SIZE=(132K,12K)'                                    00000220
//SYSLMOD  DD DSN=&USRID..&PREFIX&SUFFIX..LOAD(&MEMBER),DISP=SHR           00000230
//SYSPRINT DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=3509)           00000240
//SYSUDUMP DD DUMMY                                                        00000250
//SYSUT1   DD UNIT=3350,SPACE=(CYL,(1,1))                                  00000260
//SYSLIN   DD DSN=&&LIBMOD,DISP=(OLD,DELETE)                               00000270
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.LIB.CNTL                                          (PAT      )

//XRHSCP01 JOB (FH011,B22,5),PATDF,MSGCLASS=A,TIME=(6,),                      00000010
//   NOTIFY=XRHSC,CLASS=0                                                     00000020
//* CLASS=0:NO TAPE,A:DEFAULT,E:EVENING,F:WEEKEND.                            00000025
//*JOBPARM LINES=100,QUEUE=FETCH                                              00000030
/*PROCLIB=XRHSC.MVSPROC.CNTL                                                  00000040
//STEP1 EXEC  PAT,VOL=ANT01,FILE=1,SIZE='2048K,256K',                         00000050
// REGION.LINK=6000K,REGION.GO=3000K                                          00000060
//LINK.OBJECT DD *                                                            00000070
 INCLUDE OBJLIB(PAT)                                                          00000080
 INCLUDE NEWLIN(ZETA$)                                                        00000090
//GO.FT08F001 DD DUMMY                                                        00000100
//*GO.FT08F001 DD DUMMY                                                       00000110
//GO.DATA5  DD DSN=XRHSC.ANTENNA.DATA(CASEP01),DISP=SHR,LABEL=(,,,IN)         00000120
// EXEC NTSO,MODE=ALL                                                         00000130
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.MVSPROC.CNTL                                      (PAT      )

//ANTASIM PROC NBLK=40,OUT='*',TERMOUT='*',FILE=1,     *LA                    00000010
//          SIZE='128K,12K',OPTION=                                           00000020
//LINK    EXEC  PGM=IEWL,REGION=150K,COND=(4,LT),                             00000030
//          PARM='MAP,LIST,SIZE=(&SIZE.),&OPTION'                             00000040
//NEWLIN    DD DSN=SYS2.NEWZETA,DISP=SHR                                      00000050
//SYSLIB    DD DSN=SYS1.VLNKMLIB,DISP=SHR  XRDMS 01/06/83                     00000060
//          DD DSN=SYS1.VFORTLIB,DISP=SHR                                     00000065
//          DD DSN=XRHSC.ANTENNA.LOAD,DISP=SHR                                00000070
//          DD DSN=SYS2.WP1055,DISP=SHR                                       00000080
//          DD DSN=SYS2.IMSLS,DISP=SHR                                        00000090
//          DD DSN=SYS1.FORTSSP,DISP=SHR                                      00000095
//          DD DSN=SYS1.MVTFTLIB,DISP=SHR                                     00000100
//          DD DSN=SYS2.VFORTLIB,DISP=SHR     XRDMS 01/06/83                  00000110
//          DD DSN=SYS2.FORTLIB,DISP=SHR                                      00000120
//OBJLIB    DD DSN=XRHSC.ANTENNA.OBJ(PAT),DISP=SHR,                           00000130
//               DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)                         00000135
//SYSLMOD   DD DSN=&&LODMOD(GSFC),UNIT=DISK,SPACE=(6144,(&NBLK,20,1)),        00000140
//               DISP=(,PASS)                                                 00000150
//SYSPRINT DD SYSOUT=&OUT,DCB=(RECFM=FBM,LRECL=121,BLKSIZE=3509)              00000160
//SYSTERM   DD SYSOUT=&TERMOUT                                                00000170
//SYSUT1    DD UNIT=(DISK,SEP=(SYSLMOD,SYSPRINT)),SPACE=(CYL,(1,1))           00000180
//SYSLIN    DD DSN=&&OBJMOD,DISP=(MOD,DELETE),DCB=RECFM=FB,                   00000190
// UNIT=DISK,SPACE=(TRK,0)                                                    00000200
//          DD DDNAME=OBJECT                                                  00000210
//SYSUDUMP DD DUMMY                                                           00000220
//GO      EXEC PGM=*.LINK.SYSLMOD,COND=(4,LT),REGION=800K                     00000230
//FT05F001 DD DDNAME=DATA5                                                    00000240
//FT06F001 DD SYSOUT=&OUT,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=141,BUFNO=1)       00000250
//FT07F001 DD SYSOUT=B,DCB=(RECFM=FB,BLKSIZE=7280,LRECL=80)                   00000260
//FT08F001 DD UNIT=(6250,,DEFER),LABEL=(&FILE,NL),VOL=SER=&VOL,               00000270
//               DCB=(RECFM=FB,LRECL=64,BLKSIZE=1024,DEN=3),                  00000280
//               DISP=(NEW,KEEP)                                              00000290
//FT10F001 DD SYSOUT=&OUT,DCB=*.FT06F001                                      00000300
//SYSUDUMP DD DUMMY                                                           00000310
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.MEMO.TEXT                                        (END      )
```

The complete output file of our sample problem is shown as :

