2500-398-M

RL # 854

TECHNICAL REPORT

SUBJECT: A Bandwidth Enhancement Method for Microstrip Antennas

FROM : P. B. Katehi

Abstract:

Bandwidth enhancement methods for electromagnetically coupled microstrip dipoles are discussed in this report. It is demonstrated that if parasitic metallic strips are incorporated in the structure either coplanar and parallel to the embedded microstrip transmission line open end, or between the transmission line and the microstrip dipole, then substantial bandwidth enhancement results. Experimental verification of this model is introduced for a bandwidth definition based on the frequency range which satisfies a VSWR < 2 criterion. Also, experimental $\bar{E}$- and $\bar{H}$-plane patterns verify the theoretical model which accounts for radiation from the microstrip dipole, the parasitics and the transmission line.

TABLE OF CONTENTS

I. Introduction

It is well known that microstrip antennas exhibit very narrow bandwidth characteristics [1] through [3]. Various schemes have been proposed and/or implemented to reduce or eliminate this limitation with relatively promising results [4] through [6]. A dominant theme in these schemes is the introduction of additional capacitance which is accomplished by integrating parasitic metallic strips in the microstrip antenna structure. This is usually accomplished by stacking parasitic disks under a microstrip circular patch or a parasitic rectangular patch under the microstrip patch antenna and/or by increasing the overall substrate thickness.

This work is concerned with the bandwidth improvement of electromagnetically coupled microstrip dipoles by adapting one or two parasitic metallic strips either in a stacked fashion between the embedded microstrip transmission line open end and the radiating microstrip dipole antenna or coplanar to and near the open end of the embedded microstrip transmission line. Both of the configurations just mentioned are shown for reference in Fig. 1. For both arrangements an optimization procedure may be carried out, as to the optimum position and size of the parasitics, in order to maximize bandwidth. The methodology adopted in this work for the evaluation and optimization of the electromagnetically coupled microstrip dipole bandwidth, accounts for all the substrate effects including surface wave propagation [7] through [12]. The embedded microstrip transmission line excitation is effectively taken into account by considering it as part of the antenna, i.e., the overall module of microstrip transmission line, its excitation and the microstrip dipole are treated as a radiating system. The model allows, in addition, finite conductor thickness. The transmission line and dipole widths are taken to be a fraction of the free space wavelength λ_0, so that the longitudinal current component is the dominant contributor to the radiating system characteristics. With this restriction the transverse current component is negligibly small and its effect is omitted without introducing appreciable error, as verified in [10]. The longitudinal current distribution $J_x(x,y)$ is determined by applying Galerkin's method in the $\hat{x}$-direction of the system insuring that all interactions between the transmission line, microstrip dipole and parasitics are included; this will yield the $\hat{x}$-dependence of $J_x(x,y)$ as provided by the model. The $\hat{y}$-dependence of $J_z(x,y)$ is chosen to satisfy the edge condition at the effective width location [11]. Upon determining the current distribution, transmission line theory is invoked to evaluate the reflection coefficient and VSWR and their dependence on frequency. The bandwidth, defined here as the frequency range for which VSWR $\leq$ 2, can be found therefore as a function of the characteristics of the strip dipoles and substrate parameters. Finally, an experiment is also carried out which verifies the model with good agreement.

II. Analytical Aspects of the Model

Success in the theoretical investigation of printed circuit antennas is contingent upon a variety of factors; two of the most important factors are the accuracy of the developed mathematical model and the usefulness of the derived results. The former factor is determined by the implemented analytical method, while the latter depends significantly on the assumptions adopted with respect to the excitation mechanism of the radiating system. The following sections describe the analytical method used for the computation of the current distribution as well as the assumptions involved for the excitation mechanism.

A. Current Distribution Evaluation

The transmission line and the dipoles radiate an electric field E_r given by

$$\vec{E}_r(\vec{r}) + \vec{E}_i(\vec{r}) = \iint_S \bar{\bar{G}}(\vec{r}/\vec{r}\,') \cdot \vec{J}(\vec{r}\,')ds' \qquad (1)$$

where $\vec{E}_i(\vec{r})$ is the impressed field at the point $\vec{r} = (r,\theta,\phi)$, $\bar{\bar{G}}(\vec{r}/\vec{r}\,')$ is the dyadic Green's function and $\vec{J}(\vec{r}\,')$ is the unknown current distribution at the point $\vec{r}\,' = (\vec{r}\,',\theta' = \pi/2,\phi')$. The Green's function, pertinent to this problem, is given by the expression

$$\bar{\bar{G}}(\vec{r}/\vec{r}\,') = \int_0^\infty [k_0^2\bar{\bar{I}} + \bar{\nabla}\bar{\nabla}] \cdot J_0(\lambda|\vec{r} - \vec{r}\,'|)\bar{\bar{F}}(\lambda)\, d\lambda \qquad (2)$$

where $\bar{\bar{I}}$ is the unit dyadic, $k_0 = 2\pi/\lambda_0$ and $\bar{\bar{F}}(\lambda)$ a known dyadic function of the form

$$\bar{\bar{F}}(\lambda) = \frac{\bar{\bar{A}}(\lambda,\varepsilon_r,h)}{f_1(\lambda,\varepsilon_r,h)f_2(\lambda,\varepsilon_r,h)} \quad . \qquad (3)$$

In Eq. (3), $\bar{\bar{A}}$, f_1 and f_2 are analytic functions of the spatial frequency λ, the relative dielectric constant ε_r and the substrate thickness h, [10]-[11]. Specifically, f_1 and f_2 are in the form

$$f_1(\lambda,\varepsilon_r,h) = u_0 \sinh(uh) + u \cosh(uh) \qquad (4)$$

and

$$f_2(\lambda,\varepsilon_r,h) = \varepsilon_r u_0 \cosh(uh) + u \sinh(uh) \qquad (5)$$

with

$$u_0 = [\lambda^2 - k_0^2]^{1/2} \quad \text{and} \quad u = [\lambda^2 - \varepsilon_r k_0^2]^{1/2} \quad . \qquad (6)$$

The zeros of $f_1(\lambda,\varepsilon_r,h)$ correspond to TE surface waves while the zeros of $f_2(\lambda,\varepsilon_r,h)$ to TM surface waves respectively and their existence affects considerably the coupling between the dipoles and the microstrip line.

The widths of the strip conductors are fractions of the wavelength in the substrate. For this reason, it can be assumed that the currents are unidirectional and parallel to the $\hat{x}$-axis. Under this assumption, the current distribution in Eq. (1) may be written in the form

$$\vec{J}(\vec{r}\,') = \hat{x} f(x')g(y') \quad , \tag{7}$$

where $f(x')$ is an unknown function of x' and $g(y')$ is assumed to be given by

$$g(y') = \frac{2}{\pi w_e}\left\{1 - \left(\frac{2y'}{w_e}\right)^2\right\}^{-1/2} . \tag{8}$$

In Eq. (8), w_e is the effective strip width given by $w_e = w + 2\delta$ with δ the excess half-width. The effective strip width accounts for fringing effects due to conductor thickness [11].

From Eq. (1) the unknown current density $\vec{J}$ is evaluated by the application of the method of moments. Each section of the strip conductors is divided into a number of segments and the current is written as a finite sum

$$\vec{J}(\vec{r}\,') = \hat{x}\, g(y') \sum_{n=1}^{N} I_n f_n(x') \tag{9}$$

where N is the total number of segments considered. The expansion functions in Eq. (9) have been chosen to be piecewise sinusoidal and they are represented by

$$f_n(x') = \begin{cases} \dfrac{\sin[k_o(x' - x_{n-1})]}{\sin(k_o \ell_x)} & , \quad x_{n-1} \leq x' \leq x_n \\ \dfrac{\sin[k_o(x_{n+1} - x')]}{\sin(k_o \ell_x)} & , \quad x_n \leq x' \leq x_{n+1} \\ 0 & , \quad \text{elsewhere} \end{cases} \tag{10}$$

with ℓ_x denoting the length of each subsection.

The electric field given by Eq. (1) is projected along the axis y = 0, z = 0 using as weighting functions the basis functions (Galerkin's method). In this manner, Eq. (1) reduces to a matrix equation of the form

$$\underset{N\times N}{[Z_{mn}]} \ \underset{N\times 1}{[I_n]} = \underset{N\times 1}{[V_m]} \tag{11}$$

where $[I_n]$ is the unknown vector and $[V_m]$ is the excitation vector the latter depending critically on the impressed feed model. $[Z_{mn}]$ is the impedance matrix with elements given by

$$Z_{mn} = \delta(y)\delta(z) \int_{-w/2}^{w/2} \frac{dy'}{\left\{1 - \left(\frac{2y'}{w_e}\right)^2\right\}^{1/2}}$$

$$\cdot \int_c dx \int_c dx' \left\{k_o^2 F_{xx} + \frac{\partial^2}{\partial x^2}(F_{xx} - F_{zx})\right\} f_m(x) f_n(x') \tag{12}$$

where

$$F_{xx} = 2\left(\frac{j\omega\mu_0}{4\pi k_0^2}\right)\int_0^\infty \left(\frac{\sinh(uh)}{f_1(\lambda,\varepsilon_r,h)}\right) J_0(\lambda\rho)e^{-u_0 t}\, d\lambda \tag{13}$$

and

$$F_{zx} = 2\left(\frac{j\omega\mu_0}{4\pi k_0^2}\right)(\varepsilon_r - 1)\int_0^\infty \left(\frac{u_0 \cosh(uh)}{f_1(\lambda,\varepsilon_r,h)}\right)\left(\frac{\sinh(uh)}{f_2(\lambda,\varepsilon_r,h)}\right) J_0(\lambda\rho)e^{-u_0 t}\, d\lambda \tag{14}$$

With this expression for the elements of the impedance matrix, one can solve equation (11) for the unknown coefficients of the current distribution $\vec{J}(\vec{r}')$.

B. Excitation Mechanism

One of the difficulties encountered in the solution of this problem is the implementation of a practical excitation mechanism which can be included in the mathematical model. In most applications the microstrip line is kept very close to the ground and is excited by a coaxial line of the same characteristic impedance. As a result, a unimodal field propagates under the transmission line and the current distribution on the line, beyond an appropriate reference plane, forms standing waves of a TEM-like mode (see Fig. 2(a)). Under this assumption and for the case of zero reflections from the coax-to-microstrip transition, the microstrip line can be approximated by an ideal transmission line of the same characteristic impedance Z_0 terminated to an unknown self impedance Z_s (see Fig. 2(b)). Furthermore, the coax can be substituted by a voltage generator V_0 with internal impedance Z_0 as shown in Fig. 2(c). Since the reflection coefficient is independent of the generator's internal impedance, the line at the excitation end is left open. From the above, one can see that a voltage gap generator placed near the end of an open microstrip line can serve as a very simple and very practical excitation (see Fig. 2(d)) for developing a useful model. Another method is to eject an incident TEM mode on the microstrip transmission line [14],[15]. The former excitation mechanism is adopted here for the solution of Pocklington's integral equation and results in an excitation vector $[V_m] = [\delta_{im}]$ with

$$\delta_{im} = \begin{cases} 1 \ , & \text{at the position of the gap generator} \\ 0 \ , & \text{anywhere else} \end{cases} .$$

The quasi-TEM mode considered on the microstrip line is related to the dominant component of the current distribution derived by the method of moments. If the origin of the x-coordinate is taken at the position of Z_s, then the self impedance, normalized with respect to the characteristic impedance at the position of the first current maximum d_{max}, is given by

$$Z_s = \frac{1 + \Gamma(0)e^{j2\beta d_{max}}}{1 - \Gamma(0)e^{j2\beta d_{max}}} ,$$

where

$$\Gamma(0) = -\frac{SWR - 1}{SWR + 1} e^{j\beta|x_{max}|}$$

and x_{max} is the position of a maximum.

C. Radiation Patterns

The electromagnetic field radiated by the microstrip dipole can be readily evaluated by a saddle point method of integration applied to the expression for the total electric field. The computation of the far field pattern involves the contribution not only from the microstrip dipole but also from the parasitic strip or strips and the embedded transmission line, i.e., from the entire module. The complete radiated electric field is therefore given by

$$E_\theta = k_o^2 \Pi_\theta \quad \text{and} \quad E_\phi = k_o^2 \Pi_\phi \quad ,$$

where

$$\Pi_{\theta,\phi} = \Pi^d_{\theta,\phi} + \Pi^p_{\theta,\phi} + \Pi^{TL}_{\theta,\phi} \quad .$$

Furthermore, the Hertzian potential is denoted by

$$\Pi^s_{\theta,\phi} = \begin{matrix} \Pi^d_{\theta,\phi} \ , & (s = d \text{ for dipole}) \\ \Pi^p_{\theta,\phi} \ , & (s = p \text{ for parasitic}) \\ \Pi^{TL}_{\theta,\phi} \ , & (s = TL \text{ for transmission line}) \end{matrix}$$

with

$$\Pi^s_\theta = -\frac{j\omega\mu_o}{\pi k_o} \frac{e^{-jrk_o}}{r} I \frac{\cos(k_o \ell_x \sin\theta \cos\phi) - \cos(k\ell_x)}{k \sin(k\ell_x)\left[1 - \frac{k_o^2}{k} \sin^2\theta \cos^2\phi\right]}$$

$$\cdot \cos\phi\{\cos\theta \, F^s(\theta) + (\varepsilon_r - 1)\sin\theta \, S^s(\theta)\} \cdot \sum_{n=1}^{N_s} I^s_n$$

$$\cdot \, e^{jk_o(n\ell_x)\sin\theta\cos\phi} \cdot e^{jk_o(f^s)\sin\theta\sin\phi} \cdot e^{jk_o a^s \sin\theta\cos\phi} \qquad (15)$$

$$\Pi_\phi^s = \frac{j\omega\mu_o}{\pi k_o} \frac{e^{-jrk_o}}{r} I \frac{\cos(k_o \ell_x \sin\theta\cos\phi) - \cos(k\ell_x)}{k\sin(k\ell_x)\left[1 - \frac{k_o^2}{k^2}\sin^2\theta\cos^2\phi\right]}$$

$$\cdot \sin\phi\, F^s(\theta) \cdot \sum_{n=1}^{N_s} I_n^s\, e^{jk_o(n\ell_x)\sin\theta\cos\phi} \cdot e^{-jk_o(f^s)\sin\phi\sin\theta}$$

$$\cdot\, e^{jk_o a^s \sin\theta\cos\phi} \qquad (16)$$

where

$$I = \left\{ J_o\left(k_o \frac{w_e}{2}\sin\phi\sin\theta\right) - \frac{2}{\pi}\int_0^{u=\cos^{-1}(w/w_e)} \cos\left(k\frac{w_e}{2}\sin\phi\sin\theta\cos\sigma\right) d\sigma \right\}, \qquad (17)$$

$$F^d(\theta) = \frac{\cos\theta}{k_o} \frac{1}{\cos\theta - j\sqrt{\varepsilon_r - \sin^2\theta}\,\cot(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,h)} \qquad (18)$$

$$S^d(\theta) = F^d(\theta) \frac{\sin\theta}{\varepsilon_r\cos\theta + j\sqrt{\varepsilon_r - \sin^2\theta}\,\tan(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,h)} \qquad (19)$$

$$F^p(\theta) = \frac{\sin(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,(h - \delta))}{\sin(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,h)} F^d(\theta), \qquad (20)$$

$$S^p(\theta) = \frac{\sin(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,(h - \delta))}{\sin(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,h)} S^d(\theta), \qquad (21)$$

$$F^{TL}(\theta) = \frac{\sin(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,(h - b_s))}{\sin(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,h)} F^d(\theta), \qquad (22)$$

and

$$S^{TL}(\theta) = \frac{\sin(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,(h - b_s))}{\sin(k_o\sqrt{\varepsilon_r - \sin^2\theta}\,h)} S^d(\theta) \quad . \tag{23}$$

In addition

$$N^S + 1 = \begin{cases} N^d + 1 \equiv \text{number of dipole subsections} \\ N^p + 1 \equiv \text{number of parasitic strip subsections} \\ N^{TL} + 1 \equiv \text{number of transmission line subsections} \end{cases} ,$$

where ℓ_x denotes the length of each subsection and a^S the longitudinal displacement of the dipole or parasitic from the reference plane. For the particular applications under consideration in this article, f^S is the offset of the dipole or parasitic strip longitudinal centerline from the transmission line centerline and here it is chosen to be $f^S = 0$, i.e., all the metallic strips are aligned. Also, $a^d = a^p = a$ and $a^{TL} = 0$, while w and w_e denote the physical and effective strip widths, respectively.

D. Numerical and Experimental Results

Two different structures are investigated in this section for bandwidth improvement. The first case involves a parasitic strip located between the radiating dipole and the transmission line (see Fig. 1(a)) while the second case involves one or two parasitics embedded on the plane of the transmission line (see Fig. 1(c)). The bandwidth for these structures is defined by

$$BW = 2 \frac{f_{max} - f_{min}}{f_{max} + f_{min}} \% ,$$

where f_{max}, f_{min} denote the maximum and minimum frequencies for which VSWR $\leq$ 2. For the computational results to be shown the substrate is duroid with ε_r = 2.35 and the microstrip transmission line is kept at a distance 0.024 λ_0 from the ground plane.

Case 2

With reference to Fig. 1(a), the distance δ is varied between zero and b_s. The VSWR and equivalent impedance are evaluated as functions of frequency for each δ. Figure 3 shows the observed variation in the normalized self impedance with and without the parasitic for different frequencies and δ. For these calculations h = 0.1 λ_0, w = 0.05 λ_0 and the microstrip dipole and parasitic strip lengths are L = 0.280 λ_0.

The bandwidth and the minimum achieved VSWR are plotted as functions of the ratio δ/b_s for two different values of h (0.1 λ_0, 0.15 λ_0) with the corresponding dipole lengths equal to 0.280 λ_0 and 0.2906 λ_0 respectively (see Fig. 4). From this figure, one can see that as the substrate becomes thinner the optimum bandwidth becomes smaller and the input match worse. Also, as the substrate thickness changes from 0.1 λ_0 to 0.15 λ_0 the range of δ for VSWR $\leq$ 2 becomes smaller and is shifted towards higher values. Therefore, there is a maximum value h_{max} such that for every h larger than h_{max} the VSWR is always larger than two. It is expected that the addition of another dipole between the parasitic and the printed dipole will further reduce the VSWR and will increase the bandwidth.

Case b. <u>Parasitics of the Same Level with the Transmission Line</u>

Two different positions for the parasitic dipoles are investigated. At first, one dipole is placed along the transmission line as shown in Fig. 1(b). For this structure, the VSWR is evaluated as a function of the printed dipole length L_d, the parasitic dipole length L_p and the distance d. As an example, the variation of VSWR with L_d and d/L_d is shown in Fig. 5. However, the VSWR remains always larger than two. Subsequently, two parasitic dipoles are included and they are placed on each side of the transmission line (see Fig. 1(c)). The two parasitics have the same length, same width, same offset and overlap with respect to the transmission line. The VSWR and the self impedance Z_s are evaluated as a function of the parasitic dipole length L_p and they are plotted in Figs. 6 and 7 for $\varepsilon_r = 2.35$, $h = 0.1\lambda_o$, $b_s = 0.07587\ \lambda_o$, $w = 0.05\ \lambda_o$ and for three different lengths of the printed dipole at a frequency f = 10 GHz. From these figures, one can see that as the length of the parasitic becomes less than the printed-dipole length the VSWR goes asymptotically to values larger than two. However, there is a range of values for L_p which gives VSWR < 2 at f = 10 GHz. For six of these values and for $L_d = 0.2906\ \lambda_o$, the self impedance Z_s is evaluated at different frequencies (9.37 to 10.41 GHz) as shown in Fig. 8. From these values, the bandwidth defined by VSWR < 2 is evaluated as a function of L_p and is plotted in Fig. 9. Figures 8, 9 show that there is a specific L_p which will give VSWR = 1 at a frequency slightly above 10 GHz.

An experiment has also been performed to corroborate the theoretical model. The test fixture in this case is a collection of five duroid boards each with $\varepsilon_r = 2.17$. The overall thickness is roughly 120 mils and the boards are adhered together with vaseline. The embedded transmission line is at 30 mil, the parasitic dipole when present at 70 mil and the microstrip dipole at 120 mil from ground respectively. When there is no parasitic the bandwidth of the structure is zero, i.e., the requirement of VSWR ≤ 2 is not satisfied throughout the frequency range of interest. If the parasitic is included (in this case of the same width and length as the microstrip dipole and directly beneath it) then the theoretically calculated bandwidth, for the parameters cited in Fig.10 , is 11.6 percent, while the experimentally determined bandwidth is 10.4 percent. In addition, a frequency shift in the f_1 and f_2 points is observed between theory and experiment. Nevertheless, it is believed this is in very good agreement with the discrepancy between theory and experiment being attributed to tolerance errors in preparing the test fixture, the dissimilarity in ε_r between the duroid boards and the vaseline used to adhere them together as well as due to the contribution of a standing surface wave pattern which arises as a result of the finiteness in size of the experimental substrate boards. In addition, some error may be introduced by the coax-to-microstrip transmission line transitions which is not accounted for in the theoretical model.

Case C. Experimental Results--Radiation Patterns

The $\bar{E}$- and $\bar{H}$-plane radiation patterns of the previously discussed test fixture have also been measured at a frequency f = 9.98 GHz and the results are shown in Figs. 11 and 12. The corresponding theoretically calculated patterns by using Eqs. (15) through (23) are also superimposed for comparison. It is observed that the agreement between experiment and theory is nearly excellent in the $\bar{H}$-plane while there is some discrepancy in the $\bar{E}$-plane. It is important to note here that the theoretical model applies to an infinite substrate structure. The experimental structure, on the other hand, is finite in extent, as mentioned in the previous paragraph, and therefore diffraction from the edges will contribute to the shape of the $\bar{E}$- and $\bar{H}$-plane patterns. This contribution has been reduced by using absorbing material about the perimeter of the experimental fixture. Nevertheless, it is very difficult to eradicate these effects especially in the $\bar{E}$-plane. The physical explanation for this comment is as follows. It has been demonstrated previously [11], [13] that the TM and the TE substrate surface waves excited by a microstrip dipole influence the $\bar{E}$-plane and $\bar{H}$-plane patterns, respectively. The relative permittivity constant and material thickness of the substrate used for this test allow only the fundamental TM surface wave mode to exist. This mode propagates in the $\hat{x}$-direction (along the embedded transmission line-microstrip dipole axis) and diffracts at the substrate edges to influence the $\bar{E}$-plane radiation pattern substantially. The $\bar{H}$-plane radiation pattern is relatively clean since TE surface wave modes are not excited in this case and therefore there are no TE surface wave diffraction effects on this plane. The space wave diffraction effects at the board edges are much weaker than those due to surface waves and their contribution has been minimized substantially by the absorber layer affixed along the perimeter of the substrate board.

Conclusions

A theoretical model to improve the bandwidth of electromagnetically coupled microstrip antennas has been developed. The model either incorporates parasitic metallic strips between the microstrip dipole and the excitation microstrip transmission line embedded in the substrate or it involves parasitic strips coplanar to and adjacent to the open end of the microstrip transmission line. Substantial bandwidth improvement has been obtained for either configuration. Experimental corroboration of the theoretical model has been obtained with very satisfactory results.

References

[1] A.G.Derneryd and A. G. Lind, "Extended analysis of rectangular microstrip antennas," IEEE Trans. on Antennas and Propagat., Vol. AP-27, November 1979, pp. 846-849.

[2] J. H. Vandensonde and A. Van de Capelle, "Calculation of the bandwidth of microstrip resonator antennas," Proc. 9th European Microwave Conference, 1979, pp. 116-119.

[3] Y. T. Lo, D. Solomon and W. F. Richards, "Theory and experiment on microstrip antennas," IEEE Trans. on Antennas and Propagat., Vol. AP-27, No. 2, March 1979, pp. 137-145.

[4] H. G. Oltman and D. A. Huebner, "Electromagnetically coupled microstrip dipoles," IEEE Trans. Antennas Propagat., Vol. AP-29, January 1981, pp. 151-157.

[5] R. S. Elliott and G. J. Stern, "The design of microstrip dipole arrays including mutual coupling, Part I: Theory," IEEE Trans. on Antennas and Propagat., Vol. AP-29, pp. 757-760.

[6] G. J. Stern and R. S. Elliott, "The design of microstrip dipole arrays including mutual coupling, Part II: Experiment," IEEE Trans. Antennas and Propagat., Vol. AP-29, September 1981, pp. 761-765.

[7] P. B. Katehi and N. G. Alexopoulos, "On the effect of substrate thickness and permittivity on printed circuit dipole properties," IEEE Trans. on Antennas and Propagat., Vol. AP-31, January 1983, pp. 34-39.

[8] P. B. Katehi and N. G. Alexopoulos, "Real axis integration of Sommerfeld Integrals with applications to printed circuit antennas," Journal of Mathematical Physics, Vol. 24, No. 3, March 1983, pp. 527-533.

[9] N. G. Alexopoulos, P. B. Katehi and D. B. Rutledge, "Substrate optimization for integrated circuit antennas," IEEE Trans. Microwave Theory and Techniques, Vol. Mtt-31, No. 7, July 1983,pp. 550-557.

[10] P. B. Katehi, Ph.D. dissertation, University of California, Los Angeles, June 1984.

[11] P. B. Katehi and N. G. Alexopoulos, "On the modeling of electromagnetically coupled microstrip antennas--The printed strip dipole," IEEE Trans. on Antennas and Propagat., Vol. AP-32, No.11, Novem. 1984, pp. 1179-1185.

[12] P. B. Katehi and N. G. Alexopoulos, "Frequency-dependent characteristics of microstrip discontinuities in millimeter wave integrated circuits," IEEE Trans. on Microwave Theory and Techniques, Vol.33, No.10, October 1985, pp. 1029-1035.

[13] N. G. Alexopoulos, D. R. Jackson and P. B. Katehi, "Criteria for nearly omnidirectional radiation patterns for printed antennas," IEEE Trans. on Antennas and Propagation, Vol. AP-33, No.2, Febr. 1985, pp. 195-205.

[14] J. Castaneda, "A frequency dependent characteristic impedance for an open-circuit transmission line," UCLA Integrated Electromagnetics Laboratory, internal memorandum, June 1984.

[15] R.W Jackson and D. M. Pozar, "Full-wave analysis of microstrip open-end and gap discontinuities," IEEE Trans. on Microwave Theory and Techniques, Vol. 33, No. 10, Oct. 1985, pp. 1036-1042.

ACKNOWLEDGMENT

The authors would like to thank G. Anttila and J. Castaneda for their help in the experimental part of this research.

This research was supported by Northrop. Corp. under Research Grant 84-110-1006.

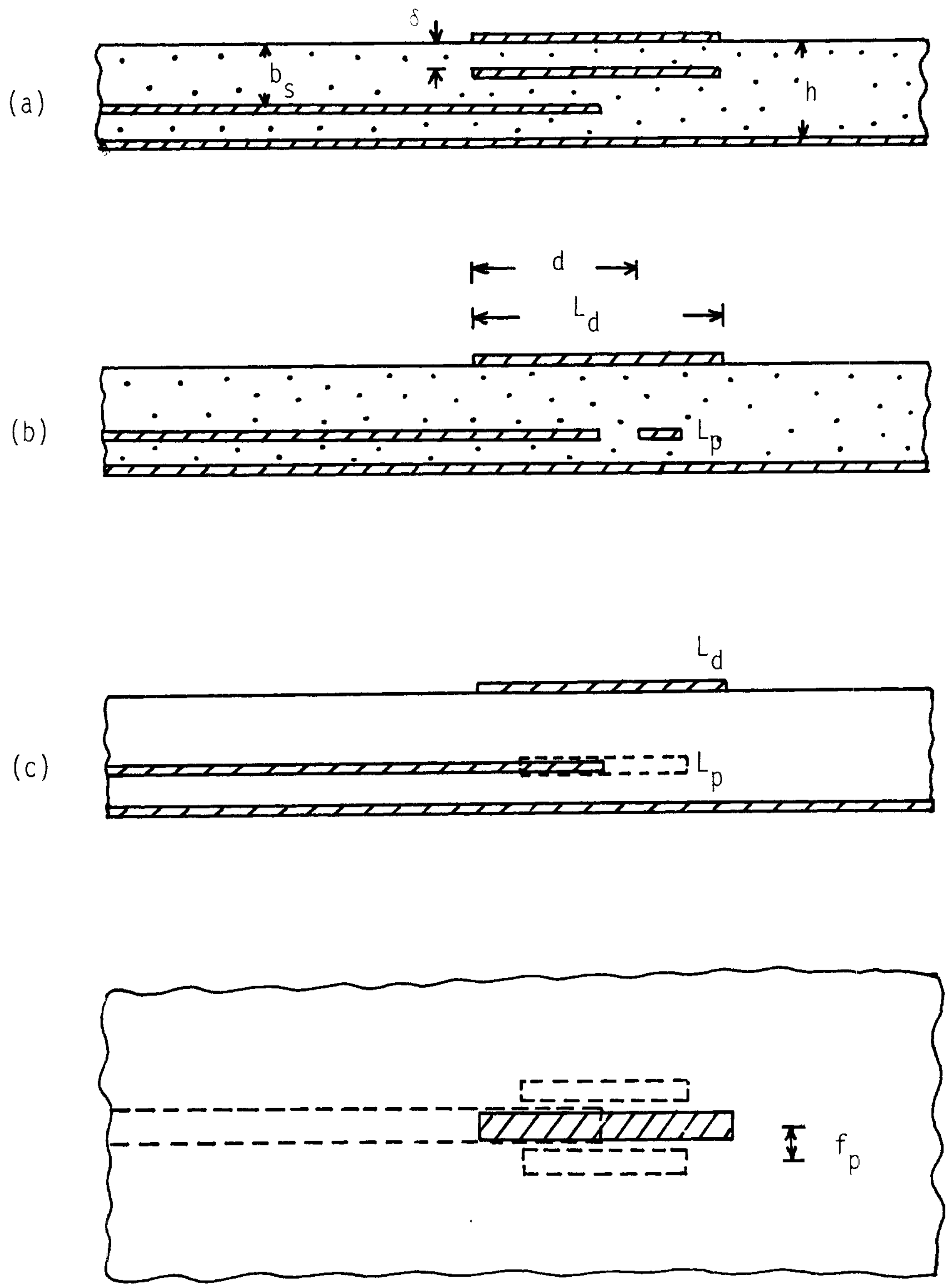


Figure 1: Microstrip Configurations for Bandwidth Improvement.

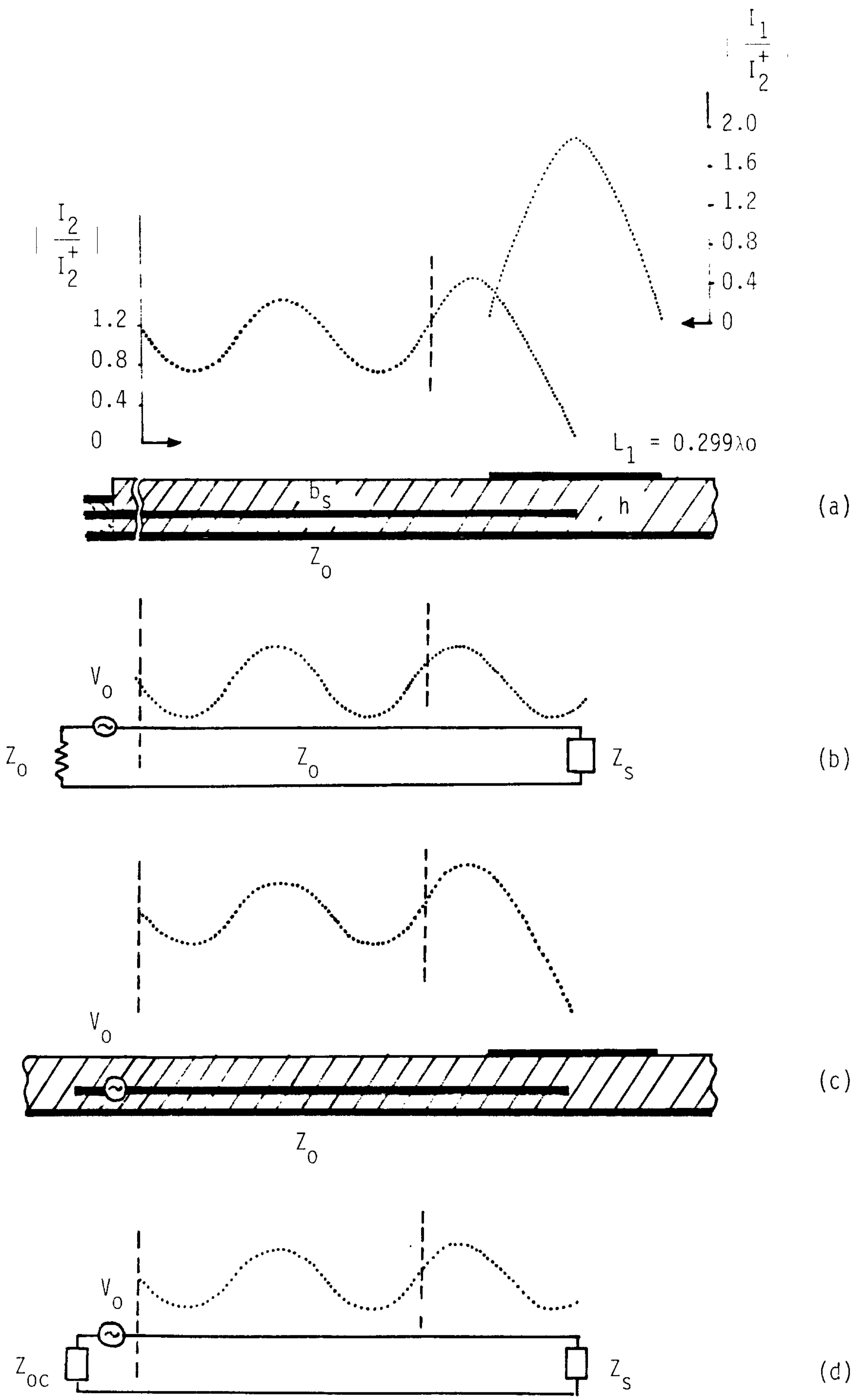


Figure 2: Analytical Model for the Excitation Mechanism.

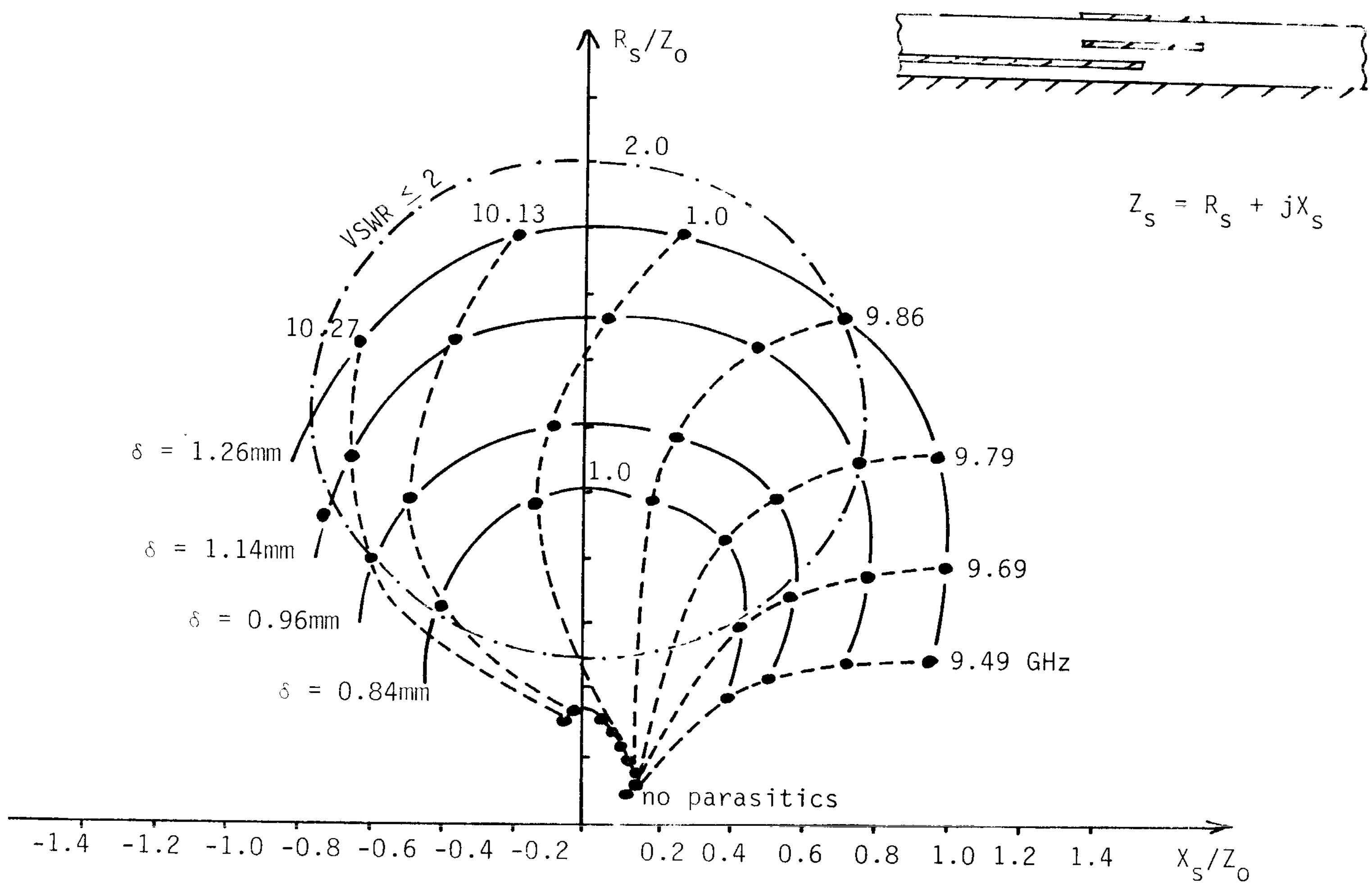


Figure 3: Normalized Self Impedance as a Function of the Embedding Distance for Various Frequencies.

