

JAERI-M

6897

自己回帰モデリングに基づく多変数システムの
同定と動特性解析コード

—D Y S A C—

1977年1月

尾熊律雄・藤井義雄・渡辺光一

この報告書は、日本原子力研究所が JAERI-M レポートとして、不定期に刊行している研究報告書です。入手、複製などのお問い合わせは、日本原子力研究所技術情報部（茨城県那珂郡東海村）あて、お申しこしてください。

JAERI-M reports, issued irregularly, describe the results of research works carried out in JAERI. Inquiries about the availability of reports and their reproduction should be addressed to Division of Technical Information, Japan Atomic Energy Research Institute, Tokai-mura, Naka-gun, Ibaraki-ken, Japan.

自己回帰モデリングに基づく多変数システムの
同定と動特性解析コード - DYSAC -

日本原子力研究所東海研究所原子炉工学部

尾熊 律雄・藤井 義雄・渡辺 光一

(1976年12月25日受理)

自己回帰モデリングに基いた多変数システムの同定と動特性解析のための計算コード“DYSAC” (Dynamic System Analysis Code) の機能と使用法を主とし、それに若干の使用結果例を加えてまとめたものである。

この計算コードは、稼動しているシステムから得られる時系列データを用いて、自己回帰式の形で多次元線型モデルを同定し、得られたモデルによってシステム動特性を解析するためのもので、ステップ応答、周波数応答及びスペクトル解析の各計算サブルーチンプログラムによって構成されている。

本コードにおいて、多変量かつ大量の実験データを解析する際、その入出力データの処理を容易にし、さらに見通しよく解析を進めることができるようにハイブリッド計算機の機能及び端末装置を有効に利用して次のような工夫がなされている。

- 実験データはアナログデータレコーダから直接ハイブリッド計算機のアナログ部に入力して処理することができる。
- 計算結果は図形の形でディスプレイ面に表示でき、必要なものをハードコピーとして得ることができる。
- 一連の操作手順がディスプレイ面に描き出され、それに従って計算機との対話形式で解析を進めていくことができる。また必要なコントロールパラメータをキーボードから入力し、ケーススタディを行いながら目的とする解析を遂行することができる。

本コードはハイブリッド計算機用に作成されたものであるが、FORTRAN IV で記述されておりハイブリッド演算機能の部分を除けば、その汎用性は極めて高い。

Computer Code DYSAC for Identification and Dynamics Analysis
of Multivariable Systems based on the Autoregressive Model

Ritsuo OGUMA, Yoshio FUJII and Koichi WATANABE

Division of Nuclear Engineering, Tokai, JAERI

(Received December 25, 1976)

Usage is described of the computer code DYSAC (Dynamic System Analysis Code) developed for a hybrid computer for the identification and the analysis of system dynamics.

A multivariable linear dynamic system is identified based on the autoregressive model using the time series data obtained from a system in operation and the system dynamics thus identified are analyzed.

This code includes subroutines for the analysis of step response, frequency response, power spectrum, etc. In order to facilitate handling a large number of various experimental data and to perform the analysis in perspective, considerations for effective utilization of hybrid computer functions and terminal devices are taken in this code, such as;

- The experimental data record in a analog data recorder are directly input to the analog part of the hybrid computer.
- The computed results can be plotted on the graphic display and its hard copy is readily available.
- A series of messages for guidance is given on the display terminal by which the analysis through man-machine interactive computation can be performed.

Thus, the required results can be obtained by performing case studies for which necessary parameters are input through the keyboard and the results displayed are checked.

目 次

I	序 文	1
II	解析手法の概説と計算アルゴリズム	3
III	解析コードの概要	9
III-1	コードの構成	9
III-2	解析コードDYSAC1	10
2.1	メインプログラム	11
2.2	解析変数選択サブルーチン(FILID1)	11
2.3	相関計算サブルーチン(CORID1)	12
2.4	同定サブルーチン(FPEID1)	12
2.5	解析ルーチン選択サブルーチン(ANSEQ1)	12
2.6	オープンループ系ステップ応答計算サブルーチン(DECOL1)	12
2.7	フィードバック系ステップ応答計算サブルーチン(DECNV1)	13
2.8	周波数応答計算サブルーチン(FREQFN)	13
2.9	モデルフィットネスチェックサブルーチン(STATES)	13
2.10	残差相関, スペクトル計算サブルーチン(RESCOR)	14
III-3	解析コードDYSAC2	14
3.1	メインプログラム	15
3.2	入力の同定サブルーチン(FPEID2)	15
3.3	入力変数のスペクトル計算サブルーチン(SINPT2)	15
3.4	スペクトルプロットサブルーチン(PLOTS2)	15
3.5	フーリエ変換サブルーチン(FTABM2)	16
3.6	オープンループ周波数応答計算サブルーチン(FRFOP2)	16
3.7	出力スペクトル計算サブルーチン(RASPE2)	16
3.8	出力スペクトルプロットサブルーチン(SPLTO2)	16
3.9	フィードバックループ周波数応答計算サブルーチン(FRFEE2)	16
III-4	サブルーチンとデータファイルの関係	16
III-5	解析コード使用に当たっての制約条件	17
IV	解析コードの使用法	36
IV-1	DYSAC1の操作手順	36
IV-2	DYSAC2の操作手順	46
V	応用例	51
VI	結 論	62
VII	参考文献	62
VIII	アペンディックス	63
VIII-1	DYSAC1のコードリスト	63
VIII-2	DYSAC2のコードリスト	88

I 序 文

本計算コードは実験で得られた時系列データをもとにシステムの動特性を解析するために作られたもので、すでに報告されている相関、スペクトル解析コード(MLCOSP), JAERI-M 6252²⁾の姉妹編である。