```
NAMES AND DESCRIPTIONS OF THE PROGRAM VARIABLES

SIGMAM =  450.000   *** ANTENNA REFLECTOR RADIUS

SIGMAO =    0.000   *** INITIAL VALUE OF SIGMA

XLI =    0.500   *** INTEGRATION CONTROL CONSTANT

XLM =   14.990   *** WAVELENGTH

FR =    2.000   *** FRENQUENCY IN GHZ UNITS

F =  500.0   *** FOCAL LENGTH OF IDEAL PARABOLOID

PLANE-WAVE ANGLES OF ARRIVAL IN INITIAL SYSTEM

ALPHAI =   90.000   *** AZIMUTHAL ANGLE
BETAI =    0.000   *** RIGHT ANGLE - POLAR ANGLE

POLARIZATION ANGLE

DELTAI =   90.000

EULERIAN ANGLES FOR ROTATION

IN ANTENNA SYSTEM

ALF1 =  180.000  BET1 =   90.000  GAM1 =   90.000

IN FENCE SYSTEM

ALF2 =   90.000  BET2 =   90.000  GAM2 =  270.000

TRANSLATION VECTOR COMPONENTS

FROM INITIAL TO ANTENNA SYSTEM

XTA =    0.0  YTA =1000.0  ZTA =    0.0

FROM INITIAL TO FENCE SYSTEM

XTF= 0.000000E+00  YTF = 0.220000E+04  ZTF = -.700000E+03
```

INDEX = 11992

PB =	93.000	VOC =	0.183448E+05	VOCL =	0.852703E+02
PB =	92.900	VOC =	0.381938E+05	VOCL =	0.916398E+02
PB =	92.800	VOC =	0.558975E+05	VOCL =	0.949479E+02
PB =	92.700	VOC =	0.692831E+05	VOCL =	0.968125E+02
PB =	92.600	VOC =	0.763894E+05	VOCL =	0.976607E+02
PB =	92.500	VOC =	0.756856E+05	VOCL =	0.975803E+02
PB =	92.400	VOC =	0.662703E+05	VOCL =	0.964264E+02
PB =	92.300	VOC =	0.480307E+05	VOCL =	0.936304E+02
PB =	92.200	VOC =	0.217458E+05	VOCL =	0.867475E+02
PB =	92.100	VOC =	0.108810E+05	VOCL =	0.807334E+02
PB =	92.000	VOC =	0.472636E+05	VOCL =	0.934905E+02
PB =	91.900	VOC =	0.840626E+05	VOCL =	0.984921E+02
PB =	91.800	VOC =	0.117380E+06	VOCL =	0.101392E+03
PB =	91.700	VOC =	0.143021E+06	VOCL =	0.103108E+03
PB =	91.600	VOC =	0.156805E+06	VOCL =	0.103907E+03
PB =	91.500	VOC =	0.154890E+06	VOCL =	0.103800E+03
PB =	91.400	VOC =	0.134104E+06	VOCL =	0.102549E+03
PB =	91.300	VOC =	0.922396E+05	VOCL =	0.992984E+02
PB =	91.200	VOC =	0.283026E+05	VOCL =	0.890365E+02
PB =	91.100	VOC =	0.573251E+05	VOCL =	0.951669E+02
PB =	91.000	VOC =	0.162807E+06	VOCL =	0.104233E+03
PB =	90.900	VOC =	0.284882E+06	VOCL =	0.109093E+03
PB =	90.800	VOC =	0.418997E+06	VOCL =	0.112444E+03
PB =	90.700	VOC =	0.559540E+06	VOCL =	0.114957E+03
PB =	90.600	VOC =	0.700155E+06	VOCL =	0.116904E+03
PB =	90.500	VOC =	0.834125E+06	VOCL =	0.118425E+03
PB =	90.400	VOC =	0.954788E+06	VOCL =	0.119598E+03
PB =	90.300	VOC =	0.105597E+07	VOCL =	0.120473E+03
PB =	90.200	VOC =	0.113237E+07	VOCL =	0.121080E+03

ORIGINAL PAGE IS
OF POOR QUALITY

PB =	90.100	VOC =	0.117994E+07	VOCL =	0.121437E+03
PB =	90.000	VOC =	0.119613E+07	VOCL =	0.121556E+03
PB =	89.900	VOC =	0.118006E+07	VOCL =	0.121438E+03
PB =	89.800	VOC =	0.113259E+07	VOCL =	0.121081E+03
PB =	89.700	VOC =	0.105627E+07	VOCL =	0.120476E+03
PB =	89.600	VOC =	0.955144E+06	VOCL =	0.119601E+03
PB =	89.500	VOC =	0.834498E+06	VOCL =	0.118429E+03
PB =	89.400	VOC =	0.700509E+06	VOCL =	0.116908E+03
PB =	89.300	VOC =	0.559841E+06	VOCL =	0.114961E+03
PB =	89.200	VOC =	0.419217E+06	VOCL =	0.112449E+03
PB =	89.100	VOC =	0.285000E+06	VOCL =	0.109097E+03
PB =	89.000	VOC =	0.162814E+06	VOCL =	0.104234E+03
PB =	88.900	VOC =	0.572242E+05	VOCL =	0.951516E+02
PB =	88.800	VOC =	0.284991E+05	VOCL =	0.890966E+02
PB =	88.700	VOC =	0.925082E+05	VOCL =	0.993236E+02
PB =	88.600	VOC =	0.134414E+06	VOCL =	0.102569E+03
PB =	88.500	VOC =	0.155207E+06	VOCL =	0.103818E+03
PB =	88.400	VOC =	0.157094E+06	VOCL =	0.103923E+03
PB =	88.300	VOC =	0.143250E+06	VOCL =	0.103122E+03
PB =	88.200	VOC =	0.117523E+06	VOCL =	0.101402E+03
PB =	88.100	VOC =	0.841043E+05	VOCL =	0.984964E+02
PB =	88.000	VOC =	0.471980E+05	VOCL =	0.934785E+02
PB =	87.900	VOC =	0.107148E+05	VOCL =	0.805997E+02
PB =	87.800	VOC =	0.219976E+05	VOCL =	0.868475E+02
PB =	87.700	VOC =	0.483404E+05	VOCL =	0.936862E+02
PB =	87.600	VOC =	0.666055E+05	VOCL =	0.964702E+02
PB =	87.500	VOC =	0.760108E+05	VOCL =	0.976175E+02
PB =	87.400	VOC =	0.766696E+05	VOCL =	0.976925E+02
PB =	87.300	VOC =	0.694880E+05	VOCL =	0.968382E+02
PB =	87.200	VOC =	0.560043E+05	VOCL =	0.949644E+02