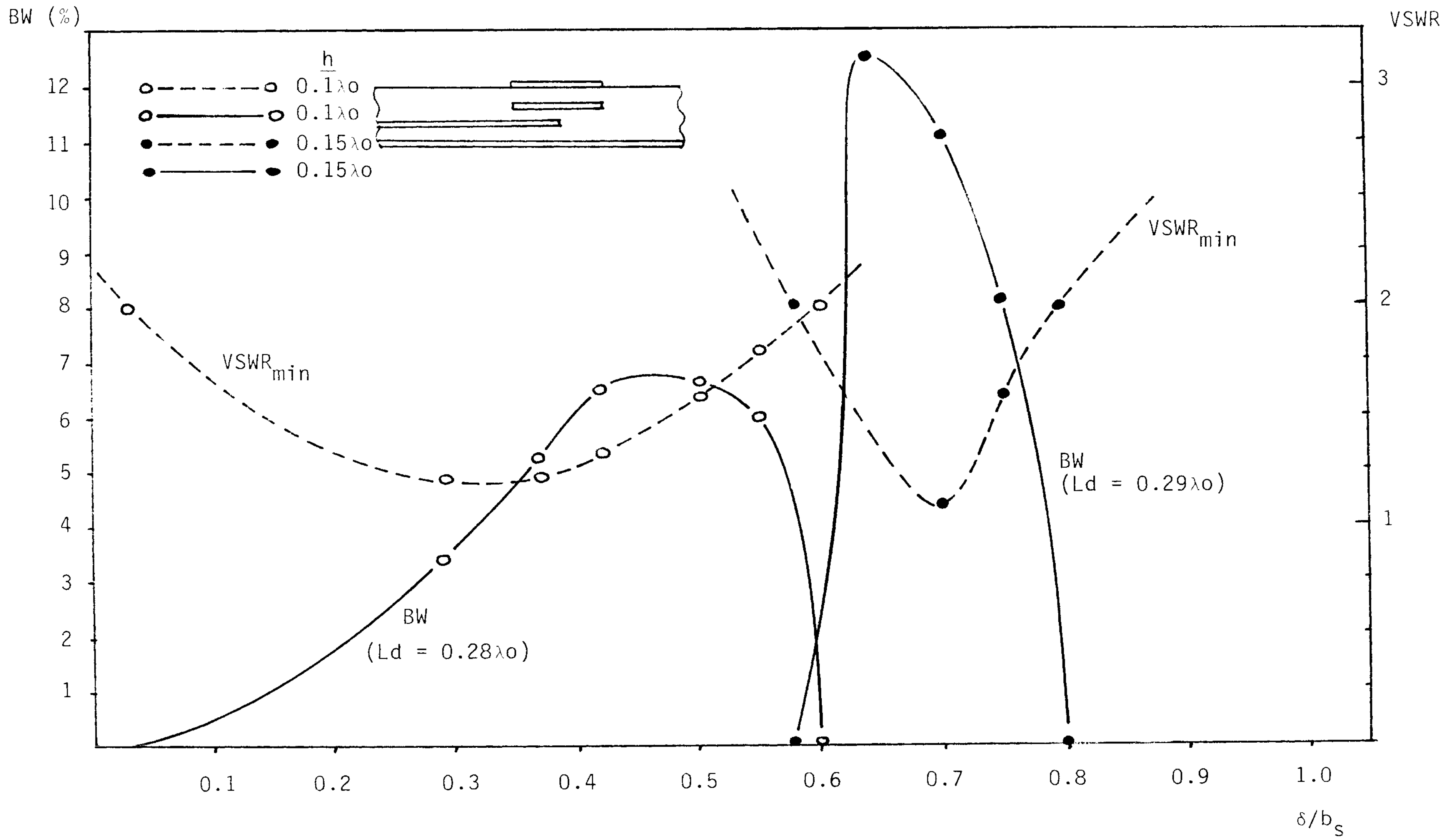


Figure 4: Bandwidth and Minimum Achieved V S W R as Functions of the Ratio δ/b_s for Two Values of the Substrate Thickness h.

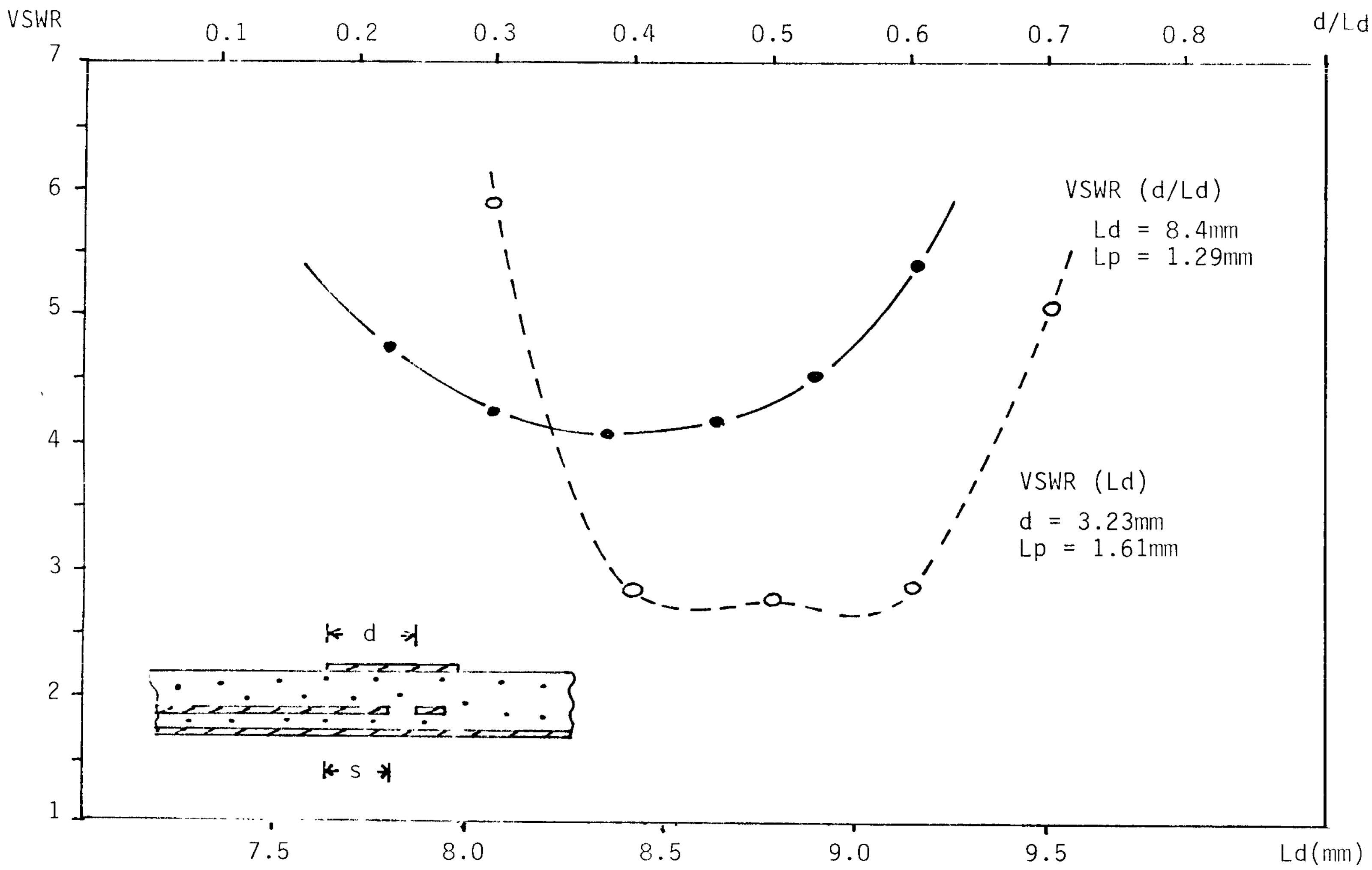


Figure 5: VSWR as a Function of L_d and d/L_d for the Case of One Parasitic of the Same Level with the Transmission Line.

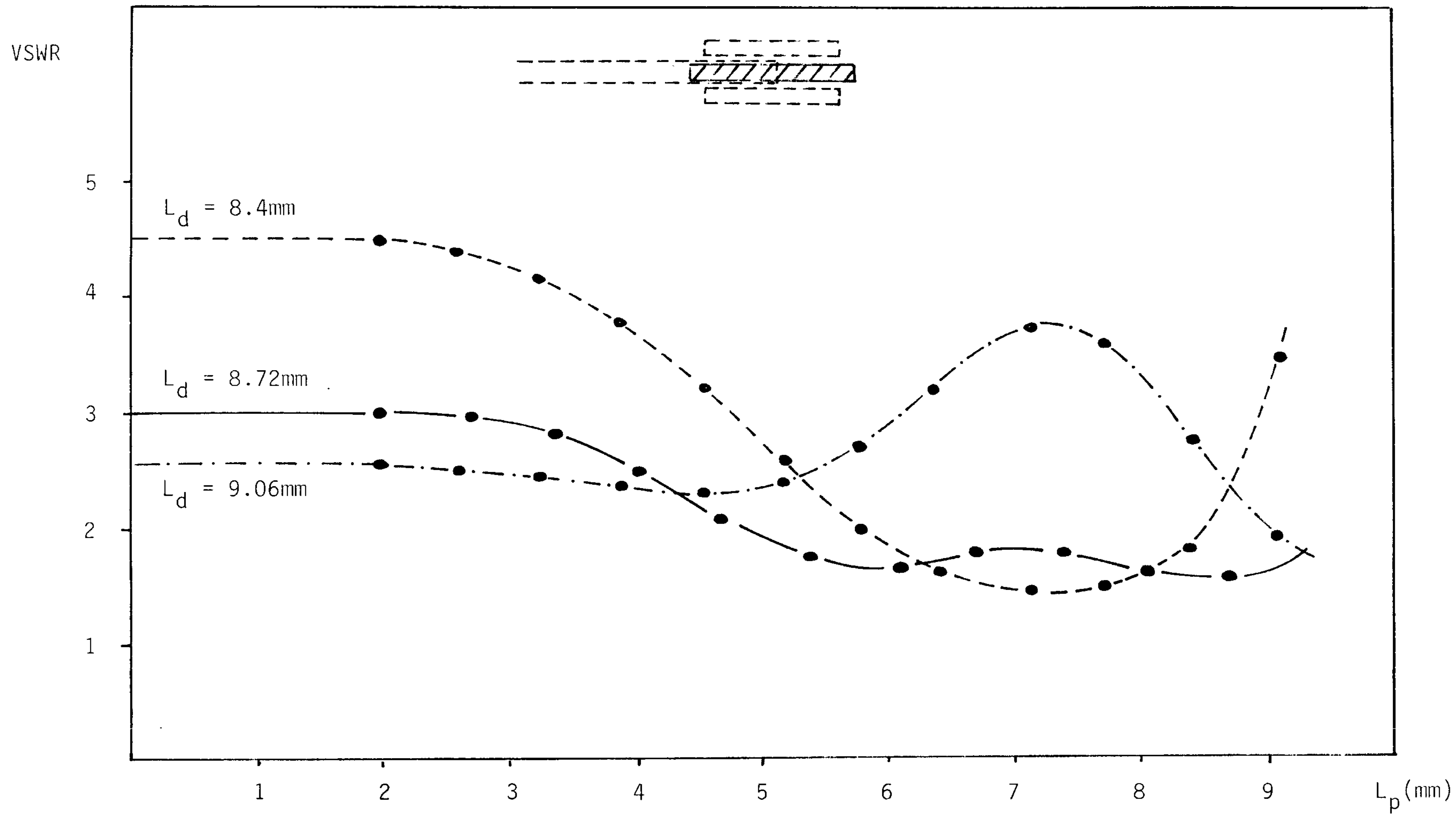


Figure 6: VSWR as a Function of L_d and L_p for the Case of Two Parasitics of the Same Level with the Transmission Line.

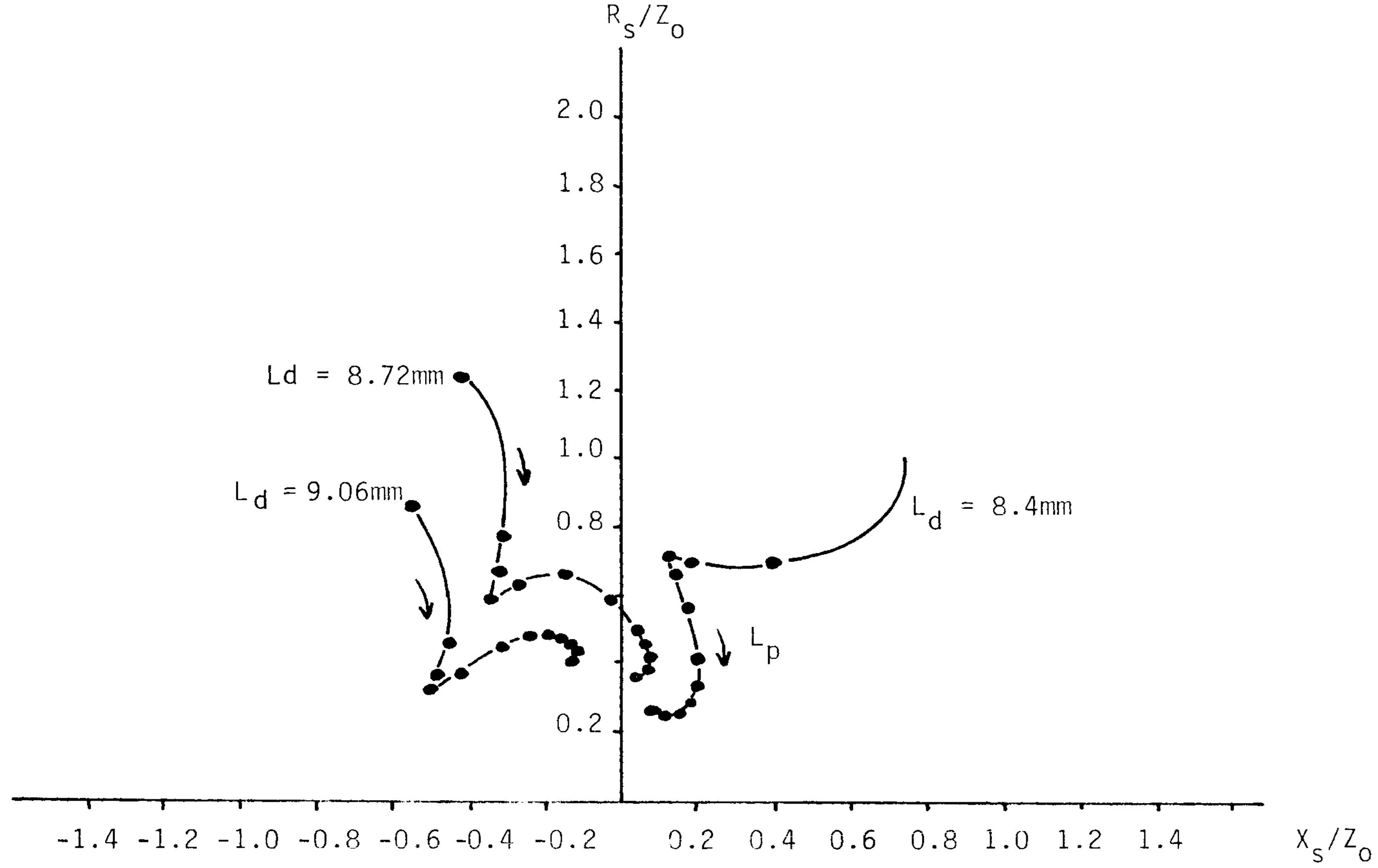


Figure 7: Self Impedance as a Function of L_d and L_p for the Case of Two Parasitics of the Same Level with the Transmission Line.

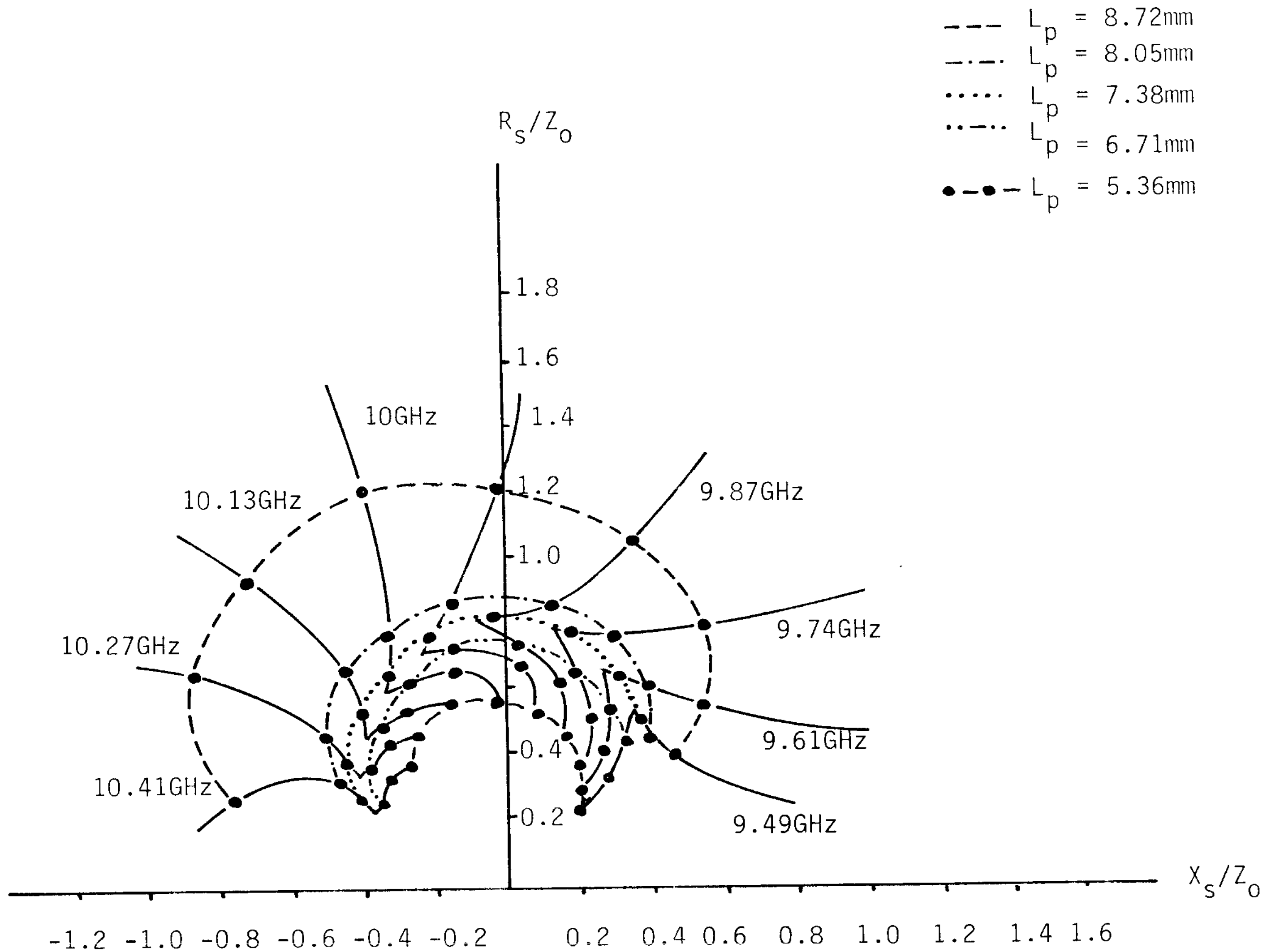


Figure 8: Self Impedance as a Function of the Parasitic Dipole Length L_p for Various Frequencies.

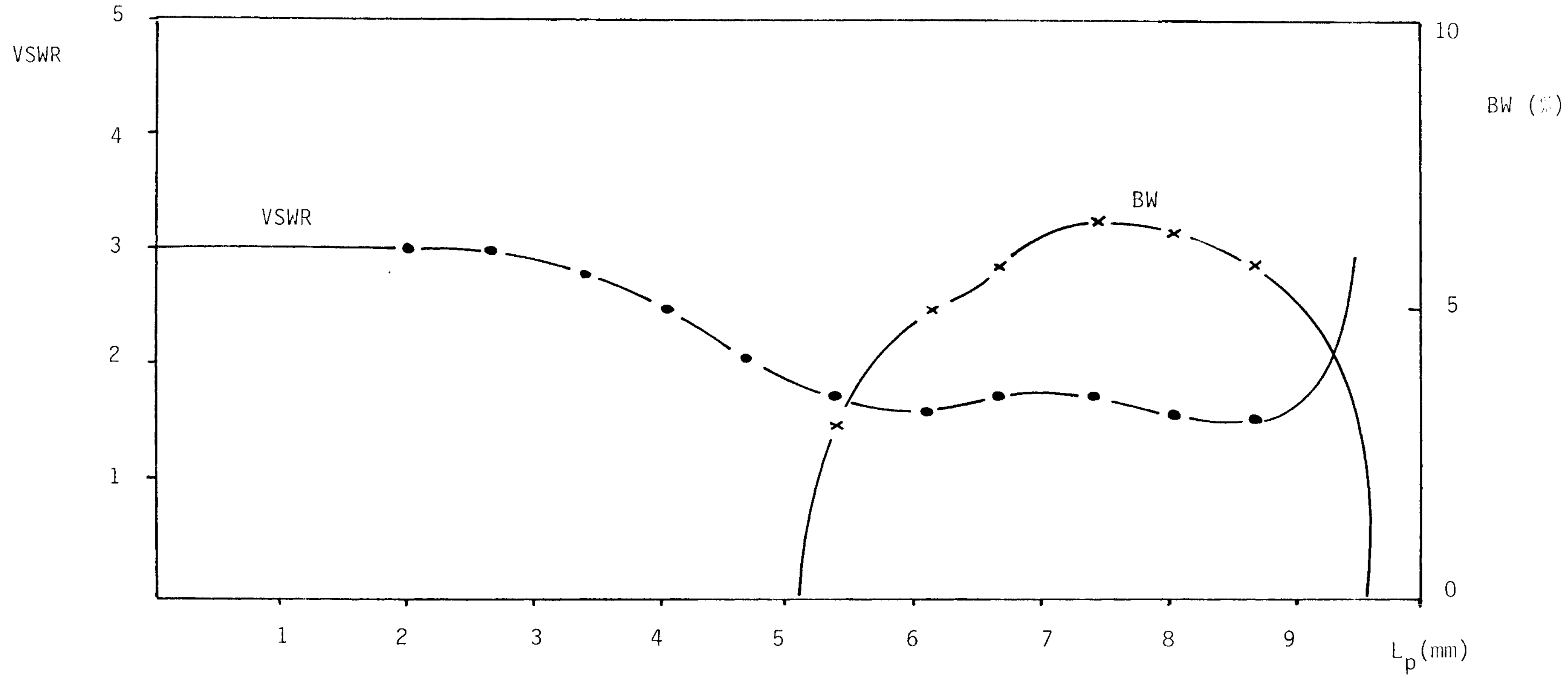


Figure 9: VSWR and Bandwidth as Functions of the Parasitic Dipole Length L_p for the Case of Two Parasitics of the Same Level with the Transmission Line.

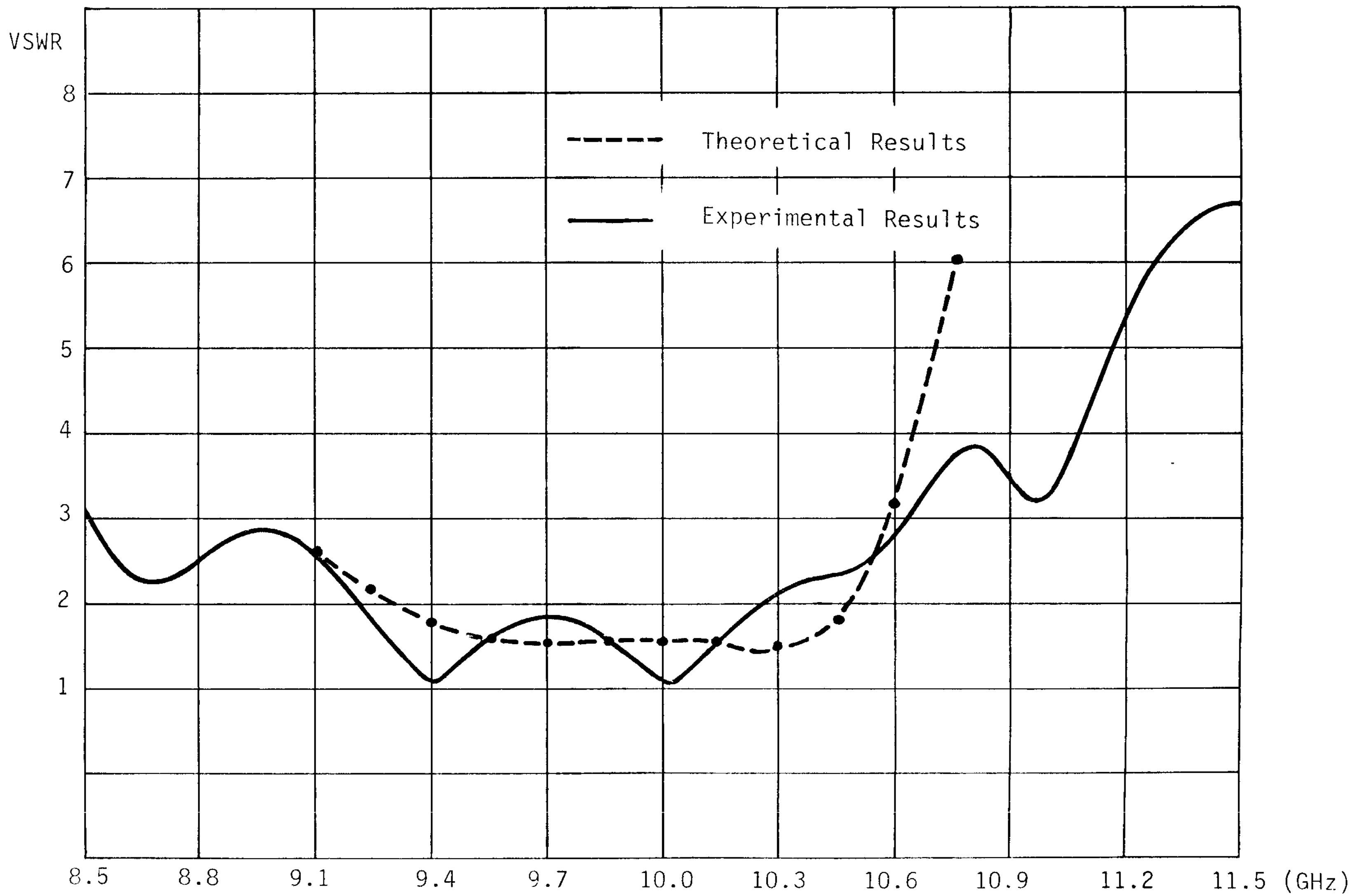


Figure 10: VSWR as a Function of Frequency for the Case of Two Parasitics of the Same Level with the Transmission Line.

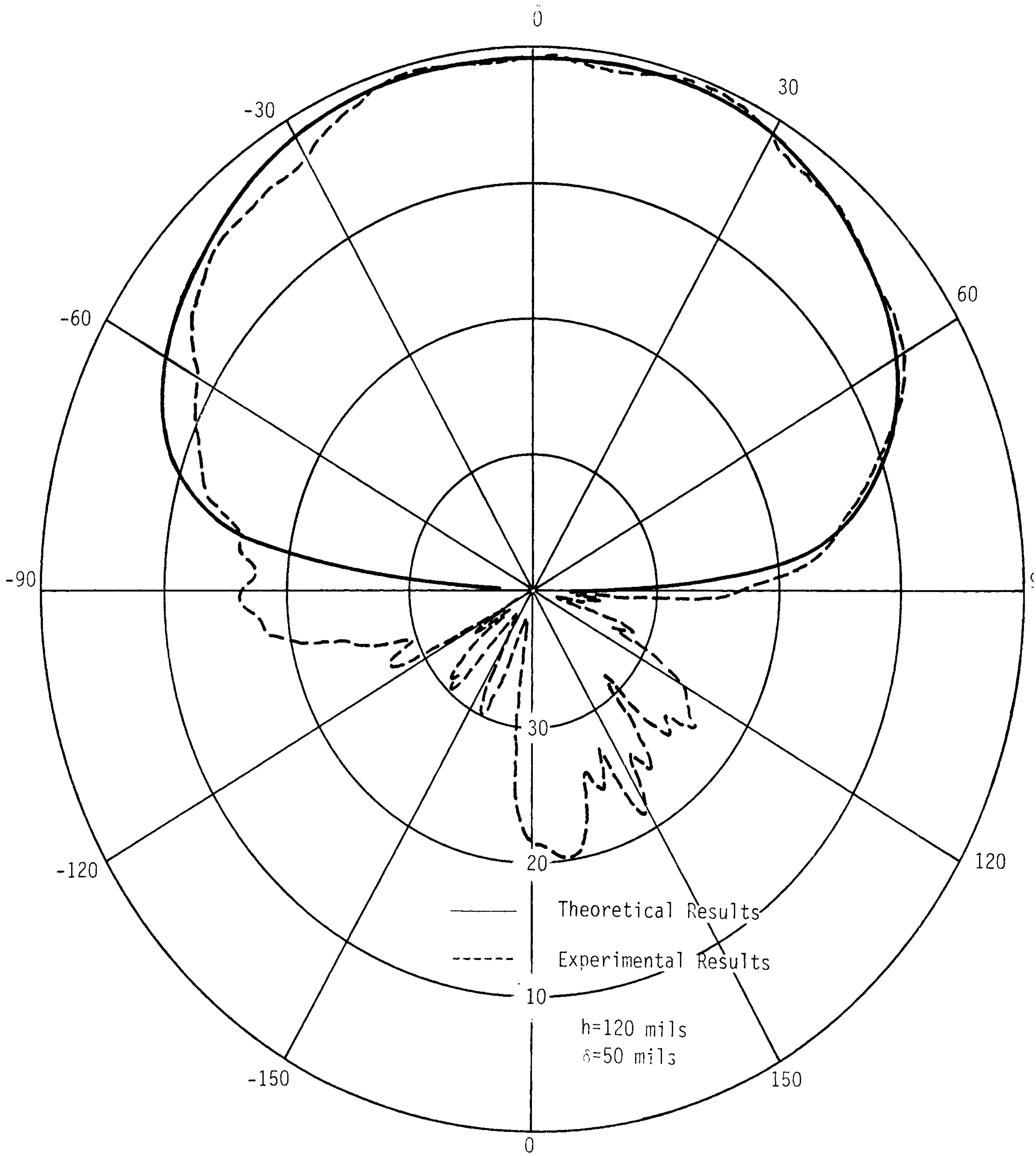


Figure 11: H-Plane Pattern for the Case of One Parasitic Between the Dipole and the Transmission Line.

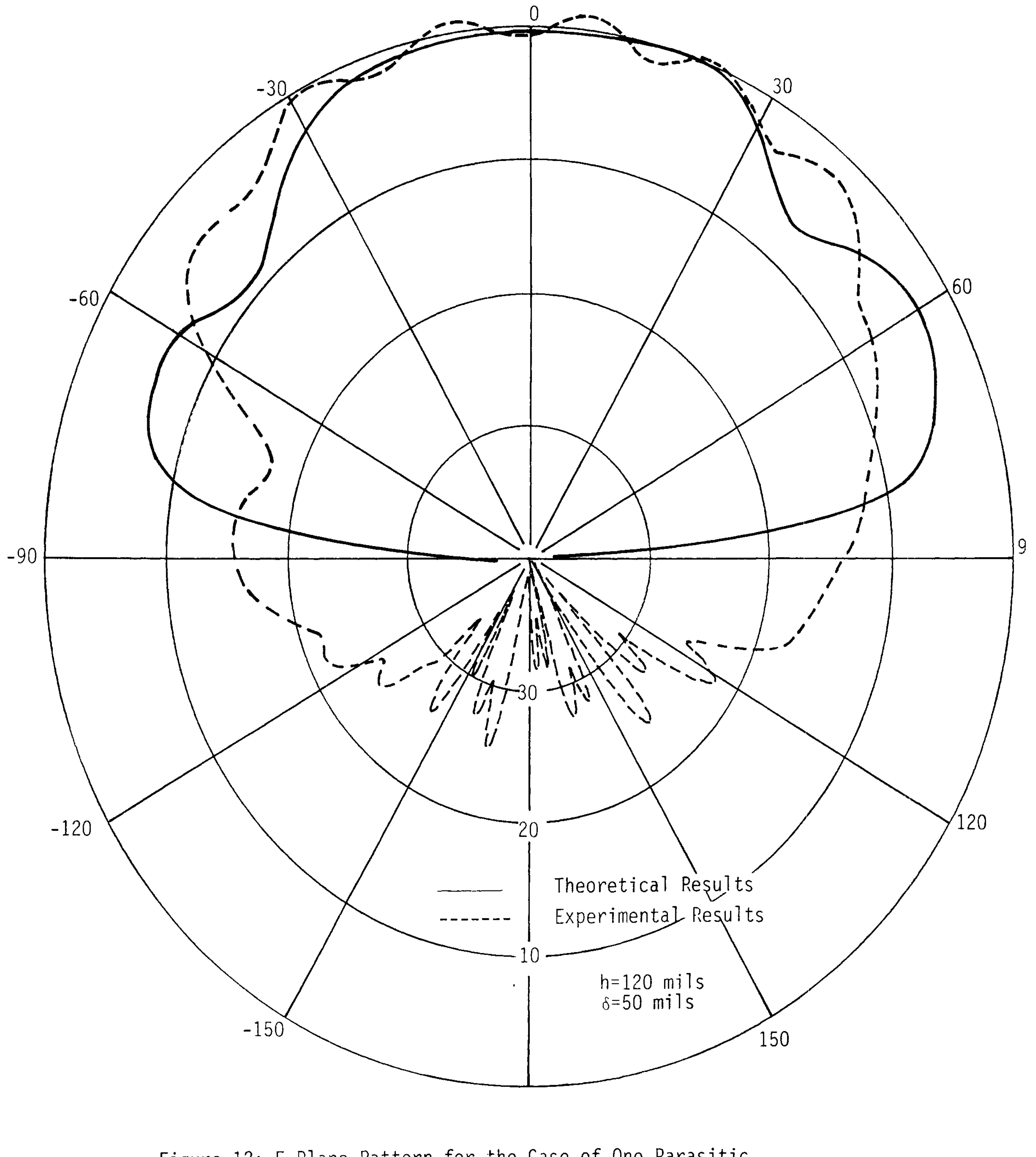


Figure 12: E-Plane Pattern for the Case of One Parasitic Between the Dipole and the Transmission Line.

COMPUTER PROGRAMS

Part a: Comments.

Part b: User's Guide.

Part c: List of Programs.

Part d: Magnetic Tape Comments.

PART A

" Comments About the Programs "

PART A; Comments about the programs.

In order to evaluate the bandwidth for the radiating structures described in the technical report, one has to submit file JOBRUN as a batch job. This file is a collection of smaller batch jobs. Each of these jobs starts with a comment statement:

```
$com   JOBRUN1: Results for the bandwidth...f=   ...
```

and evaluates the currents on the transmission line and dipoles at the frequency shown in the comment statement. Also, it evaluates the E- and H-plane radiation patterns at the same frequency. A list of such a batch job is shown in Table A.1. Preceding these jobs are a few commands which set the time limit for the whole job and create a temporary work file called DOUT. In line 1 of this program (see Table A.1), the command:

```
$sig * route=unyn t=150
```

initiates a batch job of a maximum CPU time equal to 150secs and determines the destination of any punched or printed output (route=unyn). In line 3, the program creates a file called DOUT:

```
$create dout
```

and in line 4 it empties that file in case it existed already:

```
$empty dout
```

Each of the batch subjobs (JOBRUN1, JOBRUN2,...) is a collection of various JCL commands which perform different tasks. In line 5, the job excecutes the compiled version of program FINITE (see Part c) stored in file OFIN:

```
$run ofin 5=*source* 6=dout(1) t=60
```

The program FINITE reads data from file "5" (=*source*=JOBRUN1), evaluates the elements of the impedance matrix which results from the application of the method of moments and stores these results in file DOUT starting at line 1. The maximum CPU time for this program is 60 secs. The data given in lines 6 to 30 are in a format described in Part b. The command

```
$endfile
```

in line 31 denotes the end of the data.

In the same manner, this job excecutes program FINITE (stored in OFIN) two more times (lines 32 to 58 and 59 to 85) and stores all the results in file DOUT as shown in Part c. As it was mentioned previously,the results from these three different jobs are the elements of the impedance matrix stored in vectors Z11, Z31, Z33, Z21, Z22 and Z32. The elements of these vectors correspond to the following elements of the impedance matrix.

Z11 : Self-interaction elements of the printed dipole.

Z22 : Self-interaction elements of the parasitic dipoles.

Z33 : Self-interaction elements of the transmission line.

Z21 : Mutual-interaction elements between printed dipole and parasitics.

Z31 : Mutual-interaction elemets between transmission line and printed dipole.

Z32 : Mutual-interaction elements between transmission line and parasitics.

In the case of two parasitics of the same level with the transmission line, there is one more vector given in the output as Z32 and it includes the mutual interaction elements between the two parasitics. In addition, the batch job excecutes program OINV33 (line 87)

```
$run oinv33 1=*source* 2=dout(1) 3=resgn(1) 6=result(1) t=30
```

Program OINV33 is a compiled version of programm INV33 (see Part b). It takes as input the elements of vectors Z11,Z22,Z33,Z21,Z31 and Z32, fills out the impedance matrix by using all the possible symmetries and inverts the matrix to get the current distribution on the strip conductors. The program INV33 reads input from files 1 (=*source*=JOBRUN1) and 2 (=DOUT) and stores output in files 3 (=RESGN) and 6 (=RESULT). The results in file RESULT are in such a format that the user can read them easily recognizing the currents on the transmission line, the parasitic and printed dipoles (see Part c). The results in file RESGN are in such a form that can be fed as an input to program GAIN (see Part c). The input to program INV33 read through file 1 is shown in Table A.1 from line 88 to line 94. As soon as the excecution of program INV33 is complited, the content of file DOUT is copied to file OUT1, starting at line *l+1, for later possible use.

In line 96, the batch job excecutes program OGAIN which is a compiled version of program GAIN:

```
$run ogain 1=*source* 3=resgn 6=result(*l+1) t=20
```

This program reads data from files 1 (=*source*=JOBRUN1), 3 (=RESGN) and stores results in file 6(=RESULT) starting at the *l+1 line. The results of this file are the radiated E-, H-plane patterns of the structure under consideration. This program can give plots of these patterns when compiled by a VERSATEC compiler. The data given to this program have to be in a form which is explained in details in part b.

Each of these batch subjobs (JOBRUN1, JOBRUN2,...) evaluates the currents and therefore the VSWR at some specified frequency. In order to compute the bandwidth of the radiating structure, the VSWR has to be evaluated over a range of frequencies and therefore various subjobs have to be excecuted one after the other through the batch job JOBRUN. As an example, for the evaluation of the bandwidth as shown in Figure 11 of the technical report, 14 batch subjobs had to be excecuted at the following frequencies:

9.1,9.25,9.4,9.55,9.7,9.85,10.0,10.15,10.30,10.45,10.60,10.75,
10.9 and 11.2 GHz.