時系列解析の手法は複雑かつ不規則に変動している観測データの系列をもとにシステムの特性を解析するための方法として古くから多くの分野で用いられて来た。その基本的構成はデータ処理とその結果をもとにしたシステムの特性の抽出から成り、手法の多くは近年の大型電子計算機を駆使した龐大なデータ処理を前提としている。これらの手法に関してそのデータ処理過程と解析についての最近の共通した特徴は統計的手法によるデータ処理、動特性モデルの同定及びこれをベースにしたシステム解析の各プロセスをシステム論的な観点から統一的にとらえて取扱っている点にある。さらに不規則な変動現象の中からシステムに関する情報を最大限抽出することと体系化された線型システム理論を背景に見通し良く解析を進めて行くためのモデル作りに力点が置かれている。ここに整備されたシステム同定と動特性解析のための計算コードは新しい手法の開発と上記の最近の傾向をふま次に述べる手順に従って解析を進めるよう構成されている。

i) 時系列データの統計的解析

与えられた多変数時系列データの相関、スペクトル解析、コヒーレンス計算等を行ない、変数相互間の関係と対象の動特性についての定性的評価を行なう。これらの結果をもとに以下の解析に使われる変数の選択、サンプリング間隔の決定等が行われる。このステップの計算には相関、スペクトル計算コード“MLCOSP”が用いられる。

ii) システムモデルの同定

与えられた変数の組について、i) で得られた多次元相関計算結果に対する自己回帰モデルフィッティングを行ない離散型多次元線型システムモデルを同定する。

iii) 動特性の解析

同定されたモデルをもとにシステムの動特性を解析する。ダイナミックシステムについて線型システム表現が与えられるとこれを用いて時間領域あるいは周波数領域での表現に相互変換が可能であり、対象が持っている個有の動特性を希望の領域で解析できる。与えられたシステムが多変数系であればシステムの内部構造についても議論することができる。またノイズの特性についてもこれがシステムモデルの中に反映されていればシステムとノイズを分離表現すること、逆にシステムとノイズの相互関係を明らかにすることが可能となる。本動特性解析コードではこれらの大半の問題が処理できるように作られており、ステップ応答解析、スペクトル解析、周波数応答解析、ノイズ解析の各サブルーチンが含まれ必要なルーチンについてはオープンループ系とフィードバック系に分けて解析できるようになっている。

ここでシステムモデルの同定とは、対象とするシステムに働きかけて得られた入出力変数についての観測データに対し、その動的な関係をよく説明するフィッティングモデルを求めることをいう。従来から原子炉および原子力プラントに対しても例えば動特性に関して組み立てた

理論の妥当性を実験的に確認する方法として、理論解析が困難な特性を実験的に評価する方法として、稼動している原子炉が設計通りの特性を持っているかどうか実験的に評価する方法として、さらには原子炉の安定度の推定や制御パラメータの最適調整のための手法として多くの問題に同定手法が適用されて来た。またそのための新しい手法の開発と原子炉への適用の試みもなされて来た。この様に同定手法の応用が期待される分野は動特性に関連した広い範囲にわたっており、これらに応えられるより一般性と適用性のある同定手法の開発とその原子炉への実用化は極めて大きな意義を持っている。本解析コードではこの点に留意し、以下に述べるような点で従来用いられた同定手法と異なる新しい手法が用いられている。即ち、

- 多変数システムの解析が前提となっている。
- 同定されるモデルは自己回帰 (Autoregressive ; A-R) 表現で与えられ、これを用いてシステムの特性とノイズの特性が同時に評価できる。また有色ノイズに対して適用が可能である。

○フィードバック系の解析が可能である。

- システムモデル決定のための評価関数がすぐれており思考錯誤的な決定を必要としない。

本計算コードでは、例えば実験で収録された多変量データの統計的処理、システムモデルの同定、これをベースにした時間領域、周波数領域での動特性解析等複雑かつ大量のデータ処理と解析を計算機利用者が手軽にかつ見通し良く行なえるようにプログラムには種々の工夫がなされている。主な特徴をあげてみると次の通りである。

- 本計算コードは当研究所原子炉制御研究室に設置されている E A I ハイブリッド計算機用に作られており、これに備わっている機能および端末装置を有効に利用できるようなプログラム構成ができています。例えば解析結果の大半のものはディスプレイターミナルに図形の形で出力し、さらに必要であれば、その図形をハードコピーで取ることも可能である。
- 計算上の種々のパラメータはすべてディスプレイ面に映し出された一連の解析結果を目で確かめながらキーボードから入力できるようになっておりケーススタディが容易にできる。
- 利用者が一連の解析を容易に進めて行けるようにプログラムはすべて対話形式で書かれており必要な操作手順およびコメントはすべてディスプレイ面に表示される。

本解析コードを用いてすでにいくつかのデータ解析もなされておりその有効性が確かめられている。^{4),5)} この報告書は時系列データの解析を必要とするさらに多くの利用者が本解析コードを手軽に使えるようにその使用法を中心にまとめたものである。II章では利用者が本解析手法の内容と応用の意義を理解できるよう、コードに組込まれた計算アルゴリズムとそれらの間の関係を要約して述べてある。III, IV章では解析コードの内容と実際の解析を進めて行く場合の手順と操作法を詳しく述べている。V章では解析を進めて行く上での参考として本解析コードを用いてなされた J P D R 同定実験データの解析の中から結果のいくつかを引用しておいた。尚アペンディックスとして本コードのリストを掲載しプログラム全体が参照できるようにしてある。