```
PB =   87.100 VOC = 0.381897E+05   VOCL = 0.916389E+02
PB =   87.000 VOC = 0.182293E+05   VOCL = 0.852154E+02
```

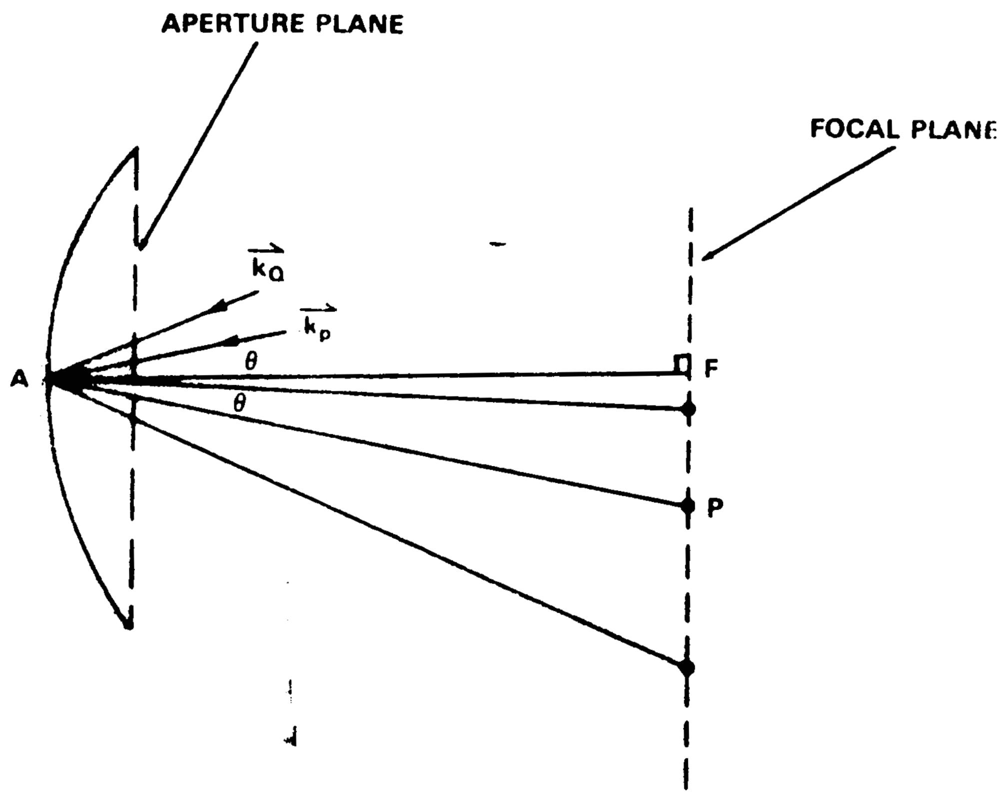
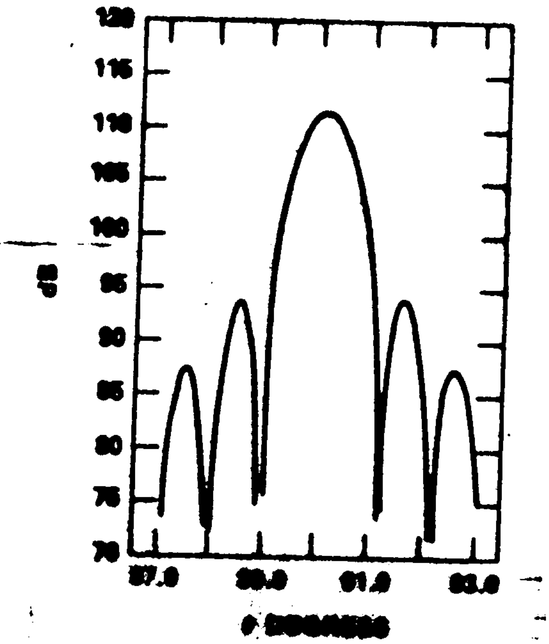
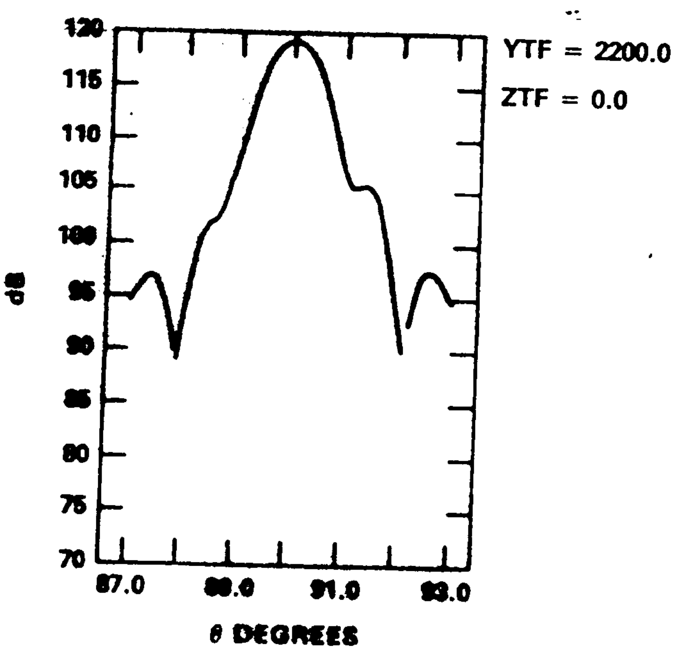
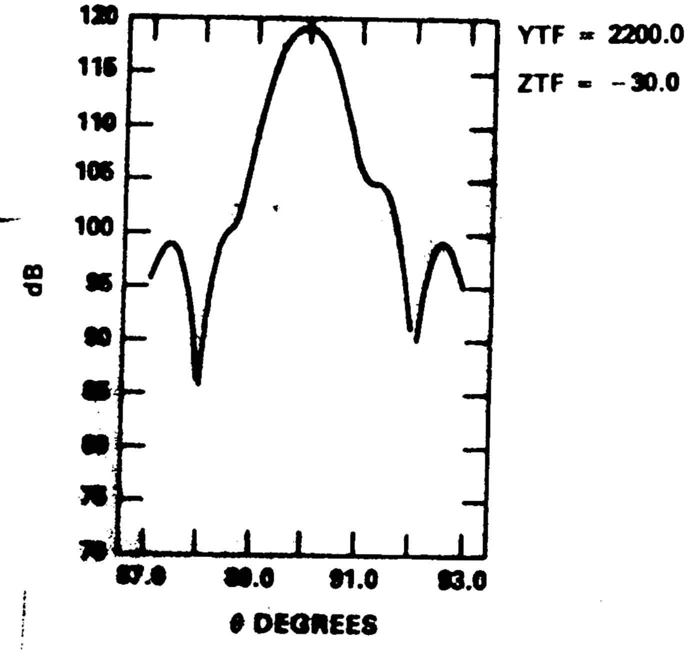


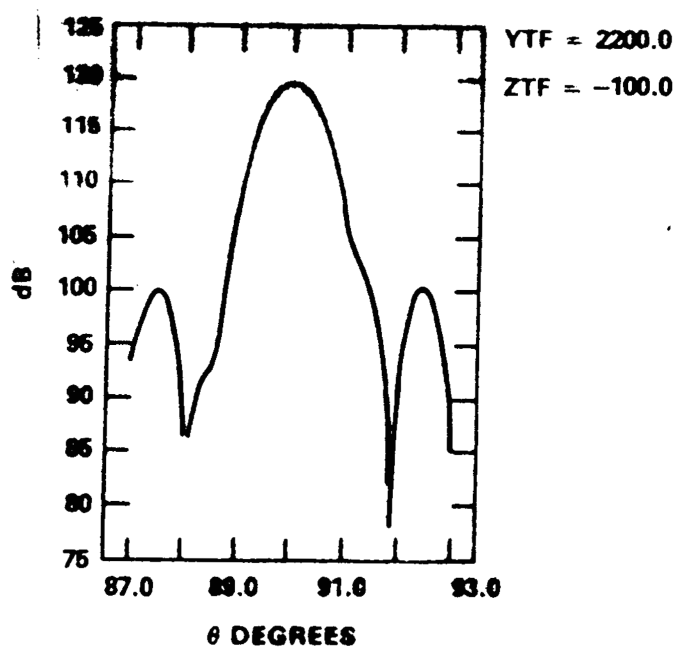
Figure A-1.

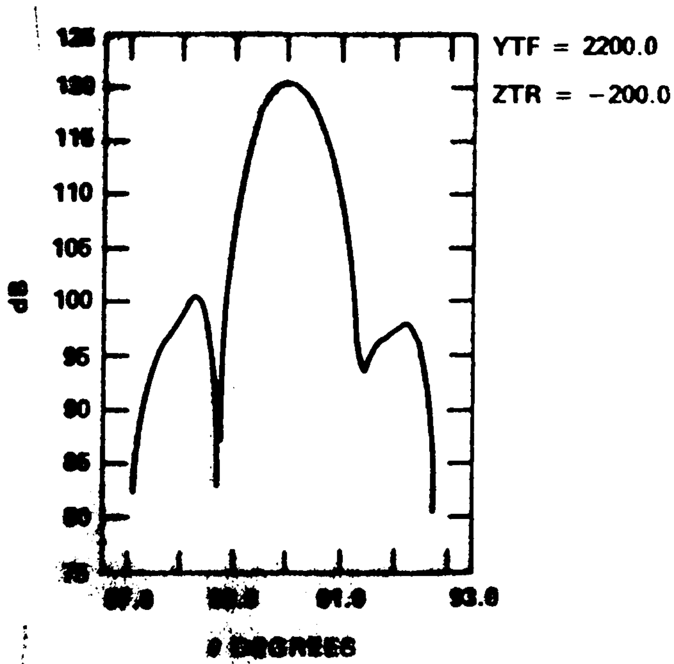

115
dB
θ DEGREES

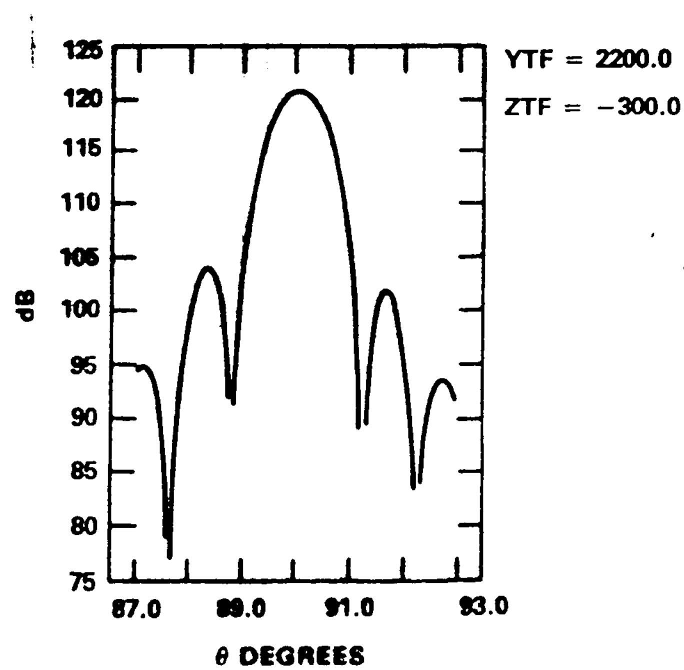

120
115
110
105
95
90
85
80
75
70
dB
87.0
91.0
93.0
θ DEGREES
YTF = 2200.0
ZTF = 0.0

ORIGINAL PAGE IS
OF POOR QUALITY

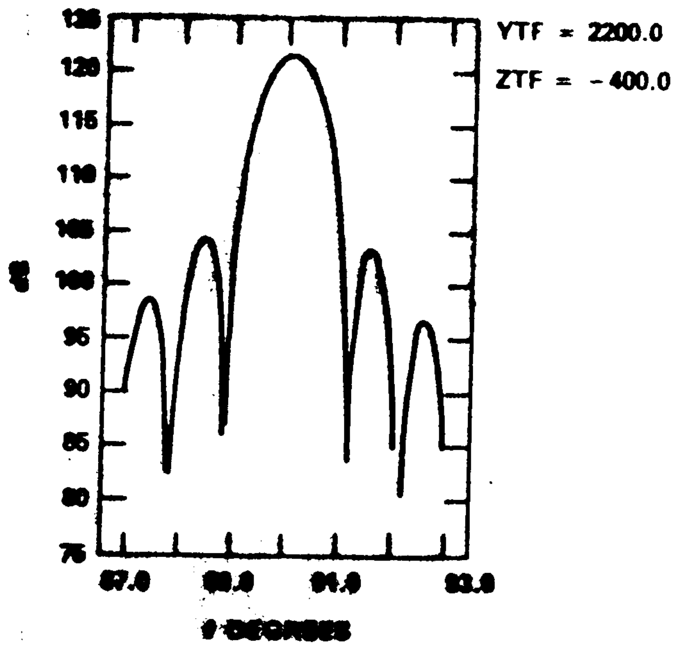


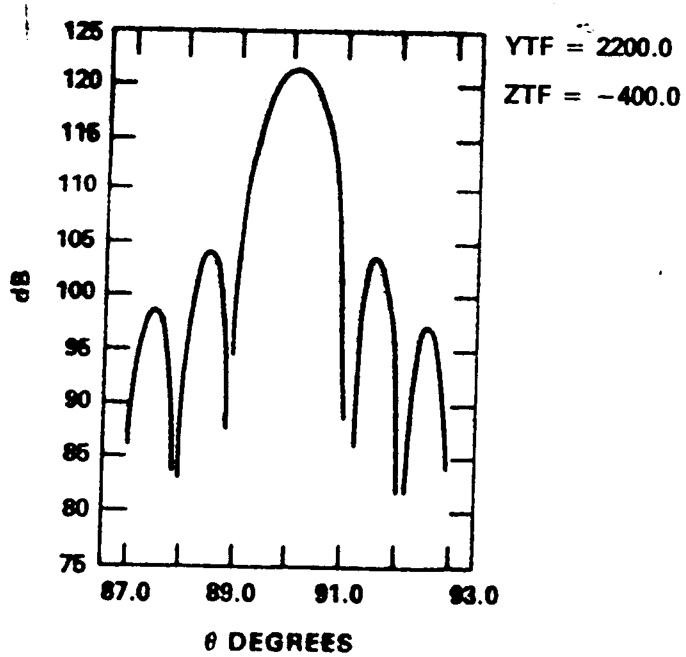


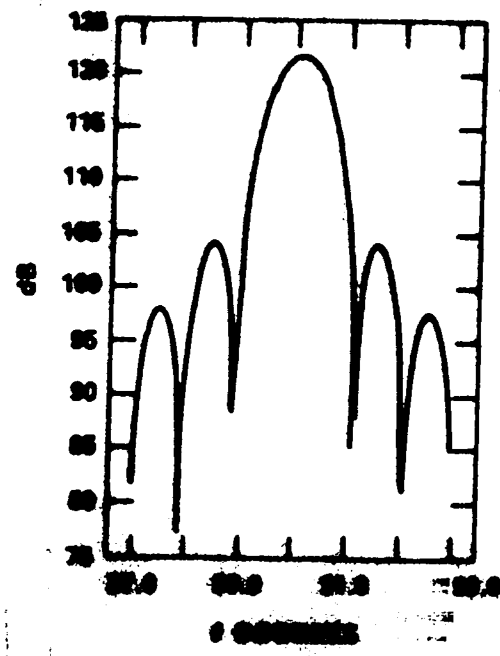

YTF = 2200.0
ZTR = −200.0
dB
110
105
100
95
90
91.0
93.0


YTF = 2200.0
ZTF = −300.0
dB
125
120
115
110
105
100
95
90
85
80
75
87.0
89.0
91.0
93.0
θ DEGREES

ORIGINAL PAGE IS
OF POOR QUALITY

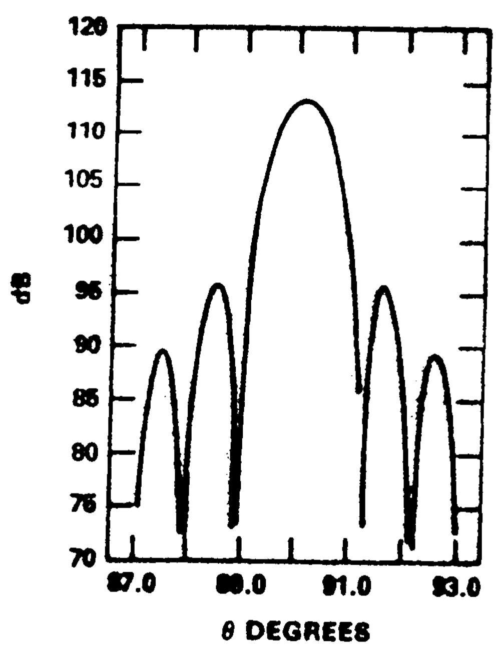






YTF = 2200.0

ZTF = −600.0

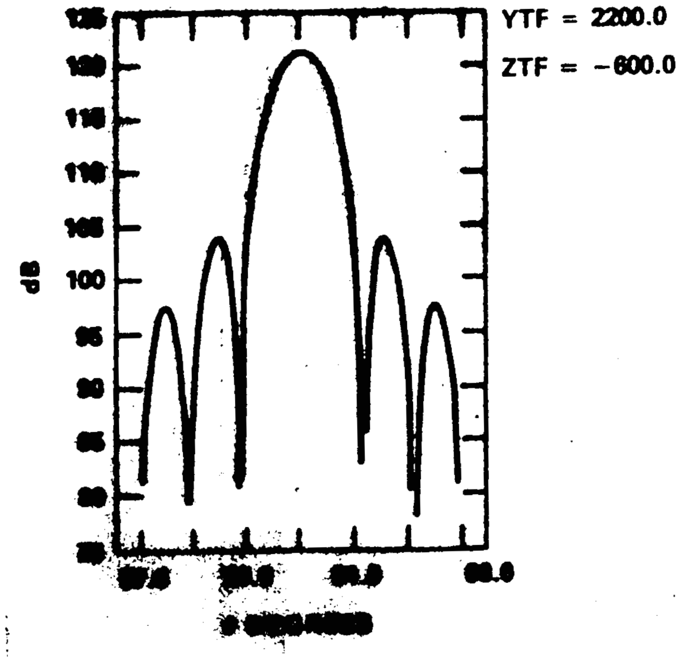


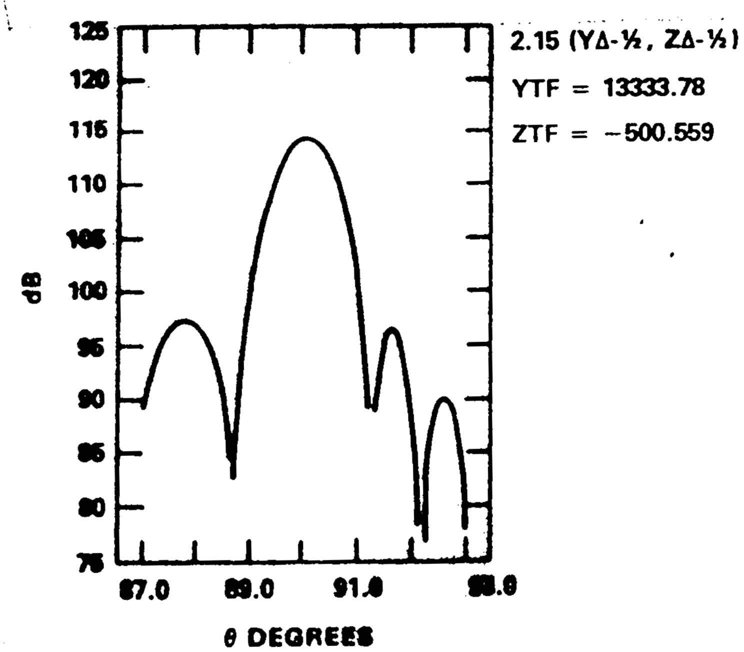
ALF$_2$ = 60.0, BET = 90.0, ZTF = 290.0

YTF = 2200.0

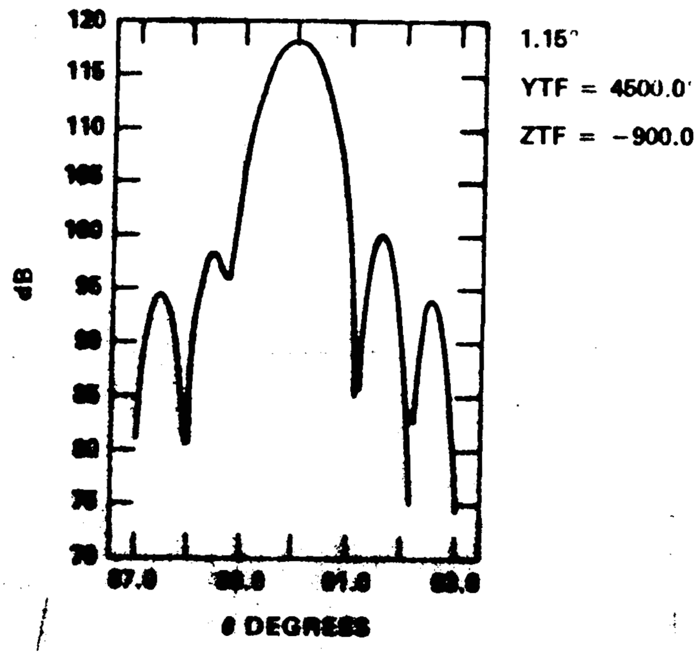
ZTF = −700.0

ORIGINAL PAGE IS
OF POOR QUALITY





ORIGINAL PAGE IS
OF POOR QUALITY