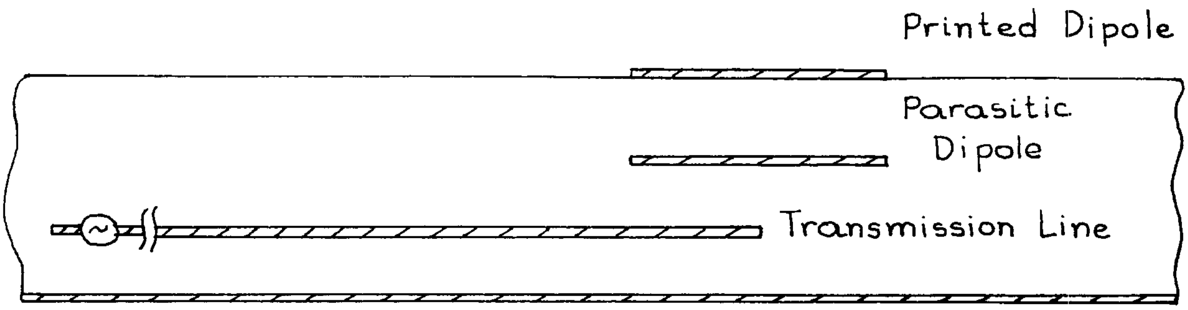
TABLE A.1

" JOBRUN1 for the Case of One Parasitic
Between the Transmission Line and the
Printed Dipole".

```
123456789 123456789 123456789 123456789 123456789 123456789 12345689


$sig * route=unyn t=150
$com ...JOBRUN1...Results for bandwidth...f=8.95GHz...
$create dout
$empty dout
$run ofin 5=*source* 6=dout(1) t=60
      8                  IIA
     15                  IDA
      1                 IOPT
      0                IFEED
    105                  NTD
     25                   ND
    116                   NF
      0                  NTE
      1                  NTM
      2               IFIRST
       2.17               ER
       2.17              EER
       0.093080            H
       0.0697205          BS
       0.0310565         DEL
       0.0000895          T1
       0.0000895          T2
       0.0152125         DLX
    100.0                  A
       3.14159265
       0.0647380           W
       0.0            OFFSET
       0.0000895      WDELTA
       6.61083218     POLTM   ---- POLTE
       6.61083218     POLES
$endfile
$run ofin 5=*source* 6=dout(*l+1) t=60
      8                  IIA
     15                  IDA
      2                 IOPT
      0                IFEED
    105                  NTP
     25                   NP
    116                   NT
      0                  NTE
      1                  NTM
      2               IFIRST
       2.17               ER
       2.17              EER
       0.093080            H
       0.0310565         DEL
       0.0310565         DEL
       0.0000895          T1
       0.0000895          T2
       0.0152125         DLX
    100.0                  A
```

```
123456789 123456789 123456789 123456789 123456789 123456789 123456789

         3.14159265
         0.064738          W
         0.0            OFPD
         0.0000895     WDELTA
         6.61083218    POLTM  ---- POLTE
         6.61083218    POLES
$endfile
$run ofin 5=*source* 6=dout(*l+1) t=60
        8                IIA
       15                IDA
        3               IOPT
        0              IFEED
      105                NTP
       25                 NP
      116                 NT
        0                NTE
        1                NTM
        2             IFIRST
         2.17             ER
         2.17            EER
         0.093080          H
         0.0697205        BS
         0.0310565       DEL
         0.0000895        T1
         0.0386640    BS-DEL
         0.0152125       DLX
       100.0               A
         3.14159265
         0.0647380         W
         0.0            OFTP
         0.0000895     WDELTA
         6.61083218    POLTM  ----- POLTE
         6.61083218    POLES
$endfile
$com ...Parasitics with length=26*DLX...
$run oinv33 1=*source* 2=dout(1) 3=resgn(1) 6=result(1) t=30
           116            NT
            25            NP
            25            ND
           105           NTP
             1           NPD
           166           NOR
            17         NFEED
$endfile
$run ogain 1=*source* 3=resgn 6=result(*l+1) t=20
        0            IPLANE
      105               NTP
      116                NT
       25               NP1
```

```
  123456789 123456789 123456789 123456789 123456789 123456789 123456789

        0                 NP2
       25                  ND
         2.17              ER
         2.17             EER
         0.093080           H
         0.0697205         BS
         0.0310565        DEL
         0.0152125        DLX
         0.0647380      WIDTH
         0.0000895     WDELTA
         0.0             OFP1
         0.0             OFP2
         0.0              OFD
$endfile
$copy dout to out1(*l+1)
$empty dout
$signoff
```



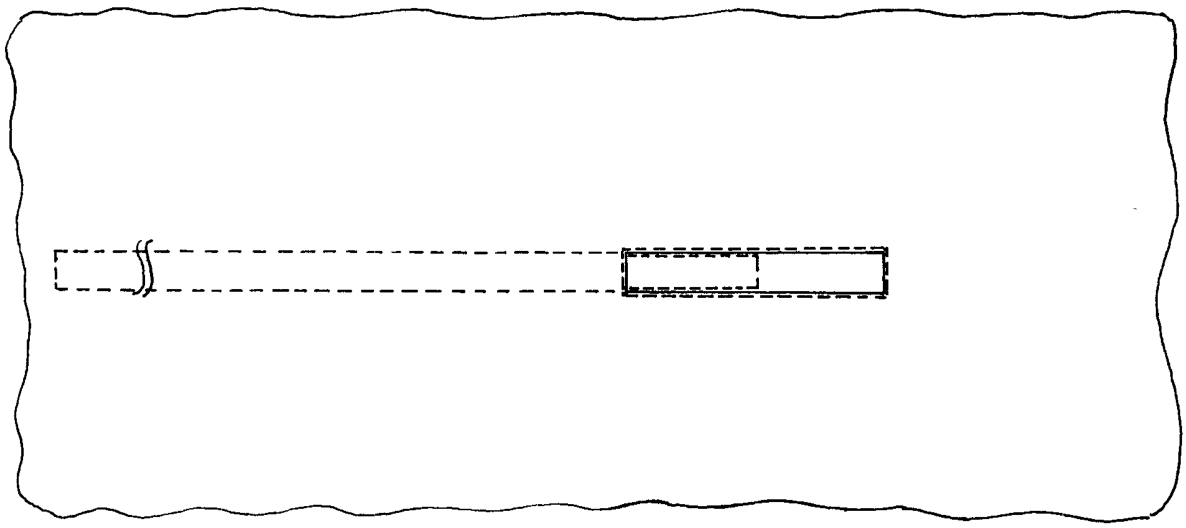


Figure A.1 : Microstrip Configuration for Bandwidth Enhancement. One Parasitic Between the Printed Dipole and the Transmission Line.

TABLE A.2

" JOBRUN1 for the Case of Two Parasitics of the Same Level with the Transmission Line".

```
123456789 123456789 123456789 123456789 123456789 123456789 123456789

$sig *route=unyn t=150
$com ....JOBRUN1....Results for bandwidth...f=8.95GHz...
$create dout
$empty dout
$run ofin 5=*source* 6=dout(1) t=60
        8                 IIA
       15                 IDA
        1                 IOPT
        0                 IFEED
      105                 NTD
       25                 ND
      116                 NT
        0                 NTE
        1                 NTM
        2                 IFIRST
          2.17            ER
          2.17            EER
          0.093080        H
          0.0697205       BS
          0.0             DEL
          0.0000895       T1
          0.0000895       T2
          0.0152125       DLX
        100.0             A
          3.14159265
          0.0647380       W
          0.0             OFTD
          0.0000895       WDELTA
          6.61083218      POLTM     -------POLTE
          6.61083218      POLES
$endfile
$run ofin 5=*source* 6=dout(*l+1) t=60
        8                 IIA
       15                 IDA
        3                 IOPT
        0                 IFEED
      105                 NTP
       25                 NP
      130                 NT
        0                 NTE
        1                 NTM
        2                 IFIRST
          2.17            ER
          2.17            EER
          0.093080        H
          0.0697205       BS
          0.0             DEL
          0.0000895       T1
          0.0000895       T2
```

```
123456789 123456789 123456789 123456789 123456789 123456789 123456789

           0.0152125        DLX
        100.0                 A
           3.14159265
           0.064738           W
           0.07            OFTP
           0.0000895     WDELTA
           6.61083218     POLTM
           6.61083218     POLES
$endfile
$run ofin 5=*source* 6=dout(*l+1) t=60
         8                  IIA
        15                  IDA
         4                 IOPT
         0                IFEED
         5                  NPD
        25                   ND
        25                   NP
         0                  NTE
         1                  NTM
         2               IFIRST
           2.17              ER
           2.17             EER
           0.093080           H
           0.0697205         BS
           0.0              DEL
           0.0000895         T1
           0.0000895         T2
           0.0152125        DLX
        100.0                 A
           3.14159265
           0.0647380          W
           0.07            OFPD
           0.0000895     WDELTA
           6.61083218     POLTM   ----- POLTE
           6.61083218     POLES
$endfile
$run ofin 5=*source* 6=dout(*l+1) t=60
         8                  IIA
        15                  IDA
         3                 IOPT
         0                IFEED
         1                  NPP
        25                  NP1
        25                  NP2
         0                  NTE
         1                  NTM
         2               IFIRST
           2.17              ER
           2.17             EER
           0.093080           H
           0.0697205         BS
           0.0              DEL
           0.0000895         T1
```

```
123456789 123456789 123456789 123456789 123456789 123456789 123456789
```

```
          0.0000895          T2
          0.0152125          DLX
        100.0                 A
          3.14159265
          0.0647380           W
          0.14             OFPP
          0.0000895      WDELTA
          6.61083218      POLTM  ------ POLTE
          6.61083218      POLES
$endfile
$com ...Parasitics with length 26*dlx ...
$run omutual 1=*source* 2=dout(1) 3=resgn(1) 6=result(1) t=30
            116              NT
             25             NP1
             25             NP2
             25              ND
            105             NTP
              1             NPP
              1             NPD
            191             NOR
             12          NFEED1
            180          NFEED2
$endfile
$run ogain 1=*source* 3=resgn 6=result(*l+1) t=20
        0                IPLANE
      105                   NTP
      116                    NT
       25                   NP1
       25                   NP2
       25                    ND
          2.17               ER
          2.17              EER
          0.093080            H
          0.0697205          BS
          0.0310565         DEL
          0.0152125         DLX
          0.0647380       WIDTH
          0.0000895      WDELTA
          0.07             OFP1
         -0.07             OFP2
          0.0              OFD
$endfile
$copy dout to out1(*l+1)
$empty dout
$signoff
```

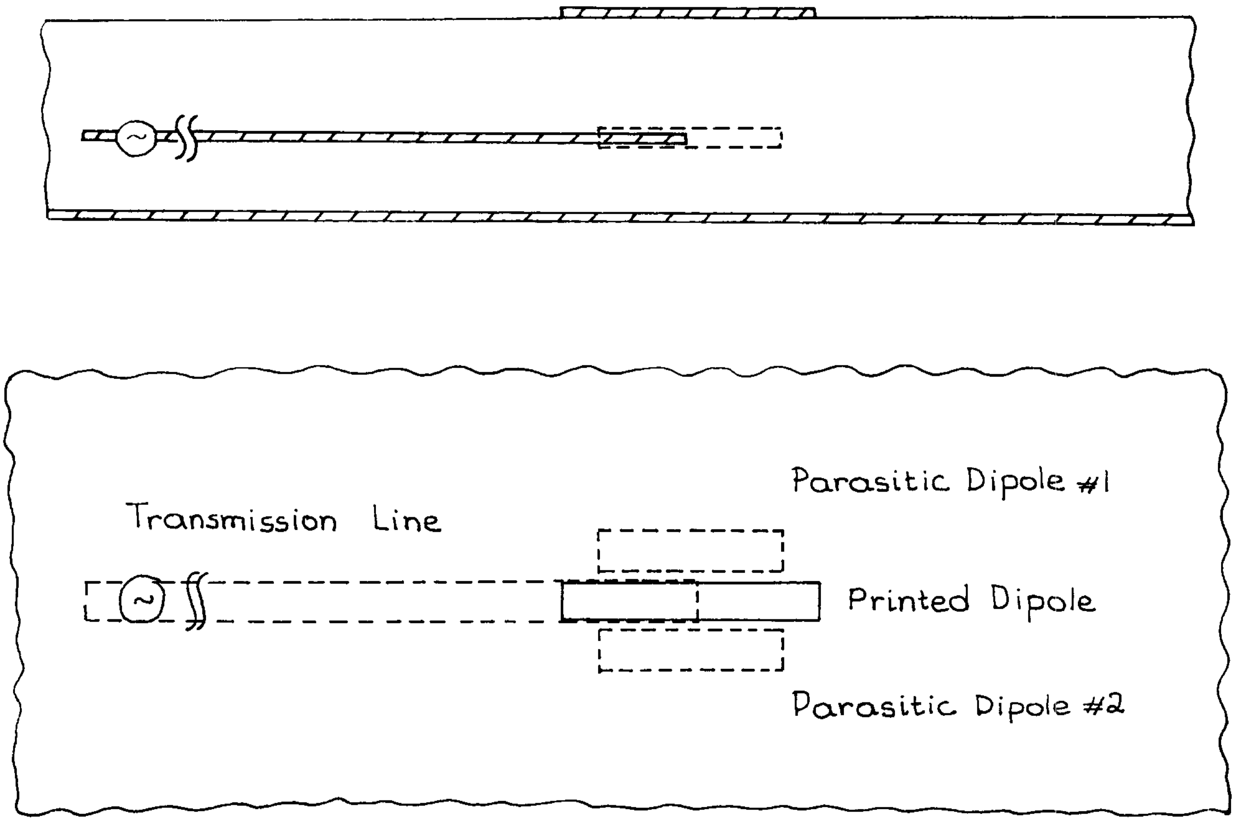


Figure A.2 : Microstrip Configuration for Bandwidth Enhancement. Two Parasitic Dipoles of the Same Level with the Transmission Line.

PART B

" User's Guide"

Part B: User's Guide.

i) Transfer the programs from the magnetic tape to your computer. On the tape there are 8 files. Five of them contain excecutable programs while the rest three are input and output data files. The files as shown on the tape are:

File name	Type of file	Description
JOBRUN	Excecutable program named JOBRUN.	A group of JCL commands which lead to the evaluation of the bandwidth for either one of the structures under consideration.(see technical report).
FOPTION0	Excecutable file named POLES.	This program evaluates the poles of the Green's function and classifies them as TE and TM.
FOPTION1	Excecutable file named FINITE.	This program evaluates the elements of the impedance matrix.
FOPTION2	Excecutable file named INV33.	This program inverts the impedance matrix and evaluates the currents on all the strip conductors. It is good only for the case of one parasitic.
FOPTION3	Excecutable file named GAIN.	This program uses the currents to evaluate the E- and H-plane radiation patterns.
FOPTION4	Data file named OUT1.	This is a permanent file that contains the elements of the impedance matrix as they were evaluated by FINITE.

File name	Type of file	Description
FOPTION5	Data file named RESULT.	This is a permanent file that contains the computed currents in a format easy to be read by the user.
FOPTION6	Data file named RESGN.	This is a permanent file that contains the computed currents. This file is used as an input to program GAIN.
FOPTION7	Excecutable file named MUTUAL.	This is a program that inverts the impedance matrix and evaluates the currents on the strip conductors. It is good only for the case of two parasitics of the same level with the T.L. (see technical report).

ii) Run program POLES by substituting the rigth values for ER and H. This program gives as output the poles of the Green's function classified as TE and TM poles . Also, it gives the same poles ordered according to their magnitude.

iii) Compile program FINITE and save it in a file named OFIN. Compile program INV33 or MUTUAL and save it in a file named OINV33 or OMUTUAL. Also, compile program GAIN and save it in a file named OGAIN.

iv) In each run of the compiled program OFIN let the variables IIA, IDA, IOPT, IFEED, A and have exactly the same values as in the example (see Tables A.1,A.2).

v) Change the rest of the variables so the correspond to the structure under consideration. The definition of each one of these variables is as follows:

NTD : (NTD-1)*DLX= Distance of the leftend of the printed dipole from the left end of the Transmission line (see figure A.1).

NPD : (NPD-1)*DLX= Distance of the left end of the printed dipole from the left end of the parasitic dipole.

NTP : (NTP-1)*DLX= Distance of the left end of the parasitic dipole from the left end of the transmission line.

NPP : (NPP-1)*DLX= Distance of the left end of parasitic #1 from the left end of parasitic #2.

NT : (NT+1)*DLX= Length of the transmission line.

ND : (ND+1)*DLX= Length of the printed dipole.

NP : (NP+1)*DLX= Length of the parasitic for the structure of figure A.1

NP1 : (NP1+1)*DLX= Length of parasitic #1 for the structure of figure A.2

NP2 : (NP2+1)*DLX= Length of parasitic #2 for the structure of figure A.2

NTE : Number of TE poles (This number comes as an output from program POLES).

NTM : Number of TM poles (This number comes as an output from program POLES).

IFIRST : This variable takes the values 0,1,2
0 : The lowest pole is a TM one.
1 : The lowest pole is a TE one.
2 : There is only one TM pole.

ER : Dielectric constant.

EER : A variable which takes values between 1 and ER. It is suggested that this variable takes the value ER.

H : Substrate thickness in λ_0

BS : Embedding distance of the transmission line in λ_0

DEL : Embedding distance of the parasitic/parasitics in λ_0

T1 : Conductor thickness of the printed and parasitic dipoles in λ_0.

T2 : Conductor thickness of the transmission line. It is suggested that you make T2 equal to T1.

DLX : Subsection length in λ_g ($\lambda_g = \lambda_0 / \sqrt{\varepsilon_r}$).

W : Width of the strip conductors.

OFTD : Offset of the printed dipole with respect to the transmission line.

OFPD : Offset of the printed dipole with respect to the parasitics.

OFTP : Offset of parasitic dipoles with respect to the transmission line.

WDELTA : Correction to the width : $w_e = w + 2\delta$. It is suggested that the value of is equal to T1 and T2.

POLTE *: This is a vector having as elements all the TE poles ordered according to their magnitude.

POLTM *: This is a vector having as elements all the TM poles ordered according to their magnitude.

POLES *: This is a vector having as elements all the poles ordered accordig to their magnitude.

NOR : Total number of unknown coefficients in the expression for the current in the integral equation (see technical report).

NFEED : NFEED*DLX= L_g= the distance of the gap generator from the left end of the microstrip transmission line. NFEED should have the following value INT(0.25/DLX).

IPLANE : This variable determines wether the E- or H-plane patterns will be evaluated by program GAIN:
0 : E-plane pattern
1 : H-plane pattern.

OFP1,2 : Offset of the 1st,2nd parasitic dipoles with respect to the transmission line.

If there is only one parasitic , OFP2 should be equal to 0.

OFD : This variable is equal to OFTD.

vi) As an example, JOBRUN is shown as it should be submitted for the evaluation of the VSWR and radiation patterns at f=8.95GHz for the two structures discussed in the technical report. Specifically, the case of one parasitic between the printed dipole and the transmission line is treated in table A.1 while the case of two parasitics of the same level with the transmission line is treated in table A.2.

* The elements of these vectors are given as output by the program POLES.

** All the lengths except of the subsection length are measured in λ_0

PART C

" List of Programs"

JOBRUN

```
$sig * route=unyn t=150
$com .....     Results for Bandwidth  ..f=8.95GHz..
$create dout
$empty dout
$run ofin 5=*source* 6=dout(1) t=60
        8
       15
        1
        0
      105
       25
      116
        0
        1
        2
          2.17
          2.17
          0.093080
          0.0697205
          0.0310565
          0.0000895
          0.0000895
          0.0152125
        100.0
          3.141592653
          0.0647380
          0.0
          0.0000895
          6.61083218
          6.61083218
$endfile
$run ofin 5=*source* 6=dout(*l+1) t=60
        8
       15
        2
        0
        1
       25
       25
        0
        1
        2
          2.17
          2.17
          0.093080
          0.0310565
          0.0310565
          0.0000895
          0.0000895
          0.0152125
        100.0
          3.141592653
          0.064738
          0.0
          0.0000895
          6.61083218
          6.61083218
$endfile
$run ofin 5=*source* 6=dout(*l+1) t=60
        8
```

```
      15
       3
       0
     105
      25
     116
       0
       1
       2
        2.17
        2.17
        0.093080
        0.0697205
        0.0310565
        0.0000895
        0.0386640
        0.0152125
      100.0
        3.141592653
        0.0647380
        0.0
        0.0000895
        6.61083218
        6.61083218
$endfile
$com .......Parasitics with length 26*dlx  ..............
$run oinv33 1=*source* 2=dout(1) 3=resgn(1) 6=result(1) t=30
          116
           25
           25
          105
            1
          166
           17
$endfile
$run ogain 1=*source* 3=resgn 6=result(*l+1) t=20
       0
     105
     116
      25
       0
      25
        2.17
        2.17
        0.093080
        0.0697205
        0.0310565
        0.0152125
        0.0647380
        0.0000895
        0.0
        0.0
        0.0
$copy dout to out1(*l+1)
$empty dout
$signoff
```

FOPTION 0

(POLES)

```
C***********************************************************************
C     THE NAME OF THIS FILE IS:................POLES..................
C***********************************************************************
C     THE DOUBLE INTEGRATION IS PERFORMED FROM XK-1 TO XK
C     SAME THING WITH THE SINGLE INTEGRATION
C
C     FROM 0 TO A WE PERFORM FIRST THE INTEGRATION WITH RESPECT TO X,X'
C     AND AFTER THAT THE INTEGRATION WITH RESPECT TO L
C
C     FROM A TO  ° WE PERFORM FIRST THE INTEGRATION WITH RESPECT TO L
C     AND AFTER THAT THE INTEGRATION WITH RESPECT TO X,X'
C
C     BUT DURING THIS SECOND INTGRATION WE ARE CAREFULL TO COMPUTE ANALY
C     TICALLY THE INTEGRALS WITH THE FAST VARYING INTEGRAND
C***********************************************************************
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION XR(40),XS(40)
C.......................................................................
C.......................................................................
C     ***********************  COMMENTS  ****************************
C
C     ER    :....DIELECTRIC CONSTANT
C
C     DP    :....HEIGHT OF THE DIELECTRIC SUBSTRATE
C
C
C     NE    :....NUMBER OF ##TE## WAVES
C
C     NM    :....NUMBER OF ##TM## WAVES
C
C     XS    :....MATRIX OF POLES CONTRIBUTING TO TE WAVES
C
C     XR    :....MATRIX OF POLES CONTRIBUTING TO TM WAVES
C
C     ERR   :....ERROR IN THE COMPUTATION OF THE POLES
C
C.......................................................................
C
      COMMON/POLE/TPO(40),LOR(40)
C.......................................................................
      ER=10.0
      ER2=ER*ER
      PI=3.141592653589D0
      PI2=PI*PI
      MAXE=5
      ERR=0.0000001D0
C.......................................................................
C.......................................................................
      DO 12 IDP=1,1
      DP=FLOAT(IDP)*0.10160
      CALL POLES (ER,PI,DP,ERR,XS,XR,NE,NM)
 12   CONTINUE
      STOP
      END
C***********************************************************************
C     THE NAME OF THIS FILE IS:................POLES....................
C***********************************************************************
C     THIS SUBROUTINE FINDS THE POLES OF THE GREEN'S FUNCTION
C                  (TE AND TM SURFACE WAVES)
C***********************************************************************
```

```
      SUBROUTINE POLES(ER,PI,DP,ERR,TEP,TMP,NE,NM)
C
C      ER=   DIELECTRIC CONSTANT
C      DP=   HEIGHT OF THE DIELECTRIC SUBSTRATE NORMALIZED TO THE WAVELE-
C            NGTH OF OPERATION
C      ERR=  ACCURACY OF CALCULATION
C      NE=   NUMBER OF TE POLES
C      NM=   NUMBER OF TM POLES
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION XS(40),XR(40),TEP(40),TMP(40)
      COMMON/POLE/TPO(40),LOR(40)
C
C      PART I : TE MODES
C
      AK0=2.D0*PI
      AK=DSQRT(ER)*AK0
      X0=DP*DSQRT(AK**2-AK0**2)
      AN=X0/PI+0.5D0
      NE=AN
      IF (NE.EQ.0) GO TO 310
      DO 2 I=1,NE
      IF (X0-(2.D0*FLOAT(I)+1.D0)*PI/2.D0) 3,3,4
    4 XS0=(2.D0*FLOAT(I)-1.D0)*PI/2.D0+ERR
      XS1=(2.D0*FLOAT(I)+1.D0)*PI/2.D0-ERR
      GO TO 5
    3 XS0=(2.D0*FLOAT(I)-1.D0)*PI/2.D0+ERR
      XS1=X0
    5 CONTINUE
      IF (DABS(XS0-XS1)-ERR) 22,7,7
    7 XSA=(XS0+XS1)/2.D0
      Y=-DTAN(XSA)*DSQRT(X0**2-XSA**2)-XSA
      IF (Y) 8,9,10
    9 XS(I)=XSA
      GO TO 222
    8 XS1=XSA
      GO TO 5
   10 XS0=XSA
      GO TO 5
   22 XS(I)=(XS0+XS1)/2.D0
  222 XS(I)=DSQRT(AK**2-XS(I)**2/DP**2)
    2 CONTINUE
      WRITE (6,301) ER,DP
  301 FORMAT('1',10X,'1)   THE TE MODES THAT CAN BE EXCITED IN A DIELECT
     *RIC SUBSTRATE WITH'/10X,'ER=',D16.9,5X,'DP=',D16.9/10X,'ARE GIVEN
     *BY:'//)
      DO 302 I=1,NE
      TEP(I)=XS(I)
      WRITE (6,303) I,TEP(I)
  303 FORMAT (10X,'ORDER OF MODE=',I4,5X,'SURFACE TE WAVE AT L=',D16.9)
C
  302 CONTINUE
      IF (NE.NE.0) GO TO 312
  310 WRITE (6,311)
  311 FORMAT('1',10X,'1)   THERE IS NOT ANY TE SURFACE WAVE'/10X,'......
     *...........................')
C
  312 CONTINUE
C
C      END OF PART I
```

```
C
C
C      PART II : TM MODES
C
      AN=X0/PI+1.D0
      NM=AN
      IF (NM.EQ.0) GO TO 320
      DO 13 I=1,NM
      IF (X0-(2.D0*FLOAT(I)+1.D0)*PI/2.D0) 14,14,15
 15   XS1=FLOAT(I)*PI-PI/3.D0-0.01D0
      GO TO 16
 14   XS1=X0
 16   XS0=FLOAT(I-1)*PI+ERR
 17   CONTINUE
      IF (DABS(XS0-XS1)-ERR) 113,19,19
 19   XRA=(XS0+XS1)/2.D0
      Y=DSQRT(ER)**2*(1.D0/DTAN(XRA))*DSQRT(X0**2-XRA**2)-XRA
      IF (Y) 20,21,24
 21   XR(I)=XRA
      GO TO 333
 20   XS1=XRA
      GO TO 17
 24   XS0=XRA
      GO TO 17
113   XR(I)=(XS0+XS1)/2.D0
333   XR(I)=DSQRT(AK**2-XR(I)**2/DP**2)
 13   CONTINUE
      WRITE (6,304) ER,DP
304   FORMAT  (//10X,'2)   THE TM MODES THAT CAN BE EXCITED IN A DIELECT
     *RIC SUBSTRATE WITH'/10X,'ER=',D16.9,5X,'DP=',D16.9/20X,'ARE GIVEN
     *BY:'//)
      DO 305 I=1,NM
      TMP(I)=XR(I)
      WRITE (6,306) I,XR(I)
306   FORMAT (10X,'ORDER OF MODE=',I4,5X,'SURFACE TM WAVE AT L=',D16.9)
C
305   CONTINUE
      IF (NM.NE.0) GO TO 322
320   WRITE (6,321)
321   FORMAT (10X,'2)   THERE IS NOT ANY TM SURFACE WAVE'/10X,'.........
     *......................................')
C
322   CONTINUE
C
      NK=NE+NM
      DO 411 IQW=1,NE
      TPO(IQW)=TEP(IQW)
      LOR(IQW)=0
 411  CONTINUE
      DO 412 IQW=1,NM
      TPO(NE+IQW)=TMP(IQW)
      LOR(NE+IQW)=1
 412  CONTINUE
      DO 515 IIL=1,NK
      WRITE (6,516) IIL,TPO(IIL),LOR(IIL)
516   FORMAT (10X,'IIL=',I4,5X,'TPO=',D16.9,5X,'LOR=',I4/)
515   CONTINUE
C
      IF (NK.EQ.1) GO TO 416
      NNK=NK-1
```

```
      DO 415 IIP=1,NNK
      IK=IIP+1
      DO 413 IIF=IK,NK
      QWR=TPO(IIP)
      IIW=LOR(IIP)
      IF (TPO(IIP).LT.TPO(IIF)) GO TO 413
      TPO(IIP)=TPO(IIF)
      LOR(IIP)=LOR(IIF)
      TPO(IIF)=QWR
      LOR(IIF)=IIW
  413 CONTINUE
  415 CONTINUE
C
  416 CONTINUE
      WRITE (6,418) (TPO(IR),LOR(IR),IR=1,NK)
  418 FORMAT (/12X,'TPO-MATRIX',10X,'LOR-MATRIX'/12X,'----------',10X,'-
     *---------'//20(10X,D16.9,5X,I4/))
      RETURN
      END
```

