

TR-O-0030

39

有限要素法による損失のある2次元導波路の  
特性解析プログラムおよびその解析法を  
用いたMMIC用低損失線路の検討

角田紀久夫 豊田一彦

1990. 7. 4.

ATR光電波通信研究所

## 概要 (Abstract)

電磁界の数値解析手法はマイクロ波回路の設計において有力な道具となっている。これらソフトウェアの充実を計るため各種解析プログラムの自作及び共同開発を進めている。

この度、有限要素法による損失を有する2次元導波路解析プログラムを作成した。この解析プログラムでは、伝送線路を構成する金属や誘電体の損失を含んだ複素伝搬定数を固有値とした解析法となっている。

さらに本解析法を用いて、MMIC用低損失伝送線路の検討を行った。

この報告書では、解析法の概要説明ならびにMMIC用低損失伝送線路の検討結果に加え、プログラムマニュアルとしての機能を持たせるためプログラムの全体構成及び使用方法、プログラムリストの内容について記述している。

## 発行時 配付先 (Initial Distribution Specifications)

## 備考 (Notes)

## 目次

1	まえがき	1
2	有限要素法による線路解析	2
2.1	ガラーキン方程式	2
2.2	ベクトル形状関数	4
2.3	線路定数の計算方法	5
3	多層化MMIC用伝送線路の解析	7
3.1	低損失スロット線路の検討	7
3.2	低損失マイクロストリップ線路の検討	10
3.3	まとめ	15
4	解析プログラムの使用方法およびプログラムコードの説明	16
4.1	プログラムシステムの概要	16
4.2	データフォーマット	17
4.3	プログラム使用方法	17
4.4	プログラムコード	25
5	まとめ	73

## 1. まえがき

ATR光電波通信研究所では、将来の移動体通信にむけて超小型無線機や高機能アンテナの実現などを主な目的としてモノリシックマイクロ波集積回路(MMIC)の研究を行っている。MMICとはガリウムヒ素などの半導体基板上にトランジスタなどの能動素子と、抵抗・容量・インダクタなどから成る受動回路を半導体プロセス技術によって一括作成するマイクロ波集積回路の総称である。MMICは、薄膜受動回路に個別のトランジスタ・容量などを実装した従来の混成マイクロ波集積回路(MIC)に比べて小型かつ信頼性が高く、電気的特性や量産性にも優れているという特徴をもつ。

ところで従来のMMICの研究開発は、MIC設計技術を高集積化・高精度化することが主であり、FETなどの能動デバイスと受動回路は別個のものとして扱われてきた。従来のMMICは半導体基板上にこれらの回路を平面的に配置する構造であって、一般に受動回路は能動デバイスに比べて大きい。このことがMMIC全体を小型化するうえでの制約事項となっていた。

このような技術状況を踏まえ、当研究所では、MMICの大幅な小型化および高機能化を目指した新しい設計コンセプトの提案を行い<sup>1)</sup>、その実現のための設計技術の確立を目指して研究を進めている。

当研究所では、MMICの大幅な小型化・高機能化を実現するためにはMMICの立体構造化が不可欠と考え、その実現を研究目標としている。そのために、共平面小型化MMIC・多層化MMIC・立体化MMICという技術的マイルストーンを設定しており、現在の研究段階は多層化MMICの基礎検討の段階である。

多層化MMICの実現のためには多層・多媒質構造を用いたマイクロ波伝送線路の高周波特性を精密に解析することが重要な課題となる。しかしながら、市販のマイクロ波回路設計用ソフトウェアには電磁界をベクトル波動として精密に取り扱ったものはほとんどなく、また、MMICのように導体の膜厚が表皮厚に比べて十分に大きくない場合には市販のソフトウェアの適用は困難である。

本研究では、多層・多媒質構造を用いたマイクロ波伝送線路の高周波特性を解析するため、ガラキン法に基づいた有限要素法による線路の電磁界解析プログラムを作成した。この線路解析プログラムは、電磁波の伝搬方向に一様な閉じた損失のある伝送線路を解析することができ、伝送モードの位相定数・減衰定数・波長短縮率・実効誘電率・特性インピーダンスを求めることができる。さらに、本解析プログラムを用いて多層化MMIC用伝送線路の損失特性を解析し、新しい低損失伝送線路構造を提案する。

以下、第2章では本解析プログラムの基となる理論と本手法の特徴について述べる。第3章では種々の多層化MMIC用の伝送線路について本解析プログラムにより数値計算を行った結果について述べる。第4章は、本解析プログラムの使用マニュアル及びプログラムコードの説明を記す。

## 参考文献

- [1] 徳満、「ATRにおける高周波回路の研究 ―超小型・高機能MMIC―」、ATRテクニカルレポート、TR-O-0023、1990年3月

## 2.有限要素法による線路解析

有限要素法(FEM)は境界要素法(BEM)とならんで任意断面形状の線路解析に適用できる数値解析法である。有限要素法はとくに線路が多く媒質によって構成されていたり、不均質な媒質を含むような場合に有用である。

有限要素法による伝送線路のベクトル波解析は従来から行われているが、実際には存在しないようなモードに対する解、すなわちスプリアス解が現れるという欠点があった。これに対し、松原らは電磁界の横成分を用いた変分表現式を提案し、それをベクトル要素を用いた有限要素法により解いた。この変分表現式は電磁界の横成分を用いた伝搬定数 $\beta$ の変分表現式であり、周波数に対して $\beta$ が直接求められるという特徴を持ち、その解析結果はスプリアス解を容易に除去できた。しかしながら、この変分表現式は線路に損失や利得がないことを条件として導かれたものであり、損失のある線路や純虚数の伝搬定数を持つエバネッセントモードの解析には適用できない。そこで松原らはさらにこの変分表現式を指標として、それをより一般化したガラーキン方程式を導き、このガラーキン方程式を有限要素法を用いて数値解析する方法を提案した<sup>[1]</sup>。このガラーキン方程式は不均質媒質・異方性媒質および分散性媒質からなる線路に対して成立するのはもちろんであるが、損失や利得のある線路に対しても成立する。

本章ではこのガラーキン方程式に基づいた有限要素法について簡単に説明を行い、また伝送線路の特性計算手順について簡単に記す。

### 2.1ガラーキン方程式

本節ではガラーキン法に基づいた有限要素法の基本式を示す。簡単のために等方性媒質から成る線路について定式化する。

まず、z軸方向に一様な線路の横断面を有限個の三角形要素に分割する。この時、電磁界を横成分 $e_t, h_t$ と縦成分 $e_z, h_z$ に分けると、Maxwellの方程式より次の微分方程式が得られる。

$$\omega \epsilon e_t + j \nabla \times h_z + \beta i_z \times h_t = 0 \quad (1a)$$

$$\omega \mu h_t - j \nabla \times e_z - \beta i_z \times e_t = 0 \quad (1b)$$

ここで $\omega$ は角周波数、 $\epsilon$ および $\mu$ はそれぞれ媒質の誘電率および透磁率、 $i_z$ はz軸方向の単位ベクトル、 $\beta$ は伝搬定数、 $j$ は虚数単位を表す。

また、境界条件は次式となる。

要素と要素の境界において

$$i_z \cdot (n \times e_t) = \text{連続} \quad i_z \cdot e_z = \text{連続} \quad (2a)$$

$$i_z \cdot (n \times h_t) = \text{連続} \quad i_z \cdot h_z = \text{連続} \quad (2b)$$

電気壁との境界において

$$i_z \cdot (n \times e_t) = 0 \quad i_z \cdot e_z = 0 \quad (2c)$$

磁気壁との境界において

$$i_z \cdot (n \times h_t) = 0 \quad i_z \cdot h_z = 0 \quad (2d)$$

インピーダンス壁との境界において

$$i_z \cdot (n_o \times e_t) = Z_t i_z \cdot h_z \quad (2e)$$

$$i_z \cdot e_z = -Z_z i_z \cdot (n_o \times h_t) \quad (2f)$$

電磁界の縦成分 $e_z$ および $h_z$ は横成分 $e_t$ および $h_t$ を用いて次式で表される。

$$j \omega \epsilon e_z = \nabla \times h_t \quad (3a)$$

$$j\omega\mu\mathbf{h}_z = -\nabla \times \mathbf{e}_t \quad (3b)$$

ここで $\mathbf{n}$ は境界に立てた単位法線ベクトル、 $\mathbf{n}_0$ はインピーダンス壁に向かう向きの単位法線ベクトルである。 $Z_t$ および $Z_z$ はインピーダンス壁の壁面インピーダンスであり、導電率 $\sigma$ の良導体の場合、次式で表される。

$$Z_t = Z_z = (1+j) \sqrt{\omega\mu/2\sigma} \quad (4)$$

さて、未知関数 $\mathbf{e}_t$ および $\mathbf{h}_t$ が局所的で $\mathbf{i}_z$ に垂直な既知ベクトル関数(すなわち基底) $\mathbf{U}_m, \mathbf{V}_m$ を用いて、

$$\mathbf{e}_t(x,y) = \sum a_m \mathbf{U}_m(x,y) \quad (5a)$$

$$\mathbf{h}_t(x,y) = \sum b_m \mathbf{V}_m(x,y) \quad (5b)$$

と表されるものと仮定する。なお、 $a_m$ および $b_m$ は未知パラメータである。ここで、基底 $\mathbf{U}_m, \mathbf{V}_m$ は境界条件として次式を満足するものとする。

要素と要素の境界において

$$\mathbf{i}_z \cdot (\mathbf{n} \times \mathbf{U}_m) = \text{連続} \quad (6a)$$

$$\mathbf{i}_z \cdot (\mathbf{n} \times \mathbf{V}_m) = \text{連続} \quad (6b)$$

電気壁との境界において

$$\mathbf{i}_z \cdot (\mathbf{n} \times \mathbf{U}_m) = 0 \quad (6c)$$

磁気壁との境界において

$$\mathbf{i}_z \cdot (\mathbf{n} \times \mathbf{V}_m) = 0 \quad (6d)$$

次に式(1)-(4)にガラーキン法を適用する。まず、式(1a)および(1b)にそれぞれ $\mathbf{U}_m$ および $\mathbf{V}_m$ を乗じ、各要素について面積分し、すべての要素について加え合わせ、さらにガウスの定理を用いて部分積分を実行すると次式を得る。

$$\sum \int \{ \omega \epsilon \mathbf{U}_m \cdot \mathbf{e}_t + j(\nabla \times \mathbf{U}_m) \cdot \mathbf{h}_z - \beta \mathbf{i}_z \cdot (\mathbf{U}_m \times \mathbf{h}_t) \} dS = j \sum \oint (\mathbf{n}_0 \times \mathbf{U}_m) \cdot \mathbf{h}_z dl \quad (7a)$$

$$\sum \int \{ \omega \mu \mathbf{V}_m \cdot \mathbf{h}_t - j(\nabla \times \mathbf{V}_m) \cdot \mathbf{e}_z + \beta \mathbf{i}_z \cdot (\mathbf{V}_m \times \mathbf{e}_t) \} dS = -j \sum \oint (\mathbf{n}_0 \times \mathbf{V}_m) \cdot \mathbf{e}_z dl \quad (7b)$$

ここで $\sum$ は全ての要素について和をとることを表し、 $\int dS$ は各要素についての面積分、 $\oint dl$ は各要素の境界についての線積分を表す。また、 $\mathbf{n}_0$ は各要素の境界に立てた外向き単位法線ベクトルである。式(2)および(6)の境界条件を用いると式(7)は次式となる。

$$\begin{aligned} \sum \int \{ \omega \epsilon \mathbf{U}_m \cdot \mathbf{e}_t + j(\nabla \times \mathbf{U}_m) \cdot \mathbf{h}_z - \beta \mathbf{i}_z \cdot (\mathbf{U}_m \times \mathbf{h}_t) \} dS \\ = j \int_C Z_t^{-1} (\mathbf{n}_0 \times \mathbf{U}_m) \cdot (\mathbf{n}_0 \times \mathbf{e}_t) dl \end{aligned} \quad (8a)$$

$$\begin{aligned} \sum \int \{ \omega \mu \mathbf{V}_m \cdot \mathbf{h}_t - j(\nabla \times \mathbf{V}_m) \cdot \mathbf{e}_z + \beta \mathbf{i}_z \cdot (\mathbf{V}_m \times \mathbf{e}_t) \} dS \\ = j \int_C Z_z (\mathbf{n}_0 \times \mathbf{V}_m) \cdot (\mathbf{n}_0 \times \mathbf{h}_t) dl \end{aligned} \quad (8b)$$

ここで、 $\int_C dl$ はインピーダンス壁にわたる線積分である。次に、式(3)および式(5)を式(8)に代入すると

$$Q_2 B = (1/\beta) P_1 A \quad (9a)$$

$$Q_1 A = (1/\beta) P_2 B \quad (9b)$$

なる行列方程式が得られる。ここでAおよびBは未知パラメータ $a_m$ および $b_m$ を要素とする列ベクトルであり、 $P_1, P_2, Q_1$ および $Q_2$ はそれぞれ $p_{1,mn}, p_{2,mn}, q_{1,mn}$ および $q_{2,mn}$ を要素とする行列である。ここで、

$$p_{1,mn} = \sum \int \{ \omega \epsilon \mathbf{U}_m \cdot \mathbf{U}_n - (1/\omega \mu) (\nabla \times \mathbf{U}_m) \cdot (\nabla \times \mathbf{U}_n) \} dS - j \int_C Z_t^{-1} (\mathbf{n}_o \times \mathbf{U}_m) \cdot (\mathbf{n}_o \times \mathbf{U}_n) dl \quad (10a)$$

$$p_{2,mn} = \sum \int \{ \omega \mu \mathbf{V}_m \cdot \mathbf{V}_n - (1/\omega \epsilon) (\nabla \times \mathbf{V}_m) \cdot (\nabla \times \mathbf{V}_n) \} dS - j \int_C Z_z (\mathbf{n}_o \times \mathbf{V}_m) \cdot (\mathbf{n}_o \times \mathbf{V}_n) dl \quad (10b)$$

$$q_{1,mn} = - \sum \int \mathbf{i}_z \cdot (\mathbf{V}_m \times \mathbf{U}_n) dS \quad (10c)$$

$$q_{2,mn} = \sum \int \mathbf{i}_z \cdot (\mathbf{U}_m \times \mathbf{V}_n) dS \quad (10d)$$

である。

式(9)の行列方程式よりAまたはBを消去して

$$(Q_2 P_2^{-1} Q_1) A = (1/\beta^2) P_1 A \quad (11a)$$

$$(Q_1 P_1^{-1} Q_2) B = (1/\beta^2) P_2 B \quad (11b)$$

なる行列の固有値問題を得る。式(11)の固有値問題を解いて固有値 $\beta$ (すなわち、伝搬定数 $\beta$ )および固有ベクトルA,B(すなわち、固有電磁界 $\mathbf{e}_t, \mathbf{h}_t, \mathbf{e}_z, \mathbf{h}_z$ )を決定することができる。

## 2.2ベクトル形状関数

式(6)の境界条件を満足する局所的な基底 $\mathbf{U}_m, \mathbf{V}_m$ を有限要素法を用いて構成する。図2-1に示す代表的な三角形要素内において $\mathbf{e}_t$ および $\mathbf{h}_t$ を2次のベクトル形状関数 $N_i, (i=1-12)$ を用いて次式で表わす。

$$\mathbf{e}_t(x,y) = \sum_{i=1}^{12} \phi_i \mathbf{N}_i(x,y) \quad (12a)$$

$$\mathbf{h}_t(x,y) = \sum_{i=1}^{12} \psi_i \mathbf{N}_i(x,y) \quad (12b)$$

ここでベクトル形状関数 $\mathbf{N}_i$ は2次のスカラー形状関数 $N_i$ を用いて次式で与えられる。

$$N_{2i-1} = \frac{n_{ij}}{\mathbf{i}_z \cdot (\mathbf{n}_{ki} \times \mathbf{n}_{ij})} N_i \quad (13a)$$

$$N_{2i} = \frac{n_{ki}}{\mathbf{i}_z \cdot (\mathbf{n}_{ij} \times \mathbf{n}_{ki})} N_i \quad (13b)$$

$$N_{i+6} = (\mathbf{i}_z \times \mathbf{n}_{ij}) N_{i+3} \quad (13c)$$

$$N_{i+9} = n_{ij} N_{i+3} \quad (13d)$$

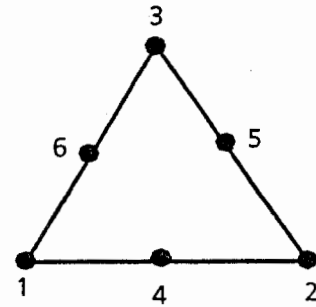


図2-1 三角形要素

また、 $\phi_i, \psi_i$ は未知パラメータである。式(13)において、 $(i,j,k)$ は $(1,2,3), (2,3,1)$ および $(3,1,2)$ とし $n_{12}, n_{23}$ および $n_{31}$ はそれぞれ辺(1-2), 辺(2-3)および辺(3-1)に立てた単位法線ベクトルである。各要素内における $\mathbf{e}_t, \mathbf{h}_t$ を式(12)の形で表わし、それらをまとめて導波路全体にわたる $\mathbf{e}_t, \mathbf{h}_t$ を構成するとそれは自動的に式(5)となる。ただし、 $\mathbf{e}_t, \mathbf{h}_t$ が式(6)と等価な境界条件

要素と要素の境界において

$$\mathbf{i}_z \cdot (\mathbf{n} \times \mathbf{e}_t) = \text{連続} \quad (14a)$$

$$\mathbf{i}_z \cdot (\mathbf{n} \times \mathbf{h}_t) = \text{連続} \quad (14b)$$

電気壁との境界において

$$\mathbf{i}_z \cdot (\mathbf{n} \times \mathbf{e}_t) = 0 \quad (14c)$$

磁気壁との境界において

$$\mathbf{i}_z \cdot (\mathbf{n} \times \mathbf{h}_t) = 0 \quad (14d)$$

を満足するように各要素のパラメータ  $\phi_i, \psi_i$  を調整するものとする。このとき、 $\phi_i, \psi_i$  はそれぞれ式(5)の  $a_m$  および  $b_m$  に対応することになり、式(5)の基底  $\mathbf{U}_m, \mathbf{V}_m$  はベクトル形状関数  $\mathbf{N}_i$  で表現されることになる。

### 2.3 線路定数の計算方法

本解析プログラムでは、損失として複素誘電率または壁面インピーダンスを与えることにより損失のある線路の位相定数、減衰定数、実効誘電率、波長短縮率、特性インピーダンスを求めることができる。

本節では前節までに示した手法により求めた固有値  $\beta$  および固有ベクトルより各線路定数を求める方法を示す。

#### ○位相定数および減衰定数

行列の固有値問題式(11)の固有値  $\beta$  が求める複素伝搬定数  $\beta$  に対応する。(ただし、実際の固有値問題の解として得られるのは真空中の波数  $k_0$  によって規格化された値、すなわち、 $\beta/k_0$  である。) この複素伝搬定数  $\beta$  の実部が位相定数、虚部が減衰定数に対応する。

#### ○実効誘電率および波長短縮率

実効誘電率  $\epsilon_{\text{eff}}$  は真空中の波数を  $k_0$  として

$$\epsilon_{\text{eff}} = (\text{Re}[\beta]/k_0)^2 \quad (15)$$

で与えられる。また、波長短縮率  $\lambda_g/\lambda_0$  は

$$\lambda_g/\lambda_0 = k_0/\text{Re}[\beta] \quad (16)$$

で与えられる。

#### ○特性インピーダンス

特性インピーダンスの定義はいくつかあるが、本プログラムでは2導体系における伝送電力  $P$  と導体に流れる電流  $I$  の比、すなわち、

$$Z_0 = P/|I|^2 \quad (17)$$

として特性インピーダンス  $Z_0$  を定義する。伝送電力  $P$  はポインティングベクトルの面積分として

$$P = \int (\mathbf{e}_t \times \mathbf{h}_t^*) \cdot \mathbf{i}_z \, dS \quad (18)$$

で与えられる。ここで、 $\int dS$  は線路断面全体にわたる面積分である。電流  $I$  は、導体を囲む閉曲線  $c$  における磁界の接線成分の積分として



$$I = \sigma / j \omega \epsilon \int_C (\mathbf{n}_0 \times \mathbf{h}_t) \cdot \mathbf{i}_z dl \quad (19)$$

で与えられる。また特に、導体が式(4)で示されるような壁面インピーダンスをもつ良導体の場合

$$I = \int_C (\mathbf{n}_0 \times \mathbf{h}_t) \cdot \mathbf{i}_z dl \quad (20)$$

となる。

#### 参考文献

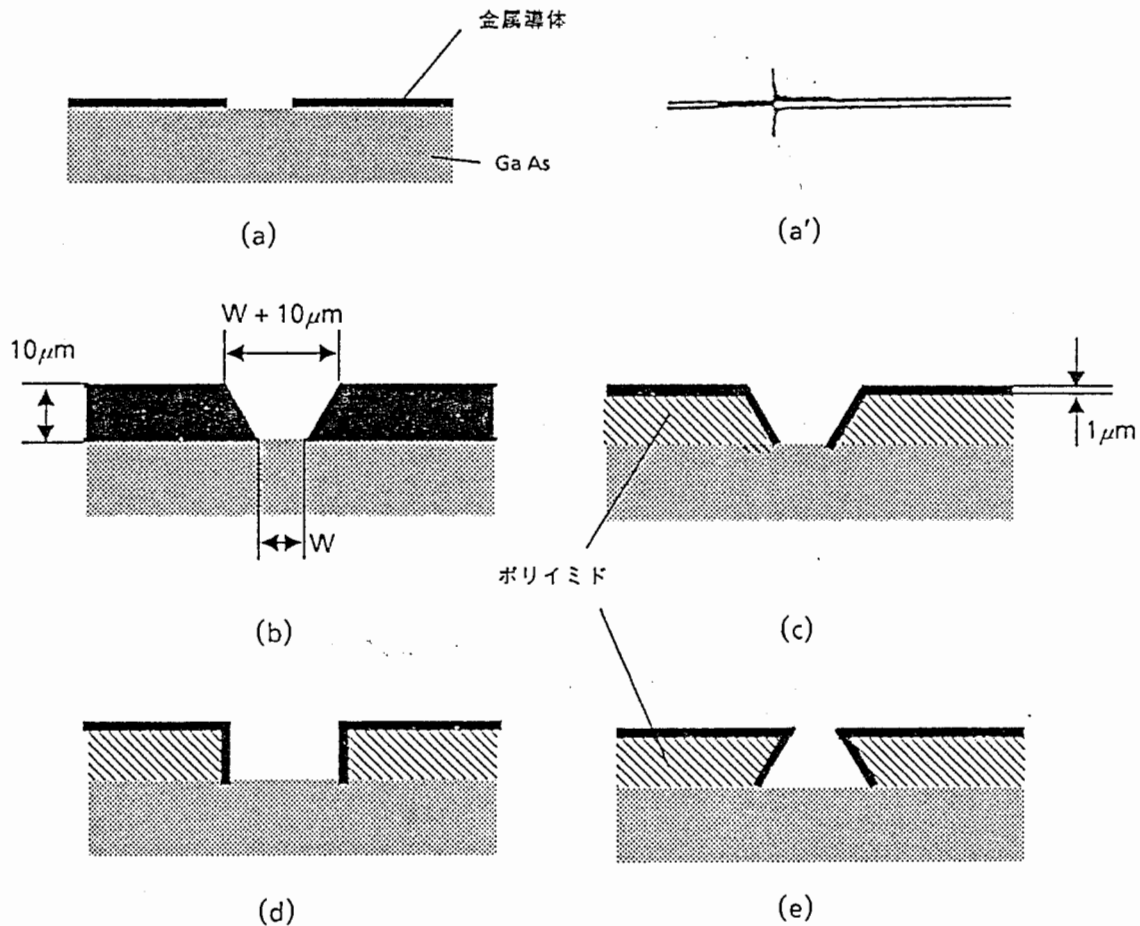
- [1]松原,アンケーオ:“損失・利得のある導波系の有限要素解析”,  
電子情報通信学会論文誌(C)、J71-C,10,pp.1398-1403(1988).