```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                   (ATMAIN  )

C***********************************************************************
C
C      PROGRAM OF ANTENNA-FENCE SIMULATION
C                TASK NO. 511
C               MARCH 27, 1986
C      GSFC ATR - RICHARD F. SCHMIDT
C      SAR TASK LEADER - DR. MACHAEL KAO
C      SAR TASK PERSONNEL - HWAI-SOON CHENG
C
C      MAIN PROGRAM : TO CONTROL THE LOGICAL FLOW OF THE PROGRAM
C
C***********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 KIX,KIY,KIZ,KFX,KFY,KFZ,KHX,KHY,KHZ
      COMMON /INPUT1/ SIGMAM,SIGMAO,XLI,FR,F
      COMMON /INPUT2/ ALPHAI,BETAI,DELTAI,ALF1,BET1,GAM1
      COMMON /INPUT3/ ALF2,BET2,GAM2,XTA,YTA,ZTA,XTF,YTF,ZTF
      COMMON /INPUTX/ ANGA,ANGB,DB,NANG,XDF
      COMMON /ANTO/   PIE,SPM,RAD,XK,XLM,PA,PB,PDB
      COMMON /ANTROP/ RIA(3,3),RIF(3,3),RFA(3,3),RAI(3,3),RFI(3,3)
      COMMON /ANTXP1/ KIX,KIY,KIZ,KFX,KFY,KFZ,KHX,KHY,KHZ
      COMMON /ANTXP2/ HIX,HIY,HIZ
      COMMON /ANTXP3/ ALPHA,BETA,SINALP,COSALP,SINBET,COSBET
      COMMON /ANTXP4/ SINDEL,COSDEL
      COMMON /ANTPA1/ XL(11993),YL(11993),ZL(11993),INDEX
      COMMON /ANTPA2/ XNI(11993),XNJ(11993),XNK(11993),DSS(11993)
      COMMON /ANTRAT/ RDPR,ZDPR,THETA
      COMMON /ANTPL/  GPR,GPI,GQR,GQI
      COMMON /ANTCY/  FHXR,FHXI,FHYR,FHYI,FHZR,FHZI
      COMMON /ANTRFA/ AHXR,AHXI,AHYR,AHYI,AHZR,AHZI
      COMMON /ANTRF1/ HXR(11993),HYR(11993),HZR(11993)
      COMMON /ANTRF2/ HXI(11993),HYI(11993),HZI(11993)
      COMMON /ANTPW1/ HXAR(11993),HXAI(11993),HYAR(11993)
      COMMON /ANTPW2/ HYAI(11993),HZAR(11993),HZAI(11993)
      COMMON /ANTLP1/ LOOP1
      CALL ATIN
      CALL ATDEF
      CALL ATEXPL
      CALL ATROP
      CALL ATXPW
      CALL ATAPER
      DO 1000 LOOP1 = 1,INDEX
              CALL ATRAT
              CALL ATPL
              CALL ATCY
              CALL ATRFA
 1000 CONTINUE
      DO 2000 LOOP2 = 1,NANG
         CALL ATPWH
         CALL ATCORR
         PB = PB - PDB
 2000 CONTINUE
      STOP
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                              (ATBKDT   )