FOPTION1

(FINITE)

```
C*****************************************************************
C     ...................FINITE ..................................
C
C     THIS PROGRAM EVALUATES THE ELEMENTS OF THE MATRIX AND VECTOR
C     IN THE MATRIX EQUATION FOR THE FOLLOWING PROBLEM:
C        "EXCITATION OF A DIPOLE ON THE AIR-DIELECTRIC
C         INTERFACE WITH A FEEDING LINE EMBEDDED IN
C         THE SUBSTRATE-GAP GENERATOR"
C
C         THIS PROGRAM IS GOOD FOR ANY EMBEDDING DISTANCE OF THE
C         TRANSMISSION LINE
C
C         MNUM=     1   WHEN OFFSET = 0.0
C                   2   WHEN OFFSET > 0.0
C
C
C*****************************************************************
      IMPLICIT REAL*8 (A-H,O-Z)
      COMPLEX*16 Z1GT(150),Z2GT(160),Z2LT(190),CI
      DIMENSION VSPE(20),VTPE(20),VSPM(20)
      COMMON/OUT/G1GT(150),G2GT(160),G2LT(190)
      COMMON/MAT/PLI,IWRITE
      COMMON/PUT/SSJ0(150),SAJ0(150),YSIN,YCOS
      COMMON/ADON/DIST(10,150,10),SERS(5),SERA(5),DARG(10,10,4),SN(10,2)
     *,WREAL,NSER,MNUM
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/DATT/COAL(20),POINT(20),CN(51),BM(51),POLTM(20),POLTE(20)
     *,AM(41),DM(41),POLES(40),NPOINT,NK0,MA,NTM,NTE,NK0K,IFIRST
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
      CALL DATA(IOPT)
      WREAL=W
      W=W*(1.D0+2.D0*WDELTA/W)
      CI=(0.D0,1.D0)
C
C     FOR THE NORMALIZATION OF THE CURRENT ALONG THE Y AXIS
                CVON=W*PI/2.D0
C
C---------------------------------------------+
C     STEP 1 : EVALUATION OF VECTOR CN        |
C              IT GIVES THE END POINTS OF THE |
C              INTERVALS CONSIDERED IN (0,K0) |
C---------------------------------------------+
      DELTA=AK0/FLOAT(NK0)
      CN(1)=0.D0
      DO 1 I=1,NK0
          CN(I+1)=DELTA*FLOAT(I)
  1   CONTINUE
C---------------------------------------------+
C     STEP 1 : EVALUATION OF VECTOR BM        |
C              IT GIVES THE END POINTS OF THE |
C              INTERVALS CONSIDERED IN (K,A)  |
C---------------------------------------------+
      DELTA=(A/DSQRT(EER)-AK)/FLOAT(MA)
```

```
      BM(1)=AK
      DO 2 I=1,MA
           BM(I+1)=DELTA*FLOAT(I)+AK
  2   CONTINUE
C-------------------------------------------------+
C     STEP 1 : EVALUATION OF THE VECTORS AM,DM    |
C              "AM" GIVES THE END POINTS AROUND   |
C                   THE TM POLES                  |
C              "DM" GIVES THE END POINTS AROUND   |
C                   THE TE POLES                  |
C                                                 |
C     IFIRST=   2   ONLY ONE TM POLE              |
C               1   TE0<TM0                       |
C               0   TM0<TE0                       |
C-------------------------------------------------+
      AM(1)=AK0
      DM(1)=AK0
      NMAX=NTE+NTM-1
      IF (IFIRST.EQ.2) GO TO 3
      DO 4 I=1,NMAX
           AM(I+1)=(POLES(I+1)+POLES(I))/2.D0
           DM(I+1)=AM(I+1)
   4  CONTINUE
      AM(NMAX+2)=AK
      DM(NMAX+2)=AK
      IF (IFIRST.EQ.1) GO TO 5
          DM(NMAX+1)=AM(NMAX+2)
          DO 6 I=1,NMAX
               DM(NMAX-I+1)=AM(NMAX-I+2)
  6       CONTINUE
          GO TO 7
  5       AM(NMAX+1)=DM(NMAX+2)
          DO 8 I=1,NMAX
               AM(NMAX-I+1)=DM(NMAX-I+2)
  8       CONTINUE
          GO TO 7
C
  3       DELTA=(AK-AK0)/FLOAT(NK0K)
          AM(1)=AK0
          DO 9 I=1,NK0K
               AM(I+1)=DELTA*FLOAT(NK0K)+AK0
  9       CONTINUE
  7       CONTINUE
C-------------------------------------------------+
C     STEP 2 : EVALUATION OF VECTORS VSPE,VTPE    |
C-------------------------------------------------+
      IF (IFIRST.EQ.2) GO TO 10
          DO 11 I=1,NTE
                ARG=POLTE(I)
                VSPE(I)=WSPE(ARG)
                VTPE(I)=WTPE(ARG)
  11      CONTINUE
  10  CONTINUE
C-------------------------------------------------+
C     STEP 2 : EVALUATION OF VECTOR VSPM          |
```

```
C-----------------------------------------------------+
      DO 12 I=1,NTM
            ARG=POLTM(I)
            VSPM(I)=WSPM(ARG)
  12  CONTINUE
C-----------------------------------------------------+
C     EVALUATION OF THE COEFFICIENTS FOR THE          |
C            FF'S FUNCTIONS                           |
C-----------------------------------------------------+
      F1XX=0.5D0/(1.D0-E2)
      F1ZX=(F1XX-1.D0/((ER+1.D0)*(1.D0-E3)))
      F2XXG=0.5D0/(1.D0+E4)
      F2ZXG=(F2XXG-1.D0/((ER+1.D0)*(1.D0+E6)))
      F2XXL=0.5D0*E4/(1.D0+E4)
      F2ZXL=(F2XXG-1.D0*ER/((1.D0+ER)*(1.D0+E6)))
      IF (IFEED.NE.1) GO TO 75
          F2XXG=F1XX
          F2ZXG=F1ZX
          F2XXL=F1XX
          F2ZXL=F1ZX
  75  F1XXL=F2XXG
      F1ZXL=F2ZXL
      F2XL=F2XXL
      F2ZL=F2ZXL
C-----------------------------------------------------+
C     STEP 2 : EVALUATION OF VECTORS Z1GT,Z2GT,       |
C              Z2LT,Z0GT,Z0LT,Z00                     |
C-----------------------------------------------------+
      V1=AK0*AK0
      V2=1.D0-EER
      V3=ER-EER
      IF (IFEED.EQ.1) V3=V2
      ER1=ER-1.D0
      YSIN=DSIN(AKK*DLX)
      YCOS=DCOS(AKK*DLX)
      SSIN2=YSIN*YSIN
      WSSIN=(2.D0/AKK)*SSIN2
      YSIN2=SSIN2/AKK
      WCOS=(1.D0-YCOS)*AKK
      NMAX=NS+ND
      CALL ARIS
      MMAX=NMAX+2
      DO 16 I=1,NPOINT
            NTTM=NTM
            IADD=1
            IF (IFIRST.NE.2) GO TO 15
                NTTM=NK0K
                IADD=0
  15        AI=COAL(I)
            TI=POINT(I)
C-----------------------------------------------------+
C     STEP 3 : EVALUATION OF INTERVALS 1 AND 2        |
C-----------------------------------------------------+
            DO 17 N=1,NK0
                  X=CN(N+1)-CN(N)
```

```
      Y=CN(N+1)+CN(N)
      ALI=0.5D0*(TI*X+Y)
C    +---------------------+
C    | EVALUATION OF GXX'S |
C    +---------------------+
      CALL RGRI(ALI)
      GCON=AI*X
      FCON=GCON
      GXXR1=GCON*(RX(1)-FRX(1))
      GXXX1=GCON*XX(1)
      GXXR2=GCON*(RX(2)-FRX(2))
      GXXX2=GCON*XX(2)
      GXXR3=GCON*(RX(3)-FRX(3))
      GXXX3=GCON*XX(3)
C    +----------------------+
C    | EVALUATION OF GZX'S  |
C    +----------------------+
      GCON=GCON*ER1
      GZXR1=GCON*RZ(1)-FCON*FRZ(1)
      GZXX1=GCON*XZ(1)
      GZXR2=GCON*RZ(2)-FCON*FRZ(2)
      GZXX2=GCON*XZ(2)
      GZXR3=GCON*RZ(3)-FCON*FRZ(3)
      GZXX3=GCON*XZ(3)
C    +-------------------------------+
C    | REFORMULATION OF GXX'S GZX'S |
C    +-------------------------------+
      VARX=V1*(V2*GXXR1+EER*GZXR1)
      VARZ=AKK*(GXXR1-GZXR1)
      GXXR1=VARX
      GZXR1=VARZ
      VARX=V1*(V2*GXXX1+EER*GZXX1)
      VARZ=AKK*(GXXX1-GZXX1)
      GXXX1=VARX
      GZXX1=VARZ
      VARX=V1*(V2*GXXR2+EER*GZXR2)
      VARZ=AKK*(GXXR2-GZXR2)
      GXXR2=VARX
      GZXR2=VARZ
      VARX=V1*(V2*GXXX2+EER*GZXX2)
      VARZ=AKK*(GXXX2-GZXX2)
      GXXX2=VARX
      GZXX2=VARZ
      VARX=V1*(V3*GXXR3+EER*GZXR3)
      VARZ=AKK*(GXXR3-GZXR3)
      GXXR3=VARX
      GZXR3=VARZ
      VARX=V1*(V3*GXXX3+EER*GZXX3)
      VARZ=AKK*(GXXX3-GZXX3)
      GXXX3=VARX
      GZXX3=VARZ
      PLI=ALI
      CALL ADONIS(MMAX)
      DO 34 K=1,NMAX
         S1=GXXR2*SSJ0(K)+GZXR2*SAJ0(K)
```

```
                        S2=GXXX2*SSJ0(K)+GZXX2*SAJ0(K)
                        Z2GT(K)=Z2GT(K)+S1-CI*S2
                        IF (K.GT.NF) GO TO 35
                            S1=GXXR3*SSJ0(K)+GZXR3*SAJ0(K)
                            S2=GXXX3*SSJ0(K)+GZXX3*SAJ0(K)
                            Z2LT(K)=Z2LT(K)+S1-CI*S2
   35                       IF (K.GT.ND) GO TO 34
                                S1=GXXR1*SSJ0(K)+GZXR1*SAJ0(K)
                                S2=GXXX1*SSJ0(K)+GZXX1*SAJ0(K)
                                Z1GT(K)=Z1GT(K)+S1-CI*S2
   34             CONTINUE
   17       CONTINUE
C-------------------------------------------------+
C      STEP 3 : EVALUATION OF INTERVAL 3          |
C-------------------------------------------------+
            IND=-IADD
            DO 18 N=1,NTTM
                  IND=IND+IADD+1
                  X=AM(IND+1)-AM(IND)
                  Y=AM(IND+1)+AM(IND)
                  ALI=0.5D0*(TI*X+Y)
C           +---------------------+
C           | EVALUATION OF GXX'S |
C           +---------------------+
                  NPOL=N
                  IF (IFIRST.EQ.2) NPOL=1
                  XTM=POLTM(NPOL)
                  CALL WGXZ (ALI,XTM)
                  GCON=AI*X
                  FCON=GCON
                  GXXR1=GCON*(RX(1)-FRX(1))
                  GXXR2=GCON*(RX(2)-FRX(2))
                  GXXR3=GCON*(RX(3)-FRX(3))
C           +---------------------+
C           | EVALUATION OF GZX'S |
C           +---------------------+
                  TMTM=(2.D0*POLTM(NPOL)-Y)/X
                  TTI=TI
                  GCON=2.D0*AI*ER1/(TTI-TMTM)
                  GZXR1=GCON*RZ(1)-FCON*FRZ(1)
                  GZXR2=GCON*RZ(2)-FCON*FRZ(2)
                  GZXR3=GCON*RZ(3)-FCON*FRZ(3)
C           +-----------------------------+
C           | REFORMULATION OF GXX'S GZX'S |
C           +-----------------------------+
                  VARX=V1*(V2*GXXR1+EER*GZXR1)
                  VARZ=AKK*(GXXR1-GZXR1)
                  GXXR1=VARX
                  GZXR1=VARZ
                  VARX=V1*(V2*GXXR2+EER*GZXR2)
                  VARZ=AKK*(GXXR2-GZXR2)
                  GXXR2=VARX
                  GZXR2=VARZ
                  VARX=V1*(V3*GXXR3+EER*GZXR3)
                  VARZ=AKK*(GXXR3-GZXR3)
```

```
               GXXR3=VARX
               GZXR3=VARZ
               PLI=ALI
               CALL ADONIS (MMAX)
               DO 36 K=1,NMAX
                    S=GXXR2*SSJ0(K)+GZXR2*SAJ0(K)
                    Z2GT(K)=Z2GT(K)+S
                    IF (K.GT.NF) GO TO 37
                        S=GXXR3*SSJ0(K)+GZXR3*SAJ0(K)
                        Z2LT(K)=Z2LT(K)+S
 37                     IF (K.GT.ND) GO TO 36
                            S=GXXR1*SSJ0(K)+GZXR1*SAJ0(K)
                            Z1GT(K)=Z1GT(K)+S
 36            CONTINUE
 18         CONTINUE
C-------------------------------------------------------+
C     STEP 3 : EVALUATION OF INTERVAL 4                 |
C-------------------------------------------------------+
            IND=-IADD
            DO 19 N=1,NTTM
               ALI=POLTM(1)
               FAN1=VSPM(1)
               IF (IFIRST.EQ.2) GO TO 20
                   ALI=POLTM(N)
                   FAN1=VSPM(N)
 20            IND=IND+IADD+1
               X=AM(IND+1)-AM(IND)
               Y=AM(IND+1)+AM(IND)
               TM=(2.D0*ALI-Y)/X
C           +---------------------+
C           | EVALUATION OF GZX'S |
C           |    GXX'S =0.D0      |
C           +---------------------+
               CALL WGZTM (FAN1,ALI)
               TTI=TI
               IF (DABS(TI-TM).LT.1.D-6) TTI=TI+1.D-5
               GCON=-2.D0*AI*ER1/(TTI-TM)
               GZXR1=GCON*RZ(1)
               GZXR2=GCON*RZ(2)
               GZXR3=GCON*RZ(3)
C           +-------------------------------+
C           | REFORMULATION OF GXX'S,GZX'S  |
C           +-------------------------------+
               VARX=V1*EER*GZXR1
               VARZ=-AKK*GZXR1
               GXXR1=VARX
               GZXR1=VARZ
               VARX=V1*EER*GZXR2
               VARZ=-AKK*GZXR2
               GXXR2=VARX
               GZXR2=VARZ
               VARX=V1*EER*GZXR3
               VARZ=-AKK*GZXR3
               GXXR3=VARX
               GZXR3=VARZ
```

```
                PLI=ALI
                CALL ADONIS(MMAX)
                DO 38 K=1,NMAX
                      S=GXXR2*SSJ0(K)+GZXR2*SAJ0(K)
                      Z2GT(K)=Z2GT(K)+S
                      IF (K.GT.NF) GO TO 39
                          S=GXXR3*SSJ0(K)+GZXR3*SAJ0(K)
                          Z2LT(K)=Z2LT(K)+S
  39                      IF (K.GT.ND) GO TO 38
                              S=GXXR1*SSJ0(K)+GZXR1*SAJ0(K)
                              Z1GT(K)=Z1GT(K)+S
  38            CONTINUE
  19      CONTINUE
          IADD=2
C--------------------------------------------------+
C     STEP 3 : EVALUATION OF INTERVAL 5            |
C--------------------------------------------------+
          IND=-1
          IF (IFIRST.EQ.2) GO TO 21
          DO 21 N=1,NTE
                IND=IND+IADD
                X=DM(IND+1)-DM(IND)
                Y=DM(IND+1)+DM(IND)
                ALI=0.5D0*(TI*X+Y)
C         +----------------------+
C         | EVALUATION OF GXX'S |
C         +----------------------+
                FCON=AI*X
                CALL WGXZE(ALI,POLTE(N))
                TMTE=(2.D0*POLTE(N)-Y)/X
                GCON=2.D0*AI/(TI-TMTE)
                GXXR1=GCON*RX(1)-FCON*FRX(1)
                GXXR2=GCON*RX(2)-FCON*FRX(2)
                GXXR3=GCON*RX(3)-FCON*FRX(3)
C         +----------------------+
C         | EVALUATION OF GZX'S  |
C         +----------------------+
                GCON=GCON*ER1
                GZXR1=GCON*RZ(1)-FCON*FRZ(1)
                GZXR2=GCON*RZ(2)-FCON*FRZ(2)
                GZXR3=GCON*RZ(3)-FCON*FRZ(3)
C         +------------------------------+
C         | REFORMULATION OF GXX'S,GZX'S |
C         +------------------------------+
                VARX=V1*(V2*GXXR1+EER*GZXR1)
                VARZ=AKK*(GXXR1-GZXR1)
                GXXR1=VARX
                GZXR1=VARZ
                VARX=V1*(V2*GXXR2+EER*GZXR2)
                VARZ=AKK*(GXXR2-GZXR2)
                GXXR2=VARX
                GZXR2=VARZ
                VARX=V1*(V3*GXXR3+EER*GZXR3)
                VARZ=AKK*(GXXR3-GZXR3)
                GXXR3=VARX
```

```
                GZXR3=VARZ
                PLI=ALI
                CALL ADONIS(MMAX)
                DO 40 K=1,NMAX
                      S=GXXR2*SSJ0(K)+GZXR2*SAJ0(K)
                      Z2GT(K)=Z2GT(K)+S
                      IF (K.GT.NF) GO TO 41
                          S=GXXR3*SSJ0(K)+GZXR3*SAJ0(K)
                          Z2LT(K)=Z2LT(K)+S
 41                       IF (K.GT.ND) GO TO 40
                              S=GXXR1*SSJ0(K)+GZXR1*SAJ0(K)
                              Z1GT(K)=Z1GT(K)+S
 40             CONTINUE
 21          CONTINUE
C-------------------------------------------------+
C     STEP 3 : EVALUATION OF INTERVALS 6,9,11     |
C-------------------------------------------------+
          IND=-1
          IF (IFIRST.EQ.2) GO TO 22
          DO 22 N=1,NTE
                ALI=POLTE(N)
                IND=IND+IADD
                X=DM(IND+1)-DM(IND)
                Y=DM(IND+1)+DM(IND)
                TM=(2.D0*ALI-Y)/X
C         +----------------------+
C         | EVALUATION OF GXX'S  |
C         +----------------------+
                TTI=TI
                IF (DABS(TI-TM).LT.1.D-6) TTI=TI+1.D-5
                FAN1=VSPE(N)
                FAN2=VTPE(N)
                CALL WGXZTE (FAN1,FAN2,POLTE(N))
                GCON=-2.D0*AI/(TTI-TM)
                GXXR1=GCON*RX(1)
                GXXR2=GCON*RX(2)
                GXXR3=GCON*RX(3)
C         +---------------------+
C         | EVALUATION OF GZX'S |
C         +---------------------+
                GCON=GCON*ER1
                GZXR1=GCON*RZ(1)
                GZXR2=GCON*RZ(2)
                GZXR3=GCON*RZ(3)
C         +-----------------------------+
C         | REFORMULATION OF GXX'S,GZX'S |
C         +-----------------------------+
                VARX=V1*(V2*GXXR1+EER*GZXR1)
                VARZ=AKK*(GXXR1-GZXR1)
                GXXR1=VARX
                GZXR1=VARZ
                VARX=V1*(V2*GXXR2+EER*GZXR2)
                VARZ=AKK*(GXXR2-GZXR2)
                GXXR2=VARX
                GZXR2=VARZ
```

```
            VARX=V1*(V3*GXXR3+EER*GZXR3)
            VARZ=AKK*(GXXR3-GZXR3)
            GXXR3=VARX
            GZXR3=VARZ
            PLI=ALI
            CALL ADONIS(MMAX)
            DO 42 K=1,NMAX
                  S=GXXR2*SSJ0(K)+GZXR2*SAJ0(K)
                  Z2GT(K)=Z2GT(K)+S
                  IF (K.GT.NF) GO TO 61
                      S=GXXR3*SSJ0(K)+GZXR3*SAJ0(K)
                      Z2LT(K)=Z2LT(K)+S
 61                   IF (K.GT.ND) GO TO 42
                          S=GXXR1*SSJ0(K)+GZXR1*SAJ0(K)
                          Z1GT(K)=Z1GT(K)+S
 42         CONTINUE
            IF (I.LT.NPOINT) GO TO 22
                VLOG=DLOG((1.D0-TM)/(1.D0+TM))
C     +--------------------+
C     | EVALUATION OF GXX'S |
C     +--------------------+
                GCON1=2.D0*VLOG
                GCON2=2.D0*PI
                GXXR11=GCON1*RX(1)
                GXXR21=GCON1*RX(2)
                GXXR31=GCON1*RX(3)
                GXXR12=GCON2*RX(1)
                GXXR22=GCON2*RX(2)
                GXXR32=GCON2*RX(3)
C     +--------------------+
C     | EVALUATION OF GZX'S |
C     +--------------------+
                GCON1=GCON1*ER1
                GCON2=GCON2*ER1
                GZXR11=GCON1*RZ(1)
                GZXR21=GCON1*RZ(2)
                GZXR31=GCON1*RZ(3)
                GZXR12=GCON2*RZ(1)
                GZXR22=GCON2*RZ(2)
                GZXR32=GCON2*RZ(3)
C     +---------------------------+
C     | REFORMATION OF GXX'S,GZX'S |
C     +---------------------------+
                VARX=V1*(V2*GXXR11+EER*GZXR11)
                VARZ=AKK*(GXXR11-GZXR11)
                GXXR11=VARX
                GZXR11=VARZ
                VARX=V1*(V2*GXXR21+EER*GZXR21)
                VARZ=AKK*(GXXR21-GZXR21)
                GXXR21=VARX
                GZXR21=VARZ
                VARX=V1*(V3*GXXR31+EER*GZXR31)
                VARZ=AKK*(GXXR31-GZXR31)
                GXXR31=VARX
                GZXR31=VARZ
```

```
                  VARX=V1*(V2*GXXR12+EER*GZXR12)
                  VARZ=AKK*(GXXR12-GZXR12)
                  GXXR12=VARX
                  GZXR12=VARZ
                  VARX=V1*(V2*GXXR22+EER*GZXR22)
                  VARZ=AKK*(GXXR22-GZXR22)
                  GXXR22=VARX
                  GZXR22=VARZ
                  VARX=V1*(V3*GXXR32+EER*GZXR32)
                  VARZ=AKK*(GXXR32-GZXR32)
                  GXXR32=VARX
                  GZXR32=VARZ
C
                  DO 43 K=1,NMAX
                        S1=GXXR21*SSJ0(K)+GZXR21*SAJ0(K)
                        S2=GXXR22*SSJ0(K)+GZXR22*SAJ0(K)
                        Z2GT(K)=Z2GT(K)+S1-CI*S2
                        IF (K.GT.NF) GO TO 62
                           S1=GXXR31*SSJ0(K)+GZXR31*SAJ0(K)
                           S2=GXXR32*SSJ0(K)+GZXR32*SAJ0(K)
                           Z2LT(K)=Z2LT(K)+S1-CI*S2
  62                       IF (K.GT.ND) GO TO 43
                               S1=GXXR11*SSJ0(K)+GZXR11*SAJ0(K)
                               S2=GXXR12*SSJ0(K)+GZXR12*SAJ0(K)
                               Z1GT(K)=Z1GT(K)+S1-CI*S2
  43              CONTINUE
  22        CONTINUE
C-----------------------------------------------+
C     STEP 3 : EVALUATION OF INTERVAL 7         |
C-----------------------------------------------+
            DO 23 N=1,MA
                  X=BM(N+1)-BM(N)
                  Y=BM(N+1)+BM(N)
                  ALI=0.5D0*(TI*X+Y)
C           +--------------------+
C           | EVALUATION OF GXX'S |
C           +--------------------+
                  CALL PGXZ(ALI)
                  GCON=AI*X
                  FCON=GCON
                  GXXR1=GCON*(RX(1)-FRX(1))
                  GXXR2=GCON*(RX(2)-FRX(2))
                  GXXR3=GCON*(RX(3)-FRX(3))
C           +--------------------+
C           | EVALUATION OF GZX'S |
C           +--------------------+
                  GCON=GCON*ER1
                  GZXR1=GCON*RZ(1)-FCON*FRZ(1)
                  GZXR2=GCON*RZ(2)-FCON*FRZ(2)
                  GZXR3=GCON*RZ(3)-FCON*FRZ(3)
C           +-----------------------------+
C           | REFORMULATION OF GXX'S,GZX'S |
C           +-----------------------------+
                  VARX=V1*(V2*GXXR1+EER*GZXR1)
                  VARZ=AKK*(GXXR1-GZXR1)
```

```
                GXXR1=VARX
                GZXR1=VARZ
                VARX=V1*(V2*GXXR2+EER*GZXR2)
                VARZ=AKK*(GXXR2-GZXR2)
                GXXR2=VARX
                GZXR2=VARZ
                VARX=V1*(V3*GXXR3+EER*GZXR3)
                VARZ=AKK*(GXXR3-GZXR3)
                GXXR3=VARX
                GZXR3=VARZ
C
                PLI=ALI
                CALL ADONIS(MMAX)
                DO 45 K=1,NMAX
                      S=GXXR2*SSJ0(K)+GZXR2*SAJ0(K)
                      Z2GT(K)=Z2GT(K)+S
                      IF (K.GT.NF) GO TO 63
                          S=GXXR3*SSJ0(K)+GZXR3*SAJ0(K)
                          Z2LT(K)=Z2LT(K)+S
   63                     IF (K.GT.ND) GO TO 45
                              S=GXXR1*SSJ0(K)+GZXR1*SAJ0(K)
                              Z1GT(K)=Z1GT(K)+S
   45           CONTINUE
   23     CONTINUE
   16 CONTINUE
C-------------------------------------------------+
C     STEP 3 : EVALUATION OF INTERVALS 8,10       |
C-------------------------------------------------+
      IND=-1
      IADD=2
      DO 25 N=1,NTM
            IF (IFIRST.NE.2) GO TO 24
                TM=(2.D0*POLTM(1)-(AK+AK0))/(AK-AK0)
                ALI=POLTM(1)
                GO TO 26
   24       ALI=POLTM(N)
            IND=IND+IADD
            X=AM(IND+1)-AM(IND)
            Y=AM(IND+1)+AM(IND)
            TM=(2.D0*ALI-Y)/X
   26       CONTINUE
C           +---------------------+
C           | EVALUATION OF GZX'S |
C           +---------------------+
            FAN1=VSPM(N)
            CALL WGZTM (FAN1,ALI)
            GCON1=2.D0*ER1*DLOG((1.D0-TM)/(1.D0+TM))
            GCON2=2.D0*ER1*PI
            GZXR11=GCON1*RZ(1)
            GZXR21=GCON1*RZ(2)
            GZXR31=GCON1*RZ(3)
            GZXR12=GCON2*RZ(1)
            GZXR22=GCON2*RZ(2)
            GZXR32=GCON2*RZ(3)
C           +-----------------------------+
```

```
C           | REFORMULATION OF GXX'S,GZX'S |
C           +------------------------------+
            VARX=V1*EER*GZXR11
            VARZ=-AKK*GZXR11
            GXXR11=VARX
            GZXR11=VARZ
            VARX=V1*EER*GZXR21
            VARZ=-AKK*GZXR21
            GXXR21=VARX
            GZXR21=VARZ
            VARX=V1*EER*GZXR31
            VARZ=-AKK*GZXR31
            GXXR31=VARX
            GZXR31=VARZ
            VARX=V1*EER*GZXR12
            VARZ=-AKK*GZXR12
            GXXR12=VARX
            GZXR12=VARZ
            VARX=V1*EER*GZXR22
            VARZ=-AKK*GZXR22
            GXXR22=VARX
            GZXR22=VARZ
            VARX=V1*EER*GZXR32
            VARZ=-AKK*GZXR32
            GXXR32=VARX
            GZXR32=VARZ
C
            PLI=ALI
            CALL ADONIS(MMAX)
            DO 47 K=1,NMAX
                  S1=GXXR21*SSJ0(K)+GZXR21*SAJ0(K)
                  S2=GXXR22*SSJ0(K)+GZXR22*SAJ0(K)
                  Z2GT(K)=Z2GT(K)+S1-CI*S2
                  IF (K.GT.NF) GO TO 64
                      S1=GXXR31*SSJ0(K)+GZXR31*SAJ0(K)
                      S2=GXXR32*SSJ0(K)+GZXR32*SAJ0(K)
                      Z2LT(K)=Z2LT(K)+S1-CI*S2
  64                  IF (K.GT.ND) GO TO 47
                          S1=GXXR11*SSJ0(K)+GZXR11*SAJ0(K)
                          S2=GXXR12*SSJ0(K)+GZXR12*SAJ0(K)
                          Z1GT(K)=Z1GT(K)+S1-CI*S2
  47        CONTINUE
  25  CONTINUE
C
      CALL TAIL
C
      CONST1=(1.D0/CVON)*15.D0*DSQRT(EER)/(PI*(YSIN*YSIN)*100.D0)
      CONST2=CONST1/ER
      IF (IOPT.NE.1) GO TO 100
      WRITE(6,66)
   66 FORMAT ('..................... Z11................')
      IF (IFEED.EQ.1) CONST2=CONST1
      WRITE (6,67) ND
   67 FORMAT (11X,I4)
      DO 50 K=1,ND
```

```
          Z1GT(K)=Z1GT(K)*CONST1
          G1GT(K)=G1GT(K)*CONST1
          Z1GT(K)=(Z1GT(K)+G1GT(K))*CI
          WRITE (6,51) Z1GT(K)
 51       FORMAT(11X,E14.7,4X,E14.7)
 50 CONTINUE
100 CONTINUE
    IF(IOPT.EQ.1) WRITE(6,101)
101 FORMAT('...................... Z31 ......................')
    IF (IOPT.EQ.2) WRITE(6,102)
102 FORMAT('...................... Z21 ......................')
    IF(IOPT.EQ.3) GO TO 103
    WRITE (6,67) NMAX
    DO 52 K=1,NMAX
          Z2GT(K)=Z2GT(K)*CONST1
          G2GT(K)=G2GT(K)*CONST1
          Z2GT(K)=(Z2GT(K)+G2GT(K))*CI
          WRITE (6,51) Z2GT(K)
 52 CONTINUE
103 CONTINUE
    IF (IOPT.EQ.1) WRITE (6,104)
104 FORMAT('...................... Z33 ......................')
    IF(IOPT.EQ.2) WRITE(6,105)
105 FORMAT('...................... Z22 ......................')
    IF (IOPT.EQ.3) WRITE(6,106)
106 FORMAT ('..................... Z32 ......................')
    WRITE (6,67) NF
    DO 54 K=1,NF
          Z2LT(K)=Z2LT(K)*CONST2
          G2LT(K)=G2LT(K)*CONST2
          Z2LT(K)=(Z2LT(K)+G2LT(K))*CI
          WRITE (6,55) Z2LT(K)
 55       FORMAT(11X,E14.7,4X,E14.7)
 54 CONTINUE
    STOP
    END
C***********************************************************************
C                          WGXZTE
C***********************************************************************
      SUBROUTINE WGXZTE(F1,F2,XTE)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
C
      X=XTE
      X2=X*X
      AK2=AK*AK
      AK02=AK0*AK0
      R1=DSQRT(AK2-X2)
      R2=DSQRT(X2-AK02)
      R3=R1*H
      R4=R2*TT1
      R5=R1*(H-BS)
```

```
      R6=R1*(H-BS+TT2)
      R7=R1*(BS-TT2)
      R8=R1*(H-BS-TT2)
      R9=R1*BS
      S1=DSIN(R3)
      C1=DCOS(R3)
      S2=DSIN(R5)
      S3=DSIN(R6)
      S4=DSIN(R7)
      S5=2.D0*S1*C1
      S6=DSIN(R8)
      S7=DSIN(R9)
      C5=DCOS(R9)
      C4=DCOS(R7)
      EX=DEXP(-R4)
      EX1=DEXP(-R2*DABS(TT2-BS))
C
      RX(1)=X*EX*S1*F1
      RX(2)=X*EX*S2*F1
      RX(3)=X*(S2/R1)*(R1*C4+R2*S4)*F1
      IF (IFEED.EQ.1) RX(3)=X*EX1*S2*F1
C
      CQ1=AK2/(X2-AK2)+1.D0
      RZ(1)=X*EX*R2*0.5D0*S5*F2
      RZ(2)=X*EX*R2*S2*C1*F2
      RZ(3)=X*R1*S2*S3*F2
      IF (IFEED.EQ.1) RZ(3)=X*EX1*R2*S2*C1*F2
      RETURN
      END
C*********************************************************************
C                              PGXZ
C*********************************************************************
      SUBROUTINE PGXZ (ALI)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
C
      X=ALI
      X2=X*X
      AK2=AK*AK
      AK02=AK0*AK0
      R1=DSQRT(X2-AK2)
      R2=DSQRT(X2-AK02)
      R3=R1*H
      R4=R2*TT1
      R5=R1*BS
      R6=R1*(BS-TT2)
      R7=R1*(BS+TT2)
      EX=DEXP(R3)
      TAN1=(EX-1.D0/EX)/(EX+1.D0/EX)
      EX=DEXP(R6)
      COSH1=0.5D0*(EX+1.D0/EX)
      SINH1=0.5D0*(EX-1.D0/EX)
```

```
      R6=R1*(H-BS+TT2)
      R7=R1*(BS-TT2)
      R8=R1*(H-BS-TT2)
      R9=R1*BS
      S1=DSIN(R3)
      C1=DCOS(R3)
      S2=DSIN(R5)
      S3=DSIN(R6)
      S4=DSIN(R7)
      S5=2.D0*S1*C1
      S6=DSIN(R8)
      S7=DSIN(R9)
      C5=DCOS(R9)
      C4=DCOS(R7)
      EX=DEXP(-R4)
      EX1=DEXP(-R2*DABS(TT2-BS))
C
      RX(1)=X*EX*S1*F1
      RX(2)=X*EX*S2*F1
      RX(3)=X*(S2/R1)*(R1*C4+R2*S4)*F1
      IF (IFEED.EQ.1) RX(3)=X*EX1*S2*F1
C
      CQ1=AK2/(X2-AK2)+1.D0
      RZ(1)=X*EX*R2*0.5D0*S5*F2
      RZ(2)=X*EX*R2*S2*C1*F2
      RZ(3)=X*R1*S2*S3*F2
      IF (IFEED.EQ.1) RZ(3)=X*EX1*R2*S2*C1*F2
      RETURN
      END
C***********************************************************************
C                                PGXZ
C***********************************************************************
      SUBROUTINE PGXZ (ALI)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
C
      X=ALI
      X2=X*X
      AK2=AK*AK
      AK02=AK0*AK0
      R1=DSQRT(X2-AK2)
      R2=DSQRT(X2-AK02)
      R3=R1*H
      R4=R2*TT1
      R5=R1*BS
      R6=R1*(BS-TT2)
      R7=R1*(BS+TT2)
      EX=DEXP(R3)
      TAN1=(EX-1.D0/EX)/(EX+1.D0/EX)
      EX=DEXP(R6)
      COSH1=0.5D0*(EX+1.D0/EX)
      SINH1=0.5D0*(EX-1.D0/EX)
```

```
      EX=DEXP(R5)
      COSH2=0.5D0*(EX+1.D0/EX)
      SINH2=0.5D0*(EX-1.D0/EX)
      EX=DEXP(R7)
      SINH3=0.5D0*(EX-1.D0/EX)
      COSH3=0.5D0*(EX+1.D0/EX)
      EX=DEXP(-R4)
      EX1=DEXP(-R2*DABS(TT2-BS))
      EXA=DEXP(-X*TT1*FA0)*FA0
      EXB=DEXP(-X*BW*FA)*FA
      EXC=DEXP(-X*TT2*FA)*FA
      EXD=DEXP(-X*BWWW*FA)*FA
      EXE=DEXP(-X*BWW*FA)*FA
      EXF=DEXP(-X*B4W*FA)*FA
      EXG=DEXP(-X*B5W*FA)*FA
C
      IF ((X-AK).LT.1.D-6) GO TO 1
          CQ1=R2*TAN1+R1
          CQ2=ER*R2+R1*TAN1
          CQ3=TAN1*COSH2-SINH2
          RX(1)=EX*TAN1*X/CQ1
          FRX(1)=F1XX*EXA
          RX(2)=EX*CQ3*X/CQ1
          FRX(2)=F2XXG*EXB
          RX(3)=CQ3*X*(R1*COSH1+R2*SINH1)/(R1*CQ1)
          FRX(3)=0.5D0*EXC-F2XXL*EXD
          IF (IFEED.EQ.1) RX(3)=EX1*CQ3*X/CQ1
          IF (IFEED.EQ.1) FRX(3)=F2XXL*EXE
C
          CQ4=CQ1*CQ2
          CQ5=AK2/(X2-AK2)+1.D0
          RZ(1)=EX*R2*TAN1*X/CQ4
          FRZ(1)=F1ZX*EXA
          RZ(2)=EX*CQ3*R2*X/CQ4
          FRZ(2)=F2ZXG*EXB
          RZ(3)=-CQ3*(TAN1*COSH1-SINH1)*R1*X/CQ4
          FRZ(3)=F2ZXL*EXD
          IF (IFEED.EQ.1) RZ(3)=EX1*CQ3*R2*X/CQ4
          IF (IFEED.EQ.1) FRZ(3)=F2ZXL*EXE
          RETURN
C
  1   CQ1=R2*H+1.D0
      CQ2=ER*R2
      CQ4=CQ1*CQ2
      RX(1)=EX*H*X/CQ1
      FRX(1)=F1XX*EXA
      RX(2)=EX*(H-BS)*X/CQ1
      FRX(2)=F2XXG*EXB
      RX(3)=X*(H-BS)*(1.D0+R2*(BS-TT2))/CQ1
      FRX(3)=0.5D0*EXC-F2XXL*EXD
      IF (IFEED.EQ.1) RX(3)=EX1*(H-BS)*X/CQ1-F2XXL*EXE
      IF (IFEED.EQ.1) FRX(3)=F2XXL*EXE
C
      RZ(1)=EX*R2*H*X/CQ4
      FRZ(1)=F1ZX*EXA
```

```
      RZ(2)=EX*R2*(H-BS)*X/CQ4
      FRZ(2)=F2ZXG*EXB
      RZ(3)=0.D0
      FRZ(3)=F2ZXL*EXD
      IF (IFEED.EQ.1) RZ(3)=EX1*R2*(H-BS)*X/CQ4
      IF (IFEED.EQ.1) FRZ(3)=F2ZXL*EXE
      RETURN
      END
C**********************************************************************
C                           WGXZE
C**********************************************************************
      SUBROUTINE WGXZE (ALI,XTE)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
C
      X=ALI
      X2=X*X
      AK2=AK*AK
      AK02=AK0*AK0
      R1=DSQRT(AK2-X2)
      R2=DSQRT(X2-AK02)
      R3=R1*H
      R4=R2*TT1
      R5=R1*(H-BS)
      R6=R1*(BS-TT2)
      R7=R1*(H-BS+TT2)
      R8=R1*(H-BS-TT2)
      R9=R1*BS
      S1=DSIN(R3)
      C1=DCOS(R3)
      C3=DCOS(R6)
      C4=DCOS(R9)
      S2=DSIN(R5)
      S3=DSIN(R6)
      S4=2.D0*S1*C1
      S5=DSIN(R7)
      S6=DSIN(R8)
      S7=DSIN(R9)
      EX=DEXP(-R4)
      EX1=DEXP(-R2*DABS(TT2-BS))
      EXA=DEXP(-X*TT1*FA0)*FA0
      EXB=DEXP(-X*BW*FA)*FA
      EXC=DEXP(-X*TT2*FA)*FA
      EXD=DEXP(-X*BWWW*FA)*FA
      EXE=DEXP(-X*BWW*FA)*FA
      EXF=DEXP(-X*B4W*FA)*FA
      EXG=DEXP(-X*B5W*FA)*FA
C
      IF ((AK-X).LT.1.D-6) GO TO 1
         CQ1=R2*S1+R1*C1
         SL=(X-XTE)/CQ1
         RX(1)=X*EX*S1*SL
```

```
      FRX(1)=F1XX*EXA
      RX(2)=X*EX*S2*SL
      FRX(2)=F2XXG*EXB
      RX(3)=X*(S2/R1)*(R1*C3+R2*S3)*SL
      FRX(3)=0.5D0*EXC-F2XXL*EXD
      IF (IFEED.EQ.1) RX(3)=X*EX1*S2*SL
      IF (IFEED.EQ.1) FRX(3)=F2XXL*EXE
C
      CQ2=ER*R2*C1-R1*S1
      CQ3=AK2/(X2-AK2)+1.D0
      TL=(X-XTE)/(CQ1*CQ2)
      RZ(1)=X*EX*R2*0.5D0*S4*TL
      FRZ(1)=F1ZX*EXA
      RZ(2)=X*EX*R2*S2*C1*TL
      FRZ(2)=F2ZXG*EXB
      RZ(3)=R1*S2*S5*X*TL
      FRZ(3)=F2ZXL*EXD
      IF (IFEED.EQ.1) RZ(3)=X*EX1*R2*S2*C1*TL
      IF (IFEED.EQ.1) FRZ(3)=F2ZXL*EXE
      RETURN
C
  1   SL=(X-XTE)/(R2*H+1.D0)
      RX(1)=X*EX*H*SL
      FRX(1)=F1XX*EXA
      RX(2)=X*EX*(H-BS)*SL
      FRX(2)=F2XXG*EXB
      RX(3)=X*EX*S2*SL
      FRX(3)=0.5D0*EXC-F2XXL*EXD
      IF (IFEED.EQ.1) RX(3)=X*EX1*(H-BS)*SL
      IF (IFEED.EQ.1) FRX(3)=F2XXL*EXE
C
      TL=SL/(ER*R2)
      RZ(1)=X*EX*R2*H*TL
      FRZ(1)=F1ZX*EXA
      RZ(2)=X*EX*R2*(H-BS)*TL
      FRZ(2)=F2ZXG*EXB
      RZ(3)=0.D0
      FRZ(3)=F2ZXL*EXD
      IF (IFEED.EQ.1) RZ(3)=X*EX1*R2*(H-BS)*TL
      IF (IFEED.EQ.1) FRZ(3)=F2ZXL*EXE
      RETURN
      END
C*********************************************************************
C                            WGZTM
C*********************************************************************
      SUBROUTINE WGZTM(F,XTM)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
C
      X=XTM
      X2=X*X
      AK2=AK*AK
```

```
      AK02=AK0*AK0
      R1=DSQRT(AK2-X2)
      R2=DSQRT(X2-AK02)
      R3=R1*H
      R4=R2*TT1
      R5=R1*(H-BS)
      R6=R1*(H-BS+TT2)
      R7=R1*(H-BS-TT2)
      S1=DSIN(R3)
      C1=DCOS(R3)
      S2=2.D0*S1*C1
      S3=DSIN(R5)
      S4=DSIN(R6)
      S5=DSIN(R7)
      EX=DEXP(-R2*TT1)
      EX1=DEXP(-R2*DABS(TT2-BS))
C
      CQ1=AK2/(X2-AK2)+1.D0
      RZ(1)=X*EX*R2*0.5D0*S2*F
      RZ(2)=X*EX*R2*S3*C1*F
      RZ(3)=X*R1*S3*S4*F
      IF (IFEED.EQ.1) RZ(3)=X*EX1*R2*S3*C1*F
      RETURN
      END
C***********************************************************************
C                            WGXZ
C***********************************************************************
      SUBROUTINE WGXZ(ALI,XTM)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
C
      X=ALI
      X2=X*X
      AK2=AK*AK
      AK02=AK0*AK0
      R1=DSQRT(AK2-X2)
      R2=DSQRT(X2-AK02)
      R3=R1*H
      R4=R1*(H-BS)
      R5=R1*(BS-TT2)
      R6=R1*(H-BS+TT2)
      R7=R1*(H-BS-TT2)
      R8=R1*BS
      S1=DSIN(R3)
      C1=DCOS(R3)
      S2=DSIN(R4)
      S3=DSIN(R5)
      C3=DCOS(R5)
      EX=DEXP(-R2*TT1)
      EX1=DEXP(-R2*DABS(TT2-BS))
      S4=2.D0*S1*C1
      S5=DSIN(R6)
```

```
      S6=DSIN(R7)
      S7=DSIN(R8)
      C4=DCOS(R8)
      EXA=DEXP(-X*TT1*FA0)*FA0
      EXB=DEXP(-X*BW*FA)*FA
      EXC=DEXP(-X*TT2*FA)*FA
      EXD=DEXP(-X*BWWW*FA)*FA
      EXE=DEXP(-X*BWW*FA)*FA
      EXF=DEXP(-X*B4W*FA)*FA
      EXG=DEXP(-X*B5W*FA)*FA
      IF ((AK-X).LT.1.D-6) GO TO 1
           CQ1=R2*S1+R1*C1
           RX(1)=EX*S1*X/CQ1
           FRX(1)=F1XX*EXA
           RX(2)=EX*S2*X/CQ1
           FRX(2)=F2XXG*EXB
           RX(3)=S2*(R1*C3+R2*S3)*X/(CQ1*R1)
           FRX(3)=0.5D0*EXC-F2XXL*EXD
           IF (IFEED.EQ.1) RX(3)=EX1*S2*X/CQ1
           IF (IFEED.EQ.1) FRX(3)=F2XXL*EXE
C
           CQ2=ER*R2*C1-R1*S1
           CQ3=AK2/(X2-AK2)+1.D0
           SL=(X-XTM)/(CQ1*CQ2)
           RZ(1)=X*EX*R2*0.5D0*S4*SL
           FRZ(1)=F1ZX*EXA
           RZ(2)=X*EX*R2*S2*C1*SL
           FRZ(2)=F2ZXG*EXB
           RZ(3)=X*R1*S2*S5*SL
           FRZ(3)=F2ZXL*EXD
           IF (IFEED.EQ.1) RZ(3)=X*EX1*R2*S2*C1*SL
           IF (IFEED.EQ.1) FRZ(3)=F2ZXL*EXE
           RETURN
C
    1 CQ1=R2*H+1.D0
      RX(1)=EX*H*X/CQ1
      FRX(1)=F1XX*EXA
      RX(2)=EX*(H-BS)*X/CQ1
      FRX(2)=F2XXG*EXB
      RX(3)=(H-BS)*(1.D0+R2*(BS-TT2))*X/CQ1
      FRX(3)=0.5D0*EXC-F2XXL*EXD
      IF (IFEED.EQ.1) RX(3)=EX1*(H-BS)*X/CQ1
      IF (IFEED.EQ.1) FRX(3)=F2XXL*EXE
C
      CQ2=ER*R2
      SL=(X-XTM)/(CQ1*CQ2)
      RZ(1)=X*EX*R2*0.5D0*H*SL
      FRZ(1)=F1ZX*EXA
      RZ(2)=X*EX*R2*(H-BS)*SL
      FRZ(2)=F2ZXG*EXB
      RZ(3)=0.D0
      FRZ(3)=F2ZXL*EXD
      IF (IFEED.EQ.1) RZ(3)=X*EX1*R2*(H-BS)*SL
      IF (IFEED.EQ.1) FRZ(3)=F2ZXL*EXE
      RETURN
```

```
      END
C***********************************************************************
C                           RGRI
C***********************************************************************
      SUBROUTINE RGRI (ALI)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
C
      X=ALI
      X2=X*X
      AK2=AK*AK
      AK02=AK0*AK0
      R1=DSQRT(AK2-X2)
      R2=DSQRT(AK02-X2)
      R3=R1*H
      R4=R2*TT1
      R5=R1*(H-BS+TT2)
      R6=R1*(-BS+TT2)
      R7=R1*(H-BS)
      R8=R1*(H-BS-TT2)
      S1=DSIN(R3)
      S12=S1*S1
      C1=DCOS(R3)
      C12=C1*C1
      S2=DSIN(R4)
      C2=DCOS(R4)
      S3=2.D0*S1*C1
      S4=DSIN(R7)
      S5=DSIN(R6)
      C3=DCOS(R5)
      S6=DSIN(R5)
      C4=DCOS(R6)
      S7=DSIN(R8)
      C5=DCOS(R7)
      S8=DSIN(R1*BS)
      CQ1=R2*R2+AK02*(ER-1.D0)*C12
      EXA=DEXP(-X*TT1*FA0)*FA0
      EXB=DEXP(-X*BW*FA)*FA
      EXC=DEXP(-X*TT2*FA)*FA
      EXD=DEXP(-X*BWWW*FA)*FA
      EXE=DEXP(-X*BWW*FA)*FA
      EXF=DEXP(-X*B4W*FA)*FA
      EXG=DEXP(-X*B5W*FA)*FA
C