II 解析手法の概説と計算アルゴリズム

ここでは与えられた時系列データ $\{X(k); k=1, 2, 3, \dots, N\}$ に対する自己回帰 (A-R) モデルの同定と動特性解析に必要な計算アルゴリズムについて述べる。後述するように A-R モデルのあてはめは、時系列データの相関計算処理によって得られる共分散関数に対して成されるが、相関解析については“相関、スペクトル解析コード (MLCOSP)” JAERI-M 6252 に述べられているのでここでは割愛する。

II-1 自己回帰モデルの同定

与えられた時系列が

$$x(k) = \sum_{m=1}^M a(m) x(k-m) + \epsilon(k) \quad \dots\dots\dots (1)$$

によって表現される時これを自己回帰過程という。ここで $\{\epsilon(k)\}$ はシステムを駆動するノイズ源で

$$* \begin{cases} E \cdot \epsilon(k) = 0 \\ E \cdot \epsilon(k) \epsilon(k-l) = \sigma^2 \cdot \delta(l) \end{cases}$$

なる白色ガウス雑音である。また $\{a(m)\}$ はシステムの特徴を示す係数である。(1)式で示されるように A-R 表現では出力 $\{x(k)\}$ の係列は雑音源 $\epsilon(k)$ とシステムの特徴 $\{a(m)\}$ の二つのものの寄与によってもたらされることを意味している。(1)式より

$$\epsilon^2(k) = x^2(k) - 2 \sum_{m=1}^M a(m) x(k) x(k-m) + \sum_{m=1}^M \sum_{l=1}^M a(l) a(m) x(k-m) x(k-l)$$

平均化操作をして共分散関数を $R(m)$ で表わすと

$$R_{\epsilon\epsilon}(0) = R_{xx}(0) - 2 \sum_{m=1}^M a(m) R_{xx}(m) + \sum_{m=1}^M \sum_{l=1}^M a(l) a(m) R_{xx}(m-l) \quad \dots\dots\dots (2)$$

ここで時系列データとシステムの次数 M が与えられると(2)式に相等する平均 2 乗誤差

$$\overline{\epsilon^2(k)} = \frac{1}{N} \sum_{k=1}^N \epsilon^2(k)$$

を最小にするように $\{a_M(m)\}$ を決めることができる。時系列データから求めた $x(k)$ の自己共分散関数を $C_{xx}(l)$ とすると最小 2 乗法の解は

$$\sum_{m=1}^M C_{xx}(l-m) a_M(m) = C_{xx}(l) \quad (l=1, 2, \dots, M) \quad \dots\dots\dots (3)$$

で与えられ係数推定値 $\{\hat{a}_M(m)\}$ を求めることができ A-R モデルの同定が可能となる。ここでシステムの次数 M の決定であるが赤池は最小 2 乗法を適用して得られる係数の推定値を用いて予測を行なった場合の one step prediction の誤差の 2 乗平均値の推定値を求めこれを最小にするような M を採用することを提案している。即ち、長さ N のデータに最小 2 乗法を適用して得られる $\{a(m)\}$ の推定値を別の新しい同一構造のデータに適用した時の one step prediction error の 2 乗平均値を Final Prediction Error (略して FPE) と定義

* $E \cdot$ は期待値操作を表わす。

するとMが真の次数より大きければ

$$FPE(M) = \left(1 + \frac{M}{N}\right) \sigma^2 \dots\dots\dots (4)$$

によって近似的に与えられる。最小2乗法を適用して得られた $\{a_M(m)\}$ の推定値を $\{\hat{a}_M(m)\}$ とし

$$S_M^2 = \frac{1}{N} \sum_{k=1}^N \left(x(k) - \sum_{\ell=1}^M \hat{a}_M(\ell) x(k-\ell) \right)^2 \dots\dots\dots (5)$$

とすると

$$\hat{\sigma}^2 \sim \left(1 - \frac{M}{N}\right)^{-1} S_M^2 \dots\dots\dots (6)$$

で与えられる。(6)を(4)に代入して

$$FPE(M) = \left(1 + \frac{M}{N}\right) \left(1 - \frac{M}{N}\right)^{-1} S_M^2 \dots\dots\dots (7)$$

がMの適否の評価基準として用いられる。Mが小さすぎるとモデルの適合不良となり S_M^2 が σ^2 に比して大きくなりMが大きすぎると推定精度の低下にもとづく誤差が大きくなる。以上の結果は多次元A-Rモデル同定に容易に拡張できる。k次元定常時系列に対して

$$X(s) = \sum_{m=1}^M A(m) X(s-m) + U(s) \dots\dots\dots (8)$$

を多次元自己回帰表現という。ここでU(s)は白色ガウス雑音で

$$\begin{cases} E \cdot U(s) = 0 \\ E \cdot U(s) U^T(s-\ell) = \Sigma \cdot \delta(\ell)^* \dots\dots\dots (9) \end{cases}$$

とする。多次元モデルの同定においても係数の推定は(3)式に対応した代数方程式の解法に帰着させられる。しかし、ここでは次に示す計算アルゴリズムがA-Rモデル同定に用いられこれによって一連の複雑な計算をいっきに処理することができる。即ち、 $A_M(m)$ ($m=1, 2, \dots, M$), d_M を $M=1, 2, 3, \dots, L$ に対して逐次、次式を適用して計算する。

$$\left\{ \begin{array}{l} d_M = C(0) - \sum_{m=1}^M A_M(m) C^T(m) \\ e_M = C(M+1) - \sum_{m=1}^M A_M(m) C(M+1-m) \\ f_M = C(0) - \sum_{m=1}^M B_M(m) C(m) \\ D_M = e_M f_M^{-1} \dots\dots\dots (10) \\ E_M = e_M^T d_M^{-1} \\ A_{M+1}(m) = A_M(m) - D_M B_M(M+1-m) \quad m = 1, 2, \dots, M \\ \quad \quad \quad = D_M \quad \quad \quad m = M+1 \\ B_{M+1}(m) = B_M(m) - E_M A_M(M+1-m) \quad m = 1, 2, \dots, M \\ \quad \quad \quad = E_M \quad \quad \quad m = M+1 \end{array} \right.$$