      BLOCK DATA
C***********************************************************************
C
C     BLOCK DATA : TO SET UP ALL THE DEFAULT VALUES
C
C***********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /INPUT1/ SIGMAM,SIGMAO,XLI,FR,F
      COMMON /INPUT2/ ALPHAI,BETAI,DELTAI,ALF1,BET1,GAM1
      COMMON /INPUT3/ ALF2,BET2,GAM2,XTA,YTA,ZTA,XTF,YTF,ZTF
      COMMON /INPUTX/ ANGA,ANGB,DB,NANG,XDF
      DATA SIGMAM/450.0/,SIGMAO/0.0/
      DATA XLI/0.5/,FR/2.0/,F/500.0/
      DATA ALPHAI/90.0/,BETAI/0.0/,DELTAI/90.0/
      DATA ALF1/180.0/,BET1/90.0/,GAM1/90.0/
      DATA ALF2/90.0/,BET2/90.0/,GAM2/270.0/
      DATA XTA/0.0/,YTA/1000.0/,ZTA/0.0/
      DATA XTF/0.0/,YTF/2200.0/,ZTF/-700.0/
      DATA ANGA/0.0/,ANGB/93.0/,DB/0.1/,NANG/61/,XDF/1.0/
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                          (ATIN     )

      SUBROUTINE ATIN
C************************************************************************
C
C     SUBROUTINE ATIN : PROVIDING THE INPUT INFORMATION TO SPECIFY
C                       THE GEOMETRICAL AND PHYSICAL REQUIREMENTS
C                       FOR THE PROGRAM
C
C************************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /INPUT1/ SIGMAM,SIGMAO,XLI,FR,F
      COMMON /INPUT2/ ALPHAI,BETAI,DELTAI,ALF1,BET1,GAM1
      COMMON /INPUT3/ ALF2,BET2,GAM2,XTA,YTA,ZTA,XTF,YTF,ZTF
      COMMON /INPUTX/ ANGA,ANGB,DB,NANG,XDF
      NAMELIST /INPUT/ SIGMAM,SIGMAO,XLI,FR,F,
     1                 ALPHAI,BETAI,DELTAI,
     2                 ALF1,BET1,GAM1,ALF2,BET2,GAM2,
     3                 XTA,YTA,ZTA,XTF,YTF,ZTF,
     4                 ANGA,ANGB,DB,NANG,XDF
      WRITE (6,70)
      READ (5,INPUT,END=30,ERR=40)
   30 WRITE (6,80)
      GO TO 60
   40 WRITE (6,90)
   60 RETURN
   70 FORMAT (1H///1H,'READ IN PROGRAM INITIAL VALUES')
   80 FORMAT ('*** END OF PROGRAM INPUT DATA ***')
   90 FORMAT ('*** WARNING ... INCORRECT INPUT !!! ')
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                    (ATDEF    )

      SUBROUTINE ATDEF
C*****************************************************************
C
C     SUBROUTINE ATDEF : TO DEFINE CONSTANT VALUE AND CONVERT PHYSICAL
C                        UNITS
C
C*****************************************************************

      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /INPUT1/ SIGMAM,SIGMAO,XLI,FR,F
      COMMON /INPUT2/ ALPHAI,BETAI,DELTAI,ALF1,BET1,GAM1
      COMMON /INPUT3/ ALF2,BET2,GAM2,XTA,YTA,ZTA,XTF,YTF,ZTF
      COMMON /INPUTX/ ANGA,ANGB,DB,NANG,XDF
      COMMON /ANT0/   PIE,SPM,RAD,XK,XLM,PA,PB,PDB
      SPM = 2.997925D10
      XLM = SPM/(FR*1.0D9)
      PIE = 3.1415926536
      RAD = PIE/180.0
      XK = (2.0*PIE)/XLM
      ALPHAI = ALPHAI*RAD
      BETAI = BETAI*RAD
      DELTAI = DELTAI*RAD
      ALF1 = ALF1*RAD
      BET1 = BET1*RAD
      GAM1 = GAM1*RAD
      ALF2 = ALF2*RAD
      BET2 = BET2*RAD
      GAM2 = GAM2*RAD
      PA = ANGA*RAD
      PB = ANGB*RAD
      PDB = DB*RAD
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                   (ATEXPL    )

      SUBROUTINE ATEXPL
C***********************************************************************
C     SUBROUTINE ATEXPL : TO PRINT OUT SIGNIFICANT INPUT PARAMETERS
C                         FOR THE USER'S INFORMATION AND RECORDS
C
C***********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /INPUT1/ SIGMAM,SIGMAO,XLI,FR,F
      COMMON /INPUT2/ ALPHAI,BETAI,DELTAI,ALF1,BET1,GAM1
      COMMON /INPUT3/ ALF2,BET2,GAM2,XTA,YTA,ZTA,XTF,YTF,ZTF
      COMMON /ANTG/   PIE,SPM,RAD,XK,XLM,PA,PB,PDB
      COMMON /INPUTX/ ANGA,ANGB,DB,NANG,XDF
      WRITE (6,10)
   10 FORMAT ('1',15X,'NAMES AND DESCRIPTIONS OF THE PROGRAM VARIABLES')
      WRITE (6,20) SIGMAM
   20 FORMAT ('0',15X,'SIGMAM = ',F8.3,'   *** ANTENNA REFLECTOR RADIUS'
     1)
      WRITE (6,30) SIGMAO
   30 FORMAT ('0',15X,'SIGMAO = ',F8.3,'   *** INITIAL VALUE OF SIGMA')
      WRITE (6,40) XLI
   40 FORMAT ('0',15X,'XLI = ',F8.3,'   *** INTEGRATION CONTROL CONSTANT
     1      ')
      WRITE (6,50) XLM
   50 FORMAT ('0',15X,'XLM = ',F8.3,'   *** WAVELENGTH')
      WRITE (6,60)FR
   60 FORMAT ('0',15X,'FR = ',F8.3,'   *** FRENQUENCY IN GHZ UNITS')
      WRITE (6,70) F
   70 FORMAT ('0',15X,'F = ',F6.1,'   *** FOCAL LENGTH OF IDEAL PARABOLO
     1ID')
      WRITE (6,80)
   80 FORMAT ('0',15X,'PLANE-WAVE ANGLES OF ARRIVAL IN INITIAL SYSTEM')
      WRITE (6,90) ALPHAI/RAD
   90 FORMAT ('0',15X,'ALPHAI = ',F8.3,'   *** AZIMUTHAL ANGLE')
      WRITE (6,100) BETAI/RAD
  100 FORMAT (15X,' BETAI = ',F8.3,'   *** RIGHT ANGLE - POLAR ANGLE')
      WRITE (6,105)
  105 FORMAT ('0',15X,'POLARIZATION ANGLE')
      WRITE (6,110) DELTAI/RAD
  110 FORMAT ('0',15X,'DELTAI = ',F8.3)
      WRITE (6,120)
  120 FORMAT ('0',15X,'EULERIAN ANGLES FOR ROTATION')
      WRITE (6,130)
  130 FORMAT ('0',15X,'IN ANTENNA SYSTEM')
      WRITE (6,140) ALF1/RAD,BET1/RAD,GAM1/RAD
  140 FORMAT ('0',15X,'ALF1 = ',F8.3,'  BET1 = ',F8.3,'  GAM1 = ',F8.3)
      WRITE (6,150)
  150 FORMAT ('0',15X,'IN FENCE SYSTEM')
      WRITE (6,160) ALF2/RAD,BET2/RAD,GAM2/RAD
  160 FORMAT ('0',15X,'ALF2 = ',F8.3,'  BET2 = ',F8.3,'  GAM2 = ',F8.3)
      WRITE (6,170)
  170 FORMAT ('0',15X,'TRANSLATION VECTOR COMPONENTS')
      WRITE (6,180)
  180 FORMAT ('0',15X,'FROM INITIAL TO ANTENNA SYSTEM')
      WRITE (6,190) XTA,YTA,ZTA
  190 FORMAT ('0',15X,'XTA = ',F6.1,'  YTA =',F6.1,'  ZTA = ',F6.1)
      WRITE (6,200)
```