### 3. 多層化MMIC用伝送線路の解析

本章では、MMICに用いる伝送線路としてスロット線路とマイクロストリップ線路を取上げ、伝送損失を小さくする線路構造についてその損失値を定量的に検討した。

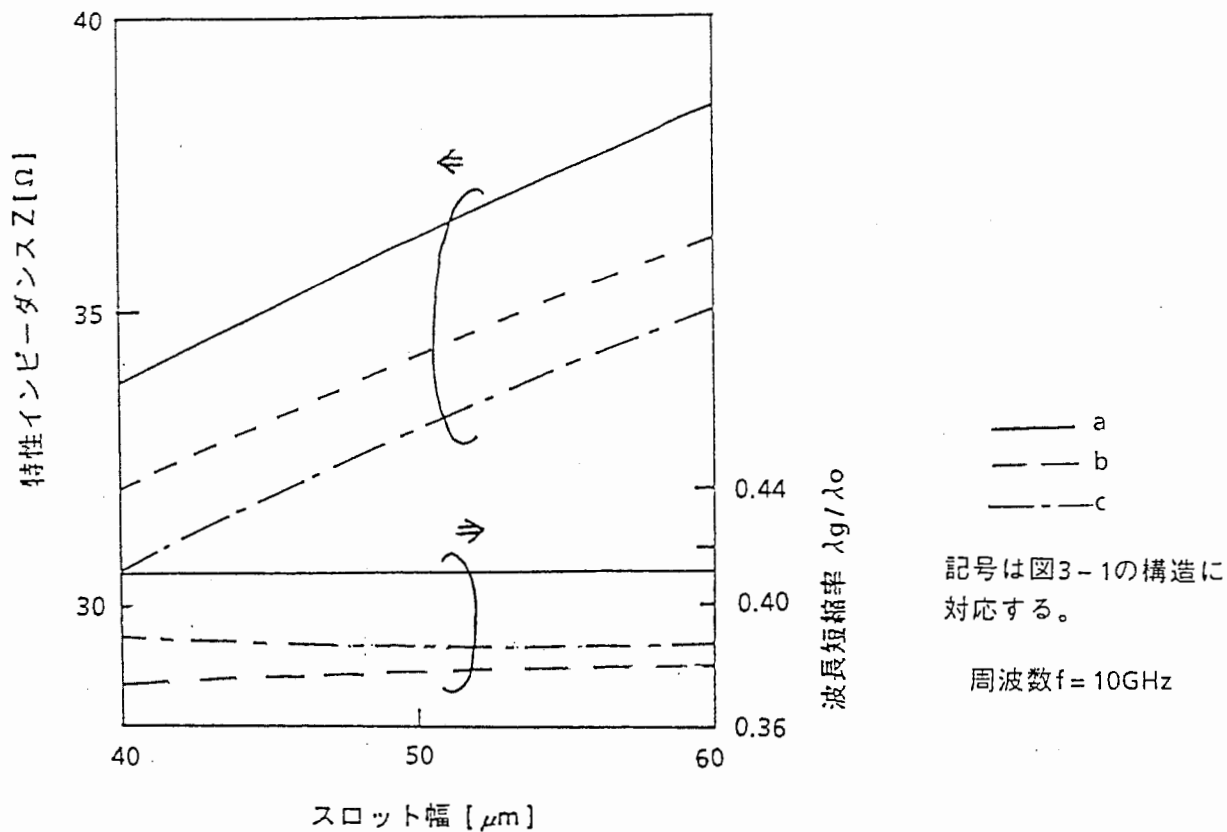
#### 3.1 低損失スロット線路の検討

通常のスロット線路は図3-1(a)に示すような構造を持つ。図3-1(a')は導体上の電流密度をヒストグラムとして表したものである。この図より、スロットを構成している導体の端部に電流が集中していることがわかる。導体により生じる損失は電流密度の二乗に比例するため、局所的に電流の集中が生じた場合は、電流が一様に分布している場合と比べ導体損失が非常に大きくなる。すなわちスロット部の電流集中を低減できれば導体損失を減らすことができる。そこで、スロットを構成する導体の向かい合う部分の面積を大きくして電磁界を広い範囲に分布させることにより電流の分散を図った線路構造、すなわち図3-1(b)-(e)に示すような構造の線路を提案し、その電気的特性を検討した。

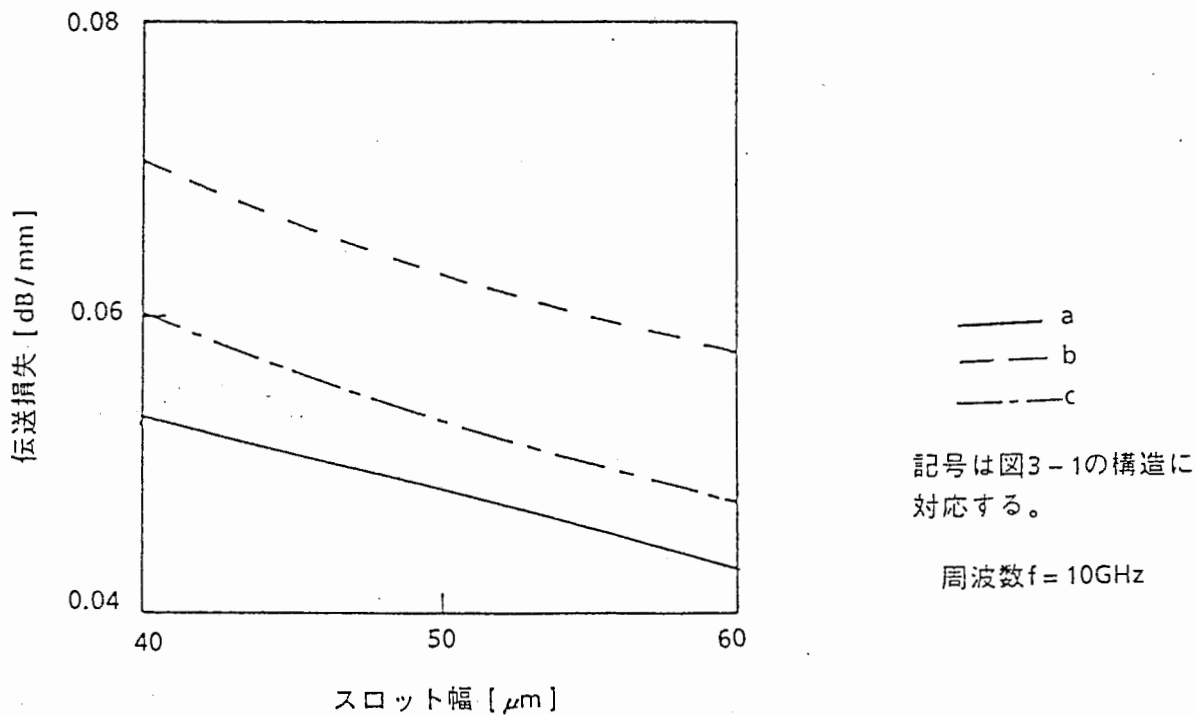


GaAs基板：比誘電率 12.9  
 ポリイミド層：比誘電率 3.3  
 金属導体：金 厚さ(c)-(e) 1 $\mu$ m、(b) 10 $\mu$ m

図 3-1 種々の形状のスロット線路



スロット幅  $W$  に対する特性インピーダンスおよび波長短縮率

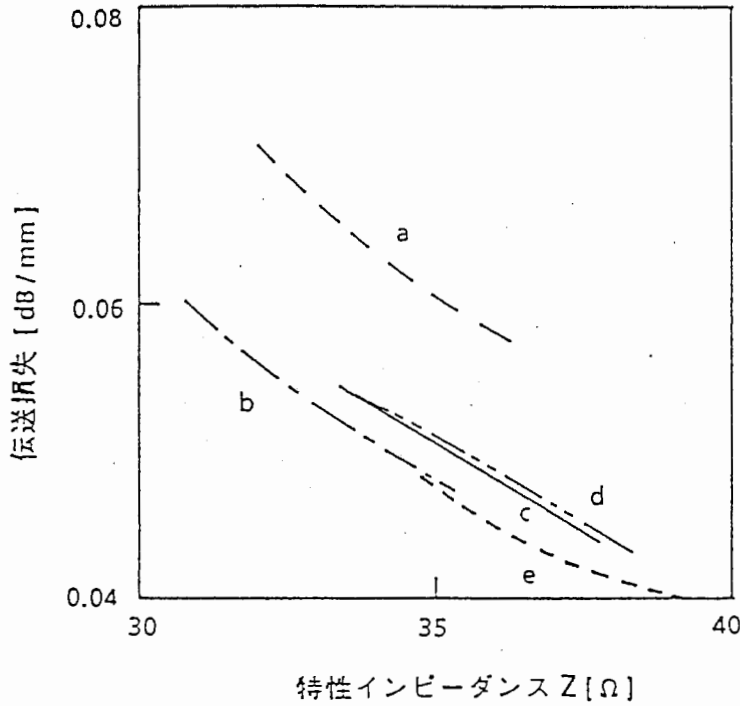


スロット幅  $W$  に対する挿入損失

図3-2 スロット線路の電気的特性

ここで数値計算において、GaAs基板の比誘電率は12.9、GaAs基板上に形成したポリイミド層の比誘電率を3.3、金属導体は金を想定し、その厚さは(c)-(e)の線路では $1\mu\text{m}$ 、(b)の線路では $10\mu\text{m}$ とした。以下の計算においては周波数を10GHzとしている。

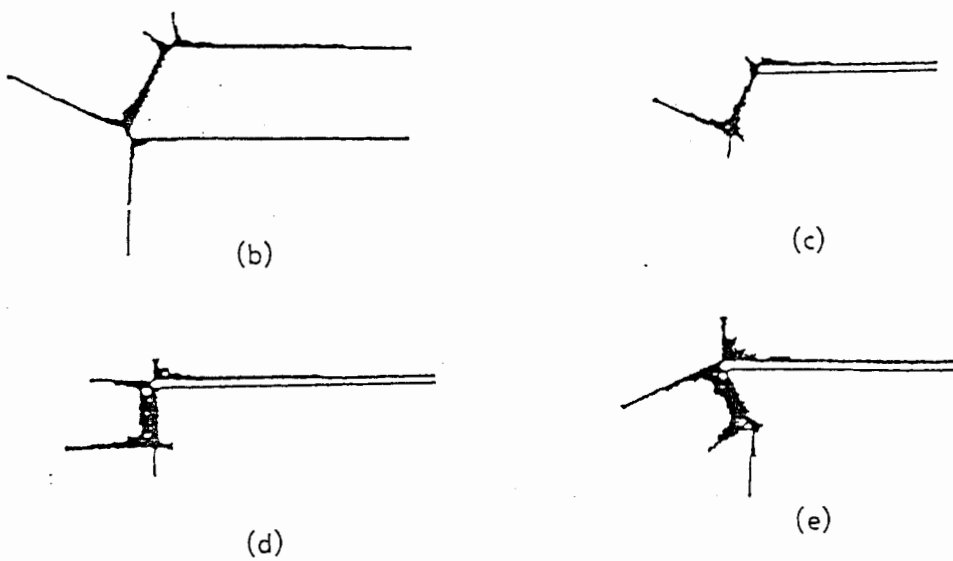
図3-2は、図3-1(a)-(c)の線路において、スロット幅 $W$ に対する特性インピーダンス・波長短縮率・伝送損失の計算値を示す。線路の伝送損失を考える場合、その特性インピーダンス $Z$ は重要なパラメータとなる。低インピーダンスの場合は電流量が大きく、また高インピーダンスの場合は逆に電流量が小さくなる。伝送損失値は総電流量の影響を受けるため、伝送損失は同じ特性インピーダンスの線路において比較しなければならない。



記号は図3-1の構造に対応する。

周波数 $f = 10\text{GHz}$

図3-3 スロット線路の伝送損失特性



記号は図3-1の構造に対応する。

図3-4 スロット線路における電流分布

図3-3は、図3-1(a)-(e)の線路における、特性インピーダンスに対する伝送損失特性を示している。図3-1(c)-(e)の線路では、損失特性が改善されたことがわかる。図3-4に電流分布を示すが、従来の構造の物と比べ、電流が分散されていることがわかる。以上より、スロットを構成する導体の向かい合う部分の面積を大きくした構造を持つ線路が低損失スロット線路となることを確認できた。

### 3.2 低損失マイクロストリップ線路の検討

図3-5(a)に示すような通常のマイクロストリップ線路では図3-5(b)に示すようにストリップ導体の縁端部に電流が集中することにより導体損失が大きくなる。(図3-5(b)にはマイクロストリップ線路の1/2領域を示した。)したがって、マイクロストリップ線路ではストリップ導体縁端部に集中する電流を分散することにより低損失化が図れる。ストリップ導体上の電流を分散させる方法として、何らかの方法で電流分布を一様分布に近づける方法と、ストリップ導体の幅を大きくして電流を分散させる方法が考えられる。以下では上記2方法について検討する。

#### 3.2.1 ストリップ導体の形状を変更したマイクロストリップ線路

まず、図3-6に示すような形状のマイクロストリップ線路について検討した。この線路はストリップ導体の端部を接地導体から遠ざけることにより電磁界の端部への集中を避け、電流の一様化をはかった線路である。導体を厚さ $1\mu\text{m}$ の金、誘電体膜は比誘電率5のSiONを想定し、その厚さは一番薄い部分が $3\mu\text{m}$ 、厚い部分が $4\mu\text{m}$ として計算した結果を図3-7に示す。また、図3-8に電流分布を示すが、電流の集中化が緩和されており、この線路では損失特性の改善がみられる。

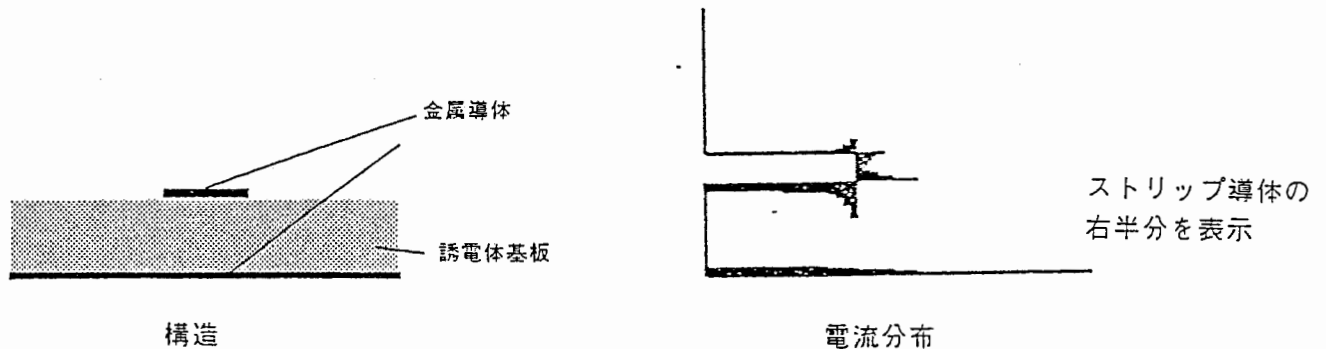


図3-5 マイクロストリップ線路

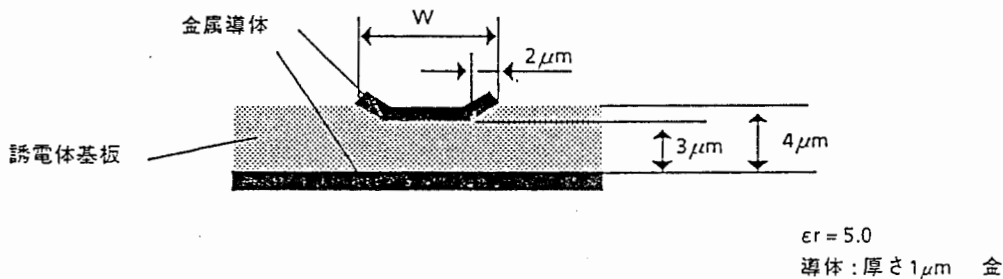
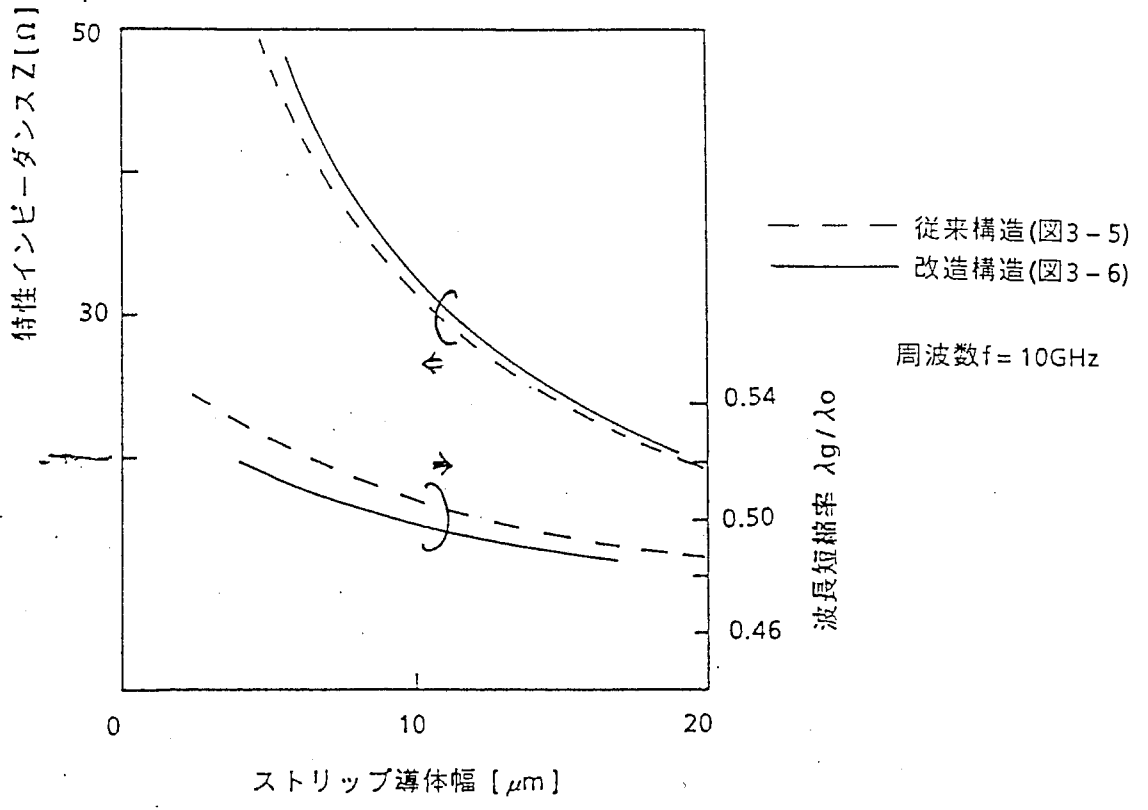


図3-6 マイクロストリップ線路の改良構造



ストリップ導体幅  $W$  に対する特性インピーダンスおよび波長短縮率

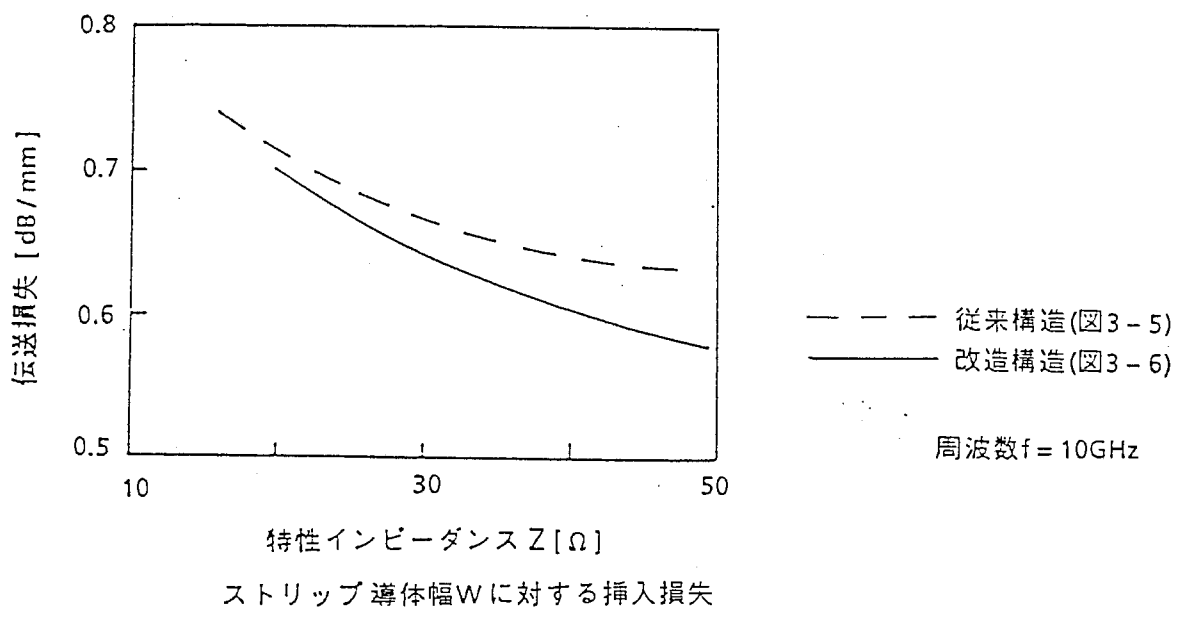
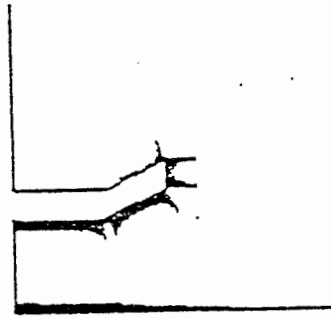


図3-7 マイクロストリップ線路の電気的特性



ストリップ導体の  
右半分を表示

図3-8 マイクロストリップ線路の電流分布

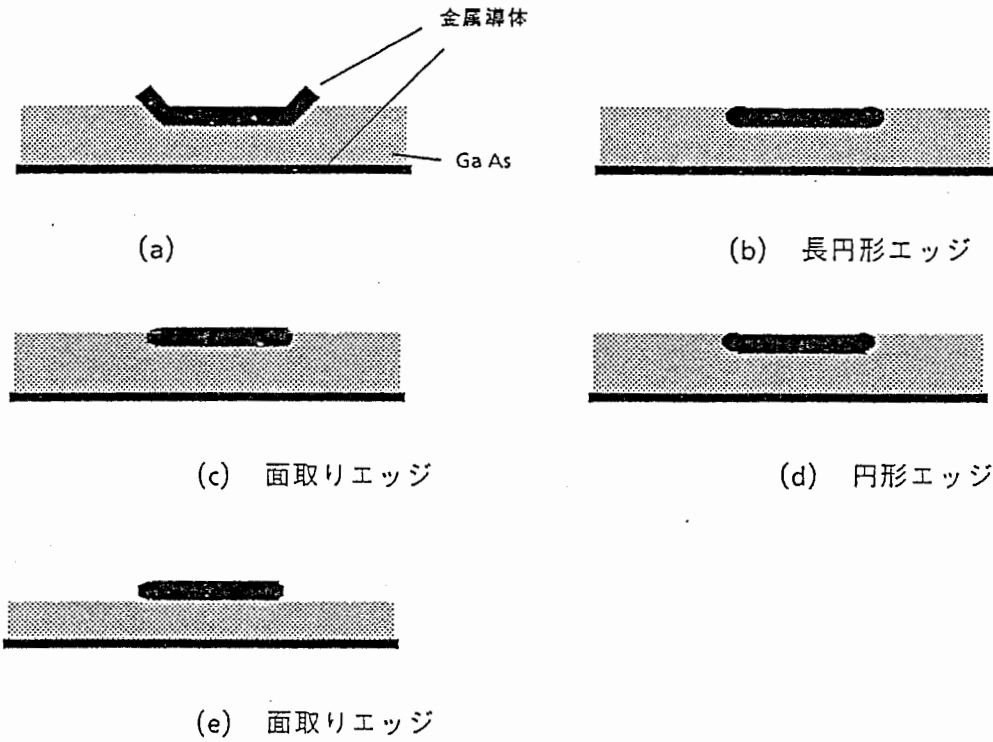


図3-9 種々のストリップ導体形状を持つ線路

表3-1 マイクロストリップ線路

線路形状	特性インピーダンス ( $\Omega$ )	波長短縮率 ( $\lambda_g/\lambda_o$ )	伝送損失 (dB/mm)
(a)	31.9	0.496	0.642
(b)	31.1	0.495	0.701
(c)	31.6	0.502	0.677
(d)	31.2	0.495	0.700
(e)	32.4	0.514	0.648

次にストリップ導体縁端部の角の部分への電磁界の集中、いわゆるエッジ効果を緩和することを目的として、図3-9(b)-(e)に示すように、ストリップ導体縁端部を面取りもしくは丸みづけしたマイクロストリップ線路の伝送損失について検討した。計算結果を表3-1に示す。これより、ストリップ導体縁端部の形状変更では伝送損失改善効果は小さい。しかし図3-9(e)に示すように、ストリップ導体縁端部が誘電体から離れて空気部に配置されたものは、先に検討した図3-9(a)の形状の物と同等な改善効果が確認された。

### 3.2.2 ストリップ導体の幅を大きくしたマイクロストリップ線路<sup>1)</sup>

本節ではストリップ導体の幅を大きくすることによる低損失化の方法について検討する。図3-10(b)の線路では誘電体膜がストリップ導体の下部にのみ形成されている。このため線路全体としての実効誘電率が低くなり、その結果特性インピーダンスが高くなる。従って、一定の特性インピーダンスを得るために従来のマイクロストリップ線路に比べてストリップ導体の幅を大きくとることができる。また、ストリップ導体の下部にのみ誘電体が形成されているため、電磁界が誘電率の高いストリップ導体の中央部に引き込まれ、その結果電流のストリップ導体端部への集中が緩和されるという効果も期待できる。

図3-11に上記線路における電気的特性の計算例を示す。導体は厚さ $1\mu\text{m}$ の金、誘電体膜は厚さ $10\mu\text{m}$ で比誘電率3.3のポリイミドとする。また、図3-10(b)の線路では誘電体はストリップ導体の縁端部から $d$ だけ内側に形成されているものとする。図3-11(a)はストリップ導体の幅 $W$ に対する特性インピーダンスおよび実効誘電率を示している。図より、導体幅を一定にした場合、線路の実効誘電率が下がり、特性インピーダンスが高くなることがわかる。従って、例えば $50\Omega$ で線路を設計する場合、従来の構造では $W/h$ は2程度であるが、 $d=5\mu\text{m}$ の新しい構造の場合 $W/h$ を3程度にすることができる。図3-11(b)に損失特性を示すが、同一の特性インピーダンスを持つ伝送線路を作成する場合、ストリップ導体の幅を大きくすることができるため、損失特性を改善することができる。 $50\Omega$ 以上の高インピーダンスの線路において損失改善の効果が大きく、ハイブリッド回路などで多用される $70\Omega$ の線路では約30%の改善効果がある。