      RX(1)=(-R2*S2*S12+0.5D0*R1*C2*S3)*X/CQ1
      XX(1)=(R2*C2*S12+0.5D0*R1*S2*S3)*X/CQ1
      FRX(1)=F1XX*EXA
C
      RX(2)=S4*(-R2*S1*S2+R1*C2*C1)*X/CQ1
      XX(2)=S4*(R2*S1*C2+R1*S2*C1)*X/CQ1
      FRX(2)=F2XXG*EXB
C
```

```
      IF (IFEED.NE.1) GO TO 1
          RX(3)=S4*(-R2*S1*S5+R1*C4*C1)*X/CQ1
          XX(3)=S4*(R2*S1*C4+R1*S5*C1)*X/CQ1
          FRX(3)=F2XXL*EXE
          GO TO 2
    1 RX(3)=(S4/R1)*(R1*R1*C3+AK02*(ER-1.D0)*S1*S5)*X/CQ1
      XX(3)=S4*R2*S6*X/CQ1
      FRX(3)=0.5D0*EXC-F2XXL*EXD
C
    2 CQ2=R1*R1+(ER*AK02*(ER-1.D0)-X2*(ER*ER-1.D0))*C12
      CQ3=1.D0-(1.D0+ER)*C12
      CQ4=0.5D0*(2.D0*AK2-X2*(1.D0+ER))
      CQ5=AK2/(X2-AK2)+1.D0
      CQ6=CQ1*CQ2
      RZ(1)=-0.5D0*R2*S3*(R1*R2*CQ3*C2+CQ4*S2*S3)*X/CQ6
      XZ(1)=0.5D0*R2*S3*(CQ4*S3*C2-R1*R2*CQ3*S2)*X/CQ6
      FRZ(1)=F1ZX*EXA
C
      RZ(2)=R2*S4*(-R1*R2*C2*CQ3-CQ4*S2*S3)*X*C1/CQ6
      XZ(2)=R2*S4*(-R1*R2*S2*CQ3+CQ4*C2*S3)*X*C1/CQ6
      FRZ(2)=F2ZXG*EXB
C
      IF (IFEED.NE.1) GO TO 3
          RZ(3)=R2*S4*(-R1*R2*C4*CQ3-CQ4*S5*S3)*X*C1/CQ6
          XZ(3)=R2*S4*(-R1*R2*S5*CQ3+CQ4*C4*S3)*X*C1/CQ6
          FRZ(3)=F2ZXL*EXE
          GO TO 4
    3 RZ(3)=-CQ4*S4*S6*S3*X*R1/CQ6
      XZ(3)=-R2*R1*R1*S4*S6*CQ3*X/CQ6
      FRZ(3)=F2ZXL*EXD
C
    4 CONTINUE
      RETURN
      END
C***********************************************************************
C               FUNCTIONS :  WSPE,WTPE,WSPM
C***********************************************************************
      FUNCTION WSPE(X)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
C
      X2=X*X
      AK02=AK0*AK0
      AK2=AK*AK
      R1=DSQRT(AK2-X2)
      R2=DSQRT(X2-AK02)
      R3=R1*H
      R4=R2*H
      SX1=DSIN(R3)/R2-DCOS(R3)/R1
      SX2=1.D0/(X*(1.D0+R4))
      WSPE=SX2/SX1
      RETURN
      END
C
```

```
C
C
      FUNCTION WTPE(X)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
C
      X2=X*X
      AK02=AK0*AK0
      AK2=AK*AK
      R1=DSQRT(AK2-X2)
      R2=DSQRT(X2-AK02)
      R3=R1*H
      R4=R2*H
      S3=DSIN(R3)
      C3=DCOS(R3)
      SX1=1.D0/(X*(1.D0+R4))
      SX2=S3/R2-C3/R1
      SX3=ER*R2*C3-R1*S3
      WTPE=SX1/(SX2*SX3)
      RETURN
      END
C
C
C
C
      FUNCTION WSPM(X)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
C
      X2=X*X
      AK02=AK0*AK0
      AK2=AK*AK
      R1=DSQRT(AK2-X2)
      R2=DSQRT(X2-AK02)
      R3=R1*H
      R4=R2*H
      S3=DSIN(R3)
      C3=DCOS(R3)
      SX1=R2*S3+R1*C3
      SX2=((ER+R4)*C3/R2+(1.D0+ER*R4)*S3/R1)*X
      WSPM=1.D0/(SX1*SX2)
      RETURN
      END
```

```
C***********************************************************************
C                                 ARIS
C***********************************************************************
      SUBROUTINE ARIS
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION AMK(4),LMK(4)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,AA,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER
     *,AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,
     *IFEED
      COMMON/ADON/DIST(10,150,10),SERS(5),SERA(5),DARG(10,10,4),S(10,2),
     *WREAL,NSER,MNUM
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
      COMMON/DATT/COAL(20),POINT(20),CN(51),BM(51),POLTM(20),POLTE(20)
     *,AM(41),DM(41),POLES(40),NPOINT,NK0,MA,NTM,NTE,NK0K,IFIRST
C
C     +-----------------------------------------+
C     | FORMATION OF VECTORS MATRICES :  DIST,  |
C     |     FUN2,FUN4,FUN6,FUN8                 |
C     |                                         |
C     +-----------------------------------------+
      W2=W/2.D0
      W4=W2*W2
      NMAX=ND+NS+2
      MNUM=2
      DELTA=PI/(2.D0*FLOAT(MNUM))
      DO 10 M=1,MNUM
            FM1=FLOAT(M-1)*DELTA
            FM2=FM1+DELTA
            X=0.5D0*(FM2-FM1)
            Y=0.5D0*(FM2+FM1)
            DO 11 J=1,NPOINT
                  FI=X*POINT(J)+Y
                  AS=DSIN(FI)
                  AC=DCOS(FI)
                  DARG(M,J,1)=W2*AC
                  DARG(M,J,2)=AC
                  DARG(M,J,3)=AS
                  DARG(M,J,4)=X
                  DO 1 K=1,NMAX
                       AXN=FLOAT(K-2)*DLX
                       DIST(M,K,J)=AXN*AS
    1             CONTINUE
   11       CONTINUE
   10 CONTINUE
C
C         +----------------------------------+
C         |  FORMATION OF THE SERIES   S     |
C         +----------------------------------+
C
      U=WREAL/W
      U=DATAN(DSQRT(1.D0/(U*U)-1.D0))
      U1=2.D0*U/FLOAT(NSER)
      DO 2 JN=1,NSER
           S2=(2.D0*FLOAT(JN)-1.D0)
```

```
         S2=S2/(2.D0*FLOAT(NSER))
         S3=DCOS(S2*U)
         S(JN,2)=S3*W/2.D0
         S(JN,1)=U1
  2   CONTINUE
C
C           +-------------------------------------------+
C           | FORMATION OF THE SERIES S(DLX); STORAGE IN |
C           |      VECTORS SERS(5),SERA(5)              |
C           +-------------------------------------------+
      ADL=AKK*DLX
      ADL2=ADL*ADL
      ADL3=ADL2*ADL
      ADL4=ADL3*ADL
      ADL5=ADL4*ADL
      ADL6=ADL5*ADL
      YSIN=DSIN(ADL)
      YCOS=DCOS(ADL)
C
      SER1=(1.D0-YCOS)*2.D0/AKK
      SER2=-YSIN/3.D0+ADL*YCOS/4.D0+ADL2*YSIN/10.D0-ADL3*YCOS/36.D0-ADL4
     *     *YSIN/168.D0+ADL5*YCOS/960.D0+ADL6*YSIN/6480.D0
      SER3=YSIN/60.D0-ADL*5.D0*YCOS/360.D0-ADL2*YSIN/168.D0+ADL3*YCOS/56
     *     0.D0+ADL4*YSIN/2592.D0-ADL5*YCOS/12960.D0-ADL6*YSIN/95040.D0
      SER4=-YSIN/2520.D0+ADL*YCOS/2880.D0+ADL2*YSIN/6480.D0-ADL3*YCOS/21
     *     600.D0-ADL4*YSIN/95040.D0+ADL5*YCOS/518400.D0
      SER5=YSIN/181440.D0-ADL*YCOS/201600.D0-ADL2*YSIN/443520.D0+ADL3*CC
     *     OS/1442775.9D0
C
      SERS(1)=SER1*SER1
      SERS(2)=DLX*2.D0*SER1*SER2
      SERS(3)=DLX*(DLX*SER2*SER2+2.D0*SER1*SER3)
      SERS(4)=DLX*(2.D0*SER1*SER4+2.D0*DLX*SER2*SER3)
      SERS(5)=DLX*(DLX*SER3*SER3+2.D0*DLX*SER2*SER4)
C
      SERA(1)=SER1
      SERA(2)=DLX*SER2
      SERA(3)=DLX*SER3
      SERA(4)=DLX*SER4
      SERA(5)=DLX*SER5
      RETURN
      END
C*********************************************************************
C                          ADONIS
C*********************************************************************
      SUBROUTINE ADONIS(NMAX)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION BJ(10,10),DERIV(9,3)
      COMMON/ADON/DIST(10,150,10),SERS(5),SERA(5),DARG(10,10,4),S(10,2),
     *WREAL,NSER,MNUM
      COMMON/PUT/SSJ0(150),SAJ0(150),YSIN,YCOS
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/MAT/PLI,IWRITE
      COMMON/BSS/ARG(10,10)
```

```
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
      COMMON/DATT/COAL(20),POINT(20),CN(51),BM(51),POLTM(20),POLTE(20)
     *,AM(41),DM(41),POLES(40),NPOINT,NK0,MA,NTM,NTE,NK0K,IFIRST
C
      W1=2.D0*YCOS
      PR1=PLI*DLX
      PR2=PR1*PR1
      PR4=PR2*PR2
      PR6=PR4*PR2
      PR8=PR6*PR2
      DO 11 M=1,MNUM
            DO 10 J=1,NPOINT
                  ARG(M,J)=PLI*DARG(M,J,1)
 10         CONTINUE
 11   CONTINUE
            DO 1 K=1,NMAX
                 SSJ0(K)=0.D0
                 SAJ0(K)=0.D0
  1         CONTINUE
      CALL BESS(BJ)
      SUMD=0.D0
      DO 23 M=1,MNUM
            DO 20 J=1,NPOINT
                  ASIN=W*DARG(M,J,4)*COAL(J)
                  AROF=PLI*OFFSET*DARG(M,J,2)
                  COFF=DCOS(AROF)
                  DO 21 NK=1,5
                        DERIV(NK,1)=0.D0
                        DERIV(NK,2)=0.D0
 21               CONTINUE
                  SSUM=0.D0
                  DO 4 JN=1,NSER
                       ARAF=PLI*S(JN,2)*DARG(M,J,2)
                       CAFF=DCOS(ARAF)
                       SSUM=SSUM+S(JN,1)*CAFF
 4                CONTINUE
                  DO 3 K=1,NMAX
                       SIN11=DARG(M,J,3)*DARG(M,J,3)
                       DO 22 NK=1,5
                             DERIV(NK,1)=DERIV(NK,2)
                             DERIV(NK,2)=DERIV(NK,3)
 22                    CONTINUE
                       COS1=DCOS(PLI*DIST(M,K,J))
                       TERM=COFF*(BJ(M,J)-SSUM/PI)*COS1
                       DERIV(1,3)=TERM
                       SIN1=SIN11
                       DERIV(2,3)=-TERM*SIN1
                       SIN1=SIN1*SIN11
                       DERIV(3,3)=TERM*SIN1
                       SIN1=SIN1*SIN11
                       DERIV(4,3)=-TERM*SIN1
                       SIN1=SIN1*SIN11
                       DERIV(5,3)=TERM*SIN1
C
```

```
                     IF (K.LT.3) GO TO 3
                     SUMS=SERS(1)*DERIV(1,2)-PR2*SERS(2)*DERIV(2,2)+PR
     *                    4*SERS(3)*DERIV(3,2)-PR6*SERS(4)*DERIV(4,2)+
     *                    PR8*SERS(5)*DERIV(5,2)
C
                     CH1=SERA(1)*(DERIV(1,1)+DERIV(1,3)-W1*DERIV(1,2))
                     CH2=SERA(2)*(DERIV(2,1)+DERIV(2,3)-W1*DERIV(2,2))
     *                   *PR2
                     CH3=SERA(3)*(DERIV(3,1)+DERIV(3,3)-W1*DERIV(3,2))
     *                   *PR4
                     CH4=SERA(4)*(DERIV(4,1)+DERIV(4,3)-W1*DERIV(4,2))
     *                   *PR6
                     CH5=SERA(5)*(DERIV(5,1)+DERIV(5,3)-W1*DERIV(5,2))
     *                   *PR8
                     SUMA=CH1-CH2+CH3-CH4+CH5
                     KJ=K-2
                     SSJ0(KJ)=SSJ0(KJ)+ASIN*SUMS
                     SAJ0(KJ)=SAJ0(KJ)+ASIN*SUMA
    3            CONTINUE
   20       CONTINUE
   23 CONTINUE
      RETURN
      END
C***********************************************************************
C                                BESS
C***********************************************************************
      SUBROUTINE BESS(BJ)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION BJ(10,10)
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
      COMMON/ADON/DIST(10,150,10),SERS(5),SERA(5),DARG(10,10,4),S(10,2),
     *WREAL,NSER,MNUM
      COMMON/BSS/ARG(10,10)
      COMMON/DATT/COAL(20),POINT(20),CN(51),BM(51),POLTM(20),POLTE(20)
     *,AM(41),DM(41),POLES(40),NPOINT,NK0,MA,NTM,NTE,NK0K,IFIRST
C
C
      PI=3.141592653589D0
      DO 20 M=1,MNUM
            DO 1 IJ=1,NPOINT
                X=ARG(M,IJ)
                IF (X.GT.0.001D0) GO TO 10
                X3=X/3.D0
                X32=X3*X3
                X34=X32*X32
                X36=X34*X32
                BJ0=1.D0-2.2499997D0*X32+1.2656208D0*X34-0.3163866D0*X36
                BJ(M,IJ)=BJ0
                GO TO 1
   10           IF (X.GT.3.D0) GO TO 12
                   X3=X/3.D0
                   X32=X3*X3
                   X34=X32*X32
                   X36=X34*X32
```

```
                X38=X36*X32
                X310=X38*X32
                X312=X310*X32
                BJ0=1.D0-2.2499997D0*X32+1.2656208D0*X34-0.3163866D0
     *              *X36+0.0444479D0*X38-0.0039444D0*X310+0.00021000
     *              D0*X312
                BJ(M,IJ)=BJ0
                GO TO 1
  12        CONTINUE
            X3=3.D0/X
            X32=X3*X3
            X33=X32*X3
            X34=X33*X3
            X35=X34*X3
            X36=X35*X3
            FJ0=0.79788456D0-0.00000077D0*X3-0.00552740D0*X32-0.0000
     *          9512D0*X33+0.00137237D0*X34-0.00072805D0*X35+0.00014
     *          476D0*X36
            TJ0=X-0.78539816D0-0.04166397D0*X3-0.00003954D0*X32+0.00
     *          262573D0*X33-0.00054125D0*X34-0.00029333D0*X35+0.000
     *          13558D0*X36
            WCON=DSQRT(1.D0/X)
            BJ(M,IJ)=WCON*FJ0*DCOS(TJ0)
   1    CONTINUE
  20  CONTINUE
      RETURN
      END
C*************************************************************************
C                         TAIL CONTRIBUTION
C*************************************************************************
      SUBROUTINE TAIL
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION Z(6),C(6),S1(4,150),D1(4,150),D2(4,150),
     *T1(3,150),T2(3,150),T3(3,150),T4(3,150)
C
      COMMON/OUT/G1GT(150),G2GT(160),G2LT(190)
      COMMON/DAT/ER,H,BS,TT1,TT2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/INT/XNS(40),CNS(40),XND(20,2),CND(20),XNT(40,3)
     *,CNT(40),MAX(3,6),IMAX(3),NDP,NTP,NSP
      COMMON/ADON/DIST(10,150,10),SERS(5),SERA(5),DARG(10,10,4),SN(10,2)
     *,WREAL,NSER,MNUM
C
C
C     This vector contains the values of t in the integrals h0
C
      Z(1)=TT1
      Z(2)=(1.D0+2.D0*E4)*TT1+BS
      IF (IFEED.EQ.1) Z(2)=TT1+BS*(1.D0-2.D0*E2)
      Z(3)=TT2
      Z(4)=2.D0*BS-TT2
      IF (IFEED.EQ.1) Z(4)=TT2-2.D0*BS*E2
      Z(5)=BS+TT2
      Z(6)=2.D0*BS+TT2
C
```

```
C     This vector contains the values of the coefficient C in
C          the integrals h0
C
      C(1)=FA0
      C(2)=FA
      C(3)=FA
      C(4)=FA
      C(5)=FA
      C(6)=FA
C
C     This vector contains the values of the coefficient A in
C             the integrals h0
C
      AKK=TPI
      MMAX=ND+NS+6
      IF ((ND+NS).LT.NF) MMAX=NF+6
C
C
      W2=W/2.D0
      THMIN=WREAL/W
      THMIN=DATAN(DSQRT(1.D0/THMIN**2-1.D0))
      THMAX=PI-THMIN
      IF (OFFSET.LT.1.D-6) THMAX=PI
      PI2=PI/2.D0
      PI4=PI/4.D0
      DLX2=DLX/2.D0
      DLX4=DLX2*DLX2
      DSP=(THMAX-THMIN)/4.D0
      DDP=DSP*DLX2
      DTP=DSP*DLX4
      COEF1=(THMAX-THMIN)/2.D0
      IF (OFFSET.LT.1.D-6) COEF1=(PI/2.D0-THMIN)/2.D0
      COEF2=(THMAX+THMIN)/2.D0
      IF (OFFSET.LT.1.D-6) COEF2=(PI/2.D0+THMIN)/2.D0
C
      YCOS=DCOS(AKK*DLX)
      CCS=DCOS(2.D0*AKK*DLX)
      YSIN=DSIN(AKK*DLX)
      SSN=DSIN(2.D0*AKK*DLX)
C
C     +---------------------------------+
C     | Evaluation of S1,S2,S3,S4,S5,S6 |
C     |      (Single Integrals)         |
C     +---------------------------------+
C
      ZP1=Z(1)*C(1)
      ZP2=Z(2)*C(2)
      ZP3=Z(3)*C(3)
      ZP4=Z(4)*C(4)
      ZP5=Z(5)*C(5)
      ZP6=Z(6)*C(6)
C
      ZP12=ZP1*ZP1
      ZP22=ZP2*ZP2
      ZP32=ZP3*ZP3
```

```
      ZP42=ZP4*ZP4
      ZP52=ZP5*ZP5
      ZP62=ZP6*ZP6
      NMAX=IMAX(1)
      DO 10 I=1,NSP
            THI=COEF1*XNS(I)+COEF2
            C1=DCOS(THI)
            C2=W2*C1
            C2=OFFSET-C2
            CW=C2*C2
            ASIN=CNS(I)*DSP
            DO 11 N=1,NMAX
                  XN=FLOAT(N-3)*DLX
                  RAD2=XN*XN+CW
                  TRAD1=DSQRT(RAD2+ZP12)
                  TRAD2=DSQRT(RAD2+ZP22)
                  TRAD3=DSQRT(RAD2+ZP32)
                  TRAD4=DSQRT(RAD2+ZP42)
                  IF (N.GT.MAX(1,1)) GO TO 12
                      S1(1,N)=S1(1,N)+DLOG(2.D0*(TRAD1+XN))*ASIN
   12             S1(2,N)=S1(2,N)+DLOG(2.D0*(TRAD2+XN))*ASIN
                  IF (N.GT.MAX(1,3)) GO TO 13
                      S1(3,N)=S1(3,N)+DLOG(2.D0*(TRAD3+XN))*ASIN
   13             S1(4,N)=S1(4,N)+DLOG(2.D0*(TRAD4+XN))*ASIN
   11       CONTINUE
   10 CONTINUE
C     +-----------------------------------------------------------+
C     | EVALUATION OF D1,D2,D4,D5                                 1
C     +-----------------------------------------------------------+
      NMAX=IMAX(2)
      DO 20 I=1,NDP
            THI=COEF1*XND(I,1)+COEF2
            XI=DLX2*(XND(I,2)+1.D0)
            C1=DCOS(THI)
            C2=W2*C1
            C2=OFFSET-C2
            CW=C2*C2
            ASIN=CND(I)*DDP
            SV1=DSIN(AKK*(DLX-XI))
            SV2=-SV1
            SV4=DSIN(AKK*XI)
            C2=DCOS(AKK*(DLX-XI))
            DO 21 N=1,NMAX
                  XNP=XI+FLOAT(N-2)*DLX
                  XNM=-XI+FLOAT(N-2)*DLX
                  RADP2=XNP*XNP+CW
                  RADM2=XNM*XNM+CW
                  TRAP1=DSQRT(RADP2+ZP12)
                  TRAP2=DSQRT(RADP2+ZP22)
                  TRAP3=DSQRT(RADP2+ZP32)
                  TRAP4=DSQRT(RADP2+ZP42)
C
                  TRAM1=DSQRT(RADM2+ZP12)
                  TRAM2=DSQRT(RADM2+ZP22)
                  TRAM3=DSQRT(RADM2+ZP32)
```

```
               TRAM4=DSQRT(RADM2+ZP42)
C
               XA1=AKK*XNP
               XA2=AKK*XNM
               XAP=DSIN(XA1)
               XAM=DSIN(XA2)
C
               SANP1=XAP*DLOG(2.D0*(TRAP1+XNP))
               SANP2=XAP*DLOG(2.D0*(TRAP2+XNP))
               SANP3=XAP*DLOG(2.D0*(TRAP3+XNP))
               SANP4=XAP*DLOG(2.D0*(TRAP4+XNP))
C
               SANM1=XAM*DLOG(2.D0*(TRAM1+XNM))
               SANM2=XAM*DLOG(2.D0*(TRAM2+XNM))
               SANM3=XAM*DLOG(2.D0*(TRAM3+XNM))
               SANM4=XAM*DLOG(2.D0*(TRAM4+XNM))
C
               XAP=DSIN(XA1/2.D0)
               XAM=DSIN(XA2/2.D0)
               SONP1=XAP/TRAP1
               SONP2=XAP/TRAP2
               SONP3=XAP/TRAP3
               SONP4=XAP/TRAP4
C
               SONM1=XAM/TRAM1
               SONM2=XAM/TRAM2
               SONM3=XAM/TRAM3
               SONM4=XAM/TRAM4
C
               Y1=-XNM/2.D0-DLX
               Y2=-XNP/2.D0+DLX
               CY1=DCOS(AKK*Y1)
               CY2=DCOS(AKK*Y2)
               SY1=DSIN(AKK*Y1)
               SY2=DSIN(AKK*Y2)
C
               IF (N.GT.MAX(2,1)) GO TO 22
                  D1(1,N)=D1(1,N)+(SANP1+SANM1)*SV2*ASIN
                  D2(1,N)=D2(1,N)+(CY1*SONP1-CY2*SONM1)*ASIN
   22          D1(2,N)=D1(2,N)+(SANP2+SANM2)*SV2*ASIN
               D2(2,N)=D2(2,N)+(CY1*SONP2-CY2*SONM2)*ASIN
   25          IF (N.GT.MAX(2,3)) GO TO 23
                  D2(3,N)=D2(3,N)+(CY1*SONP3-CY2*SONM3)*ASIN
   23          D1(4,N)=D1(4,N)+(SANP4+SANM4)*SV2*ASIN
               D2(4,N)=D2(4,N)+(CY1*SONP4-CY2*SONM4)*ASIN
   21       CONTINUE
   20    CONTINUE
C     +----------------------------------------+
C     | EVALUATION OF T1,T2,T3,T4              |
C     +----------------------------------------+
C
      NMAX=IMAX(3)
      DO 30 I=1,NTP
            THI=COEF1*XNT(I,1)+COEF2
            XI=DLX2*(XNT(I,2)+1.D0)
```

```
      XIP=DLX2*(XNT(I,3)+1.D0)
      C1=DCOS(THI)
      C2=W2*C1
      C2=OFFSET-C2
      CW=C2*C2
      SV1=DSIN(AKK*(DLX-XI))
      SV2=-SV1
      SV3=DSIN(AKK*(DLX-XIP))
      ASIN=DTP*CNT(I)
      DO 31 N=1,NMAX
            XNPP=(XI+XIP)+FLOAT(N-1)*DLX
            XNPM=(XI-XIP)+FLOAT(N-1)*DLX
            XNMP=(-XI+XIP)+FLOAT(N-1)*DLX
            XNMM=(-XI-XIP)+FLOAT(N-1)*DLX
            RADPP2=XNPP*XNPP+CW
            RADPM2=XNPM*XNPM+CW
            RADMP2=XNMP*XNMP+CW
            RADMM2=XNMM*XNMM+CW
            TAPP1=DSQRT(RADPP2+ZP12)
            TAPP2=DSQRT(RADPP2+ZP22)
            TAPP4=DSQRT(RADPP2+ZP42)
            TAPM1=DSQRT(RADPM2+ZP12)
            TAPM2=DSQRT(RADPM2+ZP22)
            TAPM4=DSQRT(RADPM2+ZP42)
            TAMP1=DSQRT(RADMP2+ZP12)
            TAMP2=DSQRT(RADMP2+ZP22)
            TAMP4=DSQRT(RADMP2+ZP42)
            TAMM1=DSQRT(RADMM2+ZP12)
            TAMM2=DSQRT(RADMM2+ZP22)
            TAMM4=DSQRT(RADMM2+ZP42)
            CST1=DCOS(AKK*(XNPM/2.D0+DLX))*DSIN(AKK*XNPP/2.D0)
            CST2=DCOS(AKK*(-XNMP/2.D0+DLX))*DSIN(AKK*XNMM/2.D0)
            CST3=DCOS(AKK*(XNMM/2.D0+DLX))*DSIN(AKK*XNMP/2.D0)
            CST4=DCOS(AKK*(-XNPP/2.D0+DLX))*DSIN(AKK*XNPM/2.D0)
            IF (N.GT.MAX(3,1)) GO TO 32
               T1(1,N)=T1(1,N)+SV2*ASIN*CST1/TAPP1
               T2(1,N)=T2(1,N)+SV1*ASIN*CST2/TAMM1
               T3(1,N)=T3(1,N)+SV1*ASIN*CST3/TAMP1
               T4(1,N)=T4(1,N)+SV2*ASIN*CST4/TAPM1
 32         T1(2,N)=T1(2,N)+SV2*ASIN*CST1/TAPP2
            T2(2,N)=T2(2,N)+SV1*ASIN*CST2/TAMM2
            T3(2,N)=T3(2,N)+SV1*ASIN*CST3/TAMP2
            T4(2,N)=T4(2,N)+SV2*ASIN*CST4/TAPM2
            IF (N.GT.MAX(3,4)) GO TO 31
               T1(3,N)=T1(3,N)+SV2*ASIN*CST1/TAPP4
               T2(3,N)=T2(3,N)+SV1*ASIN*CST2/TAMM4
               T3(3,N)=T3(3,N)+SV1*ASIN*CST3/TAMP4
               T4(3,N)=T4(3,N)+SV2*ASIN*CST4/TAPM4
 31      CONTINUE
 30   CONTINUE
      CONST1=FA0
      CONST2=FA
C
C     +----------------------+
C     | EVALUATION OF G1NI>  |
```

```
C     +----------------------+
C
      I=3
      IF (IFEED.EQ.1) I=4
      WCS=(TPI2/EER)*(1.D0/(1.D0-E2)-2.D0*EER/((1.D0+ER)*(1.D0-E3)))
      WCA=TPI*2.D0/((1.D0+ER)*(1.D0-E3))
      CS=WCS
      CA=WCA
      NMIN=1
      NMAX=ND
      DO 60 N=NMIN,NMAX
            NP1=N+2
            N0=N+1
            NM1=N
            ST1=-D1(1,NP1)+2.D0*YCOS*D1(1,N0)-D1(1,NM1)
            ST2=2.D0*(T1(1,N)+T2(1,N)-T3(1,N)-T4(1,N))
            ST=ST1+ST2
C
            MP2=N+4
            MP1=N+3
            M0=N+2
            MM1=N+1
            MM2=N
            SINP2=DSIN(AKK*FLOAT(N+1)*DLX)
            SINP1=DSIN(AKK*FLOAT(N)*DLX)
            SIN0=DSIN(AKK*FLOAT(N-1)*DLX)
            SINM1=DSIN(AKK*FLOAT(N-2)*DLX)
            SINM2=DSIN(AKK*FLOAT(N-3)*DLX)
            AT1=SINP2*S1(1,MP2)-4.D0*YCOS*SINP1*S1(1,MP1)+2.D0*(2.D0+CCS
     *         )*SIN0*S1(1,M0)-4.D0*YCOS*SINM1*S1(1,MM1)+SINM2*S1(1,MM2)
            AT2=-2.D0*(D2(1,NP1)-2.D0*YCOS*D2(1,N0)+D2(1,NM1))
            AT=AT1+AT2
            G1GT(N)=W*(CS*ST+CA*AT)*CONST1
   60 CONTINUE
C     +-----------------------+
C     |  EVALUATION OF G2NI>  |
C     +-----------------------+
C
      DS=1.D0/(1.D0+E4)-2.D0*EER/((1.D0+ER)*(1.D0+E6))
      DT=2.D0/((1.D0+ER)*(1.D0+E6))
      CS=TPI2*DS/EER
      IF (IFEED.EQ.1) CS=WCS
      CA=TPI*DT
      IF (IFEED.EQ.1) CA=WCA
      NMIN=1
      NMAX=NS+ND
      DO 61 N=NMIN,NMAX
            NP1=N+2
            N0=N+1
            NM1=N
            ST=-D1(2,NP1)+2.D0*YCOS*D1(2,N0)-D1(2,NM1)+2.D0*(T1(2,N)+T2(
     *        2,N)-T3(2,N)-T4(2,N))
            MP2=N+4
            MP1=N+3
            M0=N+2
```

```
            MM1=N+1
            MM2=N
            SINP2=DSIN(AKK*FLOAT(N+1)*DLX)
            SINP1=DSIN(AKK*FLOAT(N)*DLX)
            SIN0=DSIN(AKK*FLOAT(N-1)*DLX)
            SINM1=DSIN(AKK*FLOAT(N-2)*DLX)
            SINM2=DSIN(AKK*FLOAT(N-3)*DLX)
            AT1=SINP2*S1(2,MP2)-4.D0*YCOS*SINP1*S1(2,MP1)+2.D0*(2.D0+CCS
     *         )*SIN0*S1(2,M0)-4.D0*YCOS*SINM1*S1(2,MM1)+SINM2*S1(2,MM2)
            AT2=-2.D0*(D2(2,NP1)-2.D0*YCOS*D2(2,N0)+D2(2,NM1))
            AT=AT1+AT2
            G2GT(N)=W*(CS*ST+CA*AT)*CONST2
   61 CONTINUE
C
C      +----------------------+
C      |  EVALUATION OF G2NI<  |
C      +----------------------+
C
      DS1=ER*E4/(1.D0+E4)+EER*2.D0*ER/((1.D0+ER)*(1.D0+E6))-EER
      CS=-(TPI2/EER)*DS1
      IF (IFEED.EQ.1) CS=WCS
      CA=TPI
      IF (IFEED.EQ.1) CA=WCA
      CWW=1.D0-2.D0*ER/((1.D0+ER)*(1.D0+E6))
      IF (IFEED.EQ.1) CWW=0.D0
      CAA=TPI*CWW
      NMIN=1
      NMAX=NF
      DO 62 N=NMIN,NMAX
            NP1=N+2
            N0=N+1
            NM1=N
            ST=-D1(4,NP1)+2.D0*YCOS*D1(4,N0)-D1(4,NM1)+2.D0*(T1(3,N)+T2(
     *         3,N)-T3(3,N)-T4(3,N))
C
            MP2=N+4
            MP1=N+3
            M0=N+2
            MM1=N+1
            MM2=N
            SINP2=DSIN(AKK*FLOAT(N+1)*DLX)
            SINP1=DSIN(AKK*FLOAT(N)*DLX)
            SIN0=DSIN(AKK*FLOAT(N-1)*DLX)
            SINM1=DSIN(AKK*FLOAT(N-2)*DLX)
            SINM2=DSIN(AKK*FLOAT(N-3)*DLX)
            AT1=SINP2*S1(I,MP2)-4.D0*YCOS*SINP1*S1(I,MP1)+2.D0*(2.D0+CCS
     *         )*SIN0*S1(I,M0)-4.D0*YCOS*SINM1*S1(I,MM1)+SINM2*S1(I,MM2)
            AA1=SINP2*S1(4,MP2)-4.D0*YCOS*SINP1*S1(4,MP1)+2.D0*(2.D0+CCS
     *         )*SIN0*S1(4,M0)-4.D0*YCOS*SINM1*S1(4,MM1)+SINM2*S1(4,MM2)
            AT2=-2.D0*(D2(I,NP1)-2.D0*YCOS*D2(I,N0)+D2(I,NM1))
            AA2=-2.D0*(D2(4,NP1)-2.D0*YCOS*D2(4,N0)+D2(4,NM1))
            AT=AT1+AT2
            AA=AA1+AA2
            G2LT(N)=W*(CS*ST+CA*AT-CAA*AA)*CONST2
   62 CONTINUE
```

```
      RETURN
      END
C*********************************************************************
C     The name of this subroutine is       DATA
C     and gives all the data used by the main program and the other
C     subroutines.
C**********************************************************************
      SUBROUTINE DATA(IOPT)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION IAXX(10),DAXX(50)
      COMMON/COEF/RX(5),XX(5),RZ(5),XZ(5),FRX(5),FRZ(5),F1XX,F1ZX,F2XXG,
     *F2ZXG,F2XXL,F2ZXL,F1XXL,F1ZXL,F2XL,F2ZL
      COMMON/DATT/COAL(20),POINT(20),CN(51),BM(51),POLTM(20),POLTE(20)
     *,AM(41),DM(41),POLES(40),NPOINT,NK0,MA,NTM,NTE,NK0K,IFIRST
      COMMON/DAT/ER,H,BS,T1,T2,DLX,A,TPI,TPI2,PI,W,E1,E2,E3,E4,E5,EER,
     *AK0,AK,AKK,FA,FA0,BW,BWW,BWWW,B4W,B5W,OFFSET,WDELTA,NS,NF,ND,IFEED
      COMMON/INT/XNS(40),CNS(40),XND(20,2),CND(20),XNT(40,3)
     *,CNT(40),MAX(3,6),JMAX(3),NDP,NTP,NSP
      COMMON/ADON/DIST(10,150,10),SERS(5),SERA(5),DARG(10,10,4),S(10,2),
     *WREAL,NSER,MNUM
C     +-------------------------------------------+
C     |   VARIABLES FOR THE GEOMETRY.              |
C     |                                            |
C     |   IFEED=  1 : FEED LINE ON THE INTERFACE   |
C     |           0: FEED LINE IN THE DIELECTRIC   |
C     +-------------------------------------------+
      READ(5,80) IIA
   80 FORMAT(5X,I4)
      READ(5,80) IDA
      DO 66 I=1,IIA
           READ(5,67) IAXX(I)
   67      FORMAT (5X,I4)
   66 CONTINUE
      DO 68 I=1,IDA
            READ(5,69) DAXX(I)
   69       FORMAT (5X,D16.9)
   68 CONTINUE
      IOPT=IAXX(1)
      IFEED=IAXX(2)
      NS=IAXX(3)
      ND=IAXX(4)
      NF=IAXX(5)
      NTE=IAXX(6)
      NTM=IAXX(7)
      IFIRST=IAXX(8)
      ER=DAXX(1)
      EER=DAXX(2)
      IF (IFEED.EQ.1) EER=1.D0
      H=DAXX(3)*DSQRT(EER)
      BS=DAXX(4)*DSQRT(EER)
      DEL=DAXX(5)*DSQRT(EER)
      T1=DAXX(6)*DSQRT(EER)
      T2=DAXX(7)*DSQRT(EER)
      DLX=DAXX(8)
      A=DAXX(9)
```

```
      PI=DAXX(10)
      W=DAXX(11)*DSQRT(EER)
      OFFSET=DAXX(12)*DSQRT(EER)
      WDELTA=DAXX(13)*DSQRT(EER)
      IF (NTE.EQ.0) GO TO 71
          NPTE=13+NTE
          DO 72 NE=14,NPTE
                INE=NE-13
                POLTE(INE)=DAXX(NE)/DSQRT(EER)
C               WRITE (6,101) INE,POLTE(INE)
  101           FORMAT(10X,'INE=',I4,5X,'POLTE=',D16.9/)
  72      CONTINUE
  71  NRTM=13+NTE+1
      NPTM=13+NTE+NTM
      DO 73 NM=NRTM,NPTM
            INM=NM-NRTM+1
            POLTM(INM)=DAXX(NM)/DSQRT(EER)
C           WRITE (6,102) INM,POLTM(INM)
  102       FORMAT (10X,'INM=',I4,5X,'POLTM=',D16.9/)
  73  CONTINUE
      NMN=13+NTE+NTM+1
      NMX=13+2*(NTM+NTE)
      DO 74 NP=NMN,NMX
            NII=NP-NMN+1
            POLES(NII)=DAXX(NP)/DSQRT(EER)
C           WRITE (6,103) NII,POLES(NII)
  103       FORMAT (10X,'NII=',I4,5X,'POLES=',D16.9/)
  74  CONTINUE
C.........................................................................
      IF (IOPT.NE.1) GO TO 91
      WRITE (6,90) ER,H,BS,T1,DLX,W,PI,WDELTA
  90  FORMAT (10X,E14.7/10X,E14.7/10X,E14.7/10X,E14.7/10X,E14.7/10X,
     *E14.7/10X,E14.7/10X,E14.7/)
  91  CONTINUE
C........................................................................
      TPI=2.D0*PI
      TPI2=TPI*TPI
C     +------------------+
C     | ERROR FUNCTIONS  |
C     +------------------+
C
      AA1=A/TPI
      AA2=AA1*AA1*EER
      E1=0.5D0/(AA2-1.D0)
      E2=0.25D0*(ER-1.D0)/(AA2-1.D0)
      E3=2.D0*E2/(1.D0+ER)
      E4=0.25D0*(ER-1.D0)/(AA2-ER)
      E5=0.5D0*ER/(AA2-ER)
      E6=2.D0*ER*E4/(1.D0+ER)
C
      AK0=2.D0*PI/DSQRT(EER)
      AKK=2.D0*PI
      AK=AK0*DSQRT(ER)
      FA=DSQRT(1.D0+ER/(AA2-ER))
      FA0=DSQRT(1.D0+1.D0/(AA2-1.D0))
```

```
      FA=1.D0/FA
      FA0=1.D0/FA0
      IF (IFEED.EQ.1) FA=FA0
      BW=BS+T1*(1.D0+2.D0*E4)
      IF (IFEED.EQ.1) BW=T1+BS*(1.D0-2.D0*E2)
      BWW=T2-2.D0*BS*E2
      BWWW=2.D0*BS-T2
      B4W=BS+TT2
      B5W=2.D0*BS+TT2
C     +------------------------------------------------------+
C     |           DATA FOR THE POLES                          |
C     |  IFIRST= 0 : DOMINANT MODE IS TM WAVE (MANY POLES)   |
C     |          1 : DOMINANT MODE IS TE WAVE (MANY POLES)   |
C     |          2 : ONLY ONE TM SURFACE WAVE                 |
C     +------------------------------------------------------+
C    +-------------------------------------------------------+
C    |              Data for the Dipoles                      |
C    |                                                        |
C    |     NS      = Distance between the first points of    |
C    |               the two dipoles in dlx                   |
C    |     ND      = Length of the upper dipole in dlx        |
C    |     NF      = Length of the lower dipole               |
C    |               (feeding line) in dlx                    |
C    +-------------------------------------------------------+
C
C     VECTOR OF THE MAXIMA
C    ----------------------
C
      MAX(1,1)=ND+4
      MAX(1,2)=ND+NS+4
      MAX(1,3)=NF+4
      IF (IFEED.EQ.1) MAX(1,3)=0
      MAX(1,4)=NF+4
      MAX(1,5)=ND+NS+4
      MAX(1,6)=NF+4
      JMAX(1)=MAX(1,2)
C
      MAX(2,1)=ND+2
      MAX(2,2)=ND+NS+2
      MAX(2,3)=NF+2
      IF (IFEED.EQ.1) MAX(2,3)=0
      MAX(2,4)=NF+2
      MAX(2,5)=ND+NS+2
      MAX(2,6)=NF+2
      JMAX(2)=MAX(2,2)
C
      MAX(3,1)=ND
      MAX(3,2)=ND+NS+1
      MAX(3,4)=NF+1
      JMAX(3)=MAX(3,2)
C
C    +---------------------------+
C    | Data for the Integration |
C    +---------------------------+
      NK0=20
```

```
      NKOK=1
      MA=20
      NPOINT=10
      NSER=10
C
C     VECTOR COAL
C-----------
      COAL(1)=0.0666713443D0
      COAL(2)=0.14945134915D0
      COAL(3)=0.21908636251D0
      COAL(4)=0.26926671931D0
      COAL(5)=0.29552422471D0
      COAL(6)=COAL(5)
      COAL(7)=COAL(4)
      COAL(8)=COAL(3)
      COAL(9)=COAL(2)
      COAL(10)=COAL(1)
C
C     VECTOR POINT
C     ------------
      POINT(1)=0.973906528517D0
      POINT(2)=0.865063366688D0
      POINT(3)=0.679409568299D0
      POINT(4)=0.433395394129D0
      POINT(5)=0.148874338981D0
      POINT(6)=-POINT(5)
      POINT(7)=-POINT(4)
      POINT(8)=-POINT(3)
      POINT(9)=-POINT(2)
      POINT(10)=-POINT(1)
C
C     SINGLE INTEGRATION
C     ------------------
C
      NSP=31
      RS1=0.99708748181D0
      RS2=0.98468590966D0
      RS3=0.96250392509D0
      RS4=0.93075699789D0
      RS5=0.88976002994D0
      RS6=0.83992032014D0
      RS7=0.78173314841D0
      RS8=0.71577678458D0
      RS9=0.64270672292D0
      RS10=0.56324916140D0
      RS11=0.47819378204D0
      RS12=0.38838590160D0
      RS13=0.29471806998D0
      RS14=0.19812119933D0
      RS15=0.09955531215D0
      RS16=0.D0
C
      XNS(1)=RS1
      XNS(2)=RS2
      XNS(3)=RS3
```

```
      XNS(4)=RS4
      XNS(5)=RS5
      XNS(6)=RS6
      XNS(7)=RS7
      XNS(8)=RS8
      XNS(9)=RS9
      XNS(10)=RS10
      XNS(11)=RS11
      XNS(12)=RS12
      XNS(13)=RS13
      XNS(14)=RS14
      XNS(15)=RS15
      XNS(16)=RS16
      XNS(17)=-RS15
      XNS(18)=-RS14
      XNS(19)=-RS13
      XNS(20)=-RS12
      XNS(21)=-RS11
      XNS(22)=-RS10
      XNS(23)=-RS9
      XNS(24)=-RS8
      XNS(25)=-RS7
      XNS(26)=-RS6
      XNS(27)=-RS5
      XNS(28)=-RS4
      XNS(29)=-RS3
      XNS(30)=-RS2
      XNS(31)=-RS1
C
      CNS(1)=0.0074708315792D0
      CNS(2)=0.0173186207903D0
      CNS(3)=0.0270090191849D0
      CNS(4)=0.0364322739123D0
      CNS(5)=0.0454937075272D0
      CNS(6)=0.0541030824249D0
      CNS(7)=0.0621747865610D0
      CNS(8)=0.0696285832354D0
      CNS(9)=0.0763903865987D0
      CNS(10)=0.0823929917615D0
      CNS(11)=0.0875767406084D0
      CNS(12)=0.0918901138936D0
      CNS(13)=0.0952902429123D0
      CNS(14)=0.0977433353863D0
      CNS(15)=0.0992250112266D0
      CNS(16)=0.0997205447934D0
      CNS(17)=CNS(15)
      CNS(18)=CNS(14)
      CNS(19)=CNS(13)
      CNS(20)=CNS(12)
      CNS(21)=CNS(11)
      CNS(22)=CNS(10)
      CNS(23)=CNS(9)
      CNS(24)=CNS(8)
      CNS(25)=CNS(7)
      CNS(26)=CNS(6)
```

```
      CNS(27)=CNS(5)
      CNS(28)=CNS(4)
      CNS(29)=CNS(3)
      CNS(30)=CNS(2)
      CNS(31)=CNS(1)
C
C
C
C     2)   Double Integration
C          ------------------
C
      NDP=16
      R1=DSQRT((15.D0-2.D0*DSQRT(30.D0))/35.D0)
      R2=-R1
      S1=DSQRT((15.D0+2.D0*DSQRT(30.D0))/35.D0)
      S2=-S1
      A1=4.D0*(59.D0+6.D0*DSQRT(30.D0))/864.D0
      A2=4.D0*(59.D0-6.D0*DSQRT(30.D0))/864.D0
      A3=4.D0*49.D0/864.D0
C
      XND(1,1)=R1
      XND(1,2)=R1
      CND(1)=A1
C
      XND(2,1)=R2
      XND(2,2)=R1
      CND(2)=A1
C
      XND(3,1)=R1
      XND(3,2)=R2
      CND(3)=A1
C
      XND(4,1)=R2
      XND(4,2)=R2
      CND(4)=A1
C
      XND(5,1)=S1
      XND(5,2)=S1
      CND(5)=A2
C
      XND(6,1)=S1
      XND(6,2)=S2
      CND(6)=A2
C
      XND(7,1)=S2
      XND(7,2)=S1
      CND(7)=A2
C
      XND(8,1)=S2
      XND(8,2)=S2
      CND(8)=A2
C
      XND(9,1)=R1
      XND(9,2)=S1
      CND(9)=A3
```

```
C
      XND(10,1)=R1
      XND(10,2)=S2
      CND(10)=A3
C
      XND(11,1)=S1
      XND(11,2)=R1
      CND(11)=A3
C
      XND(12,1)=S2
      XND(12,2)=R1
      CND(12)=A3
C
      XND(13,1)=R2
      XND(13,2)=S1
      CND(13)=A3
C
      XND(14,1)=R2
      XND(14,2)=S2
      CND(14)=A3
C
      XND(15,1)=S1
      XND(15,2)=R2
      CND(15)=A3
C
      XND(16,1)=S2
      XND(16,2)=R2
      CND(16)=A3
C
C     3)  Triple Integration
C         ------------------
C
      NTP=34
      RS1=0.9317380000D0
      RS2=-RS1
      UU1=0.9167441779D0
      UU2=-UU1
      SS1=0.4086003800D0
      SS2=-SS1
      TT1=0.7398529500D0
      TT2=-TT1
      B1=8.D0*0.03558180896D0
      B2=8.D0*0.01247892770D0
      B3=8.D0*0.05286772991D0
      B4=8.D0*0.02672752182D0
C
      XNT(1,1)=RS1
      XNT(1,2)=0.D0
      XNT(1,3)=0.D0
      CNT(1)=B1
C
      XNT(2,1)=RS2
      XNT(2,2)=0.D0
      XNT(2,3)=0.D0
      CNT(2)=B1
```

```
C
      XNT(3,1)=0.D0
      XNT(3,2)=RS1
      XNT(3,3)=0.D0
      CNT(3)=B1
C
      XNT(4,1)=0.D0
      XNT(4,2)=RS2
      XNT(4,3)=0.D0
      CNT(4)=B1
C
      XNT(5,1)=0.D0
      XNT(5,2)=0.D0
      XNT(5,3)=RS1
      CNT(5)=B1
C
      XNT(6,1)=0.D0
      XNT(6,2)=0.D0
      XNT(6,3)=RS2
      CNT(6)=B1
C
      XNT(7,1)=UU1
      XNT(7,2)=UU1
      XNT(7,3)=0.D0
      CNT(7)=B2
C
      XNT(8,1)=UU2
      XNT(8,2)=UU1
      XNT(8,3)=0.D0
      CNT(8)=B2
C
      XNT(9,1)=UU1
      XNT(9,2)=UU2
      XNT(9,3)=0.D0
      CNT(9)=B2
C
      XNT(10,1)=UU2
      XNT(10,2)=UU2
      XNT(10,3)=0.D0
      CNT(10)=B2
C
      XNT(11,1)=UU1
      XNT(11,2)=0.D0
      XNT(11,3)=UU1
      CNT(11)=B2
C
      XNT(12,1)=UU1
      XNT(12,2)=0.D0
      XNT(12,3)=UU2
      CNT(12)=B2
C
      XNT(13,1)=UU2
      XNT(13,2)=0.D0
      XNT(13,3)=UU1
      CNT(13)=B2
```

```
C
      XNT(14,1)=UU2
      XNT(14,2)=0.D0
      XNT(14,3)=UU2
      CNT(14)=B2
C
      XNT(15,1)=0.D0
      XNT(15,2)=UU1
      XNT(15,3)=UU1
      CNT(15)=B2
C
      XNT(16,1)=0.D0
      XNT(16,2)=UU1
      XNT(16,3)=UU2
      CNT(16)=B2
C
      XNT(17,1)=0.D0
      XNT(17,2)=UU2
      XNT(17,3)=UU1
      CNT(17)=B2
C
      XNT(18,1)=0.D0
      XNT(18,2)=UU2
      XNT(18,3)=UU2
      CNT(18)=B2
C
      XNT(19,1)=SS1
      XNT(19,2)=SS1
      XNT(19,3)=SS1
      CNT(19)=B3
C
      XNT(20,1)=SS1
      XNT(20,2)=SS1
      XNT(20,3)=SS2
      CNT(20)=B3
C
      XNT(21,1)=SS1
      XNT(21,2)=SS2
      XNT(21,3)=SS1
      CNT(21)=B3
C
      XNT(22,1)=SS1
      XNT(22,2)=SS2
      XNT(22,3)=SS2
      CNT(22)=B3
C
      XNT(23,1)=SS2
      XNT(23,2)=SS1
      XNT(23,3)=SS1
      CNT(23)=B3
C
      XNT(24,1)=SS2
      XNT(24,2)=SS1
      XNT(24,3)=SS2
      CNT(24)=B3
```

```
C
      XNT(25,1)=SS2
      XNT(25,2)=SS2
      XNT(25,3)=SS1
      CNT(25)=B3
C
      XNT(26,1)=SS2
      XNT(26,2)=SS2
      XNT(26,3)=SS2
      CNT(26)=B3
C
      XNT(27,1)=TT1
      XNT(27,2)=TT1
      XNT(27,3)=TT1
      CNT(27)=B4
C
      XNT(28,1)=TT1
      XNT(28,2)=TT1
      XNT(28,3)=TT2
      CNT(28)=B4
C
      XNT(29,1)=TT1
      XNT(29,2)=TT2
      XNT(29,3)=TT1
      CNT(29)=B4
C
      XNT(30,1)=TT1
      XNT(30,2)=TT2
      XNT(30,3)=TT2
      CNT(30)=B4
C
      XNT(31,1)=TT2
      XNT(31,2)=TT1
      XNT(31,3)=TT1
      CNT(31)=B4
C
      XNT(32,1)=TT2
      XNT(32,2)=TT1
      XNT(32,3)=TT2
      CNT(32)=B4
C
      XNT(33,1)=TT2
      XNT(33,2)=TT2
      XNT(33,3)=TT1
      CNT(33)=B4
C
      XNT(34,1)=TT2
      XNT(34,2)=TT2
      XNT(34,3)=TT2
      CNT(34)=B4
C
      RETURN
      END
```