* Tは転置を表わす。

同時に

$$MFPE(M) = \left(1 + \frac{MK+1}{N}\right)^K \left(1 - \frac{MK+1}{N}\right)^{-K} \|d_M\| \dots\dots\dots (11)$$

を計算し、 $M=1, 2, 3, \dots, L$ の中で最小の $MFPE(M)$ を与える M をシステムの次数として採用し、この時の $A_M(m)$ がA-Rモデルの係数の推定値を与える。また $\|d_M\|$ は d_M の行列式の値を示す。この時の d_M は白色雑音 Σ の推定値を示している。

II-2 線型システムのA-R表現と他の表現との関係

同定されたA-Rモデルを用いてシステムの動特性解析が行われる。ここでは解析に使われる計算アルゴリズムとA-R表現との関係を述べる。

r 次元出力変数 $x(s)$ 、 ℓ 次元操作変数 $y(s)$ および r 次元システムノイズ $u(s)$ に対して次のシステム方程式を考える。

$$x(s) = \sum_{m=1}^M a(m) x(s-m) + \sum_{m=1}^M b(m) y(s-m) + u(s) \dots\dots\dots (12)$$

但し $\{a(m), b(m)\}$ はシステムマトリックスである。いま、

$$\begin{cases} A(m) \triangleq \begin{matrix} r \\ \ell \end{matrix} \left\{ \begin{matrix} \overbrace{a(m)}^r & \overbrace{b(m)}^\ell \\ * & * \end{matrix} \right\} \\ X(s) \triangleq \begin{matrix} r \\ \ell \end{matrix} \left\{ \begin{matrix} x(s) \\ y(s) \end{matrix} \right\} \dots\dots\dots (13) \\ U(s) \triangleq \begin{matrix} r \\ * \end{matrix} \left\{ \begin{matrix} u(s) \\ * \end{matrix} \right\}, \text{但し } r+\ell=K \end{cases}$$

とすると(12)式は(8)式で与えたA-R表現に帰着される。

$$X(s) = \sum_{m=1}^M A(m) X(s-m) + U(s) \dots\dots\dots (14)$$

$$\begin{cases} E U(s) = 0 \\ E U(s) U^T(s-\ell) = \Sigma \cdot \delta(\ell) \dots\dots\dots (15) \end{cases}$$

(i) インパルスまたはステップ応答表現

操作変数 $y(s)$ からみた出力 $x(s)$ へのインパルス応答重み系列を $\{h(m)\}$ とすると(12)式が与えられた場合、係数マトリックス $\{a(m), b(m)\}$ を用いてこれを求めることができる。

$$\begin{cases} h(1) = b(1) \\ h(s) = \sum_{j=1}^{s-1} a(j) h(s-j) + b(s) \quad \text{for } s \leq M \dots\dots\dots (16) \\ h(s) = \sum_{j=1}^M a(j) h(s-j) \quad \text{for } s > M \end{cases}$$

また(14)式において出力変数間のフィードバック効果については次の様な計算アルゴリズムが用いられる。 $x_j(s)$ を入力と見た時の $x_i(s)$ へのインパルス応答の重み系列を $\{g_{ij}(m)\}$ ($i \neq j$)とすると

$$\begin{cases} g_{ij}(1) = A_{ij}(1) \\ g_{ij}(m) = A_{ij}(m) + \sum_{\ell=1}^{m-1} A_{ii}(\ell) g_{ij}(m-\ell) \end{cases} \dots\dots\dots (17)$$

但し $A_{ij}(m) = 0$ for $m > M$

(17)式を(14)に代入して次式を得る。

$$v_i(s) = \sum_{\ell=1}^M A_{ii}(\ell) v_i(s-\ell) + u_i(s) \dots\dots\dots (18)$$

$$x_i(s) = \sum_{j=1}^M g_{ij}(m) x_j(s-m) + v_i(s) \dots\dots\dots (19)$$

(i+j)

(18)式より、A-R表現における係数マトリックスを用いて有色ノイズ $\{v_i(s)\}$ の特性をシステムの特性和分離して解析することができる。

(16)、(17)式がインパルスまたはステップ応答計算のアルゴリズムとして用いられる。また(18)式は後述するノイズ解析に使われる。

(ii) スペクトル解析

(12)式に対し(2)式で求めたのと同様にして平均化操作をほどこすと次の共分散関数とA-Rモデルとの関係式を得る。

$$\sum_{m=0}^M \sum_{n=0}^M a(m) R_{xx}(\ell+n-m) a^T(n) = \sum_{m=1}^M \sum_{n=1}^M b(m) R_{yy}(\ell+n-m) b^T(n) + R_{uu}(\ell) \dots\dots\dots (20)$$

但し $a(0) = -I$

フーリエ変換式

$$P_{xx}(f) = \sum_{\ell=-\infty}^{\infty} \exp(-i2\pi g\ell) \cdot R_{xx}(\ell), \quad (i^2 = -1)$$

を用いて(20)式を周波数領域表現に変換すると

$$a(f) P_{xx}(f) \overline{a(f)^T} = b(f) P_{yy}(f) \overline{b(f)^T} + \Sigma^* \dots\dots\dots (21)$$