```
200 FORMAT ('0',15X,'FROM INITIAL TO FENCE SYSTEM')
    WRITE (6,210) XTF,YTF,ZTF
210 FORMAT ('0',15X,'XTF= ',E12.6,'  YTF = ',E12.6,'  ZTF = ',E12.6)
    WRITE (6,230) ANGA,ANGB,DB
230 FORMAT ('0',15X,'ANGA = ',F8.3,' ANGB = ',F8.3,' DB = ',F8.3)
    WRITE (6,240) NANG,XDF
240 FORMAT ('0',15X,'NANG = ',I5,' XDF = ',F8.4)
    RETURN
    END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                  (ATROP    )

      SUBROUTINE ATROP
C***********************************************************************
C     SUBROUTINE ATROP : TO PROVIDE THE ROTATION OPERATOR FOR THE
C                        USE IN ANTENNA-INITIAL, INITIAL-FENCE
C                        SYSTEM, AND VICE VERSA
C
C***********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /INPUT2/ ALPHAI,BETAI,DELTAI,ALF1,BET1,GAM1
      COMMON /INPUT3/ ALF2,BET2,GAM2,XTA,YTA,ZTA,XTF,YTF,ZTF
      COMMON /ANTROP/ RIA(3,3),RIF(3,3),RFA(3,3),RAI(3,3),RFI(3,3)
      CALL ANGROP (ALF1,BET1,GAM1,RAI)
      CALL ANGROP (ALF2,BET2,GAM2,RIF)
      CALL TRANSP (RAI,RIA)
      CALL TRANSP (RIF,RFI)
      CALL CROSS (RIA,RFI,RFA)
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                  (ANGROP   )

      SUBROUTINE ANGROP(A,B,C,T1)
C***********************************************************************
C
C     SUBROUTINE ANGROP : TO DEFINE THE ROTATION OPERATOR FOR
C                         THE GEOMETRICAL TRANSFORMATION
C
C***********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION T1(3,3)
      SA = DSIN(A)
      CA = DCOS(A)
      SB = DSIN(B)
      CB = DCOS(B)
      SC = DSIN(C)
      CC = DCOS(C)
      T1(1,1) = CC*CA-CB*SA*SC
      T1(1,2) = CC*SA+CB*CA*SC
      T1(1,3) = SC*SB
      T1(2,1) = -SC*CA-CB*SA*CC
      T1(2,2) = -SC*SA+CB*CA*CC
      T1(2,3) = CC*SB
      T1(3,1) = SB*SA
      T1(3,2) = -SB*CA
      T1(3,3) = CB
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                    (TRANSP   )

      SUBROUTINE TRANSP(X,T2)
C***********************************************************************
C
C     SUBROUTINE TRANSP : USING ROTATION OPERATOR TO GENERATE AN
C                         INVERSE MATRIX FOR ANTI-DIRECTIONAL
C                         TRANSFORMATION
C
C***********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION X(3,3),T2(3,3)
      T2(1,1) = X(1,1)
      T2(1,2) = X(2,1)
      T2(1,3) = X(3,1)
      T2(2,1) = X(1,2)
      T2(2,2) = X(2,2)
      T2(2,3) = X(3,2)
      T2(3,1) = X(1,3)
      T2(3,2) = X(2,3)
      T2(3,3) = X(3,3)
      RETURN
      END
```

**** TSO FOREGROUND HARDCOPY ****
DSNAME=XF.HSC.ANTENNA.FORT (CROSS)

```
      SUBROUTINE CROSS(Y,Z,T3)
C**********************************************************************
C
C     SUBROUTINE CROSS : TO PERFORM MATRIX MULTIPLICATION
C
C**********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION Y(3,3),Z(3,3),T3(3,3)
      T3(1,1) = Y(1,1)*Z(1,1)+Y(1,2)*Z(2,1)+Y(1,3)*Z(3,1)
      T3(1,2) = Y(1,1)*Z(1,2)+Y(1,2)*Z(2,2)+Y(1,3)*Z(3,2)
      T3(1,3) = Y(1,1)*Z(1,3)+Y(1,2)*Z(2,3)+Y(1,3)*Z(3,3)
      T3(2,1) = Y(2,1)*Z(1,1)+Y(2,2)*Z(2,1)+Y(2,3)*Z(3,1)
      T3(2,2) = Y(2,1)*Z(1,2)+Y(2,2)*Z(2,2)+Y(2,3)*Z(3,2)
      T3(2,3) = Y(2,1)*Z(1,3)+Y(2,2)*Z(2,3)+Y(2,3)*Z(3,3)
      T3(3,1) = Y(3,1)*Z(1,1)+Y(3,2)*Z(2,1)+Y(3,3)*Z(3,1)
      T3(3,2) = Y(3,1)*Z(1,2)+Y(3,2)*Z(2,2)+Y(3,3)*Z(3,2)
      T3(3,3) = Y(3,1)*Z(1,3)+Y(3,2)*Z(2,3)+Y(3,3)*Z(3,3)
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                   (ROT      )

      SUBROUTINE ROT(D,EX,EY,EZ,RX,RY,RZ)
C***********************************************************************
C
C     SUBROUTINE ROT : TO PERFORM ROTATION FROM ONE COORDINATE
C                      SYSTEM TO ANOTHER ONE
C
C***********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION D(3,3)
      RX = D(1,1)*EX+D(2,1)*EY+D(3,1)*EZ
      RY = D(1,2)*EX+D(2,2)*EY+D(3,2)*EZ
      RZ = D(1,3)*EX+D(2,3)*EY+D(3,3)*EZ
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                  (ATXPH    )

      SUBROUTINE ATXPH
C*************************************************************************
C
C     SUBROUTINE ATXPH : TO ROTATE THE PLANE-WAVE INCIDENT ANGLES AND
C                        POLARIZATION ANGLE FROM INTIAL TO FENCE
C                        SYSTEM
C
C*************************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 KIX,KIY,KIZ,KFX,KFY,KFZ,KHX,KHY,KHZ
      COMMON /INPUT2/ ALPHAI,BETAI,DELTAI,ALF1,BET1,GAM1
      COMMON /ANTO/   PIE,SPM,RAD,XK,XLM,PA,PB,PDB
      COMMON /ANTROP/ RIA(3,3),RIF(3,3),RFA(3,3),RAI(3,3),RFI(3,3)
      COMMON /ANTXP1/ KIX,KIY,KIZ,KFX,KFY,KFZ,KHX,KHY,KHZ
      COMMON /ANTXP2/ HIX,HIY,HIZ
      COMMON /ANTXP3/ ALPHA,BETA,SINALP,COSALP,SINBET,COSBET
      COMMON /ANTXP4/ SINDEL,COSDEL
      SAI = DSIN(ALPHAI)
      CAI = DCOS(ALPHAI)
      SBI = DSIN(BETAI)
      CBI = DCOS(BETAI)
      SDI = DSIN(DELTAI)
      CDI = DCOS(DELTAI)
      KIX = CAI*CBI
      KIY = SAI*CBI
      KIZ = SBI
      CALL ROT(RIF,KIX,KIY,KIZ,KFX,KFY,KFZ)
      BETA = DASIN(KFZ)
      ALPHA = DATAN2(KFY,KFX)
      SINALP = DSIN(ALPHA)
      COSALP = DCOS(ALPHA)
      SINBET = DSIN(BETA)
      COSBET = DCOS(BETA)
      HIX = CDI*(-SAI)+SDI*(-CAI*SBI)
      HIY = CDI*CAI+SDI*(-SAI*SBI)
      HIZ = SDI*CBI
      CALL ROT(RIF,HIX,HIY,HIZ,HFX,HFY,HFZ)
      SINDEL = HFZ/COSBET
      COSDEL = (HFY+SINDEL*SINALP*SINBET)/COSALP
      WRITE (6,20) COSDEL,SINDEL
   20 FORMAT ('0',15X,'COSDEL= ',F8.3,' SINDEL= ',F8.3)
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                  (ATAPER   )