FOPTION2

(INV33)

```
C**********************************************************************
C     The name of this file is ............. INV33..................
C     It solves the problem of three strip dipoles. Two of them parasi-
C     tic and the third excited. One on the interface and two in the
C     dielectric. The exciter is in the dielectric
C**********************************************************************
      IMPLICIT REAL*4 (A-H,O-Z)
      COMPLEX*8 CUR(170),ZIN
      COMPLEX*8 BMATR,Z11(200),Z22(200),Z33(200),Z21(200),Z32(200)
     *,Z31(200)
      DIMENSION IB(170),IA(170),IDATA(10),RDATA(20)
      COMMON/MAN/BMATR(170,170)
C......................................................................
C                 DATA
C......................................................................
      DO 180 ID=1,7
             READ  (1,100)  IDATA(ID)
  180 CONTINUE
      NOEL3=IDATA(1)
      NOEL2=IDATA(2)
      NOEL1=IDATA(3)
      NS2=IDATA(4)
      NS1=IDATA(5)
      NOR=IDATA(6)
      NFEED=IDATA(7)
C
  100 FORMAT(10X,I4)
      DO 190 ID=1,8
             READ(2,200)  RDATA(ID)
  190 CONTINUE
      ER=RDATA(1)
      H=RDATA(2)
      BS=RDATA(3)
      T1=RDATA(4)
      DLX=RDATA(5)
      W=RDATA(6)
      PI=RDATA(7)
      WDELTA=RDATA(8)
  200 FORMAT(10X,E14.7)
      READ(2,310) N11
  310 FORMAT(/11X,I4)
      WRITE  (6,330)
  330 FORMAT(/,'****************************************************'/)
      DO 500 I=1,N11
             READ(2,400)  Z11(I)
C            WRITE  (6,320)  I,Z11(I)
  320        FORMAT  (2X,'Z11(',I4,')=',E14.7,3X,E14.7)
  500 CONTINUE
      READ(2,310) N31
      DO 530 I=1,N31
             READ(2,400)  Z31(I)
C            WRITE  (6,323)  I,Z31(I)
  323        FORMAT(2X,'Z31(',I4,')=',E14.7,3X,E14.7)
  530 CONTINUE
      READ(2,310) N33
      DO 540 I=1,N33
             READ(2,400)  Z33(I)
C            WRITE  (6,324)  I,Z33(I)
  324        FORMAT(2X,'Z33(',I4,')=',E14.7,3X,E14.7)
  540 CONTINUE
```

```
      READ(2,310) N21
      DO 510 I=1,N21
              READ(2,400) Z21(I)
C             WRITE (6,321) I,Z21(I)
  321         FORMAT(2X,'Z21(',I4,')=',E14.7,3X,E14.7)
  510 CONTINUE
      READ(2,310) N22
      DO 520 I=1,N22
              READ(2,400) Z22(I)
C             WRITE (6,322) I,Z22(I)
  322         FORMAT(2X,'Z22(',I4,')=',E14.7,3X,E14.7)
  520 CONTINUE
      READ(2,310) N32
      DO 550 I=1,N32
              READ(2,400) Z32(I)
C             WRITE (6,325) I,Z32(I)
  325         FORMAT(2X,'Z32(',I4,')=',E14.7,3X,E14.7)
  550 CONTINUE
  400 FORMAT (11X,E14.7,4X,E14.7)
      H=H/SQRT(ER)
      BS=BS/SQRT(ER)
      DEL=DEL/SQRT(ER)
      W=W/SQRT(ER)
      T1=T1/SQRT(ER)
      T2=T2/SQRT(ER)
      OFFSET=OFFSET/SQRT(ER)
      WDELTA=WDELTA/SQRT(ER)
C     ................................................
C
      WRITE (6,1)
  1   FORMAT (//'1',10X,'A strip dipole at the interface EM coupled '//1
     *0X,'to another printed dipole in the dielectric which'//10X,'is ex
     *cited by a gap generator'///)
      WRITE (6,3) ER,H,BS,T1,T2,DLX,W,OFFSET,WDELTA,DEL
  3   FORMAT(/10X,'ER=',E14.7,5X,'H=',E14.7/10X,'BS=',E14.7,5X,'T1=',E14
     *.7/10X,'T2=',E14.7,5X,'DLX=',E14.7/10X,'W=',E14.7/10X,'OFFSET=',E1
     *4.7,5X,'WDELTA=',E14.7/10X,'DELTA=',E14.7///)
C
C     Diagonal Matrices
C
      NS3=NS2+NS1-1
      INI=NOEL3
      IMIN=1
      IMAX=INI
      DO 4 I=IMIN,IMAX
           IXN=0
           DO 5 KI=I,IMAX
                IXN=IXN+1
                BMATR(IXN,KI)=Z33(I)
                BMATR(KI,IXN)=BMATR(IXN,KI)
  5        CONTINUE
  4   CONTINUE
      IMIN=NOEL3+1
      IMAX=NOEL3+NOEL2
      DO 6 I=IMIN,IMAX
           IXN=INI
           DO 7 KI=I,IMAX
                IXN=IXN+1
                BMATR(IXN,KI)=Z22(I-INI)
                BMATR(KI,IXN)=BMATR(IXN,KI)
```

```
    7       CONTINUE
    6   CONTINUE
        INI=NOEL3+NOEL2
        IMIN=INI+1
        IMAX=NOEL3+NOEL2+NOEL1
        DO 8 I=IMIN,IMAX
             IXN=INI
             DO 9 KI=I,IMAX
                  IXN=IXN+1
                  BMATR(IXN,KI)=Z11(I-INI)
                  BMATR(KI,IXN)=BMATR(IXN,KI)
    9       CONTINUE
    8   CONTINUE
C
C       ...1...  First off-diagonal matrix
C
C          1)   Upper Part
C
        IMIN=NOEL3+1
        IMAX=NOEL3+NOEL2
        DO 10 I=IMIN,IMAX
              IXN=0
              LXN=NS2+I-IMIN
              DO 11 KI=I,IMAX
                    IXN=IXN+1
                    BMATR(IXN,KI)=Z32(LXN)
                    BMATR(KI,IXN)=BMATR(IXN,KI)
   11         CONTINUE
   10   CONTINUE
C
C          2)  Lower Part
C
        IMI=NOEL3-NOEL2+2
        KIMIN=IMIN
        IIMAX=IMAX
        IMIN=2
        IMAX=NOEL3
        DO 12 I=IMIN,IMAX
              IXN=I-1
              KIMAX=IIMAX
              IF (I.GE.IMI) KIMAX=IIMAX-(I-IMI+1)
              LXN=IABS(NS2-I)+1
              DO 13 KI=KIMIN,KIMAX
                    IXN=IXN+1
                    BMATR(IXN,KI)=Z32(LXN)
                    BMATR(KI,IXN)=BMATR(IXN,KI)
   13         CONTINUE
   12   CONTINUE
C
C    ....2....   First off-diagonal matrix
C
C          1)  UPPER PART
C
        IMIN=NOEL3+NOEL2+1
        IMAX=NOEL3+NOEL2+NOEL1
        DO 14 I=IMIN,IMAX
              IXN=NOEL3
              LXN=NS1+I-IMIN
              DO 15 KI=I,IMAX
                    IXN=IXN+1
```

```
                 BMATR(IXN,KI)=Z21(LXN)
                 BMATR(KI,IXN)=BMATR(IXN,KI)
   15       CONTINUE
   14  CONTINUE
C
C         2)    Lower Part
C
       IMI=NOEL2-NOEL1+2+NOEL3
       KIMIN=IMIN
       IIMAX=IMAX
       IMIN=NOEL3+2
       IMAX=NOEL3+NOEL2
       DO 16 I=IMIN,IMAX
            IXN=I-1
            KIMAX=IIMAX
            IF (I.GE.IMI) KIMAX=IIMAX-(I-IMI+1)
            LXN=IABS(NS1-I+NOEL3)+1
            DO 17 KI=KIMIN,KIMAX
                 IXN=IXN+1
                 BMATR(IXN,KI)=Z21(LXN)
                 BMATR(KI,IXN)=BMATR(IXN,KI)
   17       CONTINUE
   16  CONTINUE
C
C      ....1.... Second off-diagonal matrix
C
C         1)    UPPER PART
       IMIN=NOEL3+NOEL2+1
       IMAX=NOEL3+NOEL2+NOEL1
       DO 18 I=IMIN,IMAX
            IXN=0
            LXN=NS3+I-IMIN
            DO 19 KI=I,IMAX
                 IXN=IXN+1
                 BMATR(IXN,KI)=Z31(LXN)
                 BMATR(KI,IXN)=BMATR(IXN,KI)
   19       CONTINUE
   18  CONTINUE
C
C          2)    Lower part
C
       IMI=NOEL3-NOEL1+2
       KIMIN=IMIN
       IIMAX=IMAX
       IMIN=2
       IMAX=NOEL3
       DO 20 I=IMIN,IMAX
            IXN=I-1
            KIMAX=IIMAX
            IF (I.GE.IMI) KIMAX=IIMAX-(I-IMI+1)
            LXN=IABS(NS3-I)+1
            DO 21 KI=KIMIN,KIMAX
                 IXN=IXN+1
                 BMATR(IXN,KI)=Z31(LXN)
                 BMATR(KI,IXN)=BMATR(IXN,KI)
   21       CONTINUE
   20  CONTINUE
C
C
C      IMIN=1
```

```
C     IMAX=NOEL3+NOEL2+NOEL1
C     DO 22 I=1,IMAX
C           WRITE (6,23) I,(BMATR(I,J),J=1,IMAX)
C 23        FORMAT (I2,2X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X
C    *             ,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X
C    *             ,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X
C    *             ,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X,I2,1X/)
C 22  CONTINUE
C     GO TO 1000
C
 1001 CALL MINVCD (NOR,NOR,DETA,IB,IA)
C
      DO 25 IQQ=1,NOR
            CUR(IQQ)=BMATR(IQQ,NFEED)/100.00
   25 CONTINUE
      ZIN=1.00/CUR(NFEED)
      WRITE (6,701) ZIN
 701  FORMAT (///10X,'ZIN=',(E14.7,2X,E14.7)//)
      WRITE (6,30)
  30  FORMAT (///10X,'Current distribution on the t.l.'///)
      WRITE (3,40) NOEL3
  40  FORMAT(10X,I4)
      IMIN=1
      IMAX=NOEL3+NOEL2+NOEL1
      DO 31 IQQ=IMIN,IMAX
            RECUR1=REAL(CUR(IQQ))
            ABCU1=CABS(CUR(IQQ))
            AICUR1=AIMAG(CUR(IQQ))
            PHCUR1=ATAN2(AICUR1,RECUR1)
            PHCUR1=180.00*PHCUR1/PI
            IF (IQQ.EQ.(NOEL3+1)) WRITE (6,36)
  36        FORMAT(///10X,'Current on the first parasitic dipole'////)
            IF (IQQ.EQ.(NOEL3+1)) WRITE (3,46) NOEL2
  46        FORMAT(10X,I4)
            IF (IQQ.EQ.(NOEL3+NOEL2+1)) WRITE (6,37)
  37        FORMAT(///10X,'Current on the second parasitic dipole'////)
            IF (IQQ.EQ.(NOEL3+NOEL2+1)) WRITE (3,47) NOEL1
  47        FORMAT(10X,I4)
            WRITE (6,32) IQQ,CUR(IQQ),ABCU1,PHCUR1
  32        FORMAT (7X,'CURR(',I4,')=(',(E14.7,',',E14.7),5X,E14.7,5X,
     *              E14.7/)
            WRITE (3,45) CUR(IQQ)
  45        FORMAT(10X,E14.7,1X,E14.7)
  31  CONTINUE
 1000 CONTINUE
      STOP
      END
C*********************************************************************
C     THIS SUBROUTINE INVERTS A SQUARE COMPLEX MATRIX
C*********************************************************************
      SUBROUTINE MINVCD (IA,MA,DETA,IR,IC)
      IMPLICIT REAL*4 (A-H,O-Z)
      COMPLEX*8  A,PIV,DETA,TEMP,PIV1
      DIMENSION IR(MA),IC(MA)
      COMMON/MAN/A(170,170)
      DO 1 I=1,MA
      IR(I)=0
    1 IC(I)=0
C     DETA=(1.00,0.00)
      S=0.00
```

```
      R=MA
    2 CALL SUBMCD(IA,IA,MA,MA,IR,IC,I,J)
      PIV=A(I,J)
C     DETA=PIV*DETA
      Y=CABS(PIV)
      IF (Y.EQ.0) GO TO 17
      IR(I)=J
      IC(J)=I
      PIV=(1.00,0.00)/PIV
      A(I,J)=PIV
      DO 5 K=1,MA
    5 IF (K.NE.J) A(I,K)=A(I,K)*PIV
      DO 9 K=1,MA
      IF (K.EQ.I) GO TO 9
      PIV1=A(K,J)
    6 DO 8 L=1,MA
    8 IF (L.NE.J) A(K,L)=A(K,L)-PIV1*A(I,L)
    9 CONTINUE
      DO 11 K=1,MA
   11 IF (K.NE.I) A(K,J)=-PIV*A(K,J)
      S=S+1.00
      IF (S.LT.R) GO TO 2
   12 DO 16 I=1,MA
      K=IC(I)
      M=IR(I)
      IF (K.EQ.I) GO TO 16
C     DETA=-DETA
      DO 14 L=1,MA
      TEMP=A(K,L)
      A(K,L)=A(I,L)
   14 A(I,L)=TEMP
      DO 15 L=1,MA
      TEMP=A(L,M)
      A(L,M)=A(L,I)
   15 A(L,I)=TEMP
      IC(M)=K
      IR(K)=M
   16 CONTINUE
      RETURN
   17 WRITE (6,18)I,J
   18 FORMAT (10X,'MATRIX IS SINGULAR'/10X,'I=',I4,5X,'J=',I4)
      RETURN
      END
C***********************************************************************
C.......................................................................
C***********************************************************************
      SUBROUTINE SUBMCD(IA,JA,MA,NA,IR,IC,I,J)
      IMPLICIT REAL*4 (A-H,O-Z)
      COMPLEX*8  A
      DIMENSION IR(MA),IC(NA)
      COMMON/MAN/A(170,170)
      I=0
      J=0
      TEST=0.00
      DO 5 K=1,MA
      IF (IR(K).NE.0) GO TO 5
      DO 4 L=1,NA
      IF (IC(L).NE.0) GO TO 4
      X=CABS(A(K,L))
      IF(X.LT.TEST) GO TO 4
```

```
      I=K
      J=L
      TEST=X
    4 CONTINUE
    5 CONTINUE
      RETURN
      END
```

FOPTION3

(GAIN)

```
C***********************************************************************
C     THE NAME OF THIS PROGRAM IS :     GAIN
C***********************************************************************
C     IT COMPUTES THE RADIATION PATTERN OF DIPOLES
C***********************************************************************
      IMPLICIT REAL*4 (A-H,O-Z)
      COMPLEX*8 ETH,EFI,CUR1,CUR2,CUR3,CUR4
      DIMENSION THETA(203),RGAIN(203),X(203),Y(203),XB(203),XC(203),
     *YB(203),YC(203),XPOS1(7),XPOS2(7),YPOS1(7),YPOS2(7),XA(203),
     *YA(203),XPOS3(3),XPOS4(4),YPOS3(3),YPOS4(4),ARG(201),CUR1(100),
     *CUR2(50),CUR3(50),CUR4(50),RTHETA(203)
      COMMON/DAT/PI,DLX,ER,EER,H,BS,CUR1,CUR2,S(20,2),WIDTH,
     *WREAL,NSER,ICON,CUR3,CUR4
      COMMON/PAR/OF2,OF3,OF4,NS,N1,N2,N3,N4,A2,A3,A4,DEL
C.......................................................................
C     SLOWER......Minimun dBs fo the radiation pattern you want to plot
C     DIV.........dBs per subdiv yow want to plot
C     TC1.........coef which determines the number of dB for the second
C                 subdiv circle
C     TC2.........coef which determines the number of dBs for the first
C                 subdiv circle
C     WARNING!!!!!after you change this parameters be carefull to change
C                 appropriately the corresponding symbol statements
C
C     ICON=   0   both transmission line and dipoles are included
C
C             1   only dipoles are included
C
C            -1   only transmission line is included
C
C     IPLANE= 0   E-plane radiation pattern
C
C             1   H-plane radiation pattern
C.......................................................................
      ICON=0
      SLOWER=-50.0
      DIV=10.0
      TC1=2.0
      TC2=1.0
C
      READ(1,400) IPLANE
      READ(1,400) NS
 400  FORMAT(5X,I4)
      READ(1,400) N1
      READ(1,400) N2
      READ(1,400) N3
      READ(1,400) N4
C
      PI=3.1415926535890
      READ(1,300) ER
  300 FORMAT(5X,E14.7)
      READ(1,300) EER
      READ(1,300) H
      READ(1,300) BS
      READ(1,300) DEL
      READ(1,300) DLX
      READ(1,300) WIDTH
      READ(1,300) WDELTA
C
      READ(1,300) OF2
```

```
      READ(1,300) OF3
      READ(1,300) OF4
C
      IFEED=0.25/DLX
      A2=FLOAT(NS)*DLX
      A3=A2
      A4=FLOAT(NS)*DLX
C
      WREAL=WIDTH
      WIDTH=WIDTH*(1.D0+2.D0*WDELTA/WIDTH)
C
      IF (IPLANE.EQ.1) GO TO 500
C
C     E-plane pattern
      FI=0.00
      IFI=0
      GO TO 501
C
  500 CONTINUE
C
C     H-plane pattern
      FI=PI/2.00
      IFI=1
C
  501 CONTINUE
C
C         Current distribution on the trans.line
C
      READ (3,100) NOEL1
  100 FORMAT(/I4)
      DO 110 I=1,NOEL1
            READ(3,120) CUR1(I)
  120       FORMAT(/E14.7,1X,E14.7)
  110 CONTINUE
C
C         Current distribuitons on the 1st parasitic dipole
C
      READ (3,100) NOEL2
      DO 140 I=1,NOEL2
            READ(3,120) CUR2(I)
  140 CONTINUE
C
C         Current distribuitons on the 2nd parasitic dipole
C
      IF (N3.EQ.0) GO TO 502
      READ (3,100) NOEL3
      DO 140 I=1,NOEL3
            READ(3,120) CUR3(I)
  140 CONTINUE
  502 CONTINUE
C
C        Current distribution on the printed dipole
C
      READ (3,100) NOEL4
      DO 160 I=1,NOEL4
            READ(3,120) CUR4(I)
  160 CONTINUE
C
      NSER=10
      U=(WREAL/WIDTH)
```

```
      U=ATAN(SQRT(1.0/(U*U)-1.0))
      U1=2.D0*U/FLOAT(NSER)
      DO 3 JN=1,NSER
           S2=2.D0*FLOAT(JN)-1.D0
           S2=S2/(2.D0*FLOAT(NSER))
           S3=COS(S2*U)
           S(JN,2)=S3*WIDTH/2.D0
           S(JN,1)=U1
    3 CONTINUE
      THMIN=-PI/2.0
      THMAX=PI/2.0
      MTHETA=201
      DELTH=(THMAX-THMIN)/FLOAT(MTHETA-1)
C
      DO 2 ITH=1,MTHETA
           THETA(ITH)=THMIN+FLOAT(ITH-1)*DELTH
           RTHETA(ITH)=180.00*THETA(ITH)/PI
           CALL EFIELD (THETA(ITH),FI,ETH,EFI)
           ACURR=CABS(CUR1(IFEED))
           IF (ICON.EQ.1) ACURR=CABS(CUR2(IFEED))
           ATH1=WREAL/WIDTH
           ATH1=ATAN(SQRT(1./(ATH1*ATH1)-1.0))
           ATH2=PI-ATH1
           CURIN=(1.D0/PI)*(ATH2-ATH1)
           ACURR=ACURR*CURIN
           ARG(ITH)=(960.0/EER)*(CABS(ETH)**2+CABS(EFI)**2)/ACURR
           ALARG=10.0*ALOG10(ARG(ITH))
           BLARG=180.0*THETA(ITH)/PI
           BLARG=90.0-ABS(BLARG)
           WRITE (6,4) BLARG,ARG(ITH),ALARG
    4      FORMAT(10X,E14.8,5X,E14.8,5X,E14.8)
    2 CONTINUE
C.....................................................................
C     THIS PART FINDS THE MAXIMUM ELEMENT OF THE MATRIX ARG
C.....................................................................
      KMAX=1
      R1=ARG(1)
      M1=I+1
      DO 6 K=2,MTHETA
      R2=ARG(K)
      IF (ABS(R2).LT.ABS(R1)) GO TO 6
      R1=R2
      KMAX=K
    6 CONTINUE
      ANORM=ARG(KMAX)
C
C.....................................................................
C     THIS PART FINDS THE MINIMUM ELEMENT OF THE MATRIX ARG
C.....................................................................
      LMAX=1
      R1=ARG(1)
      DO 21 K=2,MTHETA
      R2=ARG(K)
      IF (ABS(R2).GT.ABS(R1)) GO TO 21
      R1=R2
      LMAX=K
   21 CONTINUE
      BNORM=ARG(LMAX)/ANORM
      CNORM=10.0*ALOG10(BNORM)
      IF (ABS(CNORM).GE.ABS(SLOWER)) CNORM=SLOWER
```

```
C
      THETA1=THETA(KMAX)
      RMAX=10.D0*ALOG10(ANORM)
      DO 15 I=1,MTHETA
      ARGUM=ARG(I)/ANORM
      SPOWER=10.0*ALOG10(ARGUM)
      IF (ABS(SPOWER).GT.ABS(SLOWER)) SPOWER=SLOWER
      RGAIN(I)=SPOWER
      XZ=(SPOWER-CNORM)/ABS(CNORM)
      X(I)=XZ*SIN(THETA(I))
      Y(I)=XZ*COS(THETA(I))
   15 CONTINUE
      RTHET1=180.00*THETA1/PI
      WRITE (6,7) RTHET1,RMAX
    7 FORMAT (10X,'MAXIMUM GAIN OCCURS AT ANGLE:',1X,E14.8,1X,
     *//10X,'MAXIMUM GAIN IN DB:',1X,E14.8////10X,'ANGLE-THETA IN RAD',
     *10X,'NORMALIZED GAIN IN DB',10X,'X-MATRIX',10X,'Y-MATRIX'//)
C
C
      DO 8 I=1,201
      WRITE (6,9) RTHETA(I),RGAIN(I)
    9 FORMAT (10X,E14.8,10X,E14.8)
    8 CONTINUE
      FC=ABS(CNORM)/DIV
      FC1=TC1/FC
      FC2=TC2/FC
C.....................................................................
C     PLOTTING OF THE RADIATED POWER IN RECTANGULAR COORDINATES
C.....................................................................
C     CALL PLOTS (0,0,0)
C
C     CALL PLOT (0.,-2.,-13)
C     CALL PLOT (0.,3.,-13)
C     CALL PLOT (-2.,0.,-13)
C     CALL PLOT (3.,0.,-13)
C     CALL NEWPEN (1)
C     CALL SCALE (THETA,5.,201,1)
C     CALL SCALE (RGAIN,4.,201,1)
C     CALL AXIS(0.,0.,'ANGLE-THETA IN RADIANS',-22,5.,0.,THETA(202),THE
C    *TA(203))
C     CALL AXIS (0.,0.,'RADIATED POWER IN DB',20,4.,90.,RGAIN(202),RGAI
C    *N(203))
C     CALL GRID (0.,0.,10,0.5,8,0.5,3333)
C     CALL NEWPEN (5)
C     CALL LINE (THETA,RGAIN,201,1,0,5)
C     CALL NEWPEN (1)
C     IF (IFI.EQ.0) CALL SYMBOL (0.,4.,0.15,' E-PLANE PATTERN ',0.,17)
C     IF (IFI.EQ.1) CALL SYMBOL (0.,4.,0.15,' H-PLANE PATTERN ',0.,17)
C     CALL PLOT(0.,0.,+999)
C.....................................................................
C     PLOTTING OF RADIATED POWER IN POLAR COORDINATES
C.....................................................................
C     DO 11 I=1,201
C     XA(I)=SIN(THETA(I))
C     XB(I)=XA(I)*(1.0-FC2)
C     XC(I)=XA(I)*(1.0-FC1)
C     YA(I)=COS(THETA(I))
C     YB(I)=YA(I)*(1.0-FC2)
C     YC(I)=YA(I)*(1.0-FC1)
C 11  CONTINUE
```

```
C
C      CALL PLOTS(0,0,0)
C      CALL FACTOR (0.5)
C      CALL PLOT (0.,-2.,-13)
C      CALL PLOT (0.,4.,-13)
C      CALL PLOT (-2.,0.,-13)
C      CALL PLOT (4.,0.,-13)
C      X(202)=-1.0
C      X(203)=2.0/10.
C      Y(202)=Y(1)
C      Y(203)=1.0/5.0
C      XA(202)=X(202)
C      XA(203)=X(203)
C      XB(203)=X(203)
C      XC(202)=X(202)
C      XC(203)=X(203)
C      YA(202)=Y(202)
C      YA(203)=Y(203)
C      YB(202)=Y(202)
C      YB(203)=Y(203)
C      YC(202)=Y(202)
C      YC(203)=Y(203)
C
C      CALL NEWPEN (1)
C      CALL HLINE(0.,10.,0.,ZFFFF)
C      CALL VLINE (0.,5.,5.,ZFFFF)
C      CALL NEWPEN (5)
C      CALL LINE(X,Y,201,1,0,3)
C      CALL NEWPEN (1)
C      CALL LINE (XA,YA,201,1,0,3)
C      CALL LINE (XB,YB,201,1,0,3)
C      CALL LINE (XC,YC,201,1,0,3)
C
C      DO 12 I=1,7
C      ARGU=-PI/2.+PI*FLOAT(I-1)/6.
C      XPOS1(I)=5.5*SIN(ARGU)+5.
C      XPOS2(I)=7.*SIN(ARGU)+5.
C      YPOS1(I)=5.5*COS(ARGU)
C      YPOS2(I)=7.*COS(ARGU)
C   12 CONTINUE
C
C      DO 13 I=1,3
C      ARG1=-PI/2.+PI*FLOAT(I-1)/6.+PI/100.
C      ARG2=-ARG1
C      XPOS3(I)=6.75*SIN(ARG1)+5.
C      YPOS3(I)=6.75*COS(ARG1)
C      XPOS4(I)=5.75*SIN(ARG2)+5.
C 13   YPOS4(I)=5.75*COS(ARG2)
C      XPOS4(4)=4.8
C      YPOS4(4)=5.75
C
C      CALL PLOT (5.,0.,+13)
C      CALL PLOT (XPOS1(1),YPOS1(1),+13)
C      CALL PLOT (XPOS2(1),YPOS2(1),+12)
C      CALL SYMBOL (XPOS3(1),YPOS3(1),0.15,'THETA=-90',0.,9)
C      XPO=0.0
C      YPO=-0.25
C      CALL SYMBOL (XPO,YPO,0.15,'0DB',0.,3)
C      XPO=5.0*FC2
C      YPO=-0.25
```

```
C      CALL SYMBOL (XPO,YPO,0.15,'-10DB',0.,5)
C      XPO=5.0*FC1
C      CALL SYMBOL (XPO,YPO,0.15,'-20DB',0.,5)
C
C      CALL PLOT (5.,0.,+13)
C      CALL PLOT (XPOS1(2),YPOS1(2),+13)
C      CALL PLOT (XPOS2(2),YPOS2(2),+12)
C      CALL SYMBOL (XPOS3(2),YPOS3(2),0.15,'THETA=-60',330.,9)
C
C      CALL PLOT (5.,0.,+13)
C      CALL PLOT (XPOS1(3),YPOS1(3),+13)
C      CALL PLOT (XPOS2(3),YPOS2(3),+12)
C      CALL SYMBOL (XPOS3(3),YPOS3(3),0.15,'THETA=-30',300.,9)
C
C      CALL PLOT (5.,0.,+13)
C      CALL PLOT (XPOS1(4),YPOS1(4),+13)
C      CALL PLOT (XPOS2(4),YPOS2(4),+12)
C      CALL SYMBOL (XPOS4(4),YPOS4(4),0.15,'THETA=0',90.,7)
C
C      CALL PLOT (5.,0.,+13)
C      CALL PLOT (XPOS1(5),YPOS1(5),+13)
C      CALL PLOT (XPOS2(5),YPOS2(5),+12)
C      CALL SYMBOL (XPOS4(3),YPOS4(3),0.15,'THETA=30',60.,8)
C
C      CALL PLOT (5.,0.,+13)
C      CALL PLOT (XPOS1(6),YPOS1(6),+13)
C      CALL PLOT (XPOS2(6),YPOS2(6),+12)
C      CALL SYMBOL (XPOS4(2),YPOS4(2),0.15,'THETA=60',30.,8)
C
C      CALL PLOT (5.,0.,+13)
C      CALL PLOT (XPOS1(7),YPOS1(7),+13)
C      CALL PLOT (XPOS2(7),YPOS2(7),+12)
C      CALL SYMBOL (XPOS4(1),YPOS4(1),0.15,'THETA=90',0.,8)
C      XPO=10.
C      YPO=-0.25
C      CALL SYMBOL (XPO,YPO,0.15,'0DB',0.,3)
C      XPO=10.0-5.0*FC2
C      YPO=-0.25
C      CALL SYMBOL (XPO,YPO,0.15,'-10DB',0.,5)
C      XPO=10.0-5.0*FC1
C      CALL SYMBOL (XPO,YPO,0.15,'-20DB',0.,5)
C      IF (IFI.EQ.0) CALL SYMBOL (2.,7.8,0.15,' E-PLANE PATTERN ',0.,17)
C      IF (IFI.EQ.1) CALL SYMBOL (2.,7.8,0.15,' H-PLANE PATTERN ',0.,17)
C      CALL SYMBOL (4.,7.3,0.15,'ER=2.45 H=0.1127 BS=0.088 W=0.0500',0.,
C     *34)
C      CALL PLOTS (0.,0.,+999)
 2000 CONTINUE
      STOP
      END
C*********************************************************************
C      The name of this subroutine is:        EFIELD
C*********************************************************************
C      It evaluates the far field of a dipole
C*********************************************************************
      SUBROUTINE EFIELD(THETA,FI,ETH,EFI)
      IMPLICIT REAL*4(A-H,O-Z)
      COMPLEX*8 W,WN,ETH,EFI,WFL,WSL,WFD,WSD,WTL,WTD,WNL,WND,WWF,WWS,
     *W8,CUR1(100),CUR2(50),CUR3(50),CUR4(50),WND2,WND3,WND4
     *,WFP,WSP,WTP,WNP
      COMMON/DAT/PI,DLX,ER,EER,H,BS,CUR1,CUR2,S(20,2),WIDTH,
```

```
     *WREAL,NSER,ICON,CUR3,CUR4
      COMMON/PAR/OF2,OF3,OF4,NS,N1,N2,N3,N4,A2,A3,A4,DEL
C
      CK0=2.0*PI
      CKK=CK0*SQRT(EER)
      ARG=(CK0*WIDTH/2.D0)*SIN(FI)*SIN(THETA)
      CALL BESS(ARG,BJ)
      SSUM=0.D0
      DO 5 JN=1,NSER
           ARAF=CK0*S(JN,2)*SIN(FI)*SIN(THETA)
           CAFF=COS(ARAF)
           SSUM=SSUM+S(JN,1)*CAFF
  5   CONTINUE
      TERMI=(BJ-SSUM/PI)
C
      R1=CK0*SQRT(ER-SIN(THETA)**2)
      RS1=R1/CK0
      R2=R1*H
      R3=R2
      S1=SIN(R2)
      SS1=SIN(R1*(H-BS))
      SS2=SIN(R1*(H-DEL))
      S2=S1
      S3=SIN(THETA)
      S4=SIN(FI)
      C1=COS(R3)
      C2=COS(THETA)
      C3=COS(FI)
C
      W=(0.0,1.0)
      WWF=C2*TERMI/(C2*S2-W*RS1*C1)
      WWS=S3*C1*C2*TERMI/((C2*S2-W*RS1*C1)*(C2*ER*C1+W*RS1*S2))
      WFL=SS1*WWF
      WFP=SS2*WWF
      WFD=S1*WWF
      WSL=SS1*WWS
      WSP=SS2*WWS
      WSD=S1*WWS
C
      WTL=C2*WFL+(ER-1.00)*S3*WSL
      WTP=C2*WFP+(ER-1.00)*S3*WSP
      WTD=C2*WFD+(ER-1.00)*S3*WSD
C
C     For the transmission line
C
      WNL=(0.0,0.0)
      DO 1 I=1,N1
           R8=CK0*(FLOAT(I)*DLX)*S3*C3
           W8=COS(R8)+SIN(R8)*W
           WNL=WNL+CUR1(I)*W8
   1  CONTINUE
C
C     For dipole #2
C
      WND2=(0.0,0.0)
      DO 10 I=1,N2
            R8=CK0*(FLOAT(I)*DLX)*S3*C3
            W8=COS(R8)+SIN(R8)*W
            WND2=WND2+CUR2(I)*W8
  10  CONTINUE
```

```
      R8=CK0*A2*S3*C3
      W8=COS(R8)+SIN(R8)*W
      WND2=WND2*W8
      R8=CK0*OF2*S4*S3
      W8=COS(R8)+SIN(R8)*W
      WND2=WND2*W8
C
C     For dipole #3
C
      IF (N3.EQ.0) GO TO 503
      WND3=(0.0,0.0)
      DO 11 I=1,N3
             R8=CK0*FLOAT(I)*DLX*S3*C3
             W8=COS(R8)+SIN(R8)*W
             WND3=WND3+CUR3(I)*W8
   11 CONTINUE
      R8=CK0*A3*S3*C3
      W8=COS(R8)+SIN(R8)*W
      WND3=WND3*W8
      R8=CK0*OF3*S4*S3
      W8=COS(R8)+SIN(R8)*W
      WND3=WND3*W8
C
C     For dipole #4
C
  503 CONTINUE
      WND4=(0.0,0.0)
      DO 12 I=1,N4
             R8=CK0*FLOAT(I)*DLX*S3*C3
             W8=COS(R8)+SIN(R8)*W
             WND4=WND4+CUR4(I)*W8
   12 CONTINUE
      R8=CK0*A4*S3*C3
      W8=COS(R8)+SIN(R8)*W
      WND4=WND4*W8
      R8=CK0*OF4*S4*S3
      W8=COS(R8)+SIN(R8)*W
      WND4=WND4*W8
C
      WNP=WND2+WND3
      IF(N3.EQ.0) WNP=WND2
      WND=WND4
C
      IF(ICON.EQ.1) WNL=(0.D0,0.D0)
      IF(ICON.EQ.-1) WND=(0.D0,0.D0)
      IF (ABS(FI).GT.1.E-4) GO TO 2
      THER=ABS(ABS(THETA)-PI/2.0)
      IF (THER.GE.1.E-4) GO TO 2
      IF (ABS(EER-1.00).LT.1.E-6) GO TO 3
    2 R6=COS(CK0*DLX*S3*C3)-COS(CKK*DLX)
      R7=SIN(CKK*DLX)*(1.00-(CK0*S3*C3/CKK)**2)
C
      ETH=W*(-1.0)*C3*R6*(WTL*WNL+WTP*WNP+WTD*WND)/R7
      EFI=W*S4*R6*(WFL*WNL+WFP*WNP+WFD*WND)/R7
      RETURN
C
    3 R10=DLX*CK0/2.0
      ETH=(-1.0)*C3*(WTL*WNL+WTP*WNP+WTD*WND)*R10
      EFI=S4*(WFL*WNL+WFP*WNP+WFD*WND)*R10
C     WRITE (6,4) ETH,EFI
```

```
   4  FORMAT (5X,'ETH=',(E14.8,5X,E14.8),5X,'EFI=',(E14.8,5X,E14.8))
C
      RETURN
      END
C***********************************************************************
C                              BESS
C***********************************************************************
      SUBROUTINE BESS(X,BJ)
      IMPLICIT REAL*4 (A-H,O-Z)
C
      PI=3.14159265358900
      IF (X.GT.0.00100) GO TO 10
          X3=X/3.00
          X32=X3*X3
          X34=X32*X32
          X36=X34*X32
          BJ=1.00-2.249999700*X32+1.265620800*X34-0.316386600*X36
          GO TO 1
  10  IF (X.GT.3.00) GO TO 12
          X3=X/3.00
          X32=X3*X3
          X34=X32*X32
          X36=X34*X32
          X38=X36*X32
          X310=X38*X32
          X312=X310*X32
          BJ0=1.00-2.249999700*X32+1.265620800*X34-0.316386600
     *        *X36+0.044447900*X38-0.003944400*X310+0.00021000
     *        00*X312
          BJ=BJ0
          GO TO 1
   12 CONTINUE
      X3=3.00/X
      X32=X3*X3
      X33=X32*X3
      X34=X33*X3
      X35=X34*X3
      X36=X35*X3
      FJ0=0.7978845600-0.0000007700*X3-0.0055274000*X32-0.0000
     *    951200*X33+0.0013723700*X34-0.0007280500*X35+0.00014
     *    47600*X36
      TJ0=X-0.7853981600-0.0416639700*X3-0.0000395400*X32+0.00
     *    26257300*X33-0.0005412500*X34-0.0002933300*X35+0.000
     *    1355800*X36
      WCON=SQRT(1.00/X)
      BJ=WCON*FJ0*COS(TJ0)
   1  CONTINUE
      RETURN
      END
```