但し

$$a(f) = \sum_{m=0}^M a(m) \exp(-i2\pi fm) \dots\dots\dots (22)$$

$$b(f) = \sum_{m=1}^M b(m) \exp(-i2\pi fm) \dots\dots\dots (23)$$

$$\therefore P_{xx}(f) = a(f)^{-1} b(f) P_{yy}(f) \overline{b(f)^T} (\overline{a(f)^T})^{-1} + a(f)^{-1} \Sigma (\overline{a(f)^T})^{-1} \dots\dots\dots (24)$$

この式によって出力のスペクトル推定が可能となる。またこの表現式から明らかなように出力のパワースペクトルは入力のパワースペクトル $P_{yy}(f)$ の寄与する部分とノイズパワー Σ の寄与する部分の和として表わされる。従って出力のスペクトルを分解してその成因別に寄与の割合を評価することが可能である。

* — は共役複素数をあらわす。

(iii) 周波数応答表現

(i)で述べたインパルス応答に対する周波数領域での表現式を求める。(12)式において入力 $y(s)$ との相関をとると

$$\sum_{m=1}^M a(m) R_{xy}(\ell-m) = \sum_{m=1}^M b(m) R_{yy}(\ell-m) \quad \dots\dots\dots (25)$$

(25)式をフーリエ変換して

$$a(f) P_{xy}(f) = b(f) P_{yy}(f) \\ \therefore G_o(f) = a(f)^{-1} b(f) \quad \dots\dots\dots (26)$$

但し $a(f)$, $b(f)$ は (22), (23) 式によって与えられる。

(24)式においてパワースペクトルのうちノイズの寄与する部分を $P_{Nxx}(f)$ とすると

$$P_{Nxx}(f) \triangleq a(f)^{-1} \Sigma (a(f)^T)^{-1} \quad \dots\dots\dots (27)$$

この時出力 x_j から見た x_i へのフィードバック系の周波数応答は

$$G_{Fij}(f) = - \frac{(P_{Nxx}(f)^{-1})_{ij}}{(P_{Nxx}(f)^{-1})_{ii}} \quad \dots\dots\dots (28)$$

で与えられる。

(26), (28) 式がオープンループ系及びフィードバック系の周波数応答計算に用いられる。

(iv) ノイズ解析

多次元 A-R 表現ではシステムマトリックスの対角要素 $\{a_{ii}(m)\}$ およびノイズモーメント Σ にノイズ特性に関する情報が含まれている。これを利用してノイズ成分だけを抽出して解析を進めることが可能である。

(18)式のように分離されたノイズ表現式に対して相関計算をすると、

$$R_{v_i v_i}(\ell) = \sum_{m=1}^M a_{ii}(m) R_{v_i v_i}(\ell-m) + R_{u_i u_i}(\ell) \quad \dots\dots\dots (29)$$

但し

$$R_{u_i u_i}(\ell) = \begin{cases} \sigma_{ii} & \ell=0 \\ 0 & \ell \neq 0 \end{cases}$$

また(29)式に対して、周波数領域で表現すると次のノイズスペクトル計算式を得る。

$$P_{v_i v_i}(f) = \frac{\sigma_{ii}}{|a_{ii}(f)|^2} \quad \dots\dots\dots (30)$$

(19)式においてシステムノイズが無相関であるとする、

$$\Sigma_{ij} = \sigma_{ij} \cdot \delta(i-j)$$

この時の式より

$$(P_{Nxx}(f))_{ii} = \sum_{j=1}^k |(a(f)^{-1})_{ij}|^2 \sigma_{jj} \quad \dots\dots\dots (31)$$

従って出力のパワースペクトルは各ノイズ源からの寄与分の合成によって表わされる。ノイズ源 u_j から出力 x_i へのパワースペクトルの寄与分を $q_{ij}(f)$ とすると

$$q_{ij}(f) = |a(f)^{-1}_{ij}|^2 \sigma_{jj} \quad \dots\dots\dots (32)$$

正味のパワースペクトルに対する次式の比率表現をしてこれをノイズ寄与率と定義する。

$$r_{ij} \triangleq \frac{q_{ij}(f)}{(P_{N_{xx}}(f))_{ii}}, \quad (i, j=1, 2, \dots, K) \quad \dots\dots\dots (33)$$

(29), (30), (31), (33)式がノイズ解析に使われる。

(V) モデルフィットネスチェック

同定された数式モデルが実際の現象をどの程度再現してくれるか。(I)~(IV)の解析を進める前にチェックしてやる必要がある。(12)式に対して、実験で与えられた入力変数についての実現値 $\{y(s); s=1, 2, \dots, N\}$ とシステムモデル $\{\hat{a}(m), \hat{b}(m)\}$ を用いて出力の推定値 $\hat{x}(s)$ を求め、これを実験で得た実現値 $x(s)$ と比較する。

$$\hat{x}(s) = \sum_{m=1}^M \hat{a}(m) \hat{x}(s-m) + \sum_{m=1}^M \hat{b}(m) y(s-m) \quad \dots\dots\dots (34)$$

$$e(s) = x(s) - \hat{x}(s)$$

またシステムの入出力についての実験データ $\{x(s), y(s); s=1, 2, \dots, N\}$ と同定されたシステムモデル $\{\hat{a}(m), \hat{b}(m)\}$ を用いて one step prediction を評価する。

$$\hat{x}(s) = \sum_{m=1}^M \hat{a}(m) x(s-m) + \sum_{m=1}^M \hat{b}(m) y(s-m) \quad \dots\dots\dots (35)$$

$$e(s) = x(s) - \hat{x}(s)$$

さらに残差成分 $e(s)$ の相関、スペクトル解析によって同定されたモデルの妥当性が評価される。

Ⅲ 解析コードの概要

Ⅲ-1 コードの構成

本解析コードはハイブリッド計算機のディスクメモリに格納された時系列データをもとにシステムモデルの同定およびこれにもとづくシステムの動特性解析，さらに解析結果の出力処理にいたるまでをディスプレイターミナルを介して，計算機との対話形式で進めて行けるように作成されている。本コード全体の構成とⅡ章で述べた計算アルゴリズムとの関係をFig.Ⅲ-1に示している。図から明らかな様にこのコードは大きく分けて次の5つの構成要素からできている。