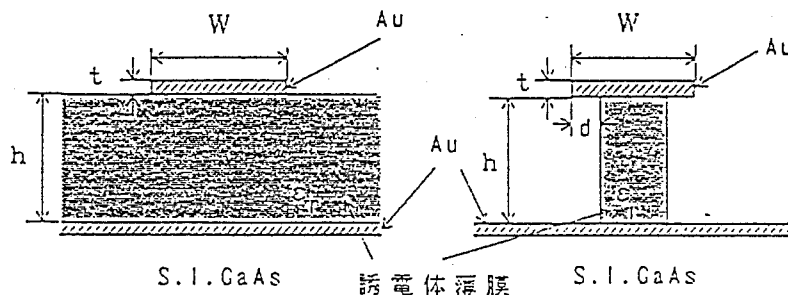
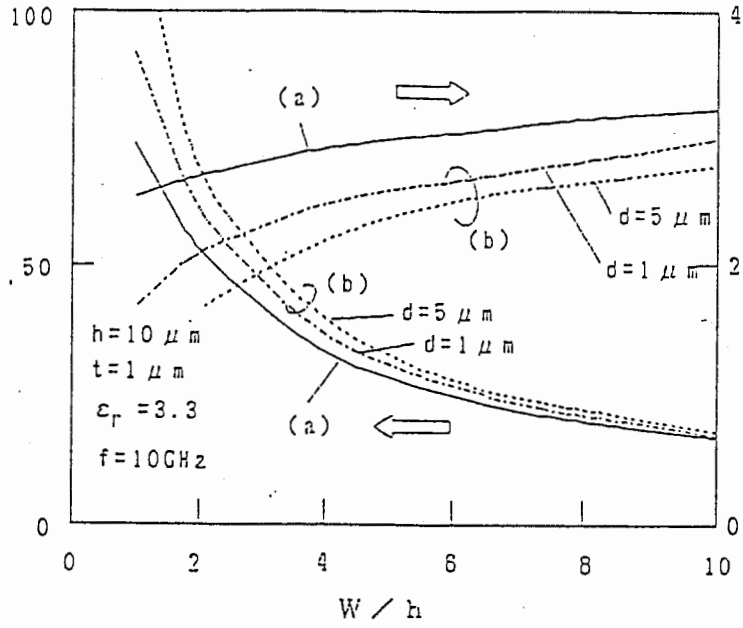
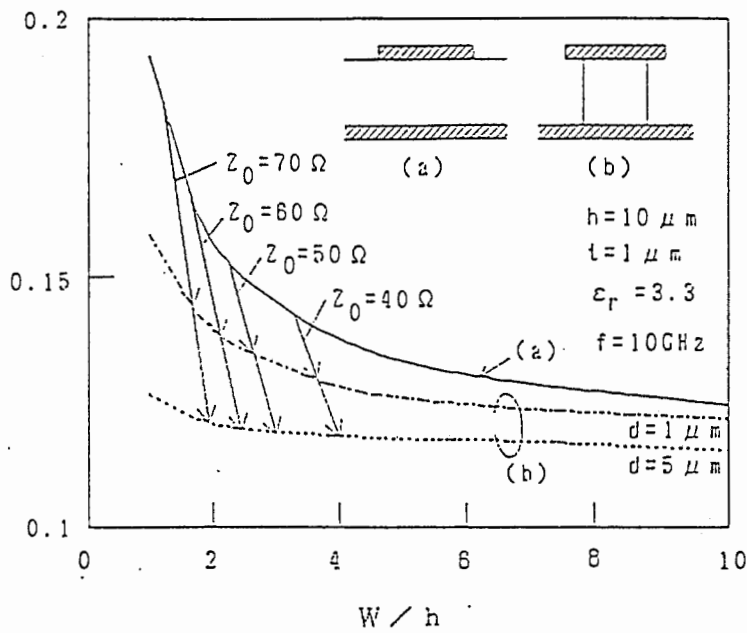


図3-10 ストリップ導体の幅を大きくしたマイクロストリップ線路





(a),(b)は図3-10の構造に対応する。



(a),(b)は図3-10の構造に対応する。

図3-11 マイクロストリップ線路の電気的特性

### 3.5 まとめ

本研究ではマイクロ波伝送線路解析プログラムを用い、スロット線路・マイクロストリップ線路の低損失化について検討した。なお計算は、図3-2,3-3,3-7はTR-O-0008にて示したプログラムを、図3-11は本解析プログラムを用いている。

スロット線路においては対向電極面積を増加した構造、マイクロストリップ線路においては誘電体をストリップ導体の下部にのみ配置した構造が低損失伝送線路実現に有効であることを示した。

なお、スロット線路で導体基板を設けた場合、スロットを構成している導体と基板導体の間にガイドされる漏洩波が生じることが報告されている<sup>[2]</sup>。このことは、本研究の計算例のように仮想境界壁を配置しなければならない場合には、その仮想境界壁の位置が計算結果に大きく影響することを意味している。したがって、この様な線路では実際の導体基板の有無、実装する場合のシールドケースの位置などを考慮して数値計算を行う必要がある。3.1節の計算例ではスロット部の中央を中心として上下左右300 $\mu$ mの位置に磁気壁を配置している。このように仮想境界壁を配置しなければならないスロット線路のような伝送線路解析には本解析法は不向きといえよう。

また、導体の厚みが表皮厚に比べて充分厚くないような場合には導体を電気壁またはインピーダンス壁として扱うことが適当でないとおもわれる。導体の厚みが表皮厚と同等あるいはそれ以下の場合には導体に複素誘電率を与えて計算する必要がある。

### 参考文献

- [1]豊田、徳満、相川：“多層化MMIC用低損失伝送線路の一構成と特性”，電子情報通信学会春季全国大会、C-761、(1989).
- [2]辻、繁沢：“ミリ波平面回路を伝搬する複素波について”，電気学会電磁界理論研資、EMT-88-76、(1988).

#### 4.解析プログラムの使用方法およびプログラムコードの説明

##### 4.1 プログラムシステムの概要

本プログラムシステムの概要を図4-1に示す。なお、これらのプログラムはVAX8650の[MUSENDAI2.FEM2DLOSS]に収納されている。

##### 使用プログラム

- 1) 対話型メッシュ作成プログラム                      MESH2.EXE  
    入力用メッシュデータを作成するプリプロセッサ  
    対話型操作によりデータの編集を行う。
- 2) FEM計算プログラム                                      FEM.EXE  
    ソルバープログラム

##### 使用データファイル

- 1) 基本メッシュ作成用データ                              MESHDATA.DAT
- 2) ソルバー入力用メッシュデータ                        FOR050.DAT
- 3) 媒質データ    ER.DAT
- 4) 計算結果出力用データ                                   標準出力

プログラム1)はグラフィック出力としてGKシステムを用いている。現在このサブルーチンパッケージはVAX8650にインストールされている。このプログラムの起動にあたっては、GKシステムの起動とドライバーの設定を前もって行わなければならない。データはすべてASCII形式で記述されており、その書式については4.2章に記述する。また、プログラムの使用方法は4.3章に、プログラムコードの説明およびソースリストは4.4章に記述する。

作業の流れは、まず基本メッシュ作成用データ(MESHDATA.DAT)をEditを用いて作成し、ついで対話型メッシュ作成プログラム(MESH2.EXE)を起動してソルバープログラム入力用データ(FOR050.DAT)を作成する。媒質の誘電率および透磁率を媒質データファイル(ER.DAT)に記述した後、ソルバープログラム入力用データを標準入力に設定してソルバープログラム(FEM.EXE)を実行すると伝送線路特性が標準出力に出力される。

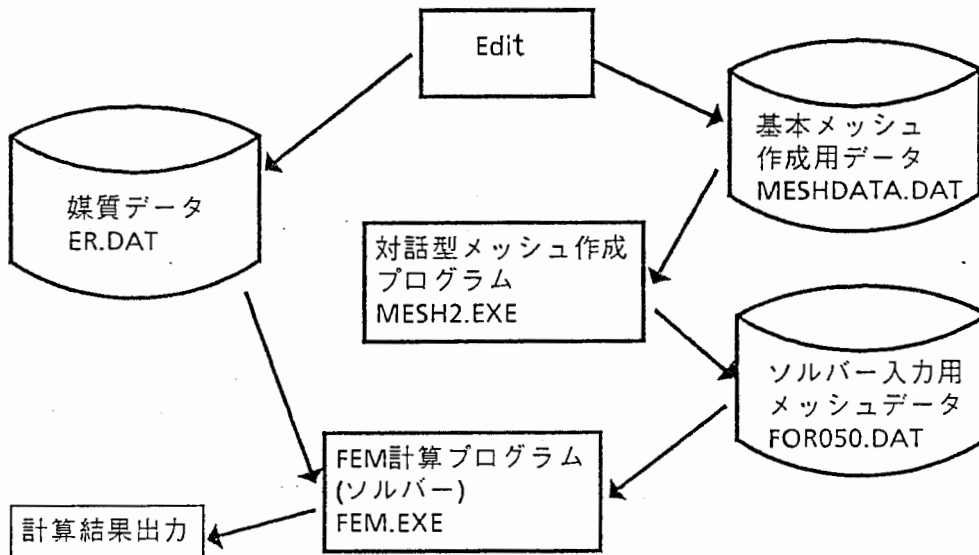


図 4-1 有限要素法による導波路の電磁界解析プログラムシステム構成図

## 4.2 データフォーマット

各プログラムにおける入出力データはすべてアスキーである。以下にその構成を示す。

### 1. 基本メッシュ作成用データ(MESHDATA.DAT)

[拡縮係数]、[計算開始周波数]、[計算終了周波数]、[周波数増分値]  
[分割パラメータ]、[金属導体(1)の比抵抗値]、[金属導体(2)の比抵抗値]  
[ノードの総数]  
[ノードのX座標値、Y座標値]  
[要素の総数]  
[要素番号]、[第1辺の分割数]、[第2辺の分割数]  
[要素を作る4つのノードの番号]  
[材質番号]、[4つの辺の境界条件コード]

(要素の数繰り返す)

### 2. ソルバー入力用メッシュデータ(FOR050.DAT)

[入力ノード総数]、[入力メッシュ総数]、[計算開始周波数]、[計算終了周波数]、[周波数増分値]  
[拡縮係数]、[分割パラメータ]、[金属導体(1)の比抵抗値]、[金属導体(2)の比抵抗値]  
[ノードのX座標値]  
[ノードのY座標値]  
[要素材質番号]  
[要素を作る3つのノードの番号]  
[要素を作る3つの辺の境界条件コード]

### 3. 媒質データ(ER.DAT)

[誘電率]、[誘電率(虚数項)]、[透磁率]、[透磁率(虚数項)] {媒質の数繰り返す}

### 4. 出力データ

[要素数]、[節点数]、[ライン数]、[電界未知数の数]、[磁界未知数の数]

[周波数]、[自由空間波長]、[自由空間波数]

[モードナンバ]  
[特性インピーダンス]  
[k0で規格化された複素伝搬定数]  
[位相定数,挿入損失]  
[ガイド波長,波長短縮率]  
[実効誘電率]

(モードの数繰り返す)

(周波数の数繰り返す)

## 4.3 プログラム使用方法

本節では本解析プログラムの使用手順ならびに使用方法について述べる。なお、本プログラムシステムは、ATRテクニカルレポート(TR-O-0008)に示したソフトウェアに準じたシステムとしており、本レポートではプリプロセッサの使用実施例を省略する。必要な場合は上記レポートを参照すること。

### 4.3.1 基本メッシュ作成用データの作成手順

このプログラムでは有限要素として三角要素を用いているが、すべての要素に関する情報を入力するのは多くの労力を要するしまた誤りも生じやすい。そのため、このシステムでは解析空間を大まかな記述方式で表現し(簡易表現データ)それに基づき自動的に三角要素を作成する

方法を用いている。この簡易表現データはEditを用いてデータファイル(MESHDATA.DAT)に記述する。以下ではこの簡易表現データの作成手順について説明する。

作業手順

- (1) 解析する導波路の断面形状・寸法、導波路内の媒質の誘電率・透磁率、金属導体の比抵抗値、境界条件、計算周波数を明確にする。

境界条件は表4-1に示した5種類が用意されている。金属導体(1)、金属導体(2)は各々ホットとアースの導体に対応する。断面外周部は全て境界条件を設定する。複素誘電率を与えた導体に流れる電流を基にして特性インピーダンスを計算する場合には導体内の要素の導体を囲む閉曲線cに一致する境界に対して境界条件コード4を与えなければならない。

金属導体の抵抗値は20°Cにおける金の抵抗値に対する比率を用いる。参考値を表4-2に示す。金属導体における壁面インピーダンス値が計算される。

計算領域を対称性を利用して1/2もしくは1/4の領域で計算する場合は、ホット導体の分割数とのかねあい分割パラメータを決める。これを表4-3に示す。なお、この値は特性インピーダンスの計算において用いる数値である。

計算周波数はGHz単位で設定する。計算開始周波数値、周波数増分値、計算終了周波数値の3つの値を設定する。

表4-1 境界条件

境界条件	使用コード
金属導体(1)	-1
金属導体(2)	1
磁気壁	2
電気壁	3
導体外周	4

[注] 金属導体(1)、金属導体(2)は各々ホットとアースに対応する。

表4-2 金に対する比抵抗値

材質	金に対する比抵抗値
銅(Cu)	0.693
銀(Ag)	0.652
チタン(Ti)	17.21

[注] 20°Cにおける値

表4-3 分割パラメータ設定値

断面の計算領域	導体の計算領域	分割パラメータ
1	1	1
1/2	1	0.5
1/2	1/2	2
1/4	1/2	1
1/4	1/4	4

- (2) 導波路の断面を、次の条件を満足するように、任意の四角形要素に分割する。
- \* 1つの四角形要素の内部は同一媒質(同じ比誘電率)であること。
  - \* 四角形要素の各辺は各々単一の境界条件であること。(1つの辺上で2つ以上の境界条件を含まないこと)
  - \* 隣り合う四角形要素同士の共通辺はその両端の点(ノード)が一致していること。(1つの辺はその辺をはさむ四角形要素の各々の辺であること)
  - \* 境界条件を設定してある辺はただ1つの四角形要素に属すること。(厚み0の金属板等をモデル化する場合はその上面と下面を独立した辺と考える。)
- (3) ノードに1番から順番に通し番号を付ける。この時、同じ座標位置でも境界条件の設定により分離されているノードは各々独立の番号付けを行う。
- (4) 座標値原点を任意の位置に設定し、各ノードの座標値を計算する。このプログラムではmmが基準単位になっており、拡張係数を設定することにより他の単位系に変換できる。  
(入力数値×拡張係数により所望の単位系に変換できる。たとえば、 $\mu\text{m}$ 単位を使用する場合は拡張係数を0.001、cm単位を用いるときは拡張係数を10とする。)
- (5) 四角形要素に1番から順番に通し番号を付ける。
- (6) 辺上の分割点数を決める。このときその辺を含む四角形要素の対向する辺は同じ分割点数とする。(この分割点数に基づき、プログラムは四角形要素内部に三角要素を作成する。)
- (7) Editを用いて上記手順により設定した入力数値を以下の書式でファイルに書き込む。

なお、入力数値間は1つ以上のブランクをおくこと。

- [1] 拡張係数、計算開始周波数値、計算終了周波数値、周波数増分値を同一行に記入。
- [2] 分割パラメータ、金属導体(1)の比抵抗値、金属導体(2)の比抵抗値を同一行に記入。
- [3] ノードの総数を記入。
- [4] ノード番号順に各ノードのX座標値、Y座標値を同一行に記入。(1行に1つのノードのデータを記入する。)
- [5] 四角形要素の総数を記入。
- [6] 四角形要素番号順に下記3行のデータを記述する。
  - \* 要素番号、第1辺の分割数、第2辺の分割数を同一行に記入。
  - \* 四角形要素を作る4つのノードの番号を反時計回り方向に順に記入。
  - \* 四角形要素内部の媒質番号、第1辺から順に4つの辺の境界条件コードを同一行に記入。

(注:ここで辺の番号は、四角形要素のノード番号記入において第1番目と第2番目のノードにより決まる辺を第1辺とし、反時計回り方向に順に数えるものとする。また境界条件コードが設定されていない辺は境界条件コードの値を0とする。媒質番号は、導波路内における媒質を区別するものであって、その誘電率・透磁率は別のファイルにて設定する。)

#### 4.3.2 対話型メッシュ作成プログラム(MESH2.EXE)の操作手順

対話型メッシュ作成プログラム(MESH2.EXE)は、入力用メッシュデータを作成するプリプロセッサであり、簡易表現データを入力としたメッシュジェネレート機能と、メッシュのグラフィック表示、メッシュデータの編集機能を持つ。なおバックアップ用に随時バックアップデータファイル(MESH.D)にデータをはきだしており、編集途中でトラブルがおきた場合はバックアップデータファイルを(FOR050.DAT)にRENAMEすれば継続できる。また入力ファイルに誤りがある場合は誤りを指摘するエラーメッセージが表示されるので、その指示に従ってデータ等の修正を行う。

メッシュデータの編集機能としては

- (1) 三角要素の向きの変更
- (2) 要素の消去
- (3) 要素の追加
- (4) ノードの座標値の変更
- (5) 境界条件の書き換え
- (6) 媒質番号の変更
- (7) 数値データの書き換え

グラフィック表示関係としては

- (8) 拡大・縮小表示
- (9) 境界条件を含むグラフィック表示
- (10) 番号表示サイズの変更

操作手順

- (1) GKシステムのLOGINとドライバーの設定をおこなう。
- (2) プログラムMESH1.EXEを起動する。
- (3) 画面に下記表示が出る。

\*\*\* Welcome to Mesh System !! \*\*\*

Sellect Data File !! Input file FOR050.DAT = 1  
MESHDATA.DAT = 2

Input No. ??

簡易表現データファイルを用いる場合は[1]、既存のデータファイルを用いる場合は[2]を入力する。

- (4) 続いて、画面に下記表示が出る。

Character size input !! ex. terminal T4014 = 0.05

ノード番号表示の文字の大きさを入力する。数値が大きいほど文字は大きくなる。

- (5) 上記操作を行うと、ターミナル画面に入力メッシュが表示される。このとき画面にはノードの総数[NODE]と要素の総数[LE]が表示される。プログラムはここで一旦ポーズがかかる。リターンキーを叩くと次に進む。(注: ターミナルによってはblankキーの場合があり、このときリターンキーを叩くとエラーが発生することがある。)
- (6) 画面に下記表示が出る。

Sellect No. !!

Change Diagonal = 1

Move Point to delet = 2

Add element = 3

Change Coordinate = 4

Change Boundary = 5

Change Element Er = 6

Change parameter = 7

Zoom = 8

Draw all data = 9

--- Change Character size = 10

END = 11

Quit = 12

作業の番号を選択する。ENDは編集データをデータファイルに書き込みプログラムを終了する。QUITは編集データをデータファイルに書き込まないでプログラムを終了する。Change Character sizeを選択すると(4)にもどる。他の作業内容は次に示す。各作業を終了すると(5)にもどる。

[作業1] 三角要素の向きの変更

指定した辺を対角線とする四角形に注目し、その対角線の向きを変更する。

Change Form ??

Input Node No. !! 0 = Return

上記表示がでるので辺の両端のノード番号を入力する。対角線の向きを変更したデータを表示し(5)に戻る。なお0を入力するとデータの変更なしに画面更新して(5)に戻る。例を図4-2に示す。上側を初期の状態とする。ここで辺の両端のノード番号として5、9を入力すると下側の状態となる。

[作業2] 要素の消去

最初に指定したノードを隣接する2番目に指定したノードに重ねて三角要素を消去する。

Move Node A to B  
No. A & B Input !!  
0 = Return

上記表示がでるので第1のノードの番号と第2のノードの番号を入力する。要素を消去したデータを表示し(5)に戻る。なお0を入力するとデータの変更なしに画面を更新して(5)に戻る。例を図4-3に示す。上側を初期の状態とする。ここでノード番号として17、16を入力すると下図のように点17が点16に移動して要素の消去が行なわれる。

[作業3] 要素の追加

指定した辺の midpoint に新たにノードを作成し、このノードを用いて要素の追加を行う。

Divided Line  
Input Node No. A & B !!

上記表示がでるので辺の両端のノード番号を入力する。要素を追加したデータを表示して(5)に戻る。なお0を入力するとデータの変更なしに画面を更新して(5)に戻る。図4-4に例を示す。左側を初期の状態とし、辺の両端のノード番号として16、12を入力すると midpoint に新たなノード18を作成して要素の追加を行い、右側の状態となる。

[作業4] ノードの座標値の変更

指定したノードの座標値を書き換える。この作業を選択すると次の表示がでる。

Input Node No. !!

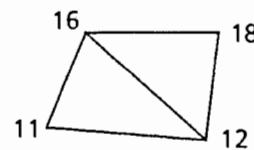
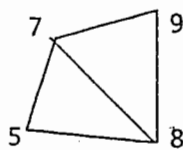
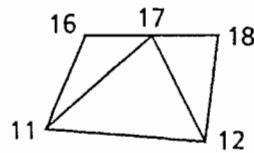
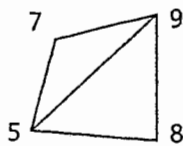


図4-2 三角要素の向きの変更

図4-3 要素の消去

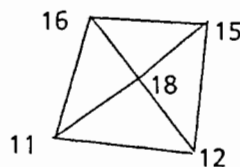
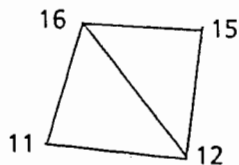


図4-4 要素の追加



ここでノード番号を入力するとその点の座標値が表示される。

```
No. : 10
X   : 10.0
Y   : 5.0
Input New Coordinate
```

X,Yの順に座標値を入力すると新しい座標値のデータを表示して(5)に戻る。

[作業5] 境界条件の書き換え

指定した辺の境界条件コードを書き換える。この作業を選択すると次の表示がでる。

```
Boundary Change
Input Node No. A & B !! 0=Return
```

辺の両端のノード番号を入力するとその辺の境界条件コードが表示される。

Boundary = 1

```
ex. 1 = Conductor(2)
    -1 = Conductor(1)
    0 = No setting
    2 = Magmetical Wall
    3 = Electrical Wall
    4 = Current area
```

Input New Condition !!

新しい境界条件コードを入力する。この作業はノード番号に0を入力するまで繰り返す。

[作業6] 媒質番号の変更

指定した要素の媒質番号を書き換える。この作業を選択すると次の表示がでる。

Input 3-Node No. !! End=0

ここで媒質を変更する要素を構成する3つのノードの番号を入力すると要素の番号とその要素の媒質番号が表示される。

Ele= 10 Er = 2

Input new Er !!

ここで新しい媒質番号をいれる。この作業はノード番号に0を入力するまで繰り返す。

[作業7] 数値データの書き換え

グラフィック表示に関係しない数値データ(拡縮係数、計算開始周波数値、周波数増分値、計算終了周波数値、分割パラメータ、金属導体(1)の比抵抗値、金属導体(2)の比抵抗値)を書き換える。全項目について順次質問してくるのでそれに答える。

[作業8] 拡大・縮小表示

画面に表示する時の縮尺を変更する。この作業を選択すると次の表示がでる。

```
Zoom Point Input !!
if 0 = Normal Size
```

拡大・縮小表示の中心となるノード番号を入れる。(0を入力すると初期設定状態になる)

Magnification ??