      SUBROUTINE ATAPER
C***********************************************************************
C
C     SUBROUTINE ATAPER : TO SUBDIVIDE THE REFLECTOR APERTURE INTO
C                         SMALL DIFFERENTIAL AREAS AND EVALUATE
C                         THE COORDINATE, UNIT NORMAL, AND DIFFEREN-
C                         TIAL AREA FOR EACH SUBDIVISION AT APERTURE
C                         SURFACE
C
C***********************************************************************
      COMMON /INPUT1/ SIGMAM,SIGMAO,XLI,FR,F
      COMMON /ANTO/   PIE,SPM,RAD,XK,XLM,PA,PB,PDB
      COMMON /ANTPA1/ XL(11993),YL(11993),ZL(11993),INDEX
      COMMON /ANTPA2/ XNI(11993),XNJ(11993),XNK(11993),DSS(11993)
      DIMENSION SZEPR(11993),CZEPR(11993)
      DSIGO = XLM*XLI
      SIGMAP = SIGMAO
      XMII = (SIGMAM-SIGMAO)/DSIGO+1.0
      IXMII = XMII
      DSIG = (SIGMAM-SIGMAO)/IXMII
      INDEX = 1
      DO 500 IPB =1,IXMII
      SIGMA = SIGMAP+DSIG
      SIGMR = SIGMA-0.5*DSIG
      XMI = 2.0*PIE*SIGMA/DSIG/4.0+1.0
      JXM = XMI
      IXMI = JXM*4
      JJXM = 2*JXM
      JJJM = 4*JXM
      DZETA = PIE/JJXM
      ZEPR =-PIE/JJJM
      IN = 0
      DO 200 INN=1,JXM
      IN = IN+1
      ZEPR = ZEPR+DZETA
      SZEPR(IN) =DSIN(ZEPR)
      CZEPR(IN) =DCOS(ZEPR)
  200 CONTINUE
      JX2 = 2*JXM
      JXM2 = JXM+1
      NK = 0
      DO 210 IN2=JXM2,JX2
      SZEPR(IN2) = SZEPR(IN-NK)
      CZEPR(IN2) = -CZEPR(IN-NK)
      NK = NK+1
  210 CONTINUE
      IN = 0
      JX3 = 3*JXM
      JXM3 = 2*JXM+1
      DO 220 IN3 =JXM3,JX3
      IN = IN+1
      SZEPR(IN3) = -SZEPR(IN)
      CZEPR(IN3) = -CZEPR(IN)
  220 CONTINUE
      JX4 = 4*JXM
      JXM4 = 3*JXM+1
```

```
      NJ = 0
      DO 230 IN4=JXM4,JX4
      SZEPR(IN4) = -SZEPR(IN-NJ)
      CZEPR(IN4) = CZEPR(IN-NJ)
      NJ = NJ+1
  230 CONTINUE
      DO 300 JPB = 1,JJJM
         XL(INDEX) = SIGMR*SZEPR(JPB)
         YL(INDEX) = -SIGMR*CZEPR(JPB)
         ZL(INDEX) = 0.0
         PAR = DSQRT(SIGMR**2+4.0*F**2)
         XNI(INDEX) = -XL(INDEX)/PAR
         XNJ(INDEX) = -YL(INDEX)/PAR
         XNK(INDEX) = 2.0*F/PAR
         DS = PIE*(SIGMA**2-SIGMAP**2)/IXMI
         DSS(INDEX) = DS
         INDEX = INDEX+1
  300 CONTINUE
      SIGMAP = SIGMA
  500 CONTINUE
      INDEX = INDEX - 1
      WRITE (6,600) INDEX
  600 FORMAT('1',20X,'INDEX = ',I10)
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                  (ATRAT    )

      SUBROUTINE ATRAT
C*********************************************************************
C
C     SUBROUTINE ATRAT : TO TRANSFORM THE POINT IN THE CARTESIAN
C                        COORDINATE SYSTEM AT REFLECTOR APERTURE
C                        TO THE POINT IN THE SPHERICAL COORDINATE
C                        SYSTEM AT FENCE
C
C*********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /INPUT3/ ALF2,BET2,GAM2,XTA,YTA,ZTA,XTF,YTF,ZTF
      COMMON /ANTROP/ RIA(3,3),RIF(3,3),RFA(3,3),RAI(3,3),RFI(3,3)
      COMMON /ANTPA1/ XL( 11993),YL( 11993),ZL( 11993),INDEX
      COMMON /ANTRAT/ RDPR,ZDPR,THETA
      COMMON /ANTLP1/ LOOP1
C     WRITE (6,10) LOOP1
   10 FORMAT ('0',1X,I5)
      TAX = XL(LOOP1)
      TAY = YL(LOOP1)
      TAZ = ZL(LOOP1)
      CALL ROT(RAI,TAX,TAY,TAZ,TIX,TIY,TIZ)
      TIX = TIX+XTA-XTF
      TIY = TIY+YTA-YTF
      TIZ = TIZ+ZTA-ZTF
      CALL ROT(RIF,TIX,TIY,TIZ,TFX,TFY,TFZ)
      RDPR = DSQRT(TFX**2+TFY**2)
      ZDPR = TFZ
      THETA = DATAN2(TFY,TFX)
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                               (ATPL     )

      SUBROUTINE ATPL
C*********************************************************************
C
C     SUBROUTINE ATPL : TO COMPUTE THE PLANE-WAVE G(P) AND G(Q)
C                       VALUES AS PART OF SOMMERFELD SOLUTION
C
C*********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /ANTO/    PIE,SPM,RAD,XK,XLM,PA,PB,PDB
      COMMON /ANTXP3/ ALPHA,BETA,SINALP,COSALP,SINBET,COSBET
      COMMON /ANTRAT/ RDPR,ZDPR,THETA
      COMMON /ANTPL/  GPR,GPI,GQR,GQI
      DIMENSION GR(2),GI(2)
      TWOPI = 2.0*PIE
      ARG1 = PIE/4.0
      C1R = DCOS(ARG1)
      C1I = DSIN(ARG1)
      AR1 = (THETA+ALPHA)/2.0
      AR2 = (THETA-ALPHA)/2.0
      Q = -DSQRT(2.0*XK*RDPR*COSBET)*DCOS(AR1)
      P = -DSQRT(2.0*XK*RDPR*COSBET)*DCOS(AR2)
      DO 50 I = 1,2
      U = 1.0
      IF (-Q.LE.0.0) U = 0.0
      SGN = 1.0
      IF (Q.LT.0.0) SGN = -1.0
      ARG2 = Q**2
      AG2TP = ARG2/TWOPI
      IAG2TP = AG2TP
      RMD = AG2TP - IAG2TP
      AG2MD = RMD*TWOPI
      C2R = DCOS(AG2MD)
      C2I = -DSIN(AG2MD)
      SQPIE = DSQRT(PIE)
      SQPIE2 = DSQRT(PIE/2.0)
      SQ2PIE = DSQRT(2.0*PIE)
      C3R = SQPIE*U*(C2R*C1R-C2I*C1I)
      C3I = SQPIE*U*(C2R*C1I+C2I*C1R)
      CALL ATCS(C,S,ARG2,AG2MD)
      C4R = SQPIE2/2.-SQPIE2*C
      C4I = SQPIE2/2.-SQPIE2*S
      C5R = SGN*(C2R*C4R-C2I*C4I)
      C5I = SGN*(C2R*C4I+C2I*C4R)
      GR(I) = C3R+C5R
      GI(I) = C3I+C5I
      Q = P
   50 CONTINUE
      GQR = GR(1)
      GQI = GI(1)
      GPR = GR(2)
      GPI = GI(2)
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XXHSC.ANTENNA.FORT                                     (ATCS     )

      SUBROUTINE ATCS(C,S,X,XMD)
      IMPLICIT REAL*8 (A-H,O-Z)
      Z=ABS(X)
      ZMD=ABS(XMD)
    2 IF(Z-4.) 3,3,4
    3 C=DSQRT(Z)
      S=Z*C
      Z=Z*Z
      ZMD=ZMD*ZMD
      C=C*((((((.50998348E-10*Z-.10140729E-7)*Z+.11605284E-5)*Z
     1 -.85224622E-4)*Z+.36938586E-2)*Z-.079788405)*Z+.79788455)
      S=S*(((((-.66777447E-9*Z+.11225331E-6)*Z-.10525853E-4)*Z
     1+.60435371E-3)*Z-.18997110E-1)*Z+.26596149)
      RETURN
    4 D=DCOS(ZMD)
      S=DSIN(ZMD)
      Z=4./Z
      A=(((((((.87682583E-3*Z-.41692894E-2)*Z+.79709430E-2)*Z-
     1.67928011E-2)*Z-.30953412E-3)*Z+.59721508E-2)*Z-.16064281E-4)*Z-
     2.024933215)*Z-.44440909E-8
      B=((((((-.66339256E-3*Z+.34014090E-2)*Z-.72716901E-2)*Z+
     1.74282459E-2)*Z-.40271450E-3)*Z-.93149105E-2)*Z-.12079984E-5)*Z+
     2.1994711
      Z=DSQRT(Z)
      C=.5+Z*(D*A+S*B)
      S=.5+Z*(S*A-D*B)
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XR.HSC.ANTENNA.FORT                                (ATCY     )