FOPTION4

(OUT1 / DOUT)

0.2170000E+01
0.1371154E+00
0.1027047E+00
0.1318417E-03
0.1521250E-01
0.9536503E-01
0.3141593E+01
0.1318417E-03

.... Z11................

25

0.4375703E-03	-0.1661283E+01
0.4372342E-03	0.3532292E+00
0.4362269E-03	0.2472233E+00
0.4345515E-03	0.9749434E-01
0.4322130E-03	0.5504017E-01
0.4292185E-03	0.3476495E-01
0.4255770E-03	0.2315949E-01
0.4212995E-03	0.1604274E-01
0.4163987E-03	0.1160677E-01
0.4108895E-03	0.8839811E-02
0.4047883E-03	0.7027479E-02
0.3981133E-03	0.5676763E-02
0.3908846E-03	0.4544296E-02
0.3831235E-03	0.3597179E-02
0.3748532E-03	0.2886684E-02
0.3660982E-03	0.2419244E-02
0.3568842E-03	0.2112645E-02
0.3472384E-03	0.1849440E-02
0.3371890E-03	0.1561405E-02
0.3267654E-03	0.1265194E-02
0.3159978E-03	0.1023717E-02
0.3049174E-03	0.8757187E-03
0.2935560E-03	0.7982550E-03
0.2819462E-03	0.7300448E-03
0.2701209E-03	0.6270357E-03

.... Z31 ....

130

0.1197214E-03	-0.5197332E-02
0.1196309E-03	-0.4600093E-02
0.1193598E-03	-0.3074783E-02
0.1189089E-03	-0.1220961E-02

0.1182795E-03	0.4122365E-03
0.1174735E-03	0.1549654E-02
0.1164933E-03	0.2172842E-02
0.1153419E-03	0.2396465E-02
0.1140227E-03	0.2362587E-02
0.1125396E-03	0.2188393E-02
0.1108971E-03	0.1953351E-02
0.1091000E-03	0.1704640E-02
0.1071537E-03	0.1467288E-02
0.1050638E-03	0.1252935E-02
0.1028367E-03	0.1065655E-02
0.1004788E-03	0.9053170E-03
0.9799706E-04	0.7695368E-03
0.9539871E-04	0.6549821E-03
0.9269133E-04	0.5583032E-03
0.8988277E-04	0.4766426E-03
0.8698112E-04	0.4077025E-03
0.8399473E-04	0.3495403E-03
0.8093213E-04	0.3003695E-03
0.7780202E-04	0.2585427E-03
0.7461323E-04	0.2226798E-03
0.7137472E-04	0.1917731E-03
0.6809551E-04	0.1651376E-03
0.6478465E-04	0.1422379E-03
0.6145124E-04	0.1225339E-03
0.5810433E-04	0.1054588E-03
0.5475293E-04	0.9051556E-04
0.5140597E-04	0.7737407E-04
0.4807227E-04	0.6586448E-04
0.4476052E-04	0.5587287E-04
0.4147921E-04	0.4723490E-04
0.3823667E-04	0.3971903E-04
0.3504097E-04	0.3310098E-04
0.3189996E-04	0.2724763E-04
0.2882119E-04	0.2212755E-04
0.2581191E-04	0.1774105E-04
0.2287906E-04	0.1403868E-04
0.2002923E-04	0.1090304E-04
0.1726862E-04	0.8204418E-05
0.1460308E-04	0.5871302E-05
0.1203805E-04	0.3907480E-05
0.9578526E-05	0.2342382E-05
0.7229109E-05	0.1165070E-05
0.4993940E-05	0.3043443E-06
0.2876706E-05	-0.3298236E-06
0.8806384E-06	-0.7868898E-06
-0.9915038E-06	-0.1055813E-05
-0.2737433E-05	-0.1101321E-05
-0.4355337E-05	-0.9191041E-06
-0.5843882E-05	-0.5569956E-06
-0.7202210E-05	-0.8437438E-07
-0.8429936E-05	0.4581213E-06
-0.9527148E-05	0.1079714E-05
-0.1049439E-04	0.1812920E-05
-0.1133268E-04	0.2666986E-05
-0.1204345E-04	0.3606538E-05
-0.1262861E-04	0.4574481E-05
-0.1309045E-04	0.5535256E-05
-0.1343169E-04	0.6496413E-05
-0.1365545E-04	0.7489060E-05

-0.1376522E-04	0.8527659E-05
-0.1376482E-04	0.9588264E-05
-0.1365846E-04	0.1062490E-04
-0.1345061E-04	0.1160664E-04
-0.1314608E-04	0.1253900E-04
-0.1274994E-04	0.1344990E-04
-0.1226748E-04	0.1435508E-04
-0.1170426E-04	0.1523690E-04
-0.1106601E-04	0.1605631E-04
-0.1035863E-04	0.1678547E-04
-0.9588199E-05	0.1742923E-04
-0.8760884E-05	0.1801534E-04
-0.7882963E-05	0.1856387E-04
-0.6960781E-05	0.1906551E-04
-0.6000723E-05	0.1948892E-04
-0.5009189E-05	0.1980912E-04
-0.3992569E-05	0.2002908E-04
-0.2957216E-05	0.2017448E-04
-0.1909418E-05	0.2026763E-04
-0.8553802E-06	0.2030602E-04
0.1988088E-06	0.2026558E-04
0.1247194E-05	0.2012452E-04
0.2283978E-05	0.1988429E-04
0.3303545E-05	0.1956837E-04
0.4300482E-05	0.1920083E-04
0.5269593E-05	0.1878607E-04
0.6205925E-05	0.1830835E-04
0.7104783E-05	0.1775027E-04
0.7961742E-05	0.1711167E-04
0.8772670E-05	0.1641110E-04
0.9533735E-05	0.1566994E-04
0.1024142E-04	0.1489563E-04
0.1089253E-04	0.1408006E-04
0.1148420E-04	0.1321333E-04
0.1201393E-04	0.1229810E-04
0.1247953E-04	0.1135014E-04
0.1287920E-04	0.1038552E-04
0.1321147E-04	0.9408292E-05
0.1347524E-04	0.8411631E-05
0.1366975E-04	0.7390990E-05
0.1379461E-04	0.6355164E-05
0.1384978E-04	0.5323179E-05
0.1383556E-04	0.4309571E-05
0.1375260E-04	0.3313106E-05
0.1360188E-04	0.2321287E-05
0.1338469E-04	0.1327297E-05
0.1310265E-04	0.3428733E-06
0.1275768E-04	-0.6065851E-06
0.1235199E-04	-0.1501423E-05
0.1188805E-04	-0.2343653E-05
0.1136862E-04	-0.3153454E-05
0.1079668E-04	-0.3948144E-05
0.1017545E-04	-0.4722531E-05
0.9508345E-05	-0.5449277E-05
0.8798992E-05	-0.6099966E-05
0.8051177E-05	-0.6668230E-05
0.7268845E-05	-0.7173762E-05
0.6456068E-05	-0.7643076E-05
0.5617034E-05	-0.8084045E-05
0.4756021E-05	-0.8477126E-05

0.3877376E-05	-0.8791225E-05
0.2985495E-05	-0.9009994E-05
0.2084802E-05	-0.9145137E-05
0.1179727E-05	-0.9225176E-05
0.2746842E-06	-0.9270175E-05
-0.6259456E-06	-0.9275015E-05

.... Z33 ....

116

0.3288023E-04	-0.1197635E+01
0.3285579E-04	0.2693736E+00
0.3278254E-04	0.1872652E+00
0.3266069E-04	0.7291131E-01
0.3249062E-04	0.3780940E-01
0.3227283E-04	0.2076203E-01
0.3200797E-04	0.1184056E-01
0.3169681E-04	0.7237307E-02
0.3134030E-04	0.4781738E-02
0.3093948E-04	0.3267499E-02
0.3049555E-04	0.2185171E-02
0.3000980E-04	0.1425656E-02
0.2948368E-04	0.9839267E-03
0.2891873E-04	0.7875281E-03
0.2831661E-04	0.6938318E-03
0.2767908E-04	0.5828647E-03
0.2700800E-04	0.4286220E-03
0.2630531E-04	0.2846170E-03
0.2557304E-04	0.2093975E-03
0.2481331E-04	0.2058157E-03
0.2402830E-04	0.2231657E-03
0.2322024E-04	0.2084661E-03
0.2239143E-04	0.1526730E-03
0.2154421E-04	0.9091299E-04
0.2068094E-04	0.6305948E-04
0.1980405E-04	0.7566818E-04
0.1891595E-04	0.9952167E-04
0.1801909E-04	0.1004154E-03
0.1711589E-04	0.7096449E-04
0.1620881E-04	0.3414655E-04
0.1530027E-04	0.1848804E-04
0.1439266E-04	0.3080254E-04
0.1348837E-04	0.5188318E-04
0.1258973E-04	0.5680772E-04
0.1169903E-04	0.3863513E-04
0.1081850E-04	0.1317317E-04
0.9950335E-05	0.2110897E-05
0.9096638E-05	0.1213789E-04
0.8259447E-05	0.2986540E-04
0.7440718E-05	0.3597083E-04
0.6642321E-05	0.2378001E-04
0.5866035E-05	0.4544152E-05
0.5113539E-05	-0.4506666E-05
0.4386409E-05	0.3167814E-05
0.3686116E-05	0.1802484E-04
0.3014017E-05	0.2458162E-04
0.2371355E-05	0.1646190E-04
0.1759254E-05	0.1834847E-05
0.1178717E-05	-0.5250128E-05
0.6306230E-06	0.1449072E-05
0.1157258E-06	0.1489077E-04
-0.3653473E-06	0.2214604E-04

-0.8120967E-06	0.1690040E-04
-0.1224151E-05	0.4948278E-05
-0.1601265E-05	-0.2086566E-05
-0.1943323E-05	0.1840845E-05
-0.2250333E-05	0.1177479E-04
-0.2522429E-05	0.1713590E-04
-0.2759865E-05	0.1219867E-04
-0.2963019E-05	0.1428127E-05
-0.3132381E-05	-0.5155865E-05
-0.3268558E-05	-0.1701040E-05
-0.3372264E-05	0.8145168E-05
-0.3444321E-05	0.1514078E-04
-0.3485650E-05	0.1343488E-04
-0.3497268E-05	0.5857418E-05
-0.3480285E-05	0.6764347E-06
-0.3435894E-05	0.3335645E-05
-0.3365368E-05	0.1130338E-04
-0.3270054E-05	0.1682667E-04
-0.3151365E-05	0.1462315E-04
-0.3010777E-05	0.6942053E-05
-0.2849816E-05	0.1234478E-05
-0.2670059E-05	0.2912714E-05
-0.2473120E-05	0.1027444E-04
-0.2260648E-05	0.1634850E-04
-0.2034318E-05	0.1566181E-04
-0.1795823E-05	0.9328992E-05
-0.1546868E-05	0.3710752E-05
-0.1289163E-05	0.4168677E-05
-0.1024415E-05	0.1002694E-04
-0.7543235E-06	0.1545943E-04
-0.4805694E-06	0.1521833E-04
-0.2048123E-06	0.9586799E-05
0.7131776E-07	0.3910619E-05
0.3462266E-06	0.3383518E-05
0.6183617E-06	0.8204093E-05
0.8862183E-06	0.1363029E-04
0.1148345E-05	0.1462865E-04
0.1403348E-05	0.1051934E-04
0.1649900E-05	0.5284293E-05
0.1886738E-05	0.3538532E-05
0.2112675E-05	0.6236906E-05
0.2326599E-05	0.1015637E-04
0.2527479E-05	0.1133338E-04
0.2714367E-05	0.8865012E-05
0.2886402E-05	0.5409839E-05
0.3042812E-05	0.4193346E-05
0.3182916E-05	0.5750362E-05
0.3306124E-05	0.7671177E-05
0.3411943E-05	0.7401960E-05
0.3499972E-05	0.5038853E-05
0.3569907E-05	0.3175008E-05
0.3621536E-05	0.4000362E-05
0.3654746E-05	0.6754503E-05
0.3669514E-05	0.8310400E-05
0.3665912E-05	0.6428076E-05
0.3644102E-05	0.2354957E-05
0.3604335E-05	-0.6664524E-07
0.3546949E-05	0.1838328E-05
0.3472364E-05	0.6599065E-05
0.3381083E-05	0.9546749E-05

0.3273687E-05	0.7104707E-05
0.3150827E-05	0.4457888E-06
0.3013228E-05	-0.5131084E-05
0.2861678E-05	-0.4832633E-05

.... Z21 ....

26

0.3070108E-03	-0.5240316E-01
0.3067773E-03	-0.3716714E-01
0.3060775E-03	-0.9469705E-02
0.3049135E-03	0.8399848E-02
0.3032889E-03	0.1449135E-01
0.3012084E-03	0.1455873E-01
0.2986784E-03	0.1259615E-01
0.2957065E-03	0.1031939E-01
0.2923014E-03	0.8313972E-02
0.2884735E-03	0.6708082E-02
0.2842341E-03	0.5456517E-02
0.2795959E-03	0.4467567E-02
0.2745726E-03	0.3665454E-02
0.2691792E-03	0.3009858E-02
0.2634316E-03	0.2484292E-02
0.2573468E-03	0.2073846E-02
0.2509427E-03	0.1752953E-02
0.2442379E-03	0.1490432E-02
0.2372521E-03	0.1263105E-02
0.2300057E-03	0.1063669E-02
0.2225195E-03	0.8959795E-03
0.2148150E-03	0.7631298E-03
0.2069145E-03	0.6593909E-03
0.1988404E-03	0.5722814E-03
0.1906156E-03	0.4913286E-03
0.1822631E-03	0.4146210E-03

.... Z22 ....

25

0.2156319E-03	-0.1212243E+01
0.2154694E-03	0.2567733E+00
0.2149826E-03	0.1797817E+00
0.2141729E-03	0.7149411E-01
0.2130427E-03	0.4123373E-01
0.2115954E-03	0.2666213E-01
0.2098353E-03	0.1802873E-01
0.2077677E-03	0.1254659E-01
0.2053987E-03	0.9106259E-02
0.2027355E-03	0.7017869E-02
0.1997859E-03	0.5677413E-02
0.1965586E-03	0.4622736E-02
0.1930634E-03	0.3648396E-02
0.1893104E-03	0.2787972E-02
0.1853107E-03	0.2158871E-02
0.1810761E-03	0.1796604E-02
0.1766191E-03	0.1603533E-02
0.1719525E-03	0.1428847E-02
0.1670900E-03	0.1188040E-02
0.1620457E-03	0.9122948E-03
0.1568341E-03	0.6942644E-03
0.1514701E-03	0.5905598E-03
0.1459691E-03	0.5708609E-03
0.1403466E-03	0.5516402E-03
0.1346185E-03	0.4738459E-03

.... Z32 ....

116

0.8398477E-04	-0.2108493E-01
0.8392193E-04	-0.1604007E-01
0.8373358E-04	-0.5742136E-02
0.8342029E-04	0.2438552E-02
0.8298300E-04	0.6244519E-02
0.8242302E-04	0.6993413E-02
0.8174200E-04	0.6343529E-02
0.8094199E-04	0.5280786E-02
0.8002536E-04	0.4239757E-02
0.7899485E-04	0.3362835E-02
0.7785350E-04	0.2665395E-02
0.7660469E-04	0.2119872E-02
0.7525213E-04	0.1693396E-02
0.7379979E-04	0.1360176E-02
0.7225194E-04	0.1101369E-02
0.7061312E-04	0.9013040E-03
0.6888812E-04	0.7453791E-03
0.6708196E-04	0.6208563E-03
0.6519988E-04	0.5187199E-03
0.6324731E-04	0.4342004E-03
0.6122987E-04	0.3652068E-03
0.5915333E-04	0.3099255E-03
0.5702360E-04	0.2654668E-03
0.5484670E-04	0.2283444E-03
0.5262876E-04	0.1959346E-03
0.5037598E-04	0.1673655E-03
0.4809460E-04	0.1430247E-03
0.4579090E-04	0.1232219E-03
0.4347119E-04	0.1072506E-03
0.4114173E-04	0.9360762E-04
0.3880877E-04	0.8101762E-04
0.3647850E-04	0.6922612E-04
0.3415703E-04	0.5884689E-04
0.3185039E-04	0.5048133E-04
0.2956446E-04	0.4397306E-04
0.2730502E-04	0.3846021E-04
0.2507767E-04	0.3309995E-04
0.2288784E-04	0.2773520E-04
0.2074078E-04	0.2288949E-04
0.1864153E-04	0.1914622E-04
0.1659489E-04	0.1653293E-04
0.1460544E-04	0.1447911E-04
0.1267750E-04	0.1234152E-04
0.1081512E-04	0.9965440E-05
0.9022074E-05	0.7755629E-05
0.7301862E-05	0.6224222E-05
0.5657669E-05	0.5473090E-05
0.4092381E-05	0.5098420E-05
0.2608571E-05	0.4578929E-05
0.1208492E-05	0.3752783E-05
-0.1059251E-06	0.2930112E-05
-0.1333078E-05	0.2557416E-05
-0.2471693E-05	0.2775465E-05
-0.3520825E-05	0.3291896E-05
-0.4479861E-05	0.3669388E-05
-0.5348512E-05	0.3733715E-05
-0.6126818E-05	0.3710012E-05
-0.6815136E-05	0.3973648E-05
-0.7414141E-05	0.4672130E-05

-0.7924815E-05	0.5582907E-05
-0.8348444E-05	0.6330278E-05
-0.8686603E-05	0.6738780E-05
-0.8941151E-05	0.6982085E-05
-0.9114220E-05	0.7396274E-05
-0.9208202E-05	0.8149983E-05
-0.9225733E-05	0.9090107E-05
-0.9169688E-05	0.9900275E-05
-0.9043158E-05	0.1040644E-04
-0.8849440E-05	0.1073196E-04
-0.8592020E-05	0.1116109E-04
-0.8274555E-05	0.1185333E-04
-0.7900860E-05	0.1268767E-04
-0.7474887E-05	0.1338084E-04
-0.7000710E-05	0.1375754E-04
-0.6482505E-05	0.1391015E-04
-0.5924533E-05	0.1410006E-04
-0.5331123E-05	0.1450311E-04
-0.4706649E-05	0.1504558E-04
-0.4055518E-05	0.1548231E-04
-0.3382147E-05	0.1563351E-04
-0.2690947E-05	0.1555046E-04
-0.1986302E-05	0.1545655E-04
-0.1272557E-05	0.1552845E-04
-0.5539957E-06	0.1573047E-04
0.1651747E-06	0.1585587E-04
0.8808392E-06	0.1572751E-04
0.1588992E-05	0.1536221E-04
0.2285749E-05	0.1494680E-04
0.2967368E-05	0.1465766E-04
0.3630255E-05	0.1450067E-04
0.4270983E-05	0.1431939E-04
0.4886302E-05	0.1395282E-04
0.5473148E-05	0.1338515E-04
0.6028658E-05	0.1274819E-04
0.6550176E-05	0.1218607E-04
0.7035263E-05	0.1172154E-04
0.7481702E-05	0.1125122E-04
0.7887508E-05	0.1066292E-04
0.8250930E-05	0.9950179E-05
0.8570457E-05	0.9210385E-05
0.8844822E-05	0.8536643E-05
0.9073000E-05	0.7920651E-05
0.9254215E-05	0.7268076E-05
0.9387934E-05	0.6508315E-05
0.9473870E-05	0.5681192E-05
0.9511978E-05	0.4905795E-05
0.9502454E-05	0.4258939E-05
0.9445725E-05	0.3688104E-05
0.9342453E-05	0.3054941E-05
0.9193519E-05	0.2275284E-05
0.9000022E-05	0.1418686E-05
0.8763273E-05	0.6610227E-06
0.8484777E-05	0.1240331E-06
0.8166234E-05	-0.2485011E-06
0.7809523E-05	-0.6530843E-06
0.7416691E-05	-0.1251186E-05

FOPTION5

(RESULT)

****

1 A strip dipole at the interface EM coupled
to another printed dipole in the dielectric which
is excited by a

```
ER= 0.2170000E+01      H= 0.9307998E-01
BS= 0.6972045E-01      T1= 0.8949997E-04
T2=-0.4745267E-76      DLX= 0.1521250E-01
W= 0.6473798E-01
OFFSET=-0.4745267E-76      WDELTA= 0.8949997E-04
DELTA=-0.4745267E-76
```

ZIN= 0.1390856E+02 -0.1946242E+02

Current distribution on the t.l.

CURR(1)=(0.5529810E-02,	0.6086718E-02	0.8223556E-02
CURR(2)=(0.7913578E-02,	0.8723304E-02	0.1177797E-01
CURR(3)=(0.1035526E-01,	0.1142906E-01	0.1542254E-01
CURR(4)=(0.1256970E-01,	0.1389077E-01	0.1873368E-01
CURR(5)=(0.1463460E-01,	0.1619454E-01	0.2182738E-01
CURR(6)=(0.1653551E-01,	0.1832530E-01	0.2468278E-01
CURR(7)=(0.1826459E-01,	0.2027472E-01	0.2728845E-01
CURR(8)=(0.1981271E-01,	0.2203275E-01	0.2963082E-01
CURR(9)=(0.2117016E-01,	0.2358915E-01	0.3169580E-01
CURR(10)=(0.2232717E-01,	0.2493553E-01	0.3347063E-01
CURR(11)=(0.2327523E-01,	0.2606709E-01	0.3494609E-01
CURR(12)=(0.2400661E-01,	0.2698093E-01	0.3611492E-01

```
CURR(  13)=( 0.2451679E-01, 0.2768240E-01      0.3697821E-01
CURR(  14)=( 0.2480210E-01, 0.2816359E-01      0.3752775E-01
CURR(  15)=( 0.2486178E-01, 0.2860489E-01      0.3789918E-01
CURR(  16)=( 0.2469550E-01, 0.2839100E-01      0.3762867E-01
CURR(  17)=( 0.2430572E-01, 0.3401132E-01      0.4180356E-01
CURR(  18)=( 0.2369601E-01, 0.2534669E-01      0.3469806E-01
CURR(  19)=( 0.2287227E-01, 0.2254402E-01      0.3211500E-01
CURR(  20)=( 0.2184313E-01, 0.1914320E-01      0.2904452E-01
CURR(  21)=( 0.2061787E-01, 0.1578638E-01      0.2596740E-01
CURR(  22)=( 0.1920843E-01, 0.1232169E-01      0.2282077E-01
CURR(  23)=( 0.1762706E-01, 0.8782301E-02      0.1969370E-01
CURR(  24)=( 0.1588852E-01, 0.5188957E-02      0.1671436E-01
CURR(  25)=( 0.1400849E-01, 0.1567575E-02      0.1409592E-01
CURR(  26)=( 0.1200465E-01,-0.2054288E-02      0.1217915E-01
CURR(  27)=( 0.9895012E-02,-0.5646870E-02      0.1139291E-01
CURR(  28)=( 0.7698670E-02,-0.9178828E-02      0.1198000E-01
CURR(  29)=( 0.5435623E-02,-0.1261922E-01      0.1374011E-01
CURR(  30)=( 0.3126045E-02,-0.1593769E-01      0.1624136E-01
CURR(  31)=( 0.7909655E-03,-0.1910619E-01      0.1912256E-01
CURR(  32)=(-0.1548504E-02,-0.2209676E-01      0.2215095E-01
CURR(  33)=(-0.3871198E-02,-0.2488352E-01      0.2518284E-01
CURR(  34)=(-0.6156076E-02,-0.2744056E-01      0.2812261E-01
CURR(  35)=(-0.8382879E-02,-0.2974601E-01      0.3090466E-01
CURR(  36)=(-0.1053148E-01,-0.3177864E-01      0.3347825E-01
CURR(  37)=(-0.1258259E-01,-0.3352074E-01      0.3580449E-01
CURR(  38)=(-0.1451793E-01,-0.3495757E-01      0.3785237E-01
CURR(  39)=(-0.1631971E-01,-0.3607548E-01      0.3959512E-01
CURR(  40)=(-0.1797234E-01,-0.3686550E-01      0.4101305E-01
CURR(  41)=(-0.1946057E-01,-0.3731932E-01      0.4208855E-01
CURR(  42)=(-0.2077153E-01,-0.3743370E-01      0.4281049E-01
```

```
CURR(  43)=(-0.2189320E-01,-0.3720727E-01      0.4317050E-01
CURR(  44)=(-0.2281630E-01,-0.3664362E-01      0.4316641E-01
CURR(  45)=(-0.2353165E-01,-0.3574622E-01      0.4279638E-01
CURR(  46)=(-0.2403376E-01,-0.3452448E-01      0.4206615E-01
CURR(  47)=(-0.2431779E-01,-0.3298843E-01      0.4098281E-01
CURR(  48)=(-0.2438155E-01,-0.3115239E-01      0.3955921E-01
CURR(  49)=(-0.2422509E-01,-0.2903360E-01      0.3781276E-01
CURR(  50)=(-0.2384919E-01,-0.2665045E-01      0.3576353E-01
CURR(  51)=(-0.2325809E-01,-0.2402529E-01      0.3343879E-01
CURR(  52)=(-0.2245648E-01,-0.2118091E-01      0.3086947E-01
CURR(  53)=(-0.2145283E-01,-0.1814394E-01      0.2809673E-01
CURR(  54)=(-0.2025533E-01,-0.1494094E-01      0.2516963E-01
CURR(  55)=(-0.1887581E-01,-0.1160165E-01      0.2215613E-01
CURR(  56)=(-0.1732635E-01,-0.8156050E-02      0.1915003E-01
CURR(  57)=(-0.1562117E-01,-0.4635349E-02      0.1629440E-01
CURR(  58)=(-0.1377605E-01,-0.1071388E-02      0.1381765E-01
CURR(  59)=(-0.1180728E-01, 0.2503756E-02      0.1206982E-01
CURR(  60)=(-0.9733178E-02, 0.6057687E-02      0.1146431E-01
CURR(  61)=(-0.7572234E-02, 0.9557996E-02      0.1219401E-01
CURR(  62)=(-0.5344339E-02, 0.1297331E-01      0.1403099E-01
CURR(  63)=(-0.3069290E-02, 0.1627221E-01      0.1655915E-01
CURR(  64)=(-0.7677733E-03, 0.1942547E-01      0.1944063E-01
CURR(  65)=( 0.1539489E-02, 0.2240423E-01      0.2245706E-01
CURR(  66)=( 0.3831622E-02, 0.2518150E-01      0.2547134E-01
CURR(  67)=( 0.6088078E-02, 0.2773252E-01      0.2839291E-01
CURR(  68)=( 0.8288421E-02, 0.3003322E-01      0.3115594E-01
CURR(  69)=( 0.1041317E-01, 0.3206373E-01      0.3371227E-01
CURR(  70)=( 0.1244302E-01, 0.3380474E-01      0.3602207E-01
CURR(  71)=( 0.1436022E-01, 0.3524167E-01      0.3805510E-01
CURR(  72)=( 0.1614732E-01, 0.3636054E-01      0.3978472E-01
```

```
CURR(  73)=( 0.1778847E-01, 0.3715133E-01     0.4119042E-01
CURR(  74)=( 0.1926944E-01, 0.3760728E-01     0.4225658E-01
CURR(  75)=( 0.2057667E-01, 0.3772321E-01     0.4297022E-01
CURR(  76)=( 0.2169960E-01, 0.3749960E-01     0.4332542E-01
CURR(  77)=( 0.2262778E-01, 0.3693730E-01     0.4331721E-01
CURR(  78)=( 0.2335395E-01, 0.3604240E-01     0.4294719E-01
CURR(  79)=( 0.2387122E-01, 0.3482158E-01     0.4221822E-01
CURR(  80)=( 0.2417680E-01, 0.3328777E-01     0.4114114E-01
CURR(  81)=( 0.2426766E-01, 0.3145348E-01     0.3972708E-01
CURR(  82)=( 0.2414458E-01, 0.2933664E-01     0.3799473E-01
CURR(  83)=( 0.2380930E-01, 0.2695629E-01     0.3596560E-01
CURR(  84)=( 0.2326557E-01, 0.2433395E-01     0.3366642E-01
CURR(  85)=( 0.2252033E-01, 0.2149462E-01     0.3113171E-01
CURR(  86)=( 0.2158127E-01, 0.1846400E-01     0.2840194E-01
CURR(  87)=( 0.2045841E-01, 0.1527042E-01     0.2552905E-01
CURR(  88)=( 0.1916457E-01, 0.1194447E-01     0.2258209E-01
CURR(  89)=( 0.1771324E-01, 0.8517347E-02     0.1965462E-01
CURR(  90)=( 0.1612000E-01, 0.5021736E-02     0.1688408E-01
CURR(  91)=( 0.1440319E-01, 0.1491607E-02     0.1448022E-01
CURR(  92)=( 0.1258180E-01,-0.2038496E-02     0.1274586E-01
CURR(  93)=( 0.1067721E-01,-0.5532678E-02     0.1202553E-01
CURR(  94)=( 0.8712891E-02,-0.8954588E-02     0.1249396E-01
CURR(  95)=( 0.6713960E-02,-0.1226678E-01     0.1398396E-01
CURR(  96)=( 0.4707947E-02,-0.1543095E-01     0.1613317E-01
CURR(  97)=( 0.2725505E-02,-0.1840834E-01     0.1860902E-01
CURR(  98)=( 0.8004336E-03,-0.2115661E-01     0.2117174E-01
CURR(  99)=(-0.1028201E-02,-0.2363015E-01     0.2365251E-01
CURR( 100)=(-0.2714849E-02,-0.2577736E-01     0.2591992E-01
CURR( 101)=(-0.4205409E-02,-0.2753685E-01     0.2785612E-01
CURR( 102)=(-0.5437352E-02,-0.2883789E-01     0.2934601E-01
```

```
CURR( 103)=(-0.6348044E-02,-0.2960837E-01     0.3028124E-01
CURR( 104)=(-0.6900232E-02,-0.2980108E-01     0.3058950E-01
CURR( 105)=(-0.7117171E-02,-0.2943631E-01     0.3028449E-01
CURR( 106)=(-0.7079169E-02,-0.2860190E-01     0.2946495E-01
CURR( 107)=(-0.6871171E-02,-0.2739449E-01     0.2824306E-01
CURR( 108)=(-0.6548587E-02,-0.2588402E-01     0.2669955E-01
CURR( 109)=(-0.6139353E-02,-0.2411075E-01     0.2488011E-01
CURR( 110)=(-0.5658176E-02,-0.2210324E-01     0.2281597E-01
CURR( 111)=(-0.5113490E-02,-0.1988231E-01     0.2052934E-01
CURR( 112)=(-0.4512072E-02,-0.1746885E-01     0.1804215E-01
CURR( 113)=(-0.3859175E-02,-0.1488047E-01     0.1537275E-01
CURR( 114)=(-0.3162184E-02,-0.1214405E-01     0.1254899E-01
CURR( 115)=(-0.2401073E-02,-0.9183709E-02     0.9492397E-02
CURR( 116)=(-0.1663873E-02,-0.6333020E-02     0.6547946E-02
```