上記表示がでるので縮尺を入力する。新しい縮尺でデータを表示し、(5)に戻る。

[作業9] 境界条件を含むグラフィック表示

境界条件・媒質条件をカラー表示する。この作業を選択すると次の表示がでる。

```
Write Node No. ??
Yes=0 No=1
```

[1]を入力するとノード番号は表示されない。

Material draw ?? yes = 0 No = 1

[1]を入力すると誘電体媒質の区分は表示されない。データを表示し、(5)に戻る。

図4-5にこれらの作業のフローを示す。

#### 4.3.3 媒質データファイルの記述方法

媒質データファイルには、上記メッシュ作成時に用いた媒質番号に対応した媒質の複素誘電率、複素透磁率を同一行に記述する。

複素誘電率、複素透磁率は各々実数項(誘電率または透磁率)を先に、虚数項(損失項)を後に記述する。なお、複素誘電率の虚部に負の数を与えられた場合には複素誘電率が直接与えられたものと見なし、正の数を与えられた場合には金に対する比抵抗率が与えられたものと見なす。正の数を与えられた場合にはその比抵抗率と周波数から複素誘電率の虚部を自動的に計算する。媒質に損失がない場合は虚数項に0を記述する。以下に記入例を示す。

1.0	0.0	1.0	0.0
2.5	0.0	1.0	0.0

上記の場合は、媒質番号1は、誘電率が  $1.0 + j0.0$ 、透磁率が  $1.0 + j0.0$ 、媒質番号2は、誘電率が  $2.5 + j0.0$ 、透磁率が  $1.0 + j0.0$  となる。

#### 4.3.4 FEM計算プログラム(FEM.EXE)の操作手順

標準入力をデータファイルに設定するだけで、特別な操作は必要ない。計算結果は標準出力へ表示される。

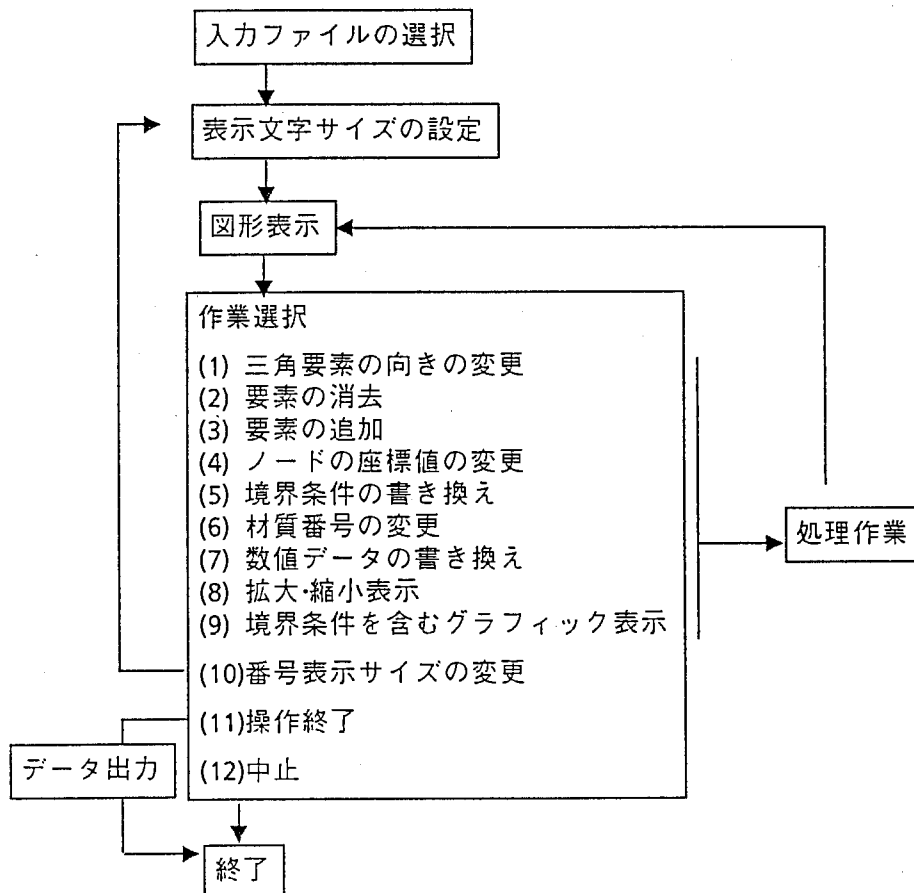


図4-5 作業フロー

#### 4.3.5 その他

本プログラムシステムでは、ポストプロセッサは作成していない。作成する場合にはATRテクニカルレポート(TR-O-0008)が参考になると思われる。

本解析プログラムfem.exeでは損失特性を精密に解析するために、複素誘電率または壁面インピーダンスを与える必要がある。誘電体損失を考慮するためにはその媒質の複素誘電率を与えなければならない。また、導体損失を考慮するためにはその導体の複素誘電率を与えるかまたは壁面インピーダンスを与える必要がある。

本プログラムにおいて、複素誘電率を与えた導体に流れる電流を基にして特性インピーダンスを計算する場合は、導体内の要素の導体を囲む閉曲線cに一致する境界に対して境界条件コード4を与えなければならない。また、壁面インピーダンスは境界条件コード1または-1を与えることにより自動的に計算される。

#### 4.4 プログラム コード

以下では、プログラムコードの簡単な説明とソースリストを示す。

##### 4.4.1 ソルバプログラム(FEM.FOR)

このプログラムはメインプログラムと12のサブルーチンプログラムからなる。フローを図4-6にしめす。なお、本プログラムではIMSLサブルーチンパッケージから以下の5つのサブルーチンを引用しておいる。(ソースコードは以下のリストには記述していない)

LEQT1C UERTST USPKD UGETIO EIGZC

本解析プログラムfem.exe中でユーザが変更する可能性のある定数について説明する。

MNOD,MELM,MUKWN,MEH

これらはプログラム中の配列の大きさを宣言するための定数である。MNODは節点数の最大値、MELMは要素数の最大値、MUKWNは電気壁や磁気壁を考慮する前の未知数の総数、MEHは電気壁や磁気壁を考慮した場合の未知数の総数である。データサイズがこれらの値を越えた場合にはプログラムはメッセージを表示して停止する。この場合、メッセージに従ってこれらの値を変更する。

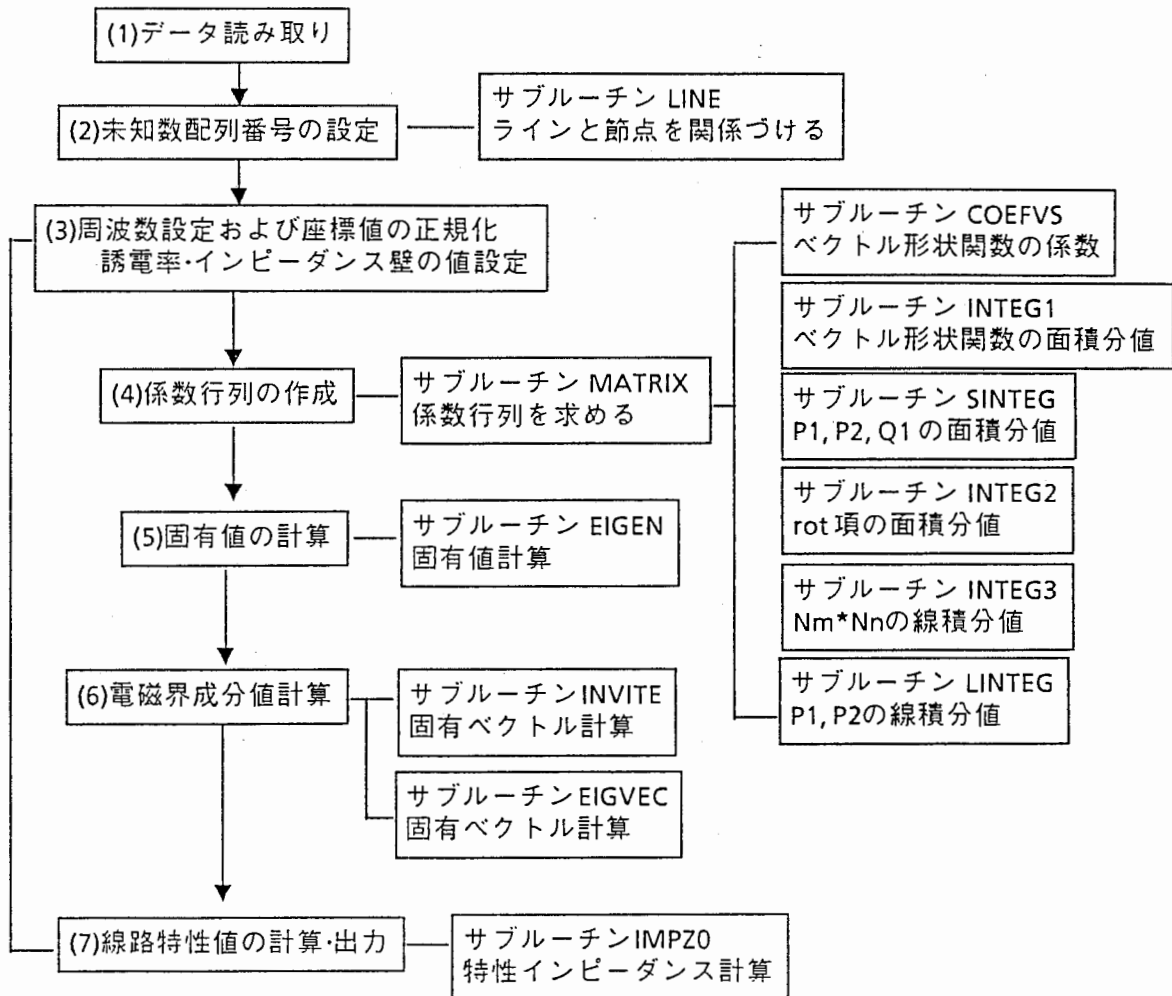


図4-6 プログラムFEMフロー

SUPRI

スプリアス解を抑制するための定数である。求められた複素伝搬定数の虚部の絶対値がこの値を越えるときにはその解がスプリアスであると見なす。したがって、極めて損失の大きい線路を解析する際にはこの値を大きくする必要がある。

以下、プログラムリストを添付する。

```

*****
*           有限要素法による導波路固有モード解析プログラム           *
*           Programed by I.Toyoda Jan. 1989                             *
*           修正 by K. Tsunoda May. 1990                               *
*****

IMPLICIT REAL*8 ( A-H, O-Z )

*   メモリサイズの設定 MNOD : 最大ノード数,   MELM : 最大要素数
*   MUKWN : 最大ライン数*4, MEH : 最大未知数数
PARAMETER ( MNOD = 150, MELM = 150, MUKWN = 730, MEH = 730 )
*****
* 変数リスト
*   IW1, IW2   インピーダンス壁識別コード
*   LC         閉曲線C識別コード
*   EP         ゼロ判定定数
*   SUPRI     スプリアス解抑制定数
*   ITE       逆べき乗法の繰り返し回数
*   KZAI      媒質の総数
*
* 物理定数
*   PI        円周率
*   VC        光速
*   EPSO      真空中の誘電率
*   MUO       真空中の透磁率
*   RHOAU     金の抵抗率
*   ZOFREE    自由空間インピーダンス
*
* 入力データ
*   NNOD      節点総数
*   NELM      要素総数
*   FREQS     計算開始周波数 (GHz)
*   FREQE     計算終了周波数
*   FSTEP     周波数ステップ幅
*   COEF      拡縮係数
*   DRATE     分割パラメータ
*   RHO1, RHO2 インピーダンス壁1,2の金に対する比抵抗率
*   XCO, YCO  節点の座標
*   EP1       要素の比誘電率に関する量
*   MU        要素の比透磁率
*   NODE      各要素を構成する節点の番号
*   BC        各要素の境界条件
*   NZAI      各要素の媒質番号
*   EP10      媒質の比誘電率
*   MUO       媒質の比透磁率
*
* 計算用変数

```

```

*   LNNOD          ラインと節点の関係を格納する配列          *
*   IE, IH         ラインと電界、磁界パラメータの関係を格納する配列  *
*   NUKWN          総ライン数*4                              *
*   NEUKWN, NHUKWN 電界、磁界パラメータ総数                  *
*   MODNO          モード番号                                *
*   FREQ           計算周波数(GHz)                          *
*   OMEGA          角周波数                                  *
*   KO             真空中の波数                              *
*   LAMO           真空中の波長                              *
*   XYNOD          各節点の規格化座標値                      *
*   EPS            各要素の複素比誘電率                      *
*   ZT, ZZ         インピーダンス壁のインピーダンス値        *
*   P1, P2, Q1     係数行列                                  *
*   BETA           複素伝搬定数                              *
*   EVECA, EVECB   固有ベクトルA,B                          *
*   ZO            特性インピーダンス                          *
*   LOSS           伝送損失                                  *
*   LAMG          ガイド波長                                *
*   EPSEFF        実効誘電率                                *
*
* 制御用変数
*   IERR          エラーフラッグ                              *
*   IFREQ        周波数掃引用カウンタ                        *
*   I, J, IELM, LN DOループ用カウンタ                        *
*
* 作業用配列
*   WK1, WK2, WK3, CWK, RWK
*****

```

```

PARAMETER ( IW1 = -1, IW2 = 1, LC = 4, EP = 1.0-12,
&          SUPRI = 1.00, ITE = 20 ,KZAI=10 )
INTEGER   BC( 3, MELM )
REAL*8    KO, LOSS, MU0, LAMO, LAMG
COMPLEX*16 ZT( 3, MELM ), ZZ( 3, MELM ), ZO( 3 ),
&         EP1( MELM ), EPS( MELM ), MU( MELM ),
&         EP10( KZAI ), MUO( KZAI ),
&         P1( MEH, MEH ), P2( MEH, MEH ), Q1( MEH, MEH ),
&         WK1( MEH, MEH ), WK2( MEH, MEH ), WK3( MEH, MEH ),
&         CWK( MEH ), BETA( MEH ), EVECA( MEH ), EVECB( MEH )
DIMENSION NODE( 3, MELM ), XYNOD( 2, MNOD ),
&         LNNOD( MNOD, MNOD, 3 ), IE( MUKWN ), IH( MUKWN ),
&         XCO( MNOD ), YCO( MNOD ), RWK( MEH )

```

\*\*\*\* 物理定数の設定 \*\*\*\*

```

PI      = 3.1415926535897932400
VC      = 2.9979245800
EPS0    = 0.8854178200
MU0     = 1.2566370600
RHOAU   = 2.4400
ZOFREE  = SQRT( MU0 / EPS0 * 1.05 )

```

\*\*\*\* 形状データの読み込み \*\*\*\*

```

READ( *, * ) NNOD, NELM, FREQS, FREQE, FSTEP
IERR = 0
IF ( NNOD .GT. MNOD ) THEN
  WRITE( *, * ) '*** MNOD >= ', NNOD, ' ****'

```

```

      IERR = 1
    ENDIF
    IF ( NELM .GT. MELM ) THEN
      WRITE( *, * ) '*** MELM >= ', NELM, ' ***'
      IERR = 1
    ENDIF
    IF ( IERR .NE. 0 ) STOP

    READ( *, * ) COEF, DRATE, RHO1, RHO2
    READ( *, * ) ( XCO( I ), I = 1, NNOD )
    READ( *, * ) ( YCO( I ), I = 1, NNOD )
    READ( *, * ) ( NZAI( I ), I = 1, NELM )
    READ( *, * ) ( ( NODE( I, J ), I = 1, 3 ), J = 1, NELM )
    READ( *, * ) ( ( BC( I, J ), I = 1, 3 ), J = 1, NELM )
CCC
    OPEN(51, FILE='ER.DAT',STATUS='OLD')
    DO I= 1, KZAI
      READ( 51, * , END=9 ) EP10( I ), MUO( I )
    ENDDO
9 CLOSE(51)
    DO I= 1, NELM
      EP1( I ) = EP10( NZAI( I ) )
      MU( I ) = MUO( NZAI( I ) )
    ENDDO

**** 未知パラメータの番号づけ ****

      CALL LINE ( NODE, BC, MELM, MNOD, MUKWN, MEH, NELM, NNOD,
&                LNNOD, IE, IH, NUKWN, NEUKWN, NHUKWN )

**** 要素数、節点数、未知数総数などの標準出力への出力 ****

      WRITE( *, 600 )
600 FORMAT( ' ', 21X, '*** LINE CHARACTERISTICS ***' / )
      WRITE( *, 610 )
610 FDRMAT( ' Parameters' )
      WRITE( *, 620 ) NELM, NNOD, NUKWN/4, NEUKWN, NHUKWN
620 FORMAT( ' ', I4, ' elements,', I4, ' nodes,', I4, ' lines,',
&          I4, ' E-unknowns,', I4, ' H-unknowns' / )

**** 周波数掃引( DOループ10 ) ****

      DO 10 IFREQ = 1, INT( ( FREQE - FREQS ) / FSTEP ) + 1
        FREQ = FREQS + FSTEP * DBLE( IFREQ - 1 )
        OMEGA = 2.DO * PI * FREQ
        KO = OMEGA / VC * 10

**** 規格化座標値の計算 ****

      DO 20 INOD = 1, NNOD
        XYNOD( 1, INOD ) = KO * COEF * 1.D-3 * XCO( INOD )
        XYNOD( 2, INOD ) = KO * COEF * 1.D-3 * YCO( INOD )
20 CONTINUE

**** 複素誘電率の設定 ****

      DO 30 IELM = 1, NELM
        IF( DIMAG( EP1( IELM ) ) .LT. EP ) THEN

```

```

        EPS( IELM ) = EP1( IELM )
    ELSE
        EPS( IELM ) = DCMPLX( DBLE( EP1( IELM ) ),
&          - 1.DO / ( OMEGA * DIMAG( EP1( IELM ) ) * RHOAU * EPS0 )
&          * 1.D10 )
    ENDIF
30    CONTINUE

```

\*\*\*\* インピーダンス壁のインピーダンス値の設定 \*\*\*\*

```

    DO 40 IELM = 1, NELM
    DO 50 LN = 1, 3
        IF( BC( LN, IELM ) .EQ. IW1 ) THEN
            ZT( LN, IELM ) = ( 1.DO, 1.DO )
&          * SQRT( OMEGA * EPS0 * RHO1 * RHOAU / 2.DO ) * 1.D-5
            ZZ( LN, IELM ) = ZT( LN, IELM )
        ELSE IF( BC( LN, IELM ) .EQ. IW2 ) THEN
            ZT( LN, IELM ) = ( 1.DO, 1.DO )
&          * SQRT( OMEGA * EPS0 * RHO2 * RHOAU / 2.DO ) * 1.D-5
            ZZ( LN, IELM ) = ZT( LN, IELM )
        ELSE
            ZT( LN, IELM ) = ( 0.DO, 0.DO )
            ZZ( LN, IELM ) = ( 0.DO, 0.DO )
        ENDIF
    50    CONTINUE
    40    CONTINUE

```

\*\*\*\* 係数行列の計算 \*\*\*\*

```

    CALL MATRIX ( NODE, EPS, MU, ZT, ZZ, BC, MELM, XYNOD, LNNOD, MNOD,
&          IE, IH, MUKWN, MEH, NELM, P1, P2, Q1 )

```

\*\*\*\* 固有値BETA(伝搬定数)の計算 \*\*\*\*

```

    CALL EIGEN ( P1, P2, Q1, WK1, WK2, WK3, RWK, CWK, BETA,
&          MEH, NEUKWN, NHUKWN )

```

\*\*\*\* 伝送線路特性の計算および出力( DOループ60 ) \*\*\*\*

```

    WRITE( *, 630 )
    630    FORMAT( ' Frequency', 13X, 'Wavelength', 15X, 'Wavenumber' )
        LAMO = VC / FREQ * 1.D2
        WRITE( *, 640 ) FREQ, LAMO, K0
    640    FORMAT( ' f =', F10.5, ' GHz,', 5X, 'λ0 =', F10.5, ' mm,',
&          6X, 'k0 =', f10.5, ' Rad/m' / )

```

```

    MODNO = 0
c$dir scalar
    DO 60 I = 1, NEUKWN
        IF( ABS( DIMAG( BETA( I ) ) ) .LT. SUPRI ) THEN
            MODNO = MODNO + 1
            WRITE( *, 650 ) MODNO
    650    FORMAT( ' mode No.', I2 )

```

\*\*\*\* 逆べき乗法による固有ベクトルの計算 \*\*\*\*

```

    BETA( I ) = 1.DO / ( BETA( I ) * BETA( I ) ) * 1.1D0
    CALL INVITE ( WK3, P1, MEH, BETA( I ), NEUKWN, ITE,

```



```

&          EVECA, WK1, WK2, RWK, CWK )
      BETA( I ) = SQRT( 1.D0 / BETA( I ) )

**** 固有ベクトルBの計算 ****

      CALL EIGVEC ( P2, Q1, MEH, BETA( I ), EVECA, EVECB,
&          NEUKWN, NHUKWN, WK1, WK2, RWK )

**** 特性インピーダンスの計算 ****

      CALL IMPZO ( P2, NODE, BC, EPS, XYNOD, IH, LNNOD, EVECB,
&          BETA( I ), MEH, MNOD, MUKWN, MELM, NELM,
&          NHUKWN, WK1, ZO )
      ZO( 1 ) = ZO( 1 ) * ZOFREE / DRATE
      ZO( 2 ) = ZO( 2 ) * ZOFREE / DRATE
      ZO( 3 ) = ZO( 3 ) * ZOFREE / DRATE

      WRITE( *, 660 )
660      FORMAT( ' Characteristic impedance' )
      WRITE( *, 670 ) IW1, ZO( 1 ), IW2, ZO( 2 ), LC, ZO( 3 )
670      FORMAT( ' ZO( 1, I2, ' ) = ( ' F10.5, ' ', ' F10.5, ' )Ω',
&          ' ZO( 2, I2, ' ) = ( ' F10.5, ' ', ' F10.5, ' )Ω' /
&          ' ZO( 3, I2, ' ) = ( ' F10.5, ' ', ' F10.5, ' )Ω' )

      WRITE( *, 680 )
680      FORMAT( ' Propagation constant' )
      WRITE( *, 690 ) DBLE( BETA( I ) ), -DIMAG( BETA( I ) )
690      FORMAT( ' β/k0 =', 1PD14.7, ' -j', 1PD14.7 )

      LOSS = -DIMAG( BETA( I ) ) * 20.D0 / LOG( 10.D0 ) * KO * 1.D-3
      WRITE( *, 700 )
700      FORMAT( ' Phase constant', 20X, 'Insertion loss' )
      WRITE( *, 710 ) DBLE( BETA( I ) ) * KO, LOSS
710      FORMAT( ' β =', F10.5, ' Rad/m', 11X, 'α =',
&          F10.5, ' dB/mm' )

      LAMG = 2.D0 * PI / DBLE( BETA( I ) ) / KO * 1.D3
      WRITE( *, 720 )
720      FORMAT( ' Guide wavelength', 18X, 'Ratio of wavelength' )
      WRITE( *, 730 ) LAMG, LAMG/LAMO
730      FORMAT( ' λg =', F10.5, ' mm', 14X, 'λg/λ0 =',
&          F10.5 )

      EPSEFF = DBLE( BETA( I ) ) ** 2
      WRITE( *, 740 )
740      FORMAT( ' Effective dielectric constant' )
      WRITE( *, 750 ) EPSEFF
750      FORMAT( ' εeff =', F10.5// )

      ENDIF
60      CONTINUE

      IF( MODNO .EQ. 0 ) THEN
        WRITE( *, * ) ' *****'
        WRITE( *, * ) ' No Propagating Modes'
        WRITE( *, * ) ' *****'
      ENDIF

```

10 CONTINUE

END

```
*****
*          ベクトル形状関数の係数を求めるサブルーチン          *
*          Programed by I.Toyoda Jan. 1989                      *
*****
* 引数リスト                                                    *
*  入力                                                            *
*  XYNOD1, XYNOD2, XYNOD3                                        *
*          節点1,2,3のx,y座標                                    *
*  出力                                                            *
*  VECN          ベクトル形状関数Niの係数                      *
*  VECSN         要素の各辺に立てた単位法線ベクトル          *
*****
* 変数リスト                                                    *
*  EP            ゼロ判定定数                                    *
*  XYNOD         各節点のx,y座標                                *
*  ABSVN         単位法線ベクトルnijの絶対値                  *
*  VNXN         ベクトルnkiとnijの外積                        *
*  I231(i)       i=1,2,3の時2,3,1となる                      *
*  I312(i)       i=1,2,3の時3,1,2となる                      *
*  IXY, I, J, K  DOループ制御変数など                        *
*****
```

SUBROUTINE COEFVS ( XYNOD1, XYNOD2, XYNOD3, VECN, VECSN )

```
IMPLICIT REAL*8 ( A-H, O-Z )
INTEGER          X, Y
PARAMETER       ( X = 1, Y = 2, EP = 1.D-12 )
DIMENSION       XYNOD1( 2 ), XYNOD2( 2 ), XYNOD3( 2 ), XYNOD( 2, 3 ),
&               VECN( 2, 12 ), VECSN( 2, 3 ), I231( 3 ), I312( 3 )
DATA I231 / 2, 3, 1 /
DATA I312 / 3, 1, 2 /
```

\*\*\*\* 各節点の座標値の入れ替え \*\*\*\*

```
DO 10 IXY = X, Y
  XYNOD( IXY, 1 ) = XYNOD1( IXY )
  XYNOD( IXY, 2 ) = XYNOD2( IXY )
  XYNOD( IXY, 3 ) = XYNOD3( IXY )
10 CONTINUE
```

\*\*\*\* 各辺の単位法線ベクトルの計算 \*\*\*\*

```
DO 20 I = 1, 3
  J = I231( I )
  VECSN( X, I ) = XYNOD( Y, J ) - XYNOD( Y, I )
  VECSN( Y, I ) = -( XYNOD( X, J ) - XYNOD( X, I ) )
  ABSVN = SQRT( VECSN( X, I )**2 + VECSN( Y, I )**2 )
  VECSN( X, I ) = VECSN( X, I ) / ABSVN
  VECSN( Y, I ) = VECSN( Y, I ) / ABSVN
  IF ( VECSN( X, I ) .GT. EP ) THEN
    CONTINUE
  ELSE IF ( VECSN( X, I ) .LT. -EP ) THEN
    VECSN( X, I ) = -VECSN( X, I )
    VECSN( Y, I ) = -VECSN( Y, I )
  ELSE
```

```

      VECSN( X, I ) = 0.00
      VECSN( Y, I ) = 1.00
    ENDIF
20  CONTINUE

```

\*\*\*\* ベクトル形状関数の係数の計算 \*\*\*\*

```

DO 30 I = 1, 3
  K = I312( I )
  VNXN = VECSN( X, K ) * VECSN( Y, I ) - VECSN( Y, K ) * VECSN( X, I )
  VECN( X, 2*I-1 ) = VECSN( X, I ) / VNXN
  VECN( Y, 2*I-1 ) = VECSN( Y, I ) / VNXN
  VECN( X, 2*I ) = -VECSN( X, K ) / VNXN
  VECN( Y, 2*I ) = -VECSN( Y, K ) / VNXN
  VECN( X, I+6 ) = -VECSN( Y, I )
  VECN( Y, I+6 ) = VECSN( X, I )
  VECN( X, I+9 ) = VECSN( X, I )
  VECN( Y, I+9 ) = VECSN( Y, I )
30  CONTINUE
  RETURN
  END