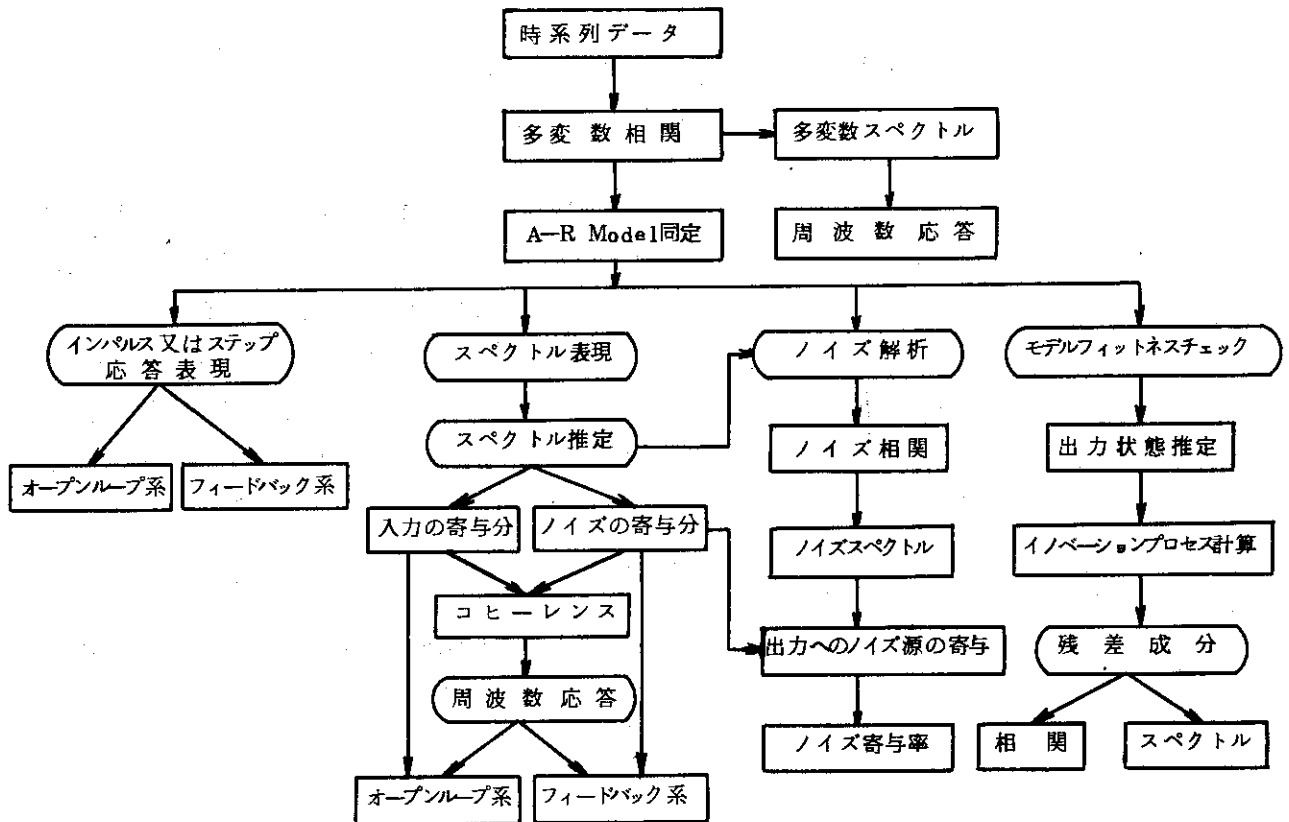


Fig. Ⅲ-1 Software construction of Dynamic System Analysis Code

(1) A-Rモデルの同定

ディスクメモリに収められているデータの中から解析に必要な変数分だけを選択し，これを用いて，多変数相関（共分散関数）計算，A-Rモデルの同定までを行なう。

(2) インパルス応答表現

オープンループ系およびフィードバック系のインパルス応答計算を行い，これをステップ応答表現に直して出力する。

(3) スペクトル表現

出力のパワースペクトル推定とその入力，ノイズの寄与分の分離，オープンループ系およびフィードバック系の周波数応答計算を行う。

(4) ノイズ解析

ノイズ相関、ノイズスペクトルおよびノイズ寄与率計算を行う。

(5) モデルフィットネスチェック

同定されたモデルを用いて出力状態推定、イノベーションプロセス計算を行い時系列データに対する再現性をチェックする。

これら一連の計算を同一の流れの中で進めて行くようにするため本コードではFig. III-1の各機能毎にサブルーチンに分割し、これらをディスクメモリに格納し、実行時に必要なサブルーチンをコアメモリに転送し実行する、いわゆるオーバーレイ方式を採用している。さらに計算機容量の関係から解析コードは次の2つに分割して構成されている。

(i) DYSAC1

(1) A-Rモデルの同定、(2)インパルス応答表現、(5)モデルフィットネスチェックの各計算プログラムで主として時間領域での解析が行われる。

(ii) DYSAC2

(3)スペクトル表現、(4)ノイズ解析の各計算プログラムで周波数領域での解析が行われる。

二つの解析コードは共にデータ処理のプロセスをコントロールするメインプログラムと解析のための各サブルーチンから出来ており、オーバーレイ方式で構成された全く独立のプログラムとなっていて、(1)で同定されたモデルの係数、次数等のデータはすべてディスクメモリに一旦格納される。(2)~(5)の各解析ではこれらのデータをコアに転送した後に計算が開始されるように作られている。従って、DYSAC2コードは必ずDYSAC1の(1)が終ったことを確認してからスタートすることになる。以下にコードの内容を二つの場合に分けて説明する。