      SUBROUTINE ATCY
C************************************************************************
C
C     SUBROUTINE ATCY : TO EVALUATE H-PLANE SOMMERFELD SOLUTION
C
C************************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /INPUTX/ ANGA,ANGB,DB,NANG,XDF
      COMMON /ANT0/   PIE,SPM,RAD,XK,XLM,PA,PB,PDB
      COMMON /ANTXP3/ ALPHA,BETA,SINALP,COSALP,SINBET,COSBET
      COMMON /ANTXP4/ SINDEL,COSDEL
      COMMON /ANTRAT/ RDPR,ZDPR,THETA
      COMMON /ANTPL/  GPR,GPI,GQR,GQI
      COMMON /ANTCY/  FHXR,FHXI,FHYR,FHYI,FHZR,FHZI
      ARG = PIE/4.0
      C1R = DCOS(-ARG)
      C1I = DSIN(-ARG)
      ARG3 = XK*(RDPR*COSBET-ZDPR*SINBET)
      C5R = DCOS(ARG3)
      C5I = DSIN(ARG3)
      C6R = (C5R*C1R-C5I*C1I)/DSQRT(PIE)
      C6I = (C5R*C1I+C5I*C1R)/DSQRT(PIE)
      IF (XDF.EQ.1.0) THEN
         C7TMP = DSQRT(2.0/(XK*RDPR*COSBET))
      ELSE
         C7TMP = 0.0
      END IF
      SAH = DSIN(ALPHA/2.0)
      CAH = DCOS(ALPHA/2.0)
      STH = DSIN(THETA/2.0)
      CTH = DCOS(THETA/2.0)
      C7ISC = C7TMP*SAH*CTH
      C7ISS = C7TMP*SAH*STH
      C7ICS = C7TMP*CAH*STH
      C7ICC = C7TMP*CAH*CTH
      C8R = SINALP*(GPR+GQR)
      C8I = SINALP*(GPI+GQI)+C7ISC
      HXER = -( C6R*C8R-C6I*C8I )
      HXEI = -( C6R*C8I+C6I*C8R )
       HXE = HXER**2+HXEI**2
      C9R = COSALP*(GPR-GQR)
      C9I = COSALP*(GPI-GQI)-C7ISS
      HYER = C6R*C9R-C6I*C9I
      HYEI = C6R*C9I+C6I*C9R
       HYE = HYER**2+HYEI**2
      HZER = 0.0
      HZEI = 0.0
      HE = DSQRT(HXE+HYE+HZE)
      C10R = COSALP*(GPR+GQR)*SINBET
      C10I = (COSALP*(GPI+GQI)+C7ICC)*SINBET
      HXHR = -( C6R*C10R-C6I*C10I )
      HXHI = -( C6R*C10I+C6I*C10R )
       HXH = DSQRT(HXHR**2+HXHI**2)
      C11R = SINALP*(GPR-GQR)*SINBET
      C11I = (SINALP*(GPI-GQI)+C7ICS)*SINBET
      HYHR = -( C6R*C11R-C6I*C11I )
```

```
      HYHI = -( C6R*C11I+C6I*C11R )
       HYH = DSQRT(HYHR**2+HYHI**2)
      C12R = (GPR+GQR)*COSBET
      C12I = (GPI+GQI)*COSBET
      HZHR = C6R*C12R-C6I*C12I
      HZHI = C6R*C12I+C6I*C12R
       HZH = DSQRT(HZHR**2+HZHI**2)
      FHXR = SINDEL*HXHR+COSDEL*HXER
      FHXI = SINDEL*HXHI+COSDEL*HXEI
      FHYR = SINDEL*HYHR+COSDEL*HYER
      FHYI = SINDEL*HYHI+COSDEL*HYEI
      FHZR = SINDEL*HZHR+COSDEL*HZER
      FHZI = SINDEL*HZHI+COSDEL*HZEI
      RETURN
      END
```

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                              (ATRFA    )

      SUBROUTINE ATRFA
C**********************************************************************
C
C     SUBROUTINE ATRFA : TO ROTATE H-PLANE SOMMERFELD SOLUTION
C                        FROM FENCE TO REFLECTOR APERTURE
C
C**********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /ANTROP/ RIA(3,3),RIF(3,3),RFA(3,3),RAI(3,3),RFI(3,3)
      COMMON /ANTCY/  FHXR,FHXI,FHYR,FHYI,FHZR,FHZI
      COMMON /ANTRFA/ AHXR,AHXI,AHYR,AHYI,AHZR,AHZI
      COMMON /ANTRF1/ HXR(11993),HYR(11993),HZR(11993)
      COMMON /ANTRF2/ HXI(11993),HYI(11993),HZI(11993)
      COMMON /ANTLP1/ LOOP1
      CALL ROT(RFA,FHXR,FHYR,FHZR,AHXR,AHYR,AHZR)
      CALL ROT(RFA,FHXI,FHYI,FHZI,AHXI,AHYI,AHZI)
      HXR(LOOP1) = AHXR
      HYR(LOOP1) = AHYR
      HZR(LOOP1) = AHZR
      HXI(LOOP1) = AHXI
      HYI(LOOP1) = AHYI
      HZI(LOOP1) = AHZI
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
**** TSO FOREGROUND HARDCOPY ****
DSNAME=XRHSC.ANTENNA.FORT                                  (ATCORR   )

      SUBROUTINE ATCORR
C***********************************************************************
C
C     SUBROUTINE ATCORR     TO CALCULATE THE CORRELATION OF THE
C                           PLANE-WAVE FUNCTION AND THE INCOMING
C                           DIFFRACTED ELECTROMAGNETIC FIELD AT
C                           ANTENNA APERTURE, THE TRANSMITTED
C                           REGION.
C
C***********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /INPUTX/ ANGA,ANGB,DB,NANG,XDF
      COMMON /ANTO/   PIE,SPM,RAD,XK,XLM,PA,PB,PDB
      COMMON /ANTPA1/ XL(11993),YL(11993),ZL(11993),INDEX
      COMMON /ANTPA2/ XNI(11993),XNJ(11993),XNK(11993),DSS(11993)
      COMMON /ANTRF1/ HXR(11993),HYR(11993),HZR(11993)
      COMMON /ANTRF2/ HXI(11993),HYI(11993),HZI(11993)
      COMMON /ANTPW1/ HXAR(11993),HXAI(11993),HYAR(11993)
      COMMON /ANTPW2/ HYAI(11993),HZAR(11993),HZAI(11993)
      VOCR = 0.0
      VOCI = 0.0
      DO 800 I = 1, INDEX
      RDNDSS = DSS(I)
      FNR = HXR(I)*HXAR(I) - HXI(I)*HXAI(I) + HYR(I)*HYAR(I)
     1        -HYI(I)*HYAI(I) + HZR(I)*HZAR(I) - HZI(I)*HZAI(I)
      FNI = HXR(I)*HXAI(I) + HXI(I)*HXAR(I) + HYR(I)*HYAI(I)
     2        +HYI(I)*HYAR(I) + HZR(I)*HZAI(I) + HZI(I)*HZAR(I)
      DVOCR = FNR * RDNDSS
      DVOCI = FNI * RDNDSS
      VOCR = VOCR + DVOCR
      VOCI = VOCI + DVOCI
  800 CONTINUE
      VOC = DSQRT(VOCR*VOCR + VOCI*VOCI)
      VOCL = 20.0*LOG10(VOC)
      WRITE (6,850) PB/RAD, VOC,VOCL
  850 FORMAT ('0',20X,'PB = ',F8.3,' VOC = ',E12.6,'   VOCL = ',E12.6)
      RETURN
      END
```

```
      SUBROUTINE ATPWH
C***********************************************************************
C
C     SUBROUTINE ATPWH : TO EVALUATE THE SHEET-CURRENT AT EVERY
C                        POINT ON THE REFLECTOR APERTURE
C
C***********************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /ANT0/   PIE,SPM,RAD,XK,XLM,PA,PB,PDB
      COMMON /ANTPA1/ XL(11993),YL(11993),ZL(11993),INDEX
      COMMON /ANTPA2/ XNI(11993),XNJ(11993),XNK(11993),DSS(11993)
      COMMON /ANTPW1/ HXAR(11993),HXAI(11993),HYAR(11993)
      COMMON /ANTPW2/ HYAI(11993),HZAR(11993),HZAI(11993)
      CPA = DCOS(PA)
      SPA = DSIN(PA)
      CPB = DCOS(PB)
      SPB = DSIN(PB)
      CACB = CPA * CPB
      CASB = CPA * SPB
      SACB = SPA * CPB
      SASB = SPA * SPB
      DO 300 LP= 1,INDEX
             PS = XL(LP)*CACB + YL(LP)*SACB + ZL(LP)*SPB
             PKS = XK * PS
             CKS = DCOS(PKS)
             SKS = DSIN(PKS)
             HXAR(LP) = -SPA * CKS
             HYAR(LP) = CPA * CKS
             HZAR(LP) = 0.0 * CKS
             HXAI(LP) = SPA * SKS
             HYAI(LP) = -CPA * SKS
             HZAI(LP) = 0.0 * SKS
  300 CONTINUE
      RETURN
      END
```