```

```

*****
*          式(15)から固有値BETAを求めるサブルーチン          *
*          Programed by I.Toyoda  Jan. 1989                    *
*****
* 引数リスト                                                    *
*  入力                                                          *
*  P1, P2, Q1          式(15)の行列 P1, P2, Q1                *
*  MEH                 整合寸法                                *
*  NEUKWN, NHUKWN     電界、磁界未知パラメータ数            *
*  出力                                                          *
*  BETA                固有値(複素伝搬定数)                  *
*  作業領域                                                    *
*  WK1, WK2, WK3, RWK, CWK                                    *
*****
* 変数リスト                                                    *
*  IERR                エラーインディケータ                  *
*  INFER               エラーインディケータ                  *
*  DUMMY               ダミー変数                            *
*  I, J, K            DOループ制御変数                        *
*****

```

```

SUBROUTINE EIGEN ( P1, P2, Q1, WK1, WK2, WK3, RWK,
&                CWK, BETA, MEH, NEUKWN, NHUKWN )

  IMPLICIT REAL*8 ( A-H, O-Z )
  COMPLEX*16      P1( MEH, MEH ), P2( MEH, MEH ), Q1( MEH, MEH ),
&                WK1( MEH, MEH ), WK2( MEH, MEH ), WK3( MEH, MEH ),
&                CWK( MEH ), BETA( MEH ), DUMMY
  DIMENSION      RWK( MEH )

```

```

**** WK1=P2, WK2=単位行列 ****
DO 10 I = 1, NHUKWN
  DO 20 J = 1, NHUKWN
    WK1( I, J ) = P2( I, J )
    WK2( I, J ) = ( 0.00, 0.00 )
20  CONTINUE
    WK2( I, I ) = ( 1.00, 0.00 )

```

```

10 CONTINUE
**** P2の逆行列を作る WK2=P2の逆行列 ****
IERR = 0
CALL LEQT1C( WK1, NHUKWN, MEH, WK2, NHUKWN, MEH, 0, RWK, IERR )
IF ( IERR .NE. 0 ) WRITE( *, * ) 'LEQT1C ERROR ** ', IERR
**** WK1 = trans( Q1 )*inv( P2 ) = Q2*inv( P2 ) ****
DO 30 I = 1, NEUKWN
  DO 40 J = 1, NHUKWN
    WK1( I, J ) = ( 0.DO, 0.DO )
    DO 50 K = 1, NHUKWN
      WK1( I, J ) = WK1( I, J ) + Q1( K, I ) * WK2( K, J )
50 CONTINUE
40 CONTINUE
30 CONTINUE
**** WK3 = Q2 * inv( P2 ) * Q1 ****
DO 60 I = 1, NEUKWN
  DO 70 J = 1, NEUKWN
    WK3( I, J ) = ( 0.DO, 0.DO )
    DO 80 K = 1, NHUKWN
      WK3( I, J ) = WK3( I, J ) + WK1( I, K ) * Q1( K, J )
80 CONTINUE
70 CONTINUE
60 CONTINUE

**** P1 および Q2*inv( P2 )*Q1 の保存 ****

DO 90 I = 1, NEUKWN
  DO 100 J = 1, NEUKWN
    WK1( I, J ) = WK3( I, J )
    WK2( I, J ) = P1( I, J )
100 CONTINUE
90 CONTINUE

**** 式(15)の第1式の一般化固有値問題を解く ****

IERR = 0
CALL EIGZC( WK1, MEH, WK2, MEH, NEUKWN, 0, CWK, BETA, DUMMY, MEH,
&          DUMMY, INFER, IERR )
IF ( ( IERR .NE. 0 ) .OR. ( INFER .NE. 0 ) )
& WRITE( *, * ) 'EIGZC ERROR ** ', IERR, INFER

DO 110 I = 1, NEUKWN
  BETA( I ) = SQRT( BETA( I ) / CWK( I ) )
110 CONTINUE

RETURN
END
*****
*          固有ベクトルAから固有ベクトルBを求めるサブルーチン          *
*          Programed by I.Toyoda Jan. 1989                               *
*****
* 引数リスト                                                                *
* 入力                                                                      *
* P2, Q1          式(14)の行列P2, Q1                                     *
* MEH             整合寸法                                             *
* BETA            複素伝搬定数                                         *
* EVECA          式(15)の固有ベクトルA                                 *
* NEUKWN, NHUKWN 電界、磁界の未知パラメータ数                       *

```

```

*   出力
*   EVECB                式(15)の固有ベクトルB
*   作業領域
*   WK1, WK2, RWK
*****
*   変数リスト
*   I, J                DOループ制御用変数
*****

```

```

SUBROUTINE EIGVEC( P2, Q1, MEH, BETA, EVECA, EVECB, NEUKWN, NHUKWN,
&                WK1, WK2, RWK )

```

```

IMPLICIT REAL*8 ( A-H, O-Z )
COMPLEX*16      P2( MEH, MEH ), Q1( MEH, MEH ), BETA,
&              EVECA( MEH ), EVECB( MEH ),
&              WK1( MEH, MEH ), WK2( MEH, MEH )
DIMENSION      RWK( MEH )

```

\*\*\*\* P2の逆行列を計算する \*\*\*\*

```

DO 10 I = 1, NHUKWN
  DO 20 J = 1, NHUKWN
    WK1( I, J ) = P2( I, J )
    WK2( I, J ) = ( 0.DO, 0.DO )
20  CONTINUE
    WK2( I, I ) = ( 1.DO, 0.DO )
10  CONTINUE

```

```

IERR = 0
CALL LEQT1C( WK1, NHUKWN, MEH, WK2, NHUKWN, MEH, 0, RWK, IERR )
IF( IERR.NE. 0 ) WRITE( *, * ) 'LEQT1C ERROR ** ', IERR

```

\*\*\*\* P2の逆行列とQ1をかける \*\*\*\*

```

DO 30 I = 1, NHUKWN
  DO 40 J = 1, NEUKWN
    WK1( I, J ) = ( 0.DO, 0.DO )
    DO 50 K = 1, NHUKWN
      WK1( I, J ) = WK1( I, J ) + WK2( I, K ) * Q1( K, J )
50  CONTINUE
40  CONTINUE
30  CONTINUE

```

\*\*\*\* B = beta\*(inv(P2)\*Q1)\*A \*\*\*\*

```

DO 60 I = 1, NHUKWN
  EVECB( I ) = ( 0.DO, 0.DO )
  DO 70 J = 1, NEUKWN
    EVECB( I ) = EVECB( I ) + WK1( I, J ) * EVECA( J )
70  CONTINUE
  EVECB( I ) = BETA * EVECB( I )
60  CONTINUE

```

```

RETURN
END

```

```

*****
*   インピーダンス壁上の電流分布と伝送電力から特性インピーダンスを
*   求めるサブルーチン
*
```

```

* 引数リスト
* 入力
*   P2          式(14)の行列P2
*   NODE        各要素を構成する節点の番号
*   BC          各要素の境界条件
*   XYNOD       各節点のx,y座標
*   IH          ラインと磁界パラメータの関係を格納する配列
*   LNNOD       ラインと節点の関係を格納する配列
*   EVECB       式(15)の固有ベクトルB
*   BETA        複素伝搬定数
*   MEH, MNOD, MUKWN, MELM
*               整合寸法
*   NELM        要素総数
*   NHUKWN      磁界未知パラメータ総数
* 出力
*   Z0          特性インピーダンス
* 作業領域
*   WK1

```

```

* 変数リスト
*   IW1, IW2    インピーダンス壁識別コード
*   LC          閉曲線C識別コード
*   EP          ゼロ判定定数
*   POWER       伝送電力
*   CRRT        インピーダンス壁上の電流
*   VECSNX, VECSNY インピーダンス壁に向かう単位法線ベクトル
*   ABSVN, LENG 辺の長さ
*   I231(i)     i=1,2,3のとき2,3,1となる
*   N1, N2      節点番号格納用テンポラリ変数
*   I1, I2, I3  パラメータ番号格納用テンポラリ変数

```

```

SUBROUTINE IMPZO ( P2, NODE, BC, EPS, XYNOD, IH, LNNOD, EVECB, BETA,
&                MEH, MNOD, MUKWN, MELM, NELM, NHUKWN, WK1, Z0 )

```

```

IMPLICIT REAL*8 ( A-H, O-Z )
COMPLEX*16      P2( MEH, MEH ), POWER, EVECB( MEH ), EPS( MELM ),
&              CRRT( 3 ), Z0( 3 ), BETA, WK1( MEH )
REAL*8         LENG
INTEGER        BC( 3, MELM ), X, Y
DIMENSION      NODE( 3, MELM ), XYNOD( 2, MNOD ), IH( MUKWN ),
&              LNNOD( MNOD, MNOD, 3 ), I231( 3 )
PARAMETER      ( X = 1, Y = 2, EP = 1.D-12 )
PARAMETER      ( IW1 = -1, IW2 = 1, LC = 4 )
DATA I231 / 2, 3, 1 /

```

```

**** P = B*P2*conjg(B)/BETA ****

```

```

DO 10 I = 1, NHUKWN
  WK1( I ) = ( 0.DO, 0.DO )
  DO 20 J = 1, NHUKWN
    WK1( I ) = WK1( I ) + P2( I, J ) * CONJG( EVECB( J ) )
20  CONTINUE
10  CONTINUE

POWER = ( 0.DO, 0.DO )

```

```

DO 30 I = 1, NHUKWN
    POWER = POWER + EVECB( I ) * WK1( I )
30 CONTINUE

```

```
POWER = POWER / BETA
```

```
**** 電流分布を求める ****
```

```

CRRT( 1 ) = ( 0.DO, 0.DO )
CRRT( 2 ) = ( 0.DO, 0.DO )
CRRT( 3 ) = ( 0.DO, 0.DO )

```

```
c$dir scalar
```

```
DO 40 I = 1, NELM
```

```
c$dir scalar
```

```
DO 50 J = 1, 3
```

```

IF( ( BC( J, I ) .EQ. IW1 ) .OR. ( BC( J, I ) .EQ. IW2 ) .OR.
& ( BC( J, I ) .EQ. LC ) ) THEN

```

```
N1 = NODE( J, I )
```

```
N2 = NODE( I231( J ), I )
```

```
VECSNX = XYNOD( Y, N2 ) - XYNOD( Y, N1 )
```

```
VECSNY = -( XYNOD( X, N2 ) - XYNOD( X, N1 ) )
```

```
ABSVN = SQRT( VECSNX**2 + VECSNY**2 )
```

```
VECSNX = VECSNX / ABSVN
```

```
VECSNY = VECSNY / ABSVN
```

```
**** COEFVS できめたnijの向きとn0の向きの相違 ****
```

```
IF( VECSNX .GT. EP ) THEN
```

```
LENG = ABSVN
```

```
ELSE IF( VECSNX .LT. -EP ) THEN
```

```
LENG = -ABSVN
```

```
ELSE
```

```
LENG = SIGN( ABSVN, VECSNY )
```

```
ENDIF
```

```
I1 = IH( LNNOD( N1, N2, 1 ) )
```

```
I2 = IH( LNNOD( N2, N1, 1 ) )
```

```
I3 = IH( LNNOD( N1, N2, 2 ) )
```

```
IF( BC( J, I ) .EQ. IW1 ) THEN
```

```
CRRT( 1 ) = CRRT( 1 )
```

```
& + ( EVECB( I1 ) + EVECB( I2 ) + EVECB( I3 ) * 4.DO )
```

```
& * LENG / 6.DO
```

```
ELSE IF( BC( J, I ) .EQ. IW2 ) THEN
```

```
CRRT( 2 ) = CRRT( 2 )
```

```
& + ( EVECB( I1 ) + EVECB( I2 ) + EVECB( I3 ) * 4.DO )
```

```
& * LENG / 6.DO
```

```
ELSE
```

```
CRRT( 3 ) = CRRT( 3 )
```

```
& + ( EVECB( I1 ) + EVECB( I2 ) + EVECB( I3 ) * 4.DO )
```

```
& * LENG / 6.DO * DIMAG( EPS( I ) ) / EPS( I )
```

```
& * ( 0.DO, 1.DO )
```

```
ENDIF
```

```
ENDIF
```

```
50 CONTINUE
```

40 CONTINUE

\*\*\*\* Z0 = P/I\*\*2 \*\*\*\*

```
IF( ABS( CRRT( 1 ) ) .GT. EP ) THEN
  Z0( 1 ) = POWER / ABS( CRRT( 1 ) )**2
ELSE
  Z0( 1 ) = ( 0.DO, 0.DO )
ENDIF
IF( ABS( CRRT( 2 ) ) .GT. EP ) THEN
  Z0( 2 ) = POWER / ABS( CRRT( 2 ) )**2
ELSE
  Z0( 2 ) = ( 0.DO, 0.DO )
ENDIF
IF( ABS( CRRT( 3 ) ) .GT. EP ) THEN
  Z0( 3 ) = POWER / ABS( CRRT( 3 ) )**2
ELSE
  Z0( 3 ) = ( 0.DO, 0.DO )
ENDIF

RETURN
END
```

```
*****
*       スカラ形状関数Nm*Nnの面積分値をもとめるサブルーチン       *
*                               Programed by I.Toyoda Jan. 1989      *
*****
* 引数リスト                                                           *
*  入力                                                                 *
*  XYNOD1, XYNOD2, XYNOD3                                             *
*                               節点1,2,3のx,y座標                     *
*  出力                                                                 *
*  SNN                          Nm*Nnの面積分値                       *
*****
* ■■リスト                                                             *
*  XYNOD                          節点1,2,3のx,y座標                 *
*  AREA                            三角形要素の面積                   *
*  A180                             面積の180分の1                   *
*  COEF                             積分値のA180分の1                 *
*                               COEF*A180が積分値になる               *
*  I231(i)                          i=1,2,3の時2,3,1となる           *
*  I312(i)                          i=1,2,3の時3,1,2となる           *
*  A, B                             作業用配列                       *
*  IXY, I, J, K                      D0ルー制御変数など               *
*****
```

SUBROUTINE INTEG1 ( XYNOD1, XYNOD2, XYNOD3, SNN )

```
IMPLICIT REAL*8 ( A-H, O-Z )
INTEGER          X, Y
PARAMETER       ( X = 1, Y = 2 )
DIMENSION       XYNOD1( 2 ), XYNOD2( 2 ), XYNOD3( 2 ), XYNOD( 2, 3 ),
&               SNN( 6, 6 ), A( 3 ), B( 3 ),
&               I231( 3 ), I312( 3 ), COEF( 6, 6 )
```

```
DATA I231 / 2, 3, 1 /
DATA I312 / 3, 1, 2 /
DATA COEF / 6.DO, -1.DO, -1.DO, 0.DO, -4.DO, 0.DO,
&          -1.DO, 6.DO, -1.DO, 0.DO, 0.DO, -4.DO,
```



```

&          -1.00, -1.00,  6.00, -4.00,  0.00,  0.00,
&          0.00,  0.00, -4.00, 32.00, 16.00, 16.00,
&          -4.00,  0.00,  0.00, 16.00, 32.00, 16.00,
&          0.00, -4.00,  0.00, 16.00, 16.00, 32.00 /

```

\*\*\*\* 各節点の座標値の入れ替え \*\*\*\*

```

DO 10 IXY = X, Y
  XYNOD( IXY, 1 ) = XYNOD1( IXY )
  XYNOD( IXY, 2 ) = XYNOD2( IXY )
  XYNOD( IXY, 3 ) = XYNOD3( IXY )
10 CONTINUE

```

\*\*\*\* 各要素の面積の計算 \*\*\*\*

```

DO 20 I = 1, 3
  J = I231( I )
  K = I312( I )
  A( I ) = XYNOD( Y, J ) - XYNOD( Y, K )
  B( I ) = -XYNOD( X, J ) + XYNOD( X, K )
20 CONTINUE
AREA = ( A( 1 ) * B( 2 ) - A( 2 ) * B( 1 ) ) / 2.00
A180 = AREA / 180.00

```

\*\*\*\* 積分値の計算 \*\*\*\*

```

DO 30 I = 1, 6
  DO 40 J = 1, 6
    SNN( I, J ) = COEF( I, J ) * A180
40 CONTINUE
30 CONTINUE

```

```

RETURN
END

```

```

*****
*   ロテーションNmの内積の面積分値を求めるサブルーチン   *
*   Programed by I.Toyoda Jan. 1989   *
*****
* 引数リスト   *
*  入力   *
*   XYNOD1, XYNOD2, XYNOD3   *
*           節点1,2,3のx,y座標   *
*  出力   *
*   SRNRN   rot(Nm).dot.rot(Nn)の面積分値   *
*****
* 変数リスト   *
*   XYNOD   節点1,2,3のx,y座標   *
*   AREA   三角形要素の面積   *
*   AREAD6 面積の12分の1   *
*   COEF   積分値の係数   *
*   AB     -xj+xkなど   *
*   I231(i) i=1,2,3の時2,3,1となる   *
*   I312(i) i=1,2,3の時3,1,2となる   *
*   ISUF   添字制御用変数   *
*   I1, I2, I3, I4 添字用テンポラリ変数   *
*   I, J, K, IXY, JXY   *
*           DOLR=制御変数など   *
*****

```

```
SUBROUTINE INTEG2 ( XYNOD1, XYNOD2, XYNOD3, SRNRN )
```

```
IMPLICIT REAL*8 ( A-H, O-Z )
INTEGER          X, Y
PARAMETER       ( X = 1, Y = 2 )
DIMENSION       XYNOD1( 2 ), XYNOD2( 2 ), XYNOD3( 2 ), XYNOD( 2, 3 ),
&               SRNRN( 6, 2, 6, 2 ), AB( 2, 3 ), COEF( 6, 6 ),
&               I231( 3 ), I312( 3 ), ISUF( 6, 6 )
```

```
DATA I231 / 2, 3, 1 /
DATA I312 / 3, 1, 2 /
DATA COEF / 3.00, -1.00, -1.00, 4.00, 0.00, 4.00,
&          -1.00, 3.00, -1.00, 4.00, 4.00, 0.00,
&          -1.00, -1.00, 3.00, 0.00, 4.00, 4.00,
&          4.00, 4.00, 0.00, 4.00, 4.00, 4.00,
&          0.00, 4.00, 4.00, 4.00, 4.00, 4.00,
&          4.00, 0.00, 4.00, 4.00, 4.00, 4.00 /
DATA ISUF / 1, 2, 3, 2, 1, 3,
&          1, 2, 3, 1, 3, 2,
&          1, 2, 3, 3, 2, 1,
&          1, 2, 3, 1, 3, 3,
&          1, 2, 3, 1, 2, 1,
&          1, 2, 3, 2, 2, 3 /
```

```
**** 各節点の座標値の入れ替え ****
```

```
DO 10 IXY = X, Y
  XYNOD( IXY, 1 ) = XYNOD1( IXY )
  XYNOD( IXY, 2 ) = XYNOD2( IXY )
  XYNOD( IXY, 3 ) = XYNOD3( IXY )
10 CONTINUE
```

```
**** 各要素の面積の計算 ****
```

```
DO 20 I = 1, 3
  J = I231( I )
  K = I312( I )
  AB( X, I ) = XYNOD( Y, J ) - XYNOD( Y, K )
  AB( Y, I ) = -XYNOD( X, J ) + XYNOD( X, K )
20 CONTINUE
AREA = ( AB( X, 1 ) * AB( Y, 2 ) - AB( X, 2 ) * AB( Y, 1 ) ) / 2.00
AREAD6 = 1.00 / 2.00 / AREA / 6.00
```

```
**** 積分値の計算 ****
```

```
DO 30 IXY = X, Y
  DO 40 JXY = X, Y
    DO 50 I = 1, 6
      DO 60 J = 1, 6
        SRNRN( I, IXY, J, JXY ) = AREAD6 * COEF( I, J )
&          * AB( IXY, ISUF( I, J ) ) * AB( JXY, ISUF( J, I ) )
60 CONTINUE
50 CONTINUE
40 CONTINUE
30 CONTINUE
```

```
DO 70 IXY = X, Y
```

```

DO 80 JXY = X, Y
DO 90 I = 4, 6
DO 100 J = 4, 6

      I1 = I-3
      I2 = I231( I-3 )
      I3 = J-3
      I4 = I231( J-3 )

      SRNRN( I, IXY, J, JXY )
&      = SRNRN( I, IXY, J, JXY ) + 4.DO * AREAD6
&      * ( AB ( IXY, I1 ) + AB ( IXY, I2 ) )
&      * ( AB ( JXY, I3 ) + AB ( JXY, I4 ) )
100      CONTINUE
90      CONTINUE
80      CONTINUE
70      CONTINUE

DO 110 IXY = X, Y
DO 120 JXY = X, Y
DO 130 I = 4, 6
      I1 = I231( I-3 )
      SRNRN( I, IXY, I, JXY )
&      = SRNRN( I, IXY, I, JXY ) + 4.DO * AREAD6
&      * AB( IXY, I1 ) * AB( JXY, I1 )
130      CONTINUE
120      CONTINUE
110      CONTINUE

RETURN
END

```

```

*****
*      スカラー形状関数Nm*Nnの線積分値をもとめるサブルーチン      *
*      Programed by I.Toyoda Jan. 1989      *
*****
* 引数リスト      *
*  入力      *
*  XYNOD1, XYNOD2   ライン両端の節点の座標      *
*  出力      *
*  LNN      Nm*Nnの線積分値      *
*****
* 変数リスト      *
*  LENG      辺の長さ      *
*  L30      辺の長さの30分の1      *
*  COEF      積分値の係数      *
*****

```

```

SUBROUTINE INTEG3 ( XYNOD1, XYNOD2, LNN )

IMPLICIT REAL*8 ( A-H, O-Z )
REAL*8      LNN( 3, 3 ), LENG, L30
INTEGER      X, Y
PARAMETER   ( X = 1, Y = 2 )
DIMENSION   XYNOD1( 2 ), XYNOD2( 2 ), COEF( 3, 3 )

DATA COEF / 4.DO, 2.DO, -1.DO,
&          2.DO, 16.DO, 2.DO,

```

```

&          -1.D0, 2.D0, 4.D0 /

LENG = SQRT( ( XYNOD2( X ) - XYNOD1( X ) ) ** 2
&          + ( XYNOD2( Y ) - XYNOD1( Y ) ) ** 2 )
L30 = LENG / 30.D0

DO 10 I = 1, 3
  DO 20 J = 1, 3
    LNN( I, J ) = COEF( I, J ) * L30
20  CONTINUE
10  CONTINUE

RETURN
END
*****
*   一般化固有値問題  $Ax=(\lambda)Bx$  の固有ベクトルを           *
*   逆べき乗法により求めるサブルーチン                       *
*   Programed by I.Toyoda Jan. 1989                            *
*****
* 引数リスト                                                    *
* 入力                                                            *
*   A                行列A                                       *
*   B                行列B                                       *
*   LD               整合寸法                                       *
*   LAMBDA           固有値                                       *
*   N                行列サイズ                                       *
*   NITE            反復回数                                       *
* 出力                                                            *
*   LAMBDA           改良された固有値                                       *
*   X                固有ベクトル                                       *
* 作業領域                                                    *
*   WK1, WK2, RWK, Z                                       *
*****
* 変数リスト                                                    *
*   I, J, K          DOループ制御用変数                                       *
*   LAMBD1, LAMBD2  作業用変数                                       *
*   IERR            エラーインディケータ                                       *
*****

SUBROUTINE INVITE( A, B, LD, LAMBDA, N, NITE, X, WK1, WK2, RWK, Z )

IMPLICIT REAL*8 ( A-H, O-Z )
COMPLEX*16      A( LD, LD ), B( LD, LD ), WK1( LD, LD ), WK2( LD, LD ),
&              X( LD ), Z( LD ), LAMBDA, LAMBD1, LAMBD2
DIMENSION      RWK( LD )

**** Xの初期化 ****

DO 10 I = 1, N
  X( I ) = ( 1.D0, 1.D0 )
10  CONTINUE

**** A-(LAMBDA)Bの逆行列を求める ****

DO 20 I = 1, N
  DO 30 J = 1, N
    WK1( I, J ) = A( I, J ) - LAMBDA * B( I, J )
    WK2( I, J ) = ( 0.D0, 0.D0 )
  
```

```

30     CONTINUE
      WK2( I, I ) = ( 1.D0, 0.D0 )
20     CONTINUE

      IERR = 0
      CALL LEQT1C( WK1, N, LD, WK2, N, LD, 0, RWK, IERR )
      IF( IERR .NE. 0 ) WRITE( *, * ) 'LEQT1C ERROR** ', IERR

      DO 40 K = 1, NITE

          DO 50 I = 1, N
              Z( I ) = ( 0.D0, 0.D0 )
              DO 60 J = 1, N
                  Z( I ) = Z( I ) + B( I, J ) * X( J )
60         CONTINUE
50     CONTINUE

          DO 70 I = 1, N
              X( I ) = ( 0.D0, 0.D0 )
              DO 80 J = 1, N
                  X( I ) = X( I ) + WK2( I, J ) * Z( J )
80         CONTINUE
70     CONTINUE

40     CONTINUE

```

\*\*\*\* 固有値の改良 \*\*\*\*

```

      LAMBD1 = ( 0.D0, 0.D0 )
      DO 90 I = 1, N
          Z( I ) = ( 0.D0, 0.D0 )
          DO 100 J = 1, N
              Z( I ) = Z( I ) + A( I, J ) * X( J )
100     CONTINUE
          LAMBD1 = LAMBD1 + X( I ) * Z( I )
90     CONTINUE

      LAMBD2 = ( 0.D0, 0.D0 )
      DO 110 I = 1, N
          Z( I ) = ( 0.D0, 0.D0 )
          DO 120 J = 1, N
              Z( I ) = Z( I ) + B( I, J ) * X( J )
120     CONTINUE
          LAMBD2 = LAMBD2 + X( I ) * Z( I )
110     CONTINUE

      LAMBDA = LAMBD1 / LAMBD2

      RETURN
      END