III-2 解析コード DYSAC1

この解析コードに含まれる主要なサブルーチン及びその機能を要約して述べると次の通りである。

- メインプログラム (DYSAC1)
プログラム全体の流れのコントロール
- 解析変数選択サブルーチン (FILID1)
ディスクに格納されているデータの中から解析に必要な変数分だけを選択する。
- 相関計算サブルーチン (CORID1)
自己および相互共分散関数計算
- 同定サブルーチン (FPEID1)
A-Rモデルフィッティング、評価関数プロット
- 解析ルーチン選択サブルーチン (ANSEQ1)
前記(2)~(5)のいずれの解析を行なうかの選択
- オープンループ系ステップ応答計算サブルーチン (DECOL1)
オープンループ系ステップ応答計算および結果のディスプレイ面へのプロット
- フィードバック系ステップ応答計算サブルーチン (DECONV1)
フィードバック系ステップ応答計算および結果のディスプレイ面へのプロット

○周波数応答計算サブルーチン (FREQFN)

DECOL1あるいはDECNV1で求めたインパルス応答をフーリエ変換して周波数応答を計算

○モデルフィットネスチェックサブルーチン (STATES)

同定されたモデルを用いた出力計算

○残差相関スペクトル計算サブルーチン (RESCOR)

STATESで求めた出力推定値とそれに対応した変数の実験データとの差分を求め残差成分の相関スペクトルを計算

以下に各サブルーチンの機能を詳細に述べる。

Ⅲ-2.1 メインプログラム (DYSAC1)

メインプログラムはあらかじめ決められた順序に従ってサブルーチン化されたプログラムをディスクからコアメモリへ転送し、そのプログラムを実行開始するものであり、極めて簡単な構成になっている。本コードの実行中はこのメインプログラムのみがコアに常駐される。尚、各サブルーチンで計算された結果はすべて一旦ディスクに確保されたデータファイルに収められ、他のサブルーチン計算でこれを必要とする場合コアに転送して用いるようになっている。Fig.Ⅲ-2に本コードの構成を、またFig.Ⅲ-3ではメインプログラムとオーバーレイ方式によるサブルーチンのメモリ配置を示す。

Ⅲ-2.2 解析変数選択サブルーチン (FILID1)

DYSAC1コードをスタートするとまず、FILID1サブルーチンがオーバーレイワーキングエリアに転送されプログラムの実行が開始される。このサブルーチンでは

○入力パラメータの読込

N ; 1チャンネル当りのデータ個数

K ; 変数の数 (チャンネル数)

IR ; 被操作変数の数

DELT ; サンプリング間隔 (SEC単位)

○オリジナルデータファイルの中から解析に必要な変数を選択

が実行される。ここでオリジナルデータファイルは利用者があらかじめ解析に必要なデータを確保しておくファイルでその作成の仕方については後述する。このファイルはプラッタ1^{*}に定義され最大数、チャンネル当り3000個、15チャンネルまでのデータを確保することができる。このサブルーチンでは本解析に用いるデータをオリジナルデータファイルの中から選択し、変数の順序付けをしてこれをデータファイルTSDATAにストアするための操作を行なう。ファイルTSDATAは本サブルーチンの実行過程でプラッタ2に定義されるもので、解析に用いる変数分のデータだけがこれにストアされる。この時ストアされるデータの配列については被

* 参考文献2 P20参照

操作変数が前にくるように順序付けが成され、最大10変数まで取ることができる。尚、本サブルーチンでは、必要に応じて2つのオリジナルデータファイルからデータを選択できるように作成されており、最大30変数の中から選ぶことができる。このサブルーチンのフローチャートをFig.Ⅲ-4に示す。

Ⅲ-2.3 相関計算サブルーチン (CORID1)

解析に必要なデータがファイルTSDATAに確保されると本サブルーチンの実行が可能となる。このサブルーチンでは各変数毎に自己および相互共分散関数がLAG=0, 1, 2, ..., 20まで計算され、プラッタ2に定義されたファイルCODATAにストアされる。計算結果は必要に応じてラインプリンタに出力できる。本ルーチンの最終に各変数の平均値がディスプレイ面に表示される。Fig.Ⅲ-5にこのサブルーチンのフローチャートを示す。

Ⅲ-2.4 同定サブルーチン (FPEID1)

本サブルーチンでは共分散関数計算結果を用いてシステムモデルの同定が行われる。同定されたモデルの係数、次数等はこのサブルーチン実行開始時に定義されるファイルIDDATAにストアされる。本ルーチンの実行が終ると評価関数の形、システムモデルの次数等がディスプレイ面に出力される。又、モデルの係数、ノイズ2次モーメント推定値等がラインプリンタに出力される。本サブルーチンのフローチャートをFig.Ⅲ-6に示す。

Ⅲ-2.5 解析ルーチン選択サブルーチン (ANSEQ1)

このサブルーチンはⅢ-1で述べた(2)~(5)のいずれの解析を行なうかを選択するためのもので、そのフローチャートをFig.Ⅲ-7に示す。ここで解析内容については次に示す番号から必要なものを選択して行なうようになっている。

- 1 ; ステップ応答計算
- 2 ; スペクトル解析
- 3 ; ノイズ解析
- 4 ; モデルフィットネスチェック
- 5 ; 計算ジョブ終了

この番号のうち2又は3を選択する場合はここでジョブを終了してDYSAC2コードの実行を開始する。このプログラムは1~4の解析が終ると再度このルーチンに戻るようになっている。