Current on the first parasitic dipole

```
CURR( 117)=(-0.7535733E-02,-0.8758638E-02     0.1155426E-01
CURR( 118)=(-0.1014305E-01,-0.1219982E-01     0.1586559E-01
CURR( 119)=(-0.1276311E-01,-0.1571526E-01     0.2024516E-01
CURR( 120)=(-0.1505901E-01,-0.1888525E-01     0.2415422E-01
CURR( 121)=(-0.1717950E-01,-0.2185617E-01     0.2779977E-01
CURR( 122)=(-0.1911406E-01,-0.2460676E-01     0.3115830E-01
CURR( 123)=(-0.2085138E-01,-0.2711692E-01     0.3420683E-01
CURR( 124)=(-0.2237806E-01,-0.2936493E-01     0.3691987E-01
CURR( 125)=(-0.2368199E-01,-0.3132947E-01     0.3927305E-01
CURR( 126)=(-0.2475377E-01,-0.3298987E-01     0.4124416E-01
CURR( 127)=(-0.2558015E-01,-0.3431433E-01     0.4279973E-01
CURR( 128)=(-0.2614281E-01,-0.3525246E-01     0.4388829E-01
```

```
CURR( 129)=(-0.2642066E-01,-0.3573780E-01       0.4444368E-01
CURR( 130)=(-0.2639570E-01,-0.3571354E-01       0.4440934E-01
CURR( 131)=(-0.2606665E-01,-0.3517430E-01       0.4378016E-01
CURR( 132)=(-0.2544660E-01,-0.3416147E-01       0.4259736E-01
CURR( 133)=(-0.2455183E-01,-0.3272875E-01       0.4091409E-01
CURR( 134)=(-0.2339638E-01,-0.3092249E-01       0.3877617E-01
CURR( 135)=(-0.2199185E-01,-0.2877880E-01       0.3621962E-01
CURR( 136)=(-0.2034729E-01,-0.2632468E-01       0.3327162E-01
CURR( 137)=(-0.1847406E-01,-0.2358643E-01       0.2996015E-01
CURR( 138)=(-0.1637485E-01,-0.2057878E-01       0.2629870E-01
CURR( 139)=(-0.1405190E-01,-0.1731761E-01       0.2230147E-01
CURR( 140)=(-0.1131796E-01,-0.1361610E-01       0.1770577E-01
CURR( 141)=(-0.8545324E-02,-0.9938192E-02       0.1310688E-01
```

Current on the second parasitic dipole

```
CURR( 142)=(-0.4317302E-02,-0.1699954E-02       0.4639927E-02
CURR( 143)=(-0.5430728E-02,-0.1728756E-02       0.5699243E-02
CURR( 144)=(-0.6494038E-02,-0.1697618E-02       0.6712258E-02
CURR( 145)=(-0.7342964E-02,-0.1578550E-02       0.7510722E-02
CURR( 146)=(-0.8093011E-02,-0.1447108E-02       0.8221366E-02
CURR( 147)=(-0.8752242E-02,-0.1316910E-02       0.8850761E-02
CURR( 148)=(-0.9327009E-02,-0.1197289E-02       0.9403542E-02
CURR( 149)=(-0.9818468E-02,-0.1093740E-02       0.9879194E-02
CURR( 150)=(-0.1022789E-01,-0.1010995E-02       0.1027773E-01
CURR( 151)=(-0.1055577E-01,-0.9528201E-03       0.1059868E-01
CURR( 152)=(-0.1080253E-01,-0.9221481E-03       0.1084182E-01
CURR( 153)=(-0.1096762E-01,-0.9207914E-03       0.1100621E-01
CURR( 154)=(-0.1104936E-01,-0.9493500E-03       0.1109006E-01
```

CURR(155)=(-0.1104738E-01,-0.1007952E-02	0.1109327E-01
CURR(156)=(-0.1096119E-01,-0.1095505E-02	0.1101580E-01
CURR(157)=(-0.1079034E-01,-0.1209857E-02	0.1085795E-01
CURR(158)=(-0.1053413E-01,-0.1347447E-02	0.1061996E-01
CURR(159)=(-0.1019086E-01,-0.1503473E-02	0.1030116E-01
CURR(160)=(-0.9757549E-02,-0.1671682E-02	0.9899706E-02
CURR(161)=(-0.9231780E-02,-0.1845369E-02	0.9414405E-02
CURR(162)=(-0.8610032E-02,-0.2016632E-02	0.8843042E-02
CURR(163)=(-0.7884189E-02,-0.2173728E-02	0.8178353E-02
CURR(164)=(-0.7042494E-02,-0.2299465E-02	0.7408388E-02
CURR(165)=(-0.5952518E-02,-0.2299824E-02	0.6381352E-02
CURR(166)=(-0.4793983E-02,-0.2219521E-02	0.5282853E-02

FOPTION6

(RESGN)

116

0.5529810E-02	0.6086718E-02
0.7913578E-02	0.8723304E-02
0.1035526E-01	0.1142906E-01
0.1256970E-01	0.1389077E-01
0.1463460E-01	0.1619454E-01
0.1653551E-01	0.1832530E-01
0.1826459E-01	0.2027472E-01
0.1981271E-01	0.2203275E-01
0.2117016E-01	0.2358915E-01
0.2232717E-01	0.2493553E-01
0.2327523E-01	0.2606709E-01
0.2400661E-01	0.2698093E-01
0.2451679E-01	0.2768240E-01
0.2480210E-01	0.2816359E-01
0.2486178E-01	0.2860489E-01
0.2469550E-01	0.2839100E-01
0.2430572E-01	0.3401132E-01
0.2369601E-01	0.2534669E-01
0.2287227E-01	0.2254402E-01
0.2184313E-01	0.1914320E-01
0.2061787E-01	0.1578638E-01
0.1920843E-01	0.1232169E-01
0.1762706E-01	0.8782301E-02
0.1588852E-01	0.5188957E-02
0.1400849E-01	0.1567575E-02
0.1200465E-01	-0.2054288E-02
0.9895012E-02	-0.5646870E-02
0.7698670E-02	-0.9178828E-02
0.5435623E-02	-0.1261922E-01
0.3126045E-02	-0.1593769E-01
0.7909655E-03	-0.1910619E-01
-0.1548504E-02	-0.2209676E-01
-0.3871198E-02	-0.2488352E-01
-0.6156076E-02	-0.2744056E-01
-0.8382879E-02	-0.2974601E-01
-0.1053148E-01	-0.3177864E-01
-0.1258259E-01	-0.3352074E-01
-0.1451793E-01	-0.3495757E-01
-0.1631971E-01	-0.3607548E-01
-0.1797234E-01	-0.3686550E-01
-0.1946057E-01	-0.3731932E-01
-0.2077153E-01	-0.3743370E-01
-0.2189320E-01	-0.3720727E-01
-0.2281630E-01	-0.3664362E-01
-0.2353165E-01	-0.3574622E-01
-0.2403376E-01	-0.3452448E-01
-0.2431779E-01	-0.3298843E-01
-0.2438155E-01	-0.3115239E-01
-0.2422509E-01	-0.2903360E-01
-0.2384919E-01	-0.2665045E-01
-0.2325809E-01	-0.2402529E-01
-0.2245648E-01	-0.2118091E-01
-0.2145283E-01	-0.1814394E-01
-0.2025533E-01	-0.1494094E-01
-0.1887581E-01	-0.1160165E-01
-0.1732635E-01	-0.8156050E-02
-0.1562117E-01	-0.4635349E-02
-0.1377605E-01	-0.1071388E-02
-0.1180728E-01	0.2503756E-02

-0.9733178E-02	0.6057687E-02
-0.7572234E-02	0.9557996E-02
-0.5344339E-02	0.1297331E-01
-0.3069290E-02	0.1627221E-01
-0.7677733E-03	0.1942547E-01
0.1539489E-02	0.2240423E-01
0.3831622E-02	0.2518150E-01
0.6088078E-02	0.2773252E-01
0.8288421E-02	0.3003322E-01
0.1041317E-01	0.3206373E-01
0.1244302E-01	0.3380474E-01
0.1436022E-01	0.3524167E-01
0.1614732E-01	0.3636054E-01
0.1778847E-01	0.3715133E-01
0.1926944E-01	0.3760728E-01
0.2057667E-01	0.3772321E-01
0.2169960E-01	0.3749960E-01
0.2262778E-01	0.3693730E-01
0.2335395E-01	0.3604240E-01
0.2387122E-01	0.3482158E-01
0.2417680E-01	0.3328777E-01
0.2426766E-01	0.3145348E-01
0.2414458E-01	0.2933664E-01
0.2380930E-01	0.2695629E-01
0.2326557E-01	0.2433395E-01
0.2252033E-01	0.2149462E-01
0.2158127E-01	0.1846400E-01
0.2045841E-01	0.1527042E-01
0.1916457E-01	0.1194447E-01
0.1771324E-01	0.8517347E-02
0.1612000E-01	0.5021736E-02
0.1440319E-01	0.1491607E-02
0.1258180E-01	-0.2038496E-02
0.1067721E-01	-0.5532678E-02
0.8712891E-02	-0.8954588E-02
0.6713960E-02	-0.1226678E-01
0.4707947E-02	-0.1543095E-01
0.2725505E-02	-0.1840834E-01
0.8004336E-03	-0.2115661E-01
-0.1028201E-02	-0.2363015E-01
-0.2714849E-02	-0.2577736E-01
-0.4205409E-02	-0.2753685E-01
-0.5437352E-02	-0.2883789E-01
-0.6348044E-02	-0.2960837E-01
-0.6900232E-02	-0.2980108E-01
-0.7117171E-02	-0.2943631E-01
-0.7079169E-02	-0.2860190E-01
-0.6871171E-02	-0.2739449E-01
-0.6548587E-02	-0.2588402E-01
-0.6139353E-02	-0.2411075E-01
-0.5658176E-02	-0.2210324E-01
-0.5113490E-02	-0.1988231E-01
-0.4512072E-02	-0.1746885E-01
-0.3859175E-02	-0.1488047E-01
-0.3162184E-02	-0.1214405E-01
-0.2401073E-02	-0.9183709E-02
-0.1663873E-02	-0.6333020E-02
25	
-0.7535733E-02	-0.8758638E-02
-0.1014305E-01	-0.1219982E-01

-0.1276311E-01	-0.1571526E-01
-0.1505901E-01	-0.1888525E-01
-0.1717950E-01	-0.2185617E-01
-0.1911406E-01	-0.2460676E-01
-0.2085138E-01	-0.2711692E-01
-0.2237806E-01	-0.2936493E-01
-0.2368199E-01	-0.3132947E-01
-0.2475377E-01	-0.3298987E-01
-0.2558015E-01	-0.3431433E-01
-0.2614281E-01	-0.3525246E-01
-0.2642066E-01	-0.3573780E-01
-0.2639570E-01	-0.3571354E-01
-0.2606665E-01	-0.3517430E-01
-0.2544660E-01	-0.3416147E-01
-0.2455183E-01	-0.3272875E-01
-0.2339638E-01	-0.3092249E-01
-0.2199185E-01	-0.2877880E-01
-0.2034729E-01	-0.2632468E-01
-0.1847406E-01	-0.2358643E-01
-0.1637485E-01	-0.2057878E-01
-0.1405190E-01	-0.1731761E-01
-0.1131796E-01	-0.1361610E-01
-0.8545324E-02	-0.9938192E-02
25	
-0.4317302E-02	-0.1699954E-02
-0.5430728E-02	-0.1728756E-02
-0.6494038E-02	-0.1697618E-02
-0.7342964E-02	-0.1578550E-02
-0.8093011E-02	-0.1447108E-02
-0.8752242E-02	-0.1316910E-02
-0.9327009E-02	-0.1197289E-02
-0.9818468E-02	-0.1093740E-02
-0.1022789E-01	-0.1010995E-02
-0.1055577E-01	-0.9528201E-03
-0.1080253E-01	-0.9221481E-03
-0.1096762E-01	-0.9207914E-03
-0.1104936E-01	-0.9493500E-03
-0.1104738E-01	-0.1007952E-02
-0.1096119E-01	-0.1095505E-02
-0.1079034E-01	-0.1209857E-02
-0.1053413E-01	-0.1347447E-02
-0.1019086E-01	-0.1503473E-02
-0.9757549E-02	-0.1671682E-02
-0.9231780E-02	-0.1845369E-02
-0.8610032E-02	-0.2016632E-02
-0.7884189E-02	-0.2173728E-02
-0.7042494E-02	-0.2299465E-02
-0.5952518E-02	-0.2299824E-02
-0.4793983E-02	-0.2219521E-02

FOPTION7

(MUTUAL)

```
C*****************************************************************
C     The name of this file is ............. MUTUAL.................
C     It solves for the mutula coupling between two dipoles
C     excited by microstrip transmission lines embedded in the
C     dielectric.
C
C     It also solves for the bandwidth of a dipole printed on a
C     dielectric substrate and excited by a microstrip transmission
C     line embedded in the dielectric in the presence of two other
C     parasitics on the same level of the transmission line
C*****************************************************************
      IMPLICIT REAL*4 (A-H,O-Z)
      COMPLEX*8 CUR(170,3)
      COMPLEX*8 BMATR,Z11(100),Z44(100),Z12(200),Z23(200),Z34(200),
     *Z24(200),Z14(200),Z13(200)
      DIMENSION IB(160),IA(160),IDATA(10),RDATA(20)
      COMMON/MAN/BMATR(160,160)
C.................................................................
C                DATA
C.................................................................
      DO 180 ID=1,10
             READ (1,100) IDATA(ID)
  180 CONTINUE
      NOEL1=IDATA(1)
      NOEL2=IDATA(2)
      NOEL3=IDATA(3)
      NOEL4=IDATA(4)
      NS12=IDATA(5)
      NS23=IDATA(6)
      NS34=IDATA(7)
      NOR=IDATA(8)
      NFEED1=IDATA(9)
      NFEED4=IDATA(10)
      WRITE(6,222) NOEL1,NOEL2,NOEL3,NOEL4,NS12,NS23,NS34
 222  FORMAT (10X,'NOEL1=',I4/10X,'NOEL2=',I4/10X,'NOEL3=',I4/
     *10X,'NOEL4=',I4/10X,'NS12=',I4/10X,'NS23=',I4/10X,'NS34=',
     *I4,//////)
  100 FORMAT(10X,I4)
      DO 190 ID=1,11
             READ(2,200) RDATA(ID)
  190 CONTINUE
      ER=RDATA(1)
      H=RDATA(2)
      BS=RDATA(3)
      T=RDATA(4)
      DLX=RDATA(5)
      W=RDATA(6)
      OF12=RDATA(7)
      OF34=RDATA(8)
      DIEL=RDATA(9)
      PI=RDATA(10)
      WDELTA=RDATA(11)
  200 FORMAT(10X,E14.7)
      READ(2,310) N44
  310 FORMAT(/11X,I4)
      WRITE (6,330)
  330 FORMAT(/,'**************************************************'/)
      DO 500 I=1,N44
             READ(2,400) Z44(I)
C            WRITE (6,320) I,Z44(I)
```

```
  320         FORMAT (2X,'Z44(',I4,')=',E14.7,3X,E14.7)
  500 CONTINUE
      READ(2,310) N14
      DO 510 I=1,N14
              READ(2,400) Z14(I)
C             WRITE (6,321) I,Z14(I)
  321         FORMAT(2X,'Z14(',I4,')=',E14.7,3X,E14.7)
  510 CONTINUE
      READ(2,310) N11
      DO 520 I=1,N11
              READ(2,400) Z11(I)
C             WRITE (6,322) I,Z11(I)
  322         FORMAT(2X,'Z11(',I4,')=',E14.7,3X,E14.7)
  520 CONTINUE
      NOEL23=NOEL2+NOEL3
      IF (NOEL23.EQ.0) GO TO 299
      READ (2,310) N12
      DO 530 I=1,N12
              READ (2,400) Z12(I)
              Z13(I)=Z12(I)
C             WRITE (6,323) I,Z12(I)
  323         FORMAT(2X,'Z12(',I4,')=',E14.7,3X,E14.7)
  530 CONTINUE
      READ (2,310) N24
      DO 540 I=1,N24
              READ(2,400) Z24(I)
C             WRITE (6,324) I,Z24(I)
  324         FORMAT(2X,'Z24(',I4,')=',E14.7,3X,E14.7)
  540 CONTINUE
      READ (2,310) N23
      DO 550 I=1,N23
              READ(2,400) Z23(I)
C             WRITE(6,325) I,Z23(I)
  325         FORMAT(2X,'Z23(',I4,')=',E14.7,3X,E14.7)
  550 CONTINUE
      READ (2,310) N34
      DO 560 I=1,N34
              READ(2,400) Z34(I)
C             WRITE (6,326) I,Z34(I)
  326         FORMAT(2X,'Z34(',I4,')=',E14.7,3X,E14.7)
  560 CONTINUE
  299 CONTINUE
  400 FORMAT (11X,E14.7,4X,E14.7)
      H=H/SQRT(ER)
      BS=BS/SQRT(ER)
      T=T/SQRT(ER)
      W=W/SQRT(ER)
      OF12=OF12/SQRT(ER)
      OF34=OF34/SQRT(ER)
      DIEL=DIEL/SQRT(ER)
      WDELTA=WDELTA/SQRT(ER)
C     ................................................
C
      WRITE (6,1)
  1   FORMAT (//'1',10X,'Mutual coupling between two strip dipoles'
     */10X,'excited by two embedded transmission lines'/10X,'OR'/
     *10X,'Bandwidth of a printed dipole in the presence of two'/
     *10X,'parasitics on the same plane with thw embedded line'///
     */)
      WRITE (6,3) ER,H,BS,T,W,OF12,OF34,DIEL,WDELTA,DLX
```

```
   3  FORMAT(/10X,'ER=',E14.7/10X,'H=',E14.7/10X,'BS=',E14.7/10X,'T='
     *,E14.7/10X,'W=',E14.7/10X,'OFFSET12=',E14.7/10X,'OFFSET34=',
     *E14.7/10X,'DIEL=',E14.7/10X,'WDELTA=',E14.7/10X,'DLX=',E14.7
     *////////)
C
C     .....First Diagonal Matrix.........
C
      IMIN=1
      IMAX=NOEL1
      DO 4 I=IMIN,IMAX
            IXN=0
            DO 5 KI=I,IMAX
                  IXN=IXN+1
                  BMATR(IXN,KI)=Z11(I)
                  BMATR(KI,IXN)=BMATR(IXN,KI)
   5        CONTINUE
   4  CONTINUE
      IF (NOEL23.EQ.0) GO TO 300
C
C     .......Second Diagonal Matrix ........
C
C
      INI=NOEL1
      IMIN=NOEL1+1
      IMAX=NOEL1+NOEL2
      DO 6 I=IMIN,IMAX
            IXN=INI
            DO 7 KI=I,IMAX
                  IXN=IXN+1
                  BMATR(IXN,KI)=Z11(I-INI)
                  BMATR(KI,IXN)=BMATR(IXN,KI)
    7       CONTINUE
    6 CONTINUE
C
C     ......Third Diagonal Matrix.....
C
C
      INI=NOEL1+NOEL2
      IMIN=INI+1
      IMAX=INI+NOEL3
      DO 8 I=IMIN,IMAX
            IXN=INI
            DO 9 KI=I,IMAX
                  IXN=IXN+1
                  BMATR(IXN,KI)=Z11(I-INI)
                  BMATR(KI,IXN)=BMATR(IXN,KI)
    9       CONTINUE
    8 CONTINUE
  300 CONTINUE
C
C     .....Fourth Diagonal Matrix.....
C
C
      INI=NOEL1+NOEL2+NOEL3
      IMIN=INI+1
      IMAX=INI+NOEL4
      DO 10 I=IMIN,IMAX
             IXN=INI
             DO 11 KI=I,IMAX
                    IXN=IXN+1
```

```
               BMATR(IXN,KI)=Z44(I-INI)
               BMATR(KI,IXN)=BMATR(IXN,KI)
   11       CONTINUE
   10 CONTINUE
      IF (NOEL23.EQ.0) GO TO 301
C
C     ...1...  First off-diagonal matrix
C
C        1)   Upper Part
C
      IAI=NOEL1-NOEL2
      IMI=IABS(IAI)+1
      IMIN=NOEL1+1
      IMAX=NOEL1+NOEL2
      DO 12 I=IMIN,IMAX
            IXN=0
            LXN=NS12+I-IMIN
            IF (IAI.LT.0) GO TO 13
                   KIMIN=I
                   KIMAX=IMAX
                   GO TO 14
  13        KIMIN=I
            KIMAX=I+NOEL1
            IF ((I-IMIN+1).GE.IMI) KIMAX=IMAX
  14        DO 15 KI=KIMIN,KIMAX
                  IXN=IXN+1
                  BMATR(IXN,KI)=Z12(LXN)
                  BMATR(KI,IXN)=BMATR(IXN,KI)
  15        CONTINUE
  12  CONTINUE
C
C   ......2) lower Part .............
C
      IMIN=2
      IMAX=NOEL1
      DO 16 I=IMIN,IMAX
            IXN=I-1
            LXN=IABS(NS12-I)+1
            IF (IAI.GT.0) GO TO 17
                  KIMIN=NOEL1+1
                  KIMAX=2*NOEL1-I+IMIN-1
                  GO TO 18
  17        KIMIN=NOEL1+1
            KIMAX=NOEL1+NOEL2
            IIMI=I-IMIN+2
            IF (IIMI.GE.IMI) KIMAX=NOEL1+NOEL2-IIMI+IMI
  18        DO 19 KI=KIMIN,KIMAX
                  IXN=IXN+1
                  BMATR(IXN,KI)=Z12(LXN)
                  BMATR(KI,IXN)=BMATR(IXN,KI)
  19        CONTINUE
  16  CONTINUE
C
C   ....2....  First off-diagonal matrix
C
C    ... 1)  Upper Part.......
C
      IAI=NOEL2-NOEL3
      IMI=IABS(IAI)+1
      IMIN=NOEL1+NOEL2+1
```

```
      IMAX=NOEL1+NOEL2+NOEL3
      DO 20 I=IMIN,IMAX
             IXN=NOEL1
             LXN=IABS(NS23)+I-IMIN+1
             IF (IAI.LT.0) GO TO 21
                     KIMIN=I
                     KIMAX=IMAX
                     GO TO 22
   21        KIMIN=I
             KIMAX=I+NOEL2-1
             IF ((I-IMIN+1).GE.IMI) KIMAX=IMAX
   22        DO 23 KI=KIMIN,KIMAX
                    IXN=IXN+1
                    BMATR(IXN,KI)=Z23(LXN)
                    BMATR(KI,IXN)=BMATR(IXN,KI)
   23        CONTINUE
   20 CONTINUE
C
C        2)    Lower Part
C
      IMIN=NOEL1+2
      IMAX=NOEL1+NOEL2
      DO 24 I=IMIN,IMAX
             IXN=I-1
             LXN=IABS(NS23-I+IMIN-1)+1
             IF (IAI.GT.0) GO TO 25
                     KIMIN=NOEL1+NOEL2+1
                     KIMAX=NOEL1+2*NOEL2-(I-IMIN)-1
                     GO TO 26
   25        KIMIN=NOEL1+NOEL2+1
             KIMAX=NOEL1+NOEL2+NOEL3
             IIMI=I-IMIN+2
             IF (IIMI.GE.IMI) KIMAX=NOEL1+NOEL2+NOEL3-IIMI+IMI
   26        DO 27 KI=KIMIN,KIMAX
                    IXN=IXN+1
                    BMATR(IXN,KI)=Z23(LXN)
                    BMATR(KI,IXN)=BMATR(IXN,KI)
   27        CONTINUE
   24 CONTINUE
C
C    ...3.... First off-diagonal matrix
C
C       1) Upper Part
C
      IAI=NOEL3-NOEL4
      IMI=IABS(IAI)+1
      IMIN=NOEL1+NOEL2+NOEL3+1
      IMAX=NOEL1+NOEL2+NOEL3+NOEL4
      DO 28 I=IMIN,IMAX
             IXN=NOEL1+NOEL2
             LXN=IABS(NS34)+I-IMIN+1
             IF (IAI.LT.0) GO TO 29
                     KIMIN=I
                     KIMAX=IMAX
                     GO TO 30
   29        KIMIN=I
             KIMAX=I+NOEL3-1
             IF ((I-IMIN+1).GE.IMI) KIMAX=IMAX
   30        DO 31 KI=KIMIN,KIMAX
                    IXN=IXN+1
```

```
                 BMATR(IXN,KI)=Z34(LXN)
                 BMATR(KI,IXN)=BMATR(IXN,KI)
   31       CONTINUE
   28 CONTINUE
C
C   ...2)... Lower Part
C
      IMIN=NOEL1+NOEL2+2
      IMAX=NOEL1+NOEL2+NOEL3
      DO 32 I=IMIN,IMAX
            IXN=I-1
            LXN=IABS(NS34-I+IMIN-1)+1
            IF (IAI.GT.0) GO TO 33
                   KIMIN=NOEL1+NOEL2+NOEL3+1
                   KIMAX=NOEL1+NOEL2+2*NOEL3-(I-IMIN)-1
                   GO TO 34
   33       KIMIN=NOEL1+NOEL2+NOEL3+1
            KIMAX=NOEL1+NOEL2+NOEL3+NOEL4
            IIMI=I-IMIN+2
            IF (IIMI.GE.IMI) KIMAX=NOEL1+NOEL2+NOEL3+NOEL4-IIMI+IMI
   34       DO 35 KI=KIMIN,KIMAX
                  IXN=IXN+1
                  BMATR(IXN,KI)=Z34(LXN)
                  BMATR(KI,IXN)=BMATR(IXN,KI)
   35       CONTINUE
   32 CONTINUE
C
C     ....1.... Second off-diagonal matrix
C
C        1)    Upper Part
C
      NS13=NS12+NS23
      IAI=NOEL1-NOEL3
      IMI=IABS(IAI)+1
      IMIN=NOEL1+NOEL2+1
      IMAX=NOEL1+NOEL2+NOEL3
      DO 36 I=IMIN,IMAX
            IXN=0
            LXN=NS13+I-IMIN
            IF (IAI.LT.0) GO TO 37
                  KIMIN=I
                  KIMAX=IMAX
                  GO TO 38
  37        KIMIN=I
            KIMAX=I+NOEL1-1
            IF ((I-IMIN+1).GE.IMI) KIMAX=IMAX
  38        DO 39 KI=KIMIN,KIMAX
                  IXN=IXN+1
                  BMATR(IXN,KI)=Z13(LXN)
                  BMATR(KI,IXN)=BMATR(IXN,KI)
  39        CONTINUE
  36  CONTINUE
C
C         2)    Lower part
C
      IMIN=2
      IMAX=NOEL1
      DO 40 I=IMIN,IMAX
            IXN=I-1
            LXN=IABS(NS13-I)+1
```

```
            IF (IAI.GT.0) GO TO 41
                  KIMIN=NOEL1+NOEL2+1
                  KIMAX=2*NOEL1+NOEL2-I+IMIN-1
                  GO TO 42
   41       KIMIN=NOEL1+NOEL2+1
            KIMAX=NOEL1+NOEL2+NOEL3
            IIMI=I-IMIN+2
            IF (IIMI.GE.IMI) KIMAX=NOEL1+NOEL2+NOEL3-IIMI+IMI
   42       DO 43 KI=KIMIN,KIMAX
                 IXN=IXN+1
                 BMATR(IXN,KI)=Z13(LXN)
                 BMATR(KI,IXN)=BMATR(IXN,KI)
   43       CONTINUE
   40  CONTINUE
C
C     ...2... Second off-diagonal matrix
C
C     1)  Upper Part
C
       NS24=NS23+NS34
       IAI=NOEL2-NOEL4
       IMI=IABS(IAI)+1
       IMIN=NOEL1+NOEL2+NOEL3+1
       IMAX=NOEL1+NOEL2+NOEL3+NOEL4
       DO 44 I=IMIN,IMAX
            IXN=NOEL1
            LXN=IABS(NS24)+I-IMIN+1
            IF (IAI.LT.0) GO TO 45
                  KIMIN=I
                  KIMAX=IMAX
                  GO TO 46
   45       KIMIN=I
            KIMAX=I+NOEL2-1
            IF ((I-IMIN+1).GE.IMI) KIMAX=IMAX
   46       DO 47 KI=KIMIN,KIMAX
                 IXN=IXN+1
                 BMATR(IXN,KI)=Z24(LXN)
                 BMATR(KI,IXN)=BMATR(IXN,KI)
   47       CONTINUE
   44  CONTINUE
C
C      2) Lower Part
C
       IMIN=NOEL1+2
       IMAX=NOEL1+NOEL2
       DO 48 I=IMIN,IMAX
            IXN=I-1
            LXN=IABS(NS24-I+IMIN-1)+1
            IF (IAI.GT.0) GO TO 49
                  KIMIN=NOEL1+NOEL2+NOEL3+1
                  KIMAX=NOEL1+2*NOEL2+NOEL3-I+IMIN-1
                  GO TO 50
   49       KIMIN=NOEL1+NOEL2+NOEL3+1
            KIMAX=NOEL1+NOEL2+NOEL3+NOEL4
            IIMI=I-IMIN+2
            IF (IIMI.GE.IMI) KIMAX=NOEL1+NOEL2+NOEL3+NOEL4-IIMI+IMI
   50       DO 51 KI=KIMIN,KIMAX
                 IXN=IXN+1
                 BMATR(IXN,KI)=Z24(LXN)
                 BMATR(KI,IXN)=BMATR(IXN,KI)
```

```
   51         CONTINUE
   48   CONTINUE
  301   CONTINUE
C
C   ...1... Third off-diagonal matrix
C
C     1) Upper part
C
        NS14=NS12+NS23+NS34
        IAI=NOEL1-NOEL4
        IMI=IABS(IAI)+1
        IMIN=NOEL1+NOEL2+NOEL3+1
        IMAX=NOEL1+NOEL2+NOEL3+NOEL4
        DO 52 I=IMIN,IMAX
              IXN=0
              LXN=NS14+I-IMIN
              IF (IAI.LT.0) GO TO 53
                     KIMIN=I
                     KIMAX=IMAX
                     GO TO 54
   53         KIMIN=I
              KIMAX=I+NOEL1-1
              IF ((I-IMIN+1).GE.IMI) KIMAX=IMAX
   54         DO 55 KI=KIMIN,KIMAX
                    IXN=IXN+1
                    BMATR(IXN,KI)=Z14(LXN)
                    BMATR(KI,IXN)=BMATR(IXN,KI)
   55         CONTINUE
   52   CONTINUE
C
C    2)  Lower Part
C
        IMIN=2
        IMAX=NOEL1
        DO 56 I=IMIN,IMAX
              IXN=I-1
              LXN=IABS(NS14-I)+1
              IF (IAI.GT.0) GO TO 57
                     KIMIN=NOEL1+NOEL2+NOEL3+1
                     KIMAX=2*NOEL1+NOEL2+NOEL3-I+IMIN-1
                     GO TO 58
   57         KIMIN=NOEL1+NOEL2+NOEL3+1
              KIMAX=NOEL1+NOEL2+NOEL3+NOEL4
              IIMI=I-IMIN+2
              IF (IIMI.GE.IMI) KIMAX=NOEL1+NOEL2+NOEL3+NOEL4-IIMI+IMI
   58         DO 59 KI=KIMIN,KIMAX
                    IXN=IXN+1
                    BMATR(IXN,KI)=Z14(LXN)
                    BMATR(KI,IXN)=BMATR(IXN,KI)
   59         CONTINUE
   56   CONTINUE
C
C
C       IMIN=1
C       IMAX=NOEL1+NOEL2+NOEL3+NOEL4
C       JMIN=1
C       DO 60 K=1,4
C             DO 61 I=1,IMAX
C                   IF (K.EQ.1) WRITE (6,62) (BMATR(I,J),J=1,12)
C                   IF (K.EQ.2) WRITE (6,63) (BMATR(I,J),J=13,17)
```

```
C                 IF (K.EQ.3) WRITE (6,64)  (BMATR(I,J),J=18,21)
C                 IF (K.EQ.4) WRITE (6,65)  (BMATR(I,J),J=22,33)
C 62              FORMAT(12(I4,2X)/)
C 63              FORMAT(5(I4,2X)/)
C 64              FORMAT(4(I4,2X)/)
C 65              FORMAT(12(I4,2X)/)
C 61          CONTINUE
C 60    CONTINUE
C
C       GO TO 1000
C
 1001 CALL MINVCD (NOR,NOR,DETA,IB,IA)
C
        DO 70 IQQ=1,NOR
C             CUR(IQQ,1)=(BMATR(IQQ,NFEED1)+BMATR(IQQ,NFEED4))/100.00
C             CUR(IQQ,2)=(BMATR(IQQ,NFEED1)-BMATR(IQQ,NFEED4))/100.00
              CUR(IQQ,3)=BMATR(IQQ,NFEED1)/100.00
   70 CONTINUE
C
        WRITE (6,71)
   71   FORMAT (///10X,'Current distribution on the strip conductors',
      *////)
        IMIN=1
        IMAX=NOEL4+NOEL3+NOEL2+NOEL1
        DO 72 K=3,3
              IF (K.EQ.1) WRITE(6,73)
              IF (K.EQ.2) WRITE(6,74)
              IF (K.EQ.3) WRITE(6,75)
   73         FORMAT(////10X,'Current Distribution for EVEN excitation'
      *               ,////)
   74         FORMAT(////10X,'Current Distribution for ODD excitation'
      *               ,////)
   75         FORMAT(////10X,'Current Distribution for UNEVEN excitat.'
      *               ,////)
              DO 76 IQQ=IMIN,IMAX
                    RECUR1=REAL(CUR(IQQ,K))
                    ABCU1=CABS(CUR(IQQ,K))
                    AICUR1=AIMAG(CUR(IQQ,K))
                    PHCUR1=ATAN2(AICUR1,RECUR1)
                    PHCUR1=180.00*PHCUR1/PI
                    IF (IQQ.EQ.1) WRITE (6,77)
   77               FORMAT(///10X,'Current on the first Transmiss. line'
      *             ,///)
                    IF (NOEL23.EQ.0) GO TO 302
                    IF (IQQ.EQ.(NOEL1+1)) WRITE (6,78)
   78               FORMAT(///10X,'Current on the first parasit.dipole'
      *             ,///)
                    IF (IQQ.EQ.(NOEL1+NOEL2+1)) WRITE (6,79)
   79               FORMAT(///10X,'Current on the second paras. dipole'
      *             ,///)
  302               CONTINUE
                    IF (IQQ.EQ.(NOEL1+NOEL2+NOEL3+1)) WRITE (6,80)
   80               FORMAT(///10X,'Current on the printed dipole'
      *             ,///)
                    WRITE (6,81) IQQ,CUR(IQQ,K),ABCU1,PHCUR1
   81               FORMAT (1X,'C(',I4,')=',(E14.7,',',E14.7),2X,
      *               E14.7,1X,E14.7/)
   76         CONTINUE
   72   CONTINUE
 1000 CONTINUE
```

```
      STOP
      END
C***********************************************************************
C     THIS SUBROUTINE INVERTS A SQUARE COMPLEX MATRIX
C***********************************************************************
      SUBROUTINE MINVCD (IA,MA,DETA,IR,IC)
      IMPLICIT REAL*4 (A-H,O-Z)
      COMPLEX*8  A,PIV,DETA,TEMP,PIV1
      DIMENSION IR(MA),IC(MA)
      COMMON/MAN/A(160,160)
      DO 1 I=1,MA
      IR(I)=0
    1 IC(I)=0
C     DETA=(1.00,0.00)
      S=0.00
      R=MA
    2 CALL SUBMCD(IA,IA,MA,MA,IR,IC,I,J)
      PIV=A(I,J)
C     DETA=PIV*DETA
      Y=CABS(PIV)
      IF (Y.EQ.0) GO TO 17
      IR(I)=J
      IC(J)=I
      PIV=(1.00,0.00)/PIV
      A(I,J)=PIV
      DO 5 K=1,MA
    5 IF (K.NE.J) A(I,K)=A(I,K)*PIV
      DO 9 K=1,MA
      IF (K.EQ.I) GO TO 9
      PIV1=A(K,J)
    6 DO 8 L=1,MA
    8 IF (L.NE.J) A(K,L)=A(K,L)-PIV1*A(I,L)
    9 CONTINUE
      DO 11 K=1,MA
   11 IF (K.NE.I) A(K,J)=-PIV*A(K,J)
      S=S+1.00
      IF (S.LT.R) GO TO 2
   12 DO 16 I=1,MA
      K=IC(I)
      M=IR(I)
      IF (K.EQ.I) GO TO 16
C     DETA=-DETA
      DO 14 L=1,MA
      TEMP=A(K,L)
      A(K,L)=A(I,L)
   14 A(I,L)=TEMP
      DO 15 L=1,MA
      TEMP=A(L,M)
      A(L,M)=A(L,I)
   15 A(L,I)=TEMP
      IC(M)=K
      IR(K)=M
   16 CONTINUE
      RETURN
   17 WRITE (6,18)I,J
   18 FORMAT (10X,'MATRIX IS SINGULAR'/10X,'I=',I4,5X,'J=',I4)
      RETURN
      END
C***********************************************************************
C.......................................................................
```

```
C**********************************************************************
      SUBROUTINE SUBMCD(IA,JA,MA,NA,IR,IC,I,J)
      IMPLICIT REAL*4 (A-H,O-Z)
      COMPLEX*8  A
      DIMENSION IR(MA),IC(NA)
      COMMON/MAN/A(160,160)
      I=0
      J=0
      TEST=0.00
      DO 5 K=1,MA
      IF (IR(K).NE.0) GO TO 5
      DO 4 L=1,NA
      IF (IC(L).NE.0) GO TO 4
      X=CABS(A(K,L))
      IF(X.LT.TEST) GO TO 4
      I=K
      J=L
      TEST=X
    4 CONTINUE
    5 CONTINUE
      RETURN
      END
```

PART D

" Magnetic Tape Comments"

Tape name= *T* 16:18:59 4 Jan. 1986

ANSI labeled 6250-bpi 9TP Volume= BANDW2 Owner= PISTIKATEHI

Rack#= C9366J LP=on BLK=in DTCHK=on RETRY=10

File	Data set name	Block count	Record count	Tapelen (feet)	Record format	Expires
1	JOBRUN	20	116	1.74	FB(500,80)	31 Dec. '99
2	FOPTION 0	34	199	2.17	FB(500,80)	31 Dec. '99
3	FOPTION1	64	2357	5.28	FB(3000,80)	31 Dec. '99
4	FOPTION2	31	367	2.27	FB(1000,80)	31 Dec. '99
5	FOPTION3	45	529	2.82	FB(1000,80)	31 Dec. '99
6	FOPTION4	39	458	2.58	FB(1000,80)	31 Dec. '99
7	FOPTION5	32	384	2.32	FB(1000,80)	31 Dec. '99
8	FOPTION6	26	307	2.07	FB(1000,80)	31 Dec. '99
9	FOPTION7	52	622	3.11	FB(1000,80)	31 Dec. '99

Total Tape Length = 24.37 feet