```

```

*****
*           ラインと節点を関係づけるサブルーチン           *
*           Programed by I.Toyoda Jan. 1989                 *
*****
* 引数リスト *
*  入力      *
*  NODE      各要素を構成する節点の番号                    *

```

```

*      BC              各要素の境界条件              *
*      MELM           整合寸法                      *
*      MNOD           整合寸法                      *
*      MUKWN          整合寸法                      *
*      MEH            整合寸法                      *
*      NELM           要素総数                      *
*      NNOD           節点総数                      *
*      出力
*      LNNOD(*,*,i)   ラインと節点の関係を格納する配列 *
*                    i=1は各ラインの端のパラメータを示す *
*                    i=2は各ラインの midpoint の接線成分を示す *
*                    i=3は各ラインの midpoint の法線成分を示す *
*      IE, IH         ラインと電界、磁界パラメータの関係を格納する配列*
*      NUKWN          ライン総数*4                  *
*      NEUKWN, NHUKWN 電界、磁界パラメータ総数      *
*****
* 変数リスト
*      EW, MW         電気壁、磁気壁識別コード      *
*      I, J           DOループ制御変数              *
*      K              未知数番号づけ用変数          *
*      N1, N2         節点番号格納用のテンポラリ変数 *
*      I231(i)        i=1,2,3の時2,3,1となる      *
*****

```

```

SUBROUTINE LINE ( NODE, BC, MELM, MNOD, MUKWN, MEH, NELM, NNOD,
&                LNNOD, IE, IH, NUKWN, NEUKWN, NHUKWN )

```

```

IMPLICIT REAL*8 ( A-H, O-Z )
INTEGER          BC( 3, MELM ), EW, MW
PARAMETER       ( EW = 3, MW = 2 )
DIMENSION       NODE( 3, MELM ), LNNOD( MNOD, MNOD, 3 ),
&               IE( MUKWN ), IH( MUKWN ), I231( 3 )
DATA I231 / 2, 3, 1 /

```

\*\*\*\* LNNODの初期化 \*\*\*\*

```

DO 10 I = 1, NNOD
  DO 20 J = 1, MNOD
    LNNOD( I, J, 1 ) = 0
    LNNOD( I, J, 2 ) = 0
    LNNOD( I, J, 3 ) = 0
20  CONTINUE
10  CONTINUE

```

\*\*\*\* 節点と各ライン上のパラメータを関係づける \*\*\*\*

```

K = 0
DO 30 I = 1, NELM
  DO 40 J = 1, 3
    N1 = NODE( J, I )
    N2 = NODE( I231( J ), I )
    IF ( LNNOD( N1, N2, 1 ) .EQ. 0 ) THEN
      LNNOD( N1, N2, 1 ) = K + 1
      LNNOD( N2, N1, 1 ) = K + 2
      LNNOD( N1, N2, 2 ) = K + 3
      LNNOD( N2, N1, 2 ) = K + 3
      LNNOD( N1, N2, 3 ) = K + 4
      LNNOD( N2, N1, 3 ) = K + 4

```

```

        K = K + 4
    ENDIF
40    CONTINUE
30    CONTINUE
    NUKWN = K

**** 配列サイズのチェック ****

    IF ( NUKWN .GT. MUKWN ) THEN
        WRITE(*,*) '*** MUKWN >= ',NUKWN,' ***'
        STOP
    ENDIF

**** IE, IH の初期化 ****

    DO 50 I = 1, NUKWN
        IE( I ) = 1
        IH( I ) = 1
50    CONTINUE

**** 電気壁では電界の接線成分と磁界の法線成分がゼロになる ****
**** 磁気壁では磁界の接線成分と電界の法線成分がゼロになる ****
**** 従ってIE, IHに0を入れて区別する ****

    DO 60 I = 1, NELM
        DO 70 J = 1, 3
            N1 = NODE( J, I )
            N2 = NODE( I231( J ), I )
            IF ( BC( J, I ) .EQ. EW ) THEN
                IE( LNNOD( N1, N2, 1 ) ) = 0
                IE( LNNOD( N2, N1, 1 ) ) = 0
                IE( LNNOD( N1, N2, 2 ) ) = 0
                IH( LNNOD( N1, N2, 3 ) ) = 0
            ELSE IF ( BC( J, I ) .EQ. MW ) THEN
                IH( LNNOD( N1, N2, 1 ) ) = 0
                IH( LNNOD( N2, N1, 1 ) ) = 0
                IH( LNNOD( N1, N2, 2 ) ) = 0
                IE( LNNOD( N1, N2, 3 ) ) = 0
            ENDIF
70    CONTINUE
60    CONTINUE

**** 電界の未知パラメータに番号づけをする ****

    K = 0
    DO 80 I = 1, NUKWN
        IF ( IE( I ) .NE. 0 ) THEN
            K = K + 1
            IE( I ) = K
        ENDIF
80    CONTINUE
    NEUKWN = K

**** 磁界の未知パラメータに番号づけをする ****

    K = 0
    DO 90 I = 1, NUKWN
        IF ( IH( I ) .NE. 0 ) THEN

```

```

      K = K + 1
      IH( I ) = K
    ENDIF
90  CONTINUE
    NHUKWN = K

```

\*\*\*\* メモリサイズのチェック \*\*\*\*

```

    IF ( ( NEUKWN .GT. MEH ) .OR. ( NHUKWN .GT. MEH ) ) THEN
      WRITE(*,*) '*** MEH >= ',MAX( NEUKWN, NHUKWN ),' ****'
      STOP
    ENDIF

    RETURN
  END

```

```

*****
*           P1, P2の線積分を計算するサブルーチン           *
*           Programed by I.Toyoda Jan. 1989                *
*****
* 引数リスト                                             *
*  入力                                                 *
*  IELM          要素番号                               *
*  LN            要素の辺の番号                         *
*  NODE          各要素を構成する節点の番号           *
*  EPS, MU       各要素の比誘電率,比透磁率           *
*  ZT, ZZ        インピーダンス壁のインピーダンス値   *
*  MELM, MNOD, MUKWN, MEH                               *
*  整合寸法                                             *
*  LNNOD         ラインと節点の関係を格納した配列     *
*  IE, IH        ラインと電界、磁界パラメータの関係を格納した配列*
*  VECN          ベクトル形状関数Niの係数             *
*  VECSN        各辺の単位法線ベクトル               *
*  LNN          Nm*Nnの線積分値                       *
*  P1, P2       P1, P2の面積分値                     *
*  出力                                                 *
*  P1, P2       式(14)の行列P1, P2                   *
*****
* 変数リスト                                             *
*  CEM, CHM, CEN, CHN                                   *
*  要素境界で電磁界の法線成分の境界条件を           *
*  入れるための変数                                   *
*  ME, MH, NE, NH  電界および磁界のパラメータ番号   *
*  MS, NS          スカラー形状関数の番号             *
*  LSCAL          ベクトル形状関数とスカラー形状関数の対応 *
*  LUU            n0 X Umの内積の線積分               *
*  LVV            n0 X Vmの内積の線積分               *
*  I731(i)        i=1,2,3の時2,3,1となる             *
*  I351(i)        i=1,2,3の時3,5,1となる             *
*  I23710         形状関数番号制御用                 *
*  I, MM, NN      DOループ制御用変数                 *
*  M, N           ベクトル形状関数番号               *
*  IWK           作業用配列                           *
*  N1, N2        節点番号格納用テンポラリ変数       *
*****

```

```

SUBROUTINE LINTEG ( IELM, LN, NODE, EPS, MYU, ZT, ZZ, MELM,
&                  LNNOD, MNOD, IE, IH, MUKWN, MEH, VECN, VECSN, LNN,
&                  P1, P2 )

```



```

IMPLICIT REAL*8 ( A-H, O-Z )
REAL*8      LNN( 3, 3 ), LUU, LVV
COMPLEX*16  P1( MEH, MEH ), P2( MEH, MEH ),
&           EPS( MELM ), MYU( MELM ),
&           ZT( 3, MELM ), ZZ( 3, MELM ),
&           CEM, CHM, CEN, CHN
INTEGER     X, Y
PARAMETER   ( X = 1, Y = 2 )
DIMENSION   NODE( 3, MELM ),
&           LNNOD( MNOD, MNOD, 3 ), IE( MUKWN ), IH( MUKWN ),
&           VECN( .2, 12 ), VECNS( 2, 3 ),
&           IWK( 12 ), I231( 3 ), I351( 3 ), I23710( 4, 3 ),
&           LSCAL( 4 )

```

```

DATA I231 / 2, 3, 1 /
DATA I351 / 3, 5, 1 /
DATA I23710 / 2, 3, 7, 10,
&           4, 5, 8, 11,
&           6, 1, 9, 12 /
DATA LSCAL / 1, 3, 2, 2 /

```

\*\*\*\* パラメータ番号の取り出し \*\*\*\*

```

DO 10 I = 1, 3
  N1 = NODE( I, IELM )
  N2 = NODE( I231( I ), IELM )
  IWK( 2 * I ) = LNNOD( N1, N2, 1 )
  IWK( I351( I ) ) = LNNOD( N2, N1, 1 )
  IWK( I + 6 ) = LNNOD( N1, N2, 2 )
  IWK( I + 9 ) = LNNOD( N1, N2, 3 )
10 CONTINUE

```

\*\*\*\* P1, P2, Q1の計算 ( DOループ20, 30 ) \*\*\*\*

```

DO 20 MM = 1, 4
  M = I23710( MM, LN )
**** 電磁界の法線成分の境界条件を入れる ****
  IF ( M .GT. 9 ) THEN
    CEM = 1.00 / EPS( IELM )
    CHM = 1.00 / MYU( IELM )
  ELSE
    CEM = ( 1.00, 0.00 )
    CHM = ( 1.00, 0.00 )
  ENDIF

```

\*\*\*\* 電界、磁界のパラメータ番号の取り出し \*\*\*\*

```

  ME = IE( IWK( M ) )
  MH = IH( IWK( M ) )
  MS = LSCAL( MM )

DO 30 NN = 1, 4
  N = I23710( NN, LN )
**** 電磁界の法線成分の境界条件を入れる ****
  IF ( N .GT. 9 ) THEN
    CEN = 1.00 / EPS( IELM )
    CHN = 1.00 / MYU( IELM )
  ELSE

```

```

      CEN = ( 1.D0, 0.D0 )
      CHN = ( 1.D0, 0.D0 )
ENDIF

```

\*\*\*\* 電界、磁界のパラメータ番号の取り出し \*\*\*\*

```

      NE = IE( IWK( N ) )
      NH = IH( IWK( N ) )
      NS = LSCAL( NN )

```

\*\*\*\* 線積分値の計算 \*\*\*\*

```

      LUU = ( VECSN( X, LN ) * VECN( Y, M ) * VECSN( X, LN )
&          * VECN( Y, N )
&          - VECSN( X, LN ) * VECN( Y, M ) * VECSN( Y, LN )
&          * VECN( X, N )
&          - VECSN( Y, LN ) * VECN( X, M ) * VECSN( X, LN )
&          * VECN( Y, N )
&          + VECSN( Y, LN ) * VECN( X, M ) * VECSN( Y, LN )
&          * VECN( X, N ) ) * LNN( MS, NS )
      LVV = LUU

      P1( ME, NE ) = P1( ME, NE )
&          - ( 0.D0, 1.D0 ) / ZT( LN, IELM ) * LUU * CEM * CEN
      P2( MH, NH ) = P2( MH, NH )
&          - ( 0.D0, 1.D0 ) * ZZ( LN, IELM ) * LVV * CHM * CHN
30      CONTINUE
20      CONTINUE

      RETURN
      END

```

```

*****
*          係数行列P1,P2,Q1を求めるサブルーチン          *
*          Programed by I.Toyoda Jan. 1989                *
*****
* 引数リスト                                             *
* 入力                                                 *
*  NODE          各要素を構成する節点の番号             *
*  EPS, MU       各要素の比誘電率,比透磁率             *
*  ZT, ZZ        インピーダンス壁のインピーダンス値     *
*  BC            各要素の境界条件                       *
*  MELM, MNOD, MUKWN, MEH                             *
*                整合寸法                               *
*  XYNOD         規格化座標値                           *
*  LNNOD         ラインと節点の関係を格納した配列       *
*  IE, IH        ラインと電界、磁界パラメータの関係を格納した配列*
*  NELM         要素総数                               *
* 出力                                                 *
*  P1, P2, Q1    式(14)の行列P1, P2, Q1                *
*****
* 変数リスト                                             *
*  IW1, IW2     インピーダンス壁1,2識別コード         *
*  VECN         ベクトル形状関数Niの係数               *
*  VECSN        要素の各辺に立てた単位法線ベクトル     *
*  SNN          Nm*Nnの面積分値                         *
*  SRNRN        rot(Nm).dot.rot(Nn)の面積分値          *
*  LNN          Nm*Nnの線積分値                         *
*  I, J, IELM, LN DOループ制御変数                     *
*  N1, N2, N3   節点番号格納用テンポラリ変数          *

```

```

*      I231(i)                i=1,2,3の時2,3,1となる                *
*****
SUBROUTINE MATRIX ( NODE, EPS, MU, ZT, ZZ, BC, MELM,
&                  XYNOD, LNNOD, MNOD, IE, IH, MUKWN, MEH, NELM,
&                  P1, P2, Q1 )

IMPLICIT REAL*8 ( A-H, O-Z )
COMPLEX*16      P1( MEH, MEH ), P2( MEH, MEH ),
&              Q1( MEH, MEH ),
&              EPS( MELM ), MU( MELM ),
&              ZT( 3, MELM ), ZZ( 3, MELM )
REAL*8         LNN( 3, 3 )
INTEGER        X, Y, BC( 3, MELM )
PARAMETER      ( X = 1, Y = 2, IW1 = -1, IW2 = 1 )
DIMENSION      NODE( 3, MELM ),
&              XYNOD( 2, MNOD ), LNNOD( MNOD, MNOD, 3 ),
&              IE( MUKWN ), IH( MUKWN ),
&              VECN( 2, 12 ), VECSN( 2, 3 ),
&              SNN( 6, 6 ), SRNRN( 6, 2, 6, 2 ), I231( 3 )
DATA I231 / 2, 3, 1 /

**** P1, P2, Q1の初期化 ****

DO 10 I = 1, MEH
  DO 20 J = 1, MEH
    P1( I, J ) = ( 0.DO, 0.DO )
    P2( I, J ) = ( 0.DO, 0.DO )
    Q1( I, J ) = ( 0.DO, 0.DO )
20  CONTINUE
10  CONTINUE

**** 式(14)の各行列の成分の計算( DOループ30 ) ****

DO 30 IELM = 1, NELM
  N1 = NODE( 1, IELM )
  N2 = NODE( 2, IELM )
  N3 = NODE( 3, IELM )

**** ベクトル形状関数の係数の計算 ****

CALL COEFVS ( XYNOD( X, N1 ), XYNOD( X, N2 ), XYNOD( X, N3 ),
&           VECN, VECSN )

**** Nm*Nnの面積分の計算 ****

CALL INTEG1 ( XYNOD( X, N1 ), XYNOD( X, N2 ), XYNOD( X, N3 ),
&           SNN )

**** rot(Nm).dot.rot(Nn)の面積分の計算 ****

CALL INTEG2 ( XYNOD( X, N1 ), XYNOD( X, N2 ), XYNOD( X, N3 ),
&           SRNRN )

**** P1, P2, Q1の面積分の計算 ****

CALL SINTEG ( IELM, NODE, EPS, MU, MELM, LNNOD, MNOD,
&           IE, IH, MUKWN, MEH, VECN, SNN, SRNRN, P1, P2, Q1 )

```

\*\*\*\* インピーダンス壁の時はP1, P2に線積分を加える( DOループ40 ) \*\*\*\*

```

DO 40 LN = 1, 3
  IF ( ( BC( LN, IELM ) .EQ. IW1 )
&      .OR. ( BC( LN, IELM ) .EQ. IW2 ) ) THEN
    N1 = NODE( LN, IELM )
    N2 = NODE( I231( LN ), IELM )

```

\*\*\*\* Nm\*Nnの線積分の計算 \*\*\*\*

```
CALL INTEG3 ( XYNOD( 1, N1 ), XYNOD( 1, N2 ), LNN )
```

\*\*\*\* P1, P2の線積分の計算 \*\*\*\*

```

CALL LINTEG ( IELM, LN, NODE, EPS, MU, ZT, ZZ, MELM, LNNOD,
&            MNOD, IE, IH, MUKWN, MEH, VECN, VECSN, LNN,
&            P1, P2 )

```

```

ENDIF
40 CONTINUE

```

```
30 CONTINUE
```

```

RETURN
END

```

```

*****
*           P1, P2, Q1の面積分を計算するサブルーチン           *
*           Programed by I.Toyoda Jan. 1989                       *
*****

```

```

* 引数リスト *
* 入力 *
*   IELM      要素番号 *
*   NODE      各要素を構成する節点の番号 *
*   EPS, MU   各要素の比誘電率,比透磁率 *
*   MELM, MNOD, MUKWN, MEH *
*           整合寸法 *
*   LNNOD     ラインと節点の関係を格納した配列 *
*   IE, IH    ラインと電界、磁界パラメータの関係を格納した配列*
*   VECN      ベクトル形状関数Niの係数 *
*   SNN       Nm*Nnの面積分値 *
*   SRNRN     rot(Nm).dot.rot(Nn)の面積分値 *
* 出力 *
*   P1, P2, Q1  式(14)の行列P1, P2, Q1の面積分値 *

```

```

*****
* 変数リスト *
*   CEM, CHM, CEN, CHN *
*           要素境界で電磁界の法線成分の境界条件を *
*           入れるための変数 *
*   ME, MH, NE, NH     電界および磁界のパラメータ番号 *
*   MS, NS              スカラー形状関数の番号 *
*   NSCAL               ベクトル形状関数とスカラー形状関数の対応 *
*   SUU                 UmとUnの内積の積分値 *
*   SVV                 VmとVnの内積の積分値 *
*   SUV, SVU           UmとVnの内積の積分値 *
*   SRURU               rot(Um)とrot(Un)の内積の積分値 *
*   SRVRV               rot(Vm)とrot(Vn)の内積の積分値 *
*   I231(i)            i=1,2,3の時2,3,1となる *

```

```

*   I351(i)           i=1,2,3の時3,5,1となる
*   LN, M, N         DOループ制御用変数
*   IWK              作業用配列
*   N1, N2           節点番号格納用テンポラリ変数
*****

```

```

SUBROUTINE SINTEG ( IELM, NODE, EPS, MYU, MELM, LNNOD, MNOD,
&                 IE, IH, MUKWN, MEH, VECN, SNN, SRNRN,
&                 P1, P2, Q1 )

```

```

IMPLICIT REAL*8 ( A-H, O-Z )
COMPLEX*16       P1( MEH, MEH ), P2( MEH, MEH ),
&               Q1( MEH, MEH ),
&               EPS( MELM ), MYU( MELM ),
&               CEM, CHM, CEN, CHN
INTEGER          X, Y
PARAMETER       ( X = 1, Y = 2 )
DIMENSION       NODE( 3, MELM ), LNNOD( MNOD, MNOD, 3 ),
&               IE( MUKWN ), IH( MUKWN ),
&               VECN( 2, 12 ), SNN( 6, 6 ), SRNRN( 6, 2, 6, 2 ),
&               IWK( 12 ), I231( 3 ), I351( 3 ), NSCAL( 12 )

```

```

DATA I231 / 2, 3, 1 /
DATA I351 / 3, 5, 1 /
DATA NSCAL / 1, 1, 2, 2, 3, 3, 4, 5, 6, 4, 5, 6 /

```

\*\*\*\* パラメータ番号の取り出し \*\*\*\*

```

DO 10 LN = 1, 3
  N1 = NODE( LN, IELM )
  N2 = NODE( I231( LN ), IELM )
  IWK( 2 * LN ) = LNNOD( N1, N2, 1 )
  IWK( I351( LN ) ) = LNNOD( N2, N1, 1 )
  IWK( LN + 6 ) = LNNOD( N1, N2, 2 )
  IWK( LN + 9 ) = LNNOD( N1, N2, 3 )
10 CONTINUE

```

\*\*\*\* P1, P2, Q1の計算 ( DOループ20, 30 ) \*\*\*\*

```

DO 20 M = 1, 12
**** 電磁界の法線成分の境界条件を入れる ****
  IF ( M .GT. 9 ) THEN
    CEM = 1.00 / EPS( IELM )
    CHM = 1.00 / MYU( IELM )
  ELSE
    CEM = ( 1.00, 0.00 )
    CHM = ( 1.00, 0.00 )
  ENDIF

```

\*\*\*\* 電界、磁界のパラメータ番号の取り出し \*\*\*\*

```

  ME = IE( IWK( M ) )
  MH = IH( IWK( M ) )
  MS = NSCAL( M )

```

```

DO 30 N = 1, 12
**** 電磁界の法線成分の境界条件を入れる ****
  IF( N .GT. 9 ) THEN
    CEN = 1.00 / EPS( IELM )

```

```

      CHN = 1.DO / MYU( IELM )
    ELSE
      CEN = ( 1.DO, 0.DO )
      CHN = ( 1.DO, 0.DO )
    ENDIF

```

\*\*\*\* 電界、磁界のパラメータ番号の取り出し \*\*\*\*

```

      NE = IE( IWK( N ) )
      NH = IH( IWK( N ) )
      NS = NSCAL( N )

```

\*\*\*\* 面積分値の計算 \*\*\*\*

```

      SUU = ( VECN( X, M ) * VECN( X, N )
&          + VECN( Y, M ) * VECN( Y, N ) ) * SNN( MS, NS )
      SVV = SUU
      SUV = ( VECN( X, M ) * VECN( Y, N )
&          - VECN( Y, M ) * VECN( X, N ) ) * SNN( MS, NS )
      SVU = SUV
      SRURU = VECN( Y, M ) * VECN( Y, N ) * SRNRN( MS, X, NS, X )
&          - VECN( Y, M ) * VECN( X, N ) * SRNRN( MS, X, NS, Y )
&          - VECN( X, M ) * VECN( Y, N ) * SRNRN( MS, Y, NS, X )
&          + VECN( X, M ) * VECN( X, N ) * SRNRN( MS, Y, NS, Y )
      SRVRV = SRURU

```

\*\*\*\* P1, P2, Q1 の構成 \*\*\*\*

```

      IF ( ( ME .NE. 0 ) .AND. ( NE .NE. 0 ) ) THEN
        P1( ME, NE ) = P1( ME, NE )
&          + ( SUU * EPS( IELM ) - SRURU / MYU( IELM ) )
&          * CEM * CEN
      ENDIF

```

```

      IF ( ( MH .NE. 0 ) .AND. ( NH .NE. 0 ) ) THEN
        P2( MH, NH ) = P2( MH, NH )
&          + ( SVV * MYU( IELM ) - SRVRV / EPS( IELM ) )
&          * CHM * CHN
      ENDIF

```

```

      IF ( ( MH .NE. 0 ) .AND. ( NE .NE. 0 ) ) THEN
        Q1( MH, NE ) = Q1( MH, NE ) - SVU * CHM * CEN
      ENDIF

```

30 CONTINUE

20 CONTINUE

```

      RETURN
      END

```

#### 4.4 対話型メッシュ作成プログラム(MESH2.FOR)

フローを図4-1にしめす。

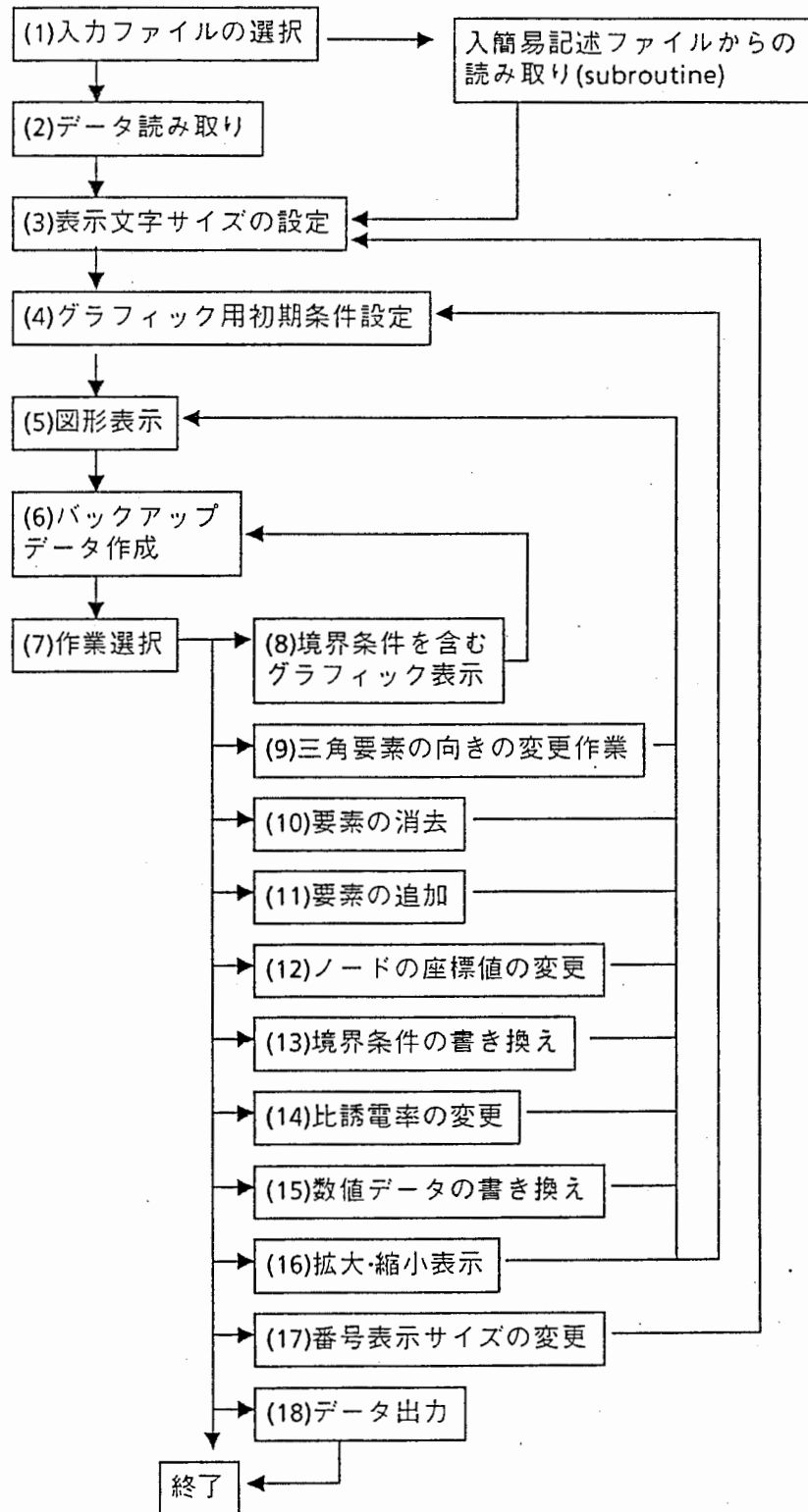


図4-1 プログラムMESH2フロー

このプログラムはメインプログラムと6つのサブルーチンプログラムからなる。またグラフィックとしてGKシステムを使用している。

メッシュの構成条件は配列MESH(NE,I)に格納される。NEは要素番号であり、2番目の引数  
はつぎの項目を表す。

I=1,2,3要素を構成する3つのノードの番号格納

I=5,6,7要素の各辺の境界条件コードの格納

I=4 要素の媒質番号を格納

またノードの座標値は配列X(NP),Y(NP)にノード番号NPのX座標値、Y座標値が格納されてい  
る。メインプログラムでは、各編集作業において入力値に該当する要素(項目)の有無の確認の  
のち、その作業内容に関して配列MESH(NE,I)、X(NP),Y(NP)の中身を変更している。