Ⅲ-2.6 オープンループ系ステップ応答計算サブルーチン (DECOL1)

上記選択のうち1を入力するとこのサブルーチンの実行が開始される。ここで最初に選択した変数の中に操作変数を含まない場合はこのジョブをスキップして次のルーチンに移る。ディスプレイ面には、ステップ応答計算の際のステップ数ならびに計算結果をラインプリンタに出力するかどうかを入力するよう指示が出される。本ルーチンでの計算可能な最大ステップ数は400、従ってステップ応答時間は400*DELTA(sec)までである。計算終了と同時に応答

波形がディスプレイ面に画かれる。ステップ数を種々に変えてみたい場合はディスプレイ面に表示される指示に従ってこのサブルーチンの頭に戻って再度計算を行うことが可能である。本ルーチンのジョブがすべて終了すると、後述する周波数応答計算を実行するか否かの選択を行う。これを行う場合は次に実行するルーチンDECNV1をスキップして先に周波数応答計算を行う。本サブルーチンのフローチャートをFig.Ⅲ-8に示す。

Ⅲ-2.7 フィードバック系ステップ応答計算サブルーチン(DECNV1)

オープンループ系の解析が終了すると本サブルーチンの実行に入る。実行手順、プログラムの形式は全くオープンループ系の場合と同じである。このプログラムのフローチャートをFig.Ⅲ-9に示す。

Ⅲ-2.8 周波数応答計算サブルーチン(FREQFN)

本サブルーチンではⅢ-2.6または2.7の結果を用いて周波数応答計算が行われる。ジョブの実行に入るとまず与えられたステップ応答をインパルス応答に変換し、これをフーリエ変換して周波数応答を求める。得られた結果のうちゲイン特性はディスプレイ面にプロットされる。また必要に応じてラインプリンタにも出力することができる。本サブルーチンのフローチャートをFig.Ⅲ-10(a), (b)に示す。ジョブの実行が終ると再びⅢ-2.5 ANSEQ1に戻って他の解析内容を選択する。

Ⅲ-2.9 モデルフィットネスチェックサブルーチン(STATS)

本サブルーチンの実行が開始されると、ファイルIDDATAにストアされたデータがコアに転送され、これを用いて出力の推定がなされる。出力推定の方法は2通りあり1つは④式のように同定されたモデルと実験データのうち入力変数に関するものだけを与えて出力を推定する方法で、もう1つは⑤式のように入出力データすべてを与えて1ステップ先の出力状態を推定する方法である。本サブルーチンではキーボードからの入力によっていずれか一方の方法を選択できるようにしている。このサブルーチンを使って出力を推定し実際の出力である実験データと比較する場合、実験データ全体にわたって比較するのではなくFig.Ⅲ-11に示すようにその1部分を取り出して評価する。またコア容量の関係からデータをさらに100個単位のブロックに区切ってそれぞれを計算しディスクメモリにストアしながら順次ブロック毎に出力推定を行うようにしてある。ここで出力推定のための条件設定として次の入力をキーボードから与える。

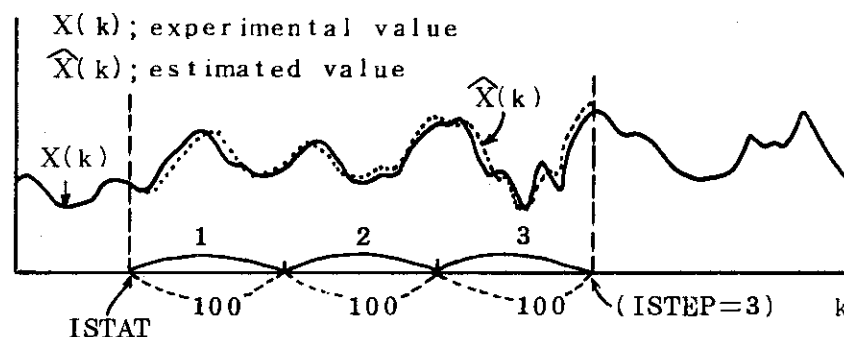


Fig.Ⅲ-11 Data length taken for the estimation(case;ISTEP=3)

ISTAT; 出力推定計算を行う場合の初期ステップを実験データのステップと対応させて与える。

ここでは $ISTAT > 20$ の制約条件がある。

この20個分のデータの確保は(84)または(85)式の初期条件として与えられるものであり、同定されるモデルの次数が最大20次で打切られていることによる。

ISTEP; 出力推定計算のステップ数で $ISTEP * 100$ ステップの計算を行なう。ここで $1 \leq ISTEP \leq 5$ なる制約条件がある。

即ち、

$ISTAT \leq \hat{X}(s) \leq ISTAT + ISTEP * 100$ なる $\hat{X}(s)$ について、これに対応した実験データと比較が行われる。出力推定結果はこれに対応した実験データと合わせてディスプレイ面に出力される。これによって直観的に同定精度を評価することができる。本サブルーチンのジョブを終えると残差成分について相関、スペクトル解析をするか否かの選択を行う。これを実行しない場合は、III-2.5 ANSEQ1にもどる。実行する場合は残差成分 $e(s)$ を計算してこのデータをディスクメモリにストアする。このサブルーチンのフローチャートをFig. III-12に示す。

III-2.10 残差相関、スペクトル計算サブルーチン (RESCOR)

2.9で残差相関計算を選択するとこのジョブが実行される。本サブルーチンの機能は直接残差相関計算をするものではなく、相関、スペクトル計算のためのデータファイルを作成することにある。データファイルが作成されると相関、スペクトル解析コードMLCOSPによってデータ解析が行われる。

サブルーチンSTATESで求めた残差系列データはプラッタ1に定義されたファイル