一方サブルーチンプログラム MESHF では入力四角形要素の辺上に配置したノードの番号を  
配列LNODE(NE,I,IB)に格納し、要素間の重複ノードに対する番号付けを行なっている。

なお、配列のサイズはパラメータ文により設定しているため、必要に応じてサイズの変更を  
行うこと。

以下、プログラムリストを添付する。

```

C      ** 2次三角要素の半自動作成プログラム **
C
C                                     1990. 5   by K. Tsunoda
C [変数リスト]
C *****
C      LB=160      出力できる分割後の三角要素の最大数
C      LA=120      出力できる分割後のノードの最大数
C *****
C 入力データ
C  SK              拡大縮小係数
C  FREQ1,FREQ2,FREQ3  計算周波数
C  DSK            計算領域分割数
C  SKM1,SKM2      金属の導電率
C  NP             ノード番号総数
C  NE            要素番号総数
C  XP(LA),YP(LA)  ノードのX座標Y座標
C  XPP(LB)        要素の誘電率
C
C 編集時に使用する変数
C  MESH(LB,7)     要素のノード,誘電率,境界条件の登録
C  ERTBL(10)      誘電体誘電率登録
C  MERTB(LB)      誘電体誘電率識別番号登録
C  IPO1,IPO2      辺指定時の入力ノード番号
C  IENO(2),LENO(2) 変更対象要素番号収納
C  INO1(7),INO2(7) 変更対象要素のデータ格納
C  AXP,AYP        辺の分割時の中点座標
C  IEADD,IP12,IPAD 番号調整時のカウンター
C  POIN2          変数入力用メモリー
C  NNE,NONE       要素番号再構成用カウンター
C
C グラフィック関係
C  IWKID          ワークのレイヤー番号
C  GMINX,GMAXX,GMINY,GMAXY  ウィンドウエリアサイズ
C  CIH            表示文字のサイズ
C  ICL           線の色設定
C  ICLTX         文字の色設定
C  ICR           塗り潰し部分の色設定
C  XXX(4),YYY(4) グラフィック表示用XY座標メモリー

```



```

C   IXY(4)           作図時の点のノード番号格納
C   AXX,AYY         表示範囲中心点座標
C   DXX            表示範囲での中心点から端までの距離
C   AXX0,AYY0      表示範囲中心点初期設定座標
C   DXX0,DYY0      表示範囲での中心点から端までの距離の初期設定値
C   XMAXM,XMINM,YMAXM,YMINM  入力座標の最大値最小値
C
C   制御用
C   JKAI           データセーブ用回数カウンター
C   JOK,JJOK       選択用変数(YES,NO判断,作業の選択)
C   I,J,JJ,JJJ,K   制御カウンター
C   IDATA0         入出力データのIO番号
C   IERR           エラーチェック
C
C
C   ***** メモリーサイズの設定 *****
C   PARAMETER (LB=500,LA=500)
C
C   ***** 配列宣言 *****
C   DIMENSION XP(LA),YP(LA),MESHD(LB,7),ERTBL(10),MERTB(LB),
C   #             IENO(2),LENO(2),INO1(7),INO2(7),XXX(4),YYY(4),IXY(4)
C
C   ***** サイズの表示
C   WRITE(6,7)LB,LA
C   7 FORMAT(' *** Welcome to Mesh Generation System !! ***',/,
C   # ' This program size = max ',I5,'meshes and max ',I5,'nodes',/,
C   # ' if you need details of this program, see manual !!',//)
C
C   ***** 初期値の設定 *****
C   JKAI=0
C   NP=0
C   NE=0
C   IERR=0
C   IWKID=1
C   ICLTX=7
C
C   ***** データブロックの選択およびデータの読み込み
C   WRITE(6,1)
C   1 FORMAT(' ***** This Program makes Meshdata *****',/,
C   # 10X,' Select Data File !! Input file FOR050.DAT=1',/,
C   # 42X,' MESHDATA.DAT=2',/,25X,' Input No. ?? ',,$)
C   READ(5,*)JOK
C
C   IF(JOK.EQ.1) THEN
C     CALL MESHE( LA,LB,NP,NE,FREQ1,FREQ2,FREQ3,SK,DSK,
C   #           SKM1,SKM2,XP,YP,MESHD,IERR )
C   ELSE
C     CALL MESHF( LB,LA,NP,NE,FREQ1,FREQ2,FREQ3,SK,DSK,
C   #           SKM1,SKM2,XP,YP,MESHD,IERR )
C   ENDIF
C   IF(IERR.NE.0)GOTO 990
C
C   ***** 表示範囲の設定
C   DO I=1,NP
C     XMAXM=MAX(XMAXM,XP(I))
C     YMAXM=MAX(YMAXM,YP(I))
C     XMINM=MIN(XMINM,XP(I))
C     YMINM=MIN(YMINM,YP(I))

```

```

ENDDO
AXX0=(XMAXM+XMINM)/2.0
AYY0=(YMAXM+YMINM)/2.0
DXX0=(XMAXM-XMINM)/2.0
DYY0=(YMAXM-YMINM)/2.0
DXX0= MAX(DXX0,DYY0)*1.1
C
C ***** グラフィック用初期データ準備
30 WRITE(6,3000)
3000 FORMAT(' Character size input !! ex. tarminal T4014 =0.05')
READ(5,*)CIH0
C
300 CONTINUE
CIH=CIH0
AXX=AXX0
AYY=AYY0
DXX=DXX0
C
C ***** グラフィックルーチン(作図)
301 CONTINUE
GMXX=AXX+DXX
GMXY=AYY+DXX
GMINX=AXX-DXX
GMINY=AYY-DXX
CALL SETGR(GMINX,GMXX,GMINY,GMXY)
C
CALL GSPLCI(1) ! 線の色選択
WLINE=1.0
CALL LINEGR(LA,LB,NE,WLINE,XP,YP,MESHD) ! 要素境界線記述
CALL NUMBGR(LA,NP,CIH,ICLTX,XP,YP) ! ノード番号記述
CALL GPAUSE(IWKID)
C
C[400]***** 作業の選択
400 CONTINUE
C ----- バックアッス
JKAI=JKAI+1
IF(JKAI.EQ.5)THEN
WRITE(6,*)' Save datas now !! '
IO=40
CALL MESHO(LA,LB,IO,NP,NE,FREQ1,FREQ2,FREQ3,SK,DSK,
# SKM1,SKM2,XP,YP,MESHD,IERR)
JKAI=0
ENDIF
C
WRITE(6,3104)NP,NE
3104 FORMAT(' NODE',/,2X,I3,/, ' ELE',/,2X,I3)
C
WRITE(6,4001)
4001 FORMAT(/, ' Select No. !! ',/,
# ' Change Diagonal =1',/, ' Move Point to delete=2',/,
# ' Add Element =3',/, ' Change Coordinate =4',/,
# ' Change Boundary =5',/, ' Change Material NO. =6',/,
# ' Change parameter =7',/, ' Zoom =8',/,
# ' Draw all data =9',/, ' -- Change Character size =10',/,
# ' END=11 ',/, ' QUIT=12')
READ(5,*)JOK
IF(JOK.EQ.10) CALL OFFGR
GOTO (450,500,700,600,650,890,860,550,750,30,800,850) JOK

```

```

C
C[450]***** 対角線の向きの変更
  450 CONTINUE
      CALL DIAGGR(LA, LB, NE, IERR, MESHD)
      IF(IERR.EQ.400) GOTO 400
      IF(IERR.EQ.-1) GOTO 950
      CALL OFFGR
      GOTO 301

C
C[500]***** ノードの重ね合わせ
  500 CONTINUE
      CALL DELGR(LA, LB, NP, NE, IERR, MESHD, XP, YP)
      IF(IERR.EQ.-1) GOTO 950
      CALL OFFGR
      GOTO 301

C
C[550]***** 図面拡大表示
  550 CONTINUE
      WRITE(6,5501)
5501 FORMAT(' Zoom Point Input !!',/, ' if 0 = Normal Size ')
      READ(5,*)IPO1
      IF(IPO1.EQ.0)GOTO 551
      WRITE(6,5502)
5502 FORMAT(' Magnification ??')
      READ(5,*)POIN2
      AXX=XP(IPO1)
      AYY=YP(IPO1)
      DXX=DXX/POIN2
      CIH=CIH/POIN2
  551 CONTINUE
      CALL OFFGR
      IF(IPO1.EQ.0)GOTO 300
      GOTO 301

C
C[600]***** 座標書き換え
  600 WRITE(6,6001)
6001 FORMAT(' Input Node No. !! ')
      READ(5,*)IPO1
      IF(IPO1.EQ.0)GOTO 601
      WRITE(6,6002)IPO1,XP(IPO1),YP(IPO1)
6002 FORMAT(' NO.:',I3,/, ' X :',F7.3,/, ' Y :',F7.3,/,
# ' Input New Cordination ',/,
# ' X Cordination ?? ')
      READ(5,*)POIN2
      XP(IPO1)=POIN2
  602 WRITE(6,6003)
6003 FORMAT(' Y Cordination ?? ')
      READ(5,*)POIN2
      YP(IPO1)=POIN2
  601 CONTINUE

C
      CALL OFFGR
      GOTO 301

C
C[650]***** 境界条件書き換え
  650 CONTINUE
      CALL BOUNGR(LA, LB, NE, IERR, MESHD)
      CALL OFFGR

```

```

        GOTO 301
C
C[700]***** 要素分割の追加
    700 CONTINUE
        CALL ADDGR(LA, LB, NP, NE, IERR, MESHD, XP, YP)
        CALL OFFGR
        GOTO 301
C
C[750]***** グラフ フルアウト
    750 CONTINUE
        WRITE(6, 7550)
    7550 FORMAT(' Write Node number ??', /, ' yes=0, NO = 1 ')
        READ(5, *) JOK
        WRITE(6, 7551)
    7551 FORMAT(' Material draw ?? yes=0 NO=1')
        READ(5, *) JJOK
C
        CALL OFFGR
        CALL SETGR(GMINX, GMAXX, GMINY, GMAXY)
C
        WLINE=2.0      ! 線幅選択
        CALL GSPLCI(1)  ! 線の色選択
        CALL LINEGR(LA, LB, NE, WLINE, XP, YP, MESHD) ! 境界線の表示
C          ノード番号表示
        IF(JOK.NE.1) CALL NUMBGR(LA, NP, CIH, ICLTX, XP, YP)
    760 WRITE(6, 3104) NP, NE
C          要素の塗り分け
        IF(JJOK.NE.1) CALL ELEMGR(LA, LB, NE, MESHD, XP, YP)
C 境界条件
        CALL ELEBGR(LA, LB, NE, MESHD, XP, YP)
C
        CALL GSLWSC(1.0) ! 線幅選択
        CALL GSPLCI(1)  ! 線の色選択
        CALL GPAUSE(IWKID)
        GOTO 400
C
C *****[850] QUIT
    850 JOK=1
        CALL OFFGR
        IF(JOK.EQ.1) GOTO 5654
C
C *****[860] other data
    860 CONTINUE
        CALL TABLGR(FREQ1, FREQ2, FREQ3, SK, DSK, SKM1, SKM2)
        GOTO 400
C
C[890] ***** 誘電率を変更する。
    890 CONTINUE
        WRITE(6, *) ' Input 3-node No. END=0'
        READ(5, *) IPO1, IPO2, IPO3
        IF(IPO1.EQ.0) GOTO 400
        DO I=1, NE
            IF((MESHD(I, 1).EQ.IPO1).AND.(MESHD(I, 2).EQ.IPO2).AND.
#           (MESHD(I, 3).EQ.IPO3) ) GOTO 891
            IF((MESHD(I, 2).EQ.IPO1).AND.(MESHD(I, 3).EQ.IPO2).AND.
#           (MESHD(I, 1).EQ.IPO3) ) GOTO 891
            IF((MESHD(I, 3).EQ.IPO1).AND.(MESHD(I, 1).EQ.IPO2).AND.
#           (MESHD(I, 2).EQ.IPO3) ) GOTO 891

```

```

ENDDO
GOTO 950
891 DIELE=MESHD(I,4)
WRITE(6,893)I,DIELE
893 FORMAT(' ELE=' ,I3, ' Material No.=' ,I3,/, ' Input new No. ')
READ(5,*)DIELE
MESHD(I,4)=DIELE
GOTO 890
C
C[800] ***** 計算されたデータをディスクに書き込む
800 CONTINUE
IO=50
CALL MESHO(LA, LB, IO, NP, NE, FREQ1, FREQ2, FREQ3, SK, DSK,
# SKM1, SKM2, XP, YP, MESHD, IERR)
C
5654 CONTINUE
STOP
C *****
C *** Error section
C *****
C ***** グラフィック時エラー
C
950 WRITE(6,9500)
9500 FORMAT(' INPUT MISS TRY AGEIN !! ')
GOTO 400
C ***** データ meshdata.dat読み込み時エラー
990 CONTINUE
STOP
END
C
SUBROUTINE ADDGR(LA, LB, NP, NE, IERR, MESHD, XP, YP)
C
C 指定した辺を2分割して新たな要素を追加する
C
DIMENSION XP(LA), YP(LA), MESHD(LB, 7),
# IENO(2), LENO(2), INO1(7), INO2(7)
C
IERR=0
C
WRITE(6,7024)
7024 FORMAT(' Divided Line ',/, ' Input Node No. A & B !! ',/,
# ' 0=Return')
READ(5,*)IPO1
IF(IPO1.EQ.0)GOTO 400
READ(5,*)IPO2
IP12=MIN(IPO1, IPO2)
AXP=(XP(IPO1)+XP(IPO2))/2.0
AYP=(YP(IPO1)+YP(IPO2))/2.0
JJ=0
DO 7025 I=1, NE
JJJ=0
IF((MESHD(I, 1).EQ.IPO1).AND.(MESHD(I, 2).EQ.IPO2)) JJJ=1
IF((MESHD(I, 2).EQ.IPO1).AND.(MESHD(I, 3).EQ.IPO2)) JJJ=2
IF((MESHD(I, 3).EQ.IPO1).AND.(MESHD(I, 1).EQ.IPO2)) JJJ=3
IF((MESHD(I, 3).EQ.IPO1).AND.(MESHD(I, 2).EQ.IPO2)) JJJ=2
IF((MESHD(I, 2).EQ.IPO1).AND.(MESHD(I, 1).EQ.IPO2)) JJJ=1
IF((MESHD(I, 1).EQ.IPO1).AND.(MESHD(I, 3).EQ.IPO2)) JJJ=3
IF(JJJ.NE.0)GOTO 7026

```

```

        IF((I.EQ.NE).AND.(JJ.NE.0))GOTO 7027
        GOTO 7025
7026  JJ=JJ+1
        IENO(JJ)=I
        LENO(JJ)=JJJ
        IF(JJ.EQ.2)GOTO 7027
7025  CONTINUE
        IF((I.EQ.NE).AND.(JJ.NE.0))GOTO 7027
        GOTO 950
C ----- 新しい番号付け
7027  CONTINUE
        IPAD=IP12+1
        DO I=NP,IPAD,-1
            XP(I+1)=XP(I)
            YP(I+1)=YP(I)
        ENDDO
        XP(IPAD)=AXP
        YP(IPAD)=AYP
        NP=NP+1
C ----- 番号変換
        DO 7038 I=1,NE
        DO 7138 J=1,3
            IF(MESHD(I,J).LT.IPAD)GOTO 7138
            MESHD(I,J)=MESHD(I,J)+1
7138  CONTINUE
7038  CONTINUE
C ----- 新しい要素の追加
        DO 7061 K=1,JJ
            IF(K.EQ.2)IENO(K)=IENO(K)+1
            IEADD=IENO(K)+1
            DO 7039 I=NE,IEADD,-1
            DO 7039 J=1,7
7039  MESHD(I+1,J)=MESHD(I,J)
            NE=NE+1
            MESHD(IEADD,4)=MESHD(IENO(K),4)
            GOTO (7051,7052,7053) LENO(K)
7051  MESHD(IEADD,1)=IPAD
            MESHD(IEADD,2)=MESHD(IENO(K),2)
            MESHD(IEADD,3)=MESHD(IENO(K),3)
            MESHD(IEADD,5)=MESHD(IENO(K),5)
            MESHD(IEADD,6)=MESHD(IENO(K),6)
            MESHD(IEADD,7)=0
            MESHD(IENO(K),1)=MESHD(IENO(K),1)
            MESHD(IENO(K),2)=IPAD
            MESHD(IENO(K),3)=MESHD(IENO(K),3)
            MESHD(IENO(K),5)=MESHD(IENO(K),5)
            MESHD(IENO(K),6)=0
            MESHD(IENO(K),7)=MESHD(IENO(K),7)
            GOTO 7060
7052  MESHD(IEADD,1)=IPAD
            MESHD(IEADD,2)=MESHD(IENO(K),3)
            MESHD(IEADD,3)=MESHD(IENO(K),1)
            MESHD(IEADD,5)=MESHD(IENO(K),6)
            MESHD(IEADD,6)=MESHD(IENO(K),7)
            MESHD(IEADD,7)=0
            MESHD(IENO(K),1)=MESHD(IENO(K),1)
            MESHD(IENO(K),2)=MESHD(IENO(K),2)
            MESHD(IENO(K),3)=IPAD

```

```

MESH(D(IENO(K),5)=MESH(D(IENO(K),5)
MESH(D(IENO(K),6)=MESH(D(IENO(K),6)
MESH(D(IENO(K),7)=0
GOTO 7060
7053 MESH(D(IEADD,1)=IPAD
MESH(D(IEADD,2)=MESH(D(IENO(K),2)
MESH(D(IEADD,3)=MESH(D(IENO(K),3)
MESH(D(IEADD,5)=0
MESH(D(IEADD,6)=MESH(D(IENO(K),6)
MESH(D(IEADD,7)=MESH(D(IENO(K),7)
MESH(D(IENO(K),1)=MESH(D(IENO(K),1)
MESH(D(IENO(K),2)=MESH(D(IENO(K),2)
MESH(D(IENO(K),3)=IPAD
MESH(D(IENO(K),5)=MESH(D(IENO(K),5)
MESH(D(IENO(K),6)=0
MESH(D(IENO(K),7)=MESH(D(IENO(K),7)
7060 CONTINUE
7061 CONTINUE
C
7040 CONTINUE
C
RETURN
C
400 IERR=400
RETURN
950 IERR=-1
RETURN
END
C
SUBROUTINE BOUNGR(LA, LB, NE, IERR, MESH(D)
C
C 境界条件値の書き換え
C
DIMENSION MESH(D(LB, 7)
C
IERR=0
C
650 WRITE(6, 6501)
6501 FORMAT(' Boundary Change ', /,
#' Input Node No. A & B !! 0=Return ')
READ(5, *) IPO1
IF(IPO1.EQ.0)GOTO 699
READ(5, *) IPO2
JJ=0
DO 6025 I=1, NE
IF((MESH(D(I, 1).EQ.IPO1).AND.(MESH(D(I, 2).EQ.IPO2))GOTO 6026
IF((MESH(D(I, 2).EQ.IPO1).AND.(MESH(D(I, 3).EQ.IPO2))GOTO 6027
IF((MESH(D(I, 3).EQ.IPO1).AND.(MESH(D(I, 1).EQ.IPO2))GOTO 6028
IF((MESH(D(I, 3).EQ.IPO1).AND.(MESH(D(I, 2).EQ.IPO2))GOTO 6027
IF((MESH(D(I, 2).EQ.IPO1).AND.(MESH(D(I, 1).EQ.IPO2))GOTO 6026
IF((MESH(D(I, 1).EQ.IPO1).AND.(MESH(D(I, 3).EQ.IPO2))GOTO 6028
6025 CONTINUE
GOTO 950
6026 IPO2=5
GOTO 6029
6027 IPO2=6
GOTO 6029
6028 IPO2=7

```

```

6029 WRITE(6,6030)MESH(I,IPO2)
6030 FORMAT(' BOUNDARY= ',I3,/, ' ex. 1=Conductor( 1)',/,
# 7X, '-1=Conductor(-1)',/,8X, '0=No setting',/,
# 8X, '2=Magmetical Wall',/,8X, '3=Electrical Wall',/,
# 8X, '4=Current area',/, ' Input New Condition !!')
READ(5,*)MESH(I,IPO2)
GOTO 650

C
699 CONTINUE
C
RETURN
950 IERR=-1
RETURN
END

C
SUBROUTINE DELGR(LA, LB, NP, NE, IERR, MESH, XP, YP)
C
C 指定したノードを隣接ノードに重ねる形で要素を消去する
C
DIMENSION XP(LA), YP(LA), MESH(LB, 7), IENO(2), LENO(2),
# INO1(7), INO2(7)
C
IERR=0
C
WRITE(6,5001)
5001 FORMAT(' Move Node A. to B',/, ' No.A & B Input !! ',/,
# ' 0=Return')
READ(5,*)IPO1
IF(IPO1.EQ.0)GOTO 501
READ(5,*)IPO2
C
IENO(1)=0
IENO(2)=0
JJ=0
DO 501 I=1, NE
IF((MESH(I, 1).EQ.IPO1).AND.(MESH(I, 2).EQ.IPO2))GOTO 502
IF((MESH(I, 2).EQ.IPO1).AND.(MESH(I, 3).EQ.IPO2))GOTO 502
IF((MESH(I, 3).EQ.IPO1).AND.(MESH(I, 1).EQ.IPO2))GOTO 502
IF((MESH(I, 3).EQ.IPO1).AND.(MESH(I, 2).EQ.IPO2))GOTO 502
IF((MESH(I, 2).EQ.IPO1).AND.(MESH(I, 1).EQ.IPO2))GOTO 502
IF((MESH(I, 1).EQ.IPO1).AND.(MESH(I, 3).EQ.IPO2))GOTO 502
IF((I.EQ.NE).AND.(JJ.NE.0))GOTO 503
GOTO 501
502 CONTINUE
JJ=JJ+1
IENO(JJ)=I
IF(JJ.EQ.2)GOTO 503
501 CONTINUE
GOTO 950

C
503 CONTINUE
DO 504 I=1, NE
DO 504 J=1, 3
504 IF(MESH(I, J).EQ.IPO1)MESH(I, J)=IPO2
C
NNE=0
DO 507 I=1, NE
IF( (I.EQ.IENO(1)).OR.(I.EQ.IENO(2)) )GOTO 507

```



```

      NNE=NNE+1
      DO 508 K=1,7
508      MESHD(NNE,K)=MESHD(I,K)
507 CONTINUE
      NE=NNE
C
      DO 510 I=1,NE
      DO 510 J=1,3
510 IF(MESHD(I,J).GT.IPO1)MESHD(I,J)=MESHD(I,J)-1
C
      DO 509 I=IPO1+1,NP
      XP(I-1)=XP(I)
509  YP(I-1)=YP(I)
      NP=NP-1
C
      RETURN
C
950 IERR=-1
      RETURN
      END
C
      SUBROUTINE DIAGGR(LA, LB, NE, IERR, MESHD)
C
      対角線の向きを変更する。
C
      DIMENSION MESHD(LB,7), IENO(2), ILNO(2), INO1(7), INO2(7)
C
      IERR=0
C
      WRITE(6,101)
101 FORMAT(' Change Form ?? ',/, ' Input Node No. !! 0=Return')
      READ(5,*)IPO1
      IF(IPO1.EQ.0)GOTO 400
      READ(5,*)IPO2
C
      JJ=0
      DO 10 I=1,NE
      IL=0
      IF((MESHD(I,1).EQ.IPO1).AND.(MESHD(I,2).EQ.IPO2)) IL=1
      IF((MESHD(I,2).EQ.IPO1).AND.(MESHD(I,1).EQ.IPO2)) IL=1
      IF((MESHD(I,2).EQ.IPO1).AND.(MESHD(I,3).EQ.IPO2)) IL=2
      IF((MESHD(I,3).EQ.IPO1).AND.(MESHD(I,2).EQ.IPO2)) IL=2
      IF((MESHD(I,3).EQ.IPO1).AND.(MESHD(I,1).EQ.IPO2)) IL=3
      IF((MESHD(I,1).EQ.IPO1).AND.(MESHD(I,3).EQ.IPO2)) IL=3
      IF(IL.NE.0) THEN
      JJ=JJ+1
      IENO(JJ)=I
      ILNO(JJ)=IL
      IF(JJ.EQ.2)GOTO 11
      ENDIF
10 CONTINUE
      GOTO 950
C
11 IF(MESHD(IENO(1),4).NE.MESHD(IENO(2),4)) GOTO 960
C
      DO 20 I=1,2
      IF(ILNO(I).EQ.1) THEN
      INO1(1)=MESHD(IENO(I),3)

```

```

    MESHD(IENO(I),3)=MESHD(IENO(I),2)
    MESHD(IENO(I),2)=MESHD(IENO(I),1)
    MESHD(IENO(I),1)=INO1(1)
    INO1(1)=MESHD(IENO(I),7)
    MESHD(IENO(I),7)=MESHD(IENO(I),6)
    MESHD(IENO(I),6)=MESHD(IENO(I),5)
    MESHD(IENO(I),5)=INO1(1)
ENDIF
IF(ILNO(I).EQ.3) THEN
    INO1(1)=MESHD(IENO(I),1)
    MESHD(IENO(I),1)=MESHD(IENO(I),2)
    MESHD(IENO(I),2)=MESHD(IENO(I),3)
    MESHD(IENO(I),3)=INO1(1)
    INO1(1)=MESHD(IENO(I),5)
    MESHD(IENO(I),5)=MESHD(IENO(I),6)
    MESHD(IENO(I),6)=MESHD(IENO(I),7)
    MESHD(IENO(I),7)=INO1(1)
ENDIF
20 CONTINUE
C
DO 30 I=1,7
    INO1(I)=MESHD(IENO(1),I)
    INO2(I)=MESHD(IENO(2),I)
30 CONTINUE
C
MESHD(IENO(1),1)=INO1(2)
MESHD(IENO(1),2)=INO2(1)
MESHD(IENO(1),3)=INO1(1)
MESHD(IENO(1),4)=INO1(4)
MESHD(IENO(1),5)=INO2(7)
MESHD(IENO(1),6)=INO1(6)
MESHD(IENO(1),7)=INO1(5)
C
MESHD(IENO(2),1)=INO2(2)
MESHD(IENO(2),2)=INO1(1)
MESHD(IENO(2),3)=INO2(1)
MESHD(IENO(2),4)=INO2(4)
MESHD(IENO(2),5)=INO1(7)
MESHD(IENO(2),6)=INO2(6)
MESHD(IENO(2),7)=INO2(5)
C
RETURN
C
C
400 IERR=400
    RETURN
950 IERR=-1
    RETURN
960 IERR=-1
    WRITE(6,*) ' Material is not same !!!'
    RETURN
END
C
SUBROUTINE ELEBGR(LA,LB,NE,MESHD,XP,YP)
C
C   境界条件ごとに色を塗り分ける
C
DIMENSION XP(LA),YP(LA),MESHD(LB,7),XXX(4),YYY(4),IXY(4)

```

```

C
CALL GSLWSC(3.0) ! 線幅選択
DO 757 I=1,NE ! 境界線の表示
DO 758 J=1,3
IF(MESHD(I,J+4).EQ.0)GOTO 758
IF(MESHD(I,J+4).EQ.-1)ICL=7
IF(MESHD(I,J+4).EQ.1)ICL=4
IF(MESHD(I,J+4).EQ.2)ICL=3
IF(MESHD(I,J+4).EQ.3)ICL=2
CALL GSPLCI(ICL) ! 線の色選択
IXY(1)=MESHD(I,J)
IXY(2)=MESHD(I,J+1)
IF(J.EQ.3)IXY(2)=MESHD(I,1)
DO K=1,2
XXX(K)=XP(IXY(K))
YYY(K)=YP(IXY(K))
ENDDO
CALL GPL(2,XXX,YYY)
758 CONTINUE
757 CONTINUE
RETURN
END

C
SUBROUTINE ELEMGR(LA, LB, NE, MESHD, XP, YP)
C
C 要素の媒質種類ごとに色分けする。
C
DIMENSION XP(LA), YP(LA), MESHD(LB, 7), XXX(4), YYY(4), IXY(4)
C
CALL GSLWSC(1.0)
CALL GSFAIS(3)
CALL GSFASI(24)
C
DO 10 I=1, NE
IXY(1)=MESHD(I, 1)
IXY(2)=MESHD(I, 2)
IXY(3)=MESHD(I, 3)
IXY(4)=MESHD(I, 1)
ICR=MESHD(I, 4)+1
CALL GSFACI(ICR)
DO 11 J=1, 4
XXX(J)=XP(IXY(J))
11 YYY(J)=YP(IXY(J))
CALL GFA(4, XXX, YYY)
10 CONTINUE
C
RETURN
END

C
SUBROUTINE LINEGR(LA, LB, NE, WLINE, XP, YP, MESHD)
C
C メッシュの輪郭を作図する。
C
DIMENSION XP(LA), YP(LA), MESHD(LB, 7), XXX(4), YYY(4)
C
C ライン幅の指定
CALL GSLWSC(WLINE)
C

```

```

DO 10 I=1,NE
  DO 11 J=1,3
    XXX(J)=XP(MESHD(I,J))
11    YYY(J)=YP(MESHD(I,J))
    XXX(4)=XXX(1)
    YYY(4)=YYY(1)
    CALL GPL(4,XXX,YYY)
10 CONTINUE
C
  RETURN
  END
C
  SUBROUTINE MESHE(LA, LB, NP, NE, FREQ1, FREQ2, FREQ3, SK, DSK,
# SKM1, SKM2, XP, YP, MESHD, IERR)
C
C   ディスクからデータを読み取る
C
C ***** 配列宣言 *****
C   DIMENSION XP(LA), YP(LA), MESHD(LB, 7)
C
C   IDATA0=11
C   OPEN(IDATA0, FILE='FOR050.DAT', STATUS='OLD')
C   READ(IDATA0, *) NP, NE, FREQ1, FREQ2, FREQ3
C   READ(IDATA0, *) SK, DSK, SKM1, SKM2
C   READ(IDATA0, *) (XP(J), J=1, NP)
C   READ(IDATA0, *) (YP(J), J=1, NP)
C   READ(IDATA0, *) (MESHD(J, 4), J=1, NE)
C   READ(IDATA0, *) (MESHD(J, 1), MESHD(J, 2), MESHD(J, 3), J=1, NE)
C   READ(IDATA0, *) (MESHD(J, 5), MESHD(J, 6), MESHD(J, 7), J=1, NE)
C   CLOSE(IDATA0)
C
C   RETURN
C   END
C
  SUBROUTINE MESHF(LB, LA, NP, NE, FREQ1, FREQ2, FREQ3, SK, DSK,
# SKM1, SKM2, XP, YP, MESHD, IERR)
C
C   メッシュ簡易作成プログラム
C
C [変数リスト]
C *****
C   INOMA=50      入力するノードの最大数
C   IBLMA=60      入力する四角形ブロックの最大数
C   IBUN=10       入力要素の各辺の最大分割数
C   LB=(メインから) 出力できる分割後の三角要素の最大数
C   LA=(メインから) 出力できる分割後のノードの最大数
C *****
C 入力データ
C   SK            拡大縮小係数
C   FREQ1, FREQ2, FREQ3  計算周波数
C   DSK          計算領域分割数
C   SKM1, SKM2   金属の導電率
C   NNODE       入力ノード数
C   XX(INOMA)   入力ノードデータのX座標
C   YY(INOMA)   入力ノードデータのY座標
C   IBOUN1, IBOUN2, IBOUN3, IBOUN4  入力要素の辺の境界条件
C   NBLOK      入力要素数

```

```

C  NX,NY          入力要素の辺の分割数
C  AX(4),AY(4)   入力要素の4つのノードのX座標Y座標
C  X(IBUN,IBUN)  入力要素内での分割後のX座標
C  Y(IBUN,IBUN)  入力要素内での分割後のY座標
C  XP(IPONA)     出カノードデータのX座標
C  YP(LA)        出カノードデータのY座標
C  NOIN(INOMA)   入力ノードにおけるノード番号
C  ID(IBUN,IBUN) 入力要素内での分割後ノードの位置識別用メモリー
C  YED           入力要素の誘電率
C  NP            分割後ノード番号
C  NE            分割後要素番号
C  NUMBL         入力データの読み込み要素番号
C  XX1,YY1,XX2,YY2 計算時の内部メモリー
C  LINE(4,2)     共通辺の識別用メモリー
C  LNODE(IBLMA,4,IBUN) 辺上のノード番号登録用メモリー
C  MESH(LB,7)    分割後の要素のノード,誘電率,境界条件の登録
C  NODE(5,IBLMA) 入力要素のノード番号格納
C  I,J,JJ,K,N    制御用カウンター
C  IDATA0        入出力データのIO番号
C  IERR          エラーチェック
C
C  ***** 最大値の設定 *****
      PARAMETER (INOMA=50,IBLMA=60,IBUN=10)
C  *****
      DIMENSION X(IBUN,IBUN),Y(IBUN,IBUN),ID(IBUN,IBUN),
      #AX(4),AY(4),XX(INOMA),YY(INOMA),XP(LA),YP(LA),
      #LINE(4,2),LNODE(IBLMA,4,IBUN),
      #NOIN(INOMA),MESH(LB,7),NODE(5,IBLMA)
C  ***** 初期値の設定 *****
      NP=0
      NE=0
      IERR=0
C[100]***** ブロックデータの読み込み
      IDATA0=11
      OPEN(IDATA0,FILE='MESHDATA.DAT',STATUS='OLD')
C  ***** ブロックの節点座標を読み込む
      READ(IDATA0,*)SK,FREQ1,FREQ2,FREQ3
      READ(IDATA0,*)DSK,SKM1,SKM2
      READ(IDATA0,*)NNODE          ! NNODE --- ブロックの節点数
      IF(NNODE.LE.0)GOTO 900
      IF(NNODE.GT.INOMA)GOTO 901
      DO I=1,NNODE
      NOIN(I)=0
      ENDDO
      DO I=1,NNODE
      READ(IDATA0,*)XX(I),YY(I)    ! XX(I),YY(I) --- 節点座標
      ENDDO
C  ***** ブロックの構成を読み込む
      READ(IDATA0,*)NBLOK
      DO 199 JJ=1,NBLOK
      READ(IDATA0,*,END=900)NUMBL,NX,NY
      IF(NUMBL.NE.JJ)GOTO 900
      IF(NUMBL.GT.NBLOK)GOTO 900
      READ(IDATA0,*)(NODE(J,JJ),J=1,4)
      NODE(5,JJ)=NODE(1,JJ)
C  ----- ブロックの頂点座標を設定する
      DO J=1,4
      AX(J)=XX(NODE(J,JJ)) ! NODE(J,I) I番目のブロックでJ番目の辺

```

```

      AY(J)=YY(NODE(J,JJ))      !          のノード番号
      ENDDO
C ----- ブロックの誘電率を読み込む
C ***** 境界条件を読み込みデータを作成する
      READ(IDATA0,*)YED,IBOUN1,IBOUN2,IBOUN3,IBOUN4
CCC      YED=YED*100.
C ----- ブロック内分割後節点の座標を計算
      XX1=(AX(2)-AX(1))/FLOAT(NX)
      YY1=(AY(2)-AY(1))/FLOAT(NX)
      XX2=(AX(3)-AX(4))/FLOAT(NX)
      YY2=(AY(3)-AY(4))/FLOAT(NX)
      X(1,1)=AX(1)
      Y(1,1)=AY(1)
      X(1,NY+1)=AX(4)
      Y(1,NY+1)=AY(4)
      DO I=1,NX
      X(I+1,1)=X(I,1)+XX1      ! 第1辺と第3辺の分割後における
      Y(I+1,1)=Y(I,1)+YY1      ! 節点座標を求める
      X(I+1,NY+1)=X(I,NY+1)+XX2
      Y(I+1,NY+1)=Y(I,NY+1)+YY2
      ENDDO
      DO 101 I=1,NX+1
      XX1=(X(I,NY+1)-X(I,1))/FLOAT(NY)      ! 全節点における座標を計算する
      YY1=(Y(I,NY+1)-Y(I,1))/FLOAT(NY)      ! 結果は配列 X(I,J)と Y(I,J)
      DO J=2,NY
      X(I,J)=X(I,J-1)+XX1
      Y(I,J)=Y(I,J-1)+YY1
      ENDDO
101 CONTINUE
C ----- すでに入力されたブロックと重なる辺を判定する
      DO 102 J=1,2
      DO 102 I=1,4
102 LINE(I,J)=0
      DO 103 J=1,4
      DO 103 I=1,JJ-1
      DO 103 K=4,1,-1
      IF((NODE(J,JJ).NE.NODE(K+1,I)).OR.
      * (NODE(J+1,JJ).NE.NODE(K,I)))GOTO 103
      LINE(J,1)=I
      LINE(J,2)=K      ! J番目の辺がIブロックのK辺と重なる
103 CONTINUE
C ----- 重なる辺を考慮して各ノードにナンバ付けをし、座標を取る。
      DO J=1,NY+1
      DO I=1,NX+1
      ID(I,J)=0
      ENDDO
      ENDDO
      IF(LINE(1,1).NE.0)THEN
      DO I=1,NX+1
      ID(I,1)=LNODE(LINE(1,1),LINE(1,2),NX+2-I)
      ENDDO
      ENDIF
      IF(LINE(2,1).NE.0)THEN
      DO J=1,NY+1
      ID(NX+1,J)=LNODE(LINE(2,1),LINE(2,2),NY+2-J)
      ENDDO
      ENDIF
      IF(LINE(3,1).NE.0)THEN

```

```

DO I=NX+1,1,-1
  ID(I,NY+1)=LNODE(LINE(3,1),LINE(3,2),I)
ENDDO
ENDIF
IF(LINE(4,1).NE.0)THEN
DO J=NY+1,1,-1
  ID(1,J)=LNODE(LINE(4,1),LINE(4,2),J)
ENDDO
ENDIF
IF(NOIN(NODE(1,JJ)).NE.0)ID(1,1)=NOIN(NODE(1,JJ))
IF(NOIN(NODE(2,JJ)).NE.0)ID(NX+1,1)=NOIN(NODE(2,JJ))
IF(NOIN(NODE(3,JJ)).NE.0)ID(NX+1,NY+1)=NOIN(NODE(3,JJ))
IF(NOIN(NODE(4,JJ)).NE.0)ID(1,NY+1)=NOIN(NODE(4,JJ))
DO 120 J=1,NY+1
DO 120 I=1,NX+1
  IF(ID(I,J).NE.0)GOTO 120
  NP=NP+1
  IF(NP.GT.LA)GOTO 903
  ID(I,J)=NP
  XP(NP)=X(I,J)
  YP(NP)=Y(I,J)

```

120 CONTINUE

C ----- ブロックの各辺上にある節点番号を登録する

```

DO I=1,NX+1
  LNODE(JJ,1,I)=ID(I,1)
  LNODE(JJ,3,I)=ID(NX+2-I,NY+1)
ENDDO
DO I=1,NY+1
  LNODE(JJ,2,I)=ID(NX+1,I)
  LNODE(JJ,4,I)=ID(1,NY+2-I)
ENDDO
NOIN(NODE(1,JJ))=ID(1,1)
NOIN(NODE(2,JJ))=ID(NX+1,1)
NOIN(NODE(3,JJ))=ID(NX+1,NY+1)
NOIN(NODE(4,JJ))=ID(1,NY+1)

```

C[130]----- 三角要素の頂点番号列を作成する

```

DO 130 J=1,NY
DO 130 I=1,NX
  NE=NE+1
  MESHD(NE,1)=ID(I,J)           ! MESHD(1)-MESHD(3) ノード番号
  MESHD(NE,2)=ID(I+1,J)         ! MESHD(4) 誘電率
  MESHD(NE,3)=ID(I,J+1)
  MESHD(NE,4)=YED
  MESHD(NE,5)=0.0
  IF(J.EQ.1)MESHD(NE,5)=IBOUN1
  MESHD(NE,6)=0.0
  MESHD(NE,7)=0.0
  IF(I.EQ.1)MESHD(NE,7)=IBOUN4
  NE=NE+1
  MESHD(NE,1)=ID(I+1,J+1)
  MESHD(NE,2)=ID(I,J+1)
  MESHD(NE,3)=ID(I+1,J)
  MESHD(NE,4)=YED
  MESHD(NE,5)=0.0
  IF(J.EQ.NY)MESHD(NE,5)=IBOUN3
  MESHD(NE,6)=0.0
  MESHD(NE,7)=0.0
  IF(I.EQ.NX)MESHD(NE,7)=IBOUN2

```

```

1300 FORMAT(2X,3(I4,2X))
130 CONTINUE
199 CONTINUE
    CLOSE(IDATA0)
    RETURN
C ***** テータ meshdata.dat 読み込み時エラー
900 WRITE(6,9001)
9001 FORMAT(' Error !! Data is nothing or incorrect',
* ' !! confirm [MESHDATA.DAT] !!')
    CLOSE(IDATA0)
    RETURN
901 WRITE(6,9002)INOMA,N
9002 FORMAT(' Error !! Main Node Over !! max=[',I3,
#'] you input [' ,I4,' ] !! ')
    GOTO 990
902 WRITE(6,9003)IBLMA,NUMBL
9003 FORMAT(' Error !! Main Block Over !! max=[',I3,
#'] you input [' ,I4,' ] !! ')
    GOTO 990
903 WRITE(6,9004)LA
9004 FORMAT(' Error !! Divide node Over !! max=[',I3,' ] !! ')
    GOTO 990
904 WRITE(6,9005)LB
9005 FORMAT(' Error !! Divide element Over !! max=[',I3,' ] !! ')
990 CLOSE(IDATA0)
    RETURN
    END
C
    SUBROUTINE MESHO(LA, LB, IO, NP, NE, FREQ1, FREQ2, FREQ3, SK, DSK,
# SKM1, SKM2, XP, YP, MESHD, IERR)
C
C     データをディスクに出力する。
C
C ***** 配列宣言 *****
    DIMENSION XP(LA), YP(LA), MESHD(LB, 7)
C
C     出力先ファイル名の設定
    IF( IO.EQ.50) THEN
        OPEN(IO, FILE='FOR050.DAT', STATUS='UNKNOWN')
    ELSE
        OPEN(IO, FILE='MESHD.DAT:1', STATUS='UNKNOWN')
    ENDIF
C
C     データの読み込み
    WRITE(IO, *) NP, NE, FREQ1, FREQ2, FREQ3
    WRITE(IO, *) SK, DSK, SKM1, SKM2
    WRITE(IO, *) (XP(J), J=1, NP)
    WRITE(IO, *) (YP(J), J=1, NP)
    WRITE(IO, *) (MESHD(J, 4), J=1, NE)
    WRITE(IO, *) (MESHD(J, 1), MESHD(J, 2), MESHD(J, 3), J=1, NE)
    WRITE(IO, *) (MESHD(J, 5), MESHD(J, 6), MESHD(J, 7), J=1, NE)
C
    CLOSE(IO)
C
    RETURN
    END
C
    SUBROUTINE NUMBGR(LA, NP, CIH, ICLTX, XP, YP)

```



```

C
C     ノード番号の表示を行う
C
C     CHARA1,CHARA2,CHARA3  表示用数字用メモリー
C     ICLTX                  文字の色設定
C     CIH                    表示文字のサイズ
C     NP                     ノード総数
C     XP(*),YP(*)           ノードのX,Y座標値
C
C     DIMENSION XP(LA),YP(LA)
C     CHARACTER*3 CHARA3
C     CHARACTER*2 CHARA2
C     CHARACTER*1 CHARA1
C
C     101 FORMAT(I1)
C     102 FORMAT(I2)
C     103 FORMAT(I3)
C
C     CALL GSTXCI(ICLTX)
C     CALL GSCHH(CIH)
C
C     DO 10 I=1,9
C         IF( I .GT. NP ) GOTO 20
C         WRITE(CHARA1,101)I
C     10  CALL GTX(XP(I),YP(I),CHARA1)
C         DO 11 I=10,99
C             IF( I .GT. NP ) GOTO 20
C             WRITE(CHARA2,102)I
C     11  CALL GTX(XP(I),YP(I),CHARA2)
C         DO 12 I=100,999
C             IF( I .GT. NP ) GOTO 20
C             WRITE(CHARA3,103)I
C     12  CALL GTX(XP(I),YP(I),CHARA3)
C
C     20 CONTINUE
C     RETURN
C     END
C
C     SUBROUTINE OFFGR
C
C     グラフィック画面の消去およびGKシステムオフ
C
C     IWKID=1
C     CALL GDAWK(IWKID)
C     CALL GCLWK(IWKID)
C     CALL GCLKS
C
C     RETURN
C     END
C
C     SUBROUTINE SETGR(GMINX,GMAXX,GMINY,GMAXY)
C
C     GKシステムオンおよびグラフィック画面の設定
C     ワールド座標系として(GMINX,GMINY),(GMAXX,GMAXY)を設定
C
C     IERR=6
C     ICONID=0
C     IWKID=1

```

```

ITRAN=1
CALL GOPKS(IERR)
CALL GQEWK(1,IERRIN,NUMBER,ITYPE)
CALL GOPWK(IWKID,ICONID,ITYPE)
CALL GACWK(IWKID)
CALL GQMD$ (ITYPE,IERRIN,DCUNIT,RX,RY,LX,LY)
CALL GSWN(ITRAN,GMINX,GMAXX,GMINY,GMAXY)
CALL GSV$P(ITRAN,0.0,1.0,0.0,1.0)
CALL GSELNT(ITRAN)
C
CALL GSWKWN(IWKID,0.0,1.0,0.0,1.0)
RX0=MIN(RX,RY)
RX0=RX0*0.9
RX2=RX*0.95
RX1=RX*0.95-RX0
RY1=0.0
RY2=RX0
CALL GSWKVP(IWKID,RX1,RX2,RY1,RY2)
CALL GUWK(IWKID,1)
C
RETURN
END
C
SUBROUTINE TABLGR(FREQ1,FREQ2,FREQ3,SK,DSK,SKM1,SKM2)
C
C   計算条件の設定を行う。
C
C   *****   周波数
WRITE(6,101)FREQ1,FREQ2,FREQ3
101 FORMAT(' Frequency start=',F9.3,'GHz',/,12X,' stop=',F9.3,'GHz',/,
# 12X,' step=',F9.3,'GHz',/)
WRITE(6,*) ' Change Frequency ?? yes=1'
READ(5,*)JOK
IF(JOK.EQ.1)THEN
WRITE(6,*) ' Input Frequency start,stop,step'
READ(5,*)FREQ1,FREQ2,FREQ3
ENDIF
C
C   *****   寸法の単位系
WRITE(6,102)SK
102 FORMAT('Magnification=',F9.6)
WRITE(6,*) ' Change Magnification ?? yes=1'
READ(5,*)JOK
IF(JOK.EQ.1)THEN
WRITE(6,*) ' Input Magnification'
READ(5,*)SK
ENDIF
C
C   *****   計算領域設定
WRITE(6,103)DSK
103 FORMAT('Cal. area =',F4.1)
WRITE(6,*) ' Change Cal. area ?? yes=1'
READ(5,*)JOK
IF(JOK.EQ.1)THEN
WRITE(6,*) ' Input Cal. area full=1 half=2'
READ(5,*)DSK
ENDIF
C

```

```

C *****  金属の種類
      WRITE(6,104)SKM1,SKM2 .
104  FORMAT(' Metal resistanse (-1),(+1)=' ,2(F7.3,2X),/,
# ' Au = 1.000, Cu = 0.693 Ag = 0.652 Ti = 17.21')
      WRITE(6.*) ' Change resistanse ?? yes=1'
      READ(5,*)JOK
      IF(JOK.EQ.1)THEN
          WRITE(6.*) ' Input resistanse -1 & +1 '
          READ(5,*)SKM1,SKM2
      ENDIF
C
      RETURN
      END

```

## 5 まとめ

2次元伝送線路の電気的特性を計算するため、ガラーキン法に基づいた有限要素法による線路の電磁界解析プログラムを作成した。この線路解析プログラムは、電磁波の伝搬方向に一律な閉じた損失のある伝送線路を解析し、伝送モードの位相定数・減衰定数・波長短縮率・実効誘電率・特性インピーダンスを求めることができる。また、多層・多媒質構造に対して適用できる。

さらに、本解析プログラムを用いて多層化MMIC用伝送線路の損失特性を解析し、新しい低損失伝送線路構造を提案した。

## 改訂記録等

1990年6月30日初版作成

### \*\*\*\*\* プログラム使用者へのお願い

このマニュアルは、FUJI XEROX J Star で作成しており、文書ファイルはフロッピーに納めています。プログラムの修正・改良等変更事項があった場合は、改訂記録を追加するとともに本文の修正も行って下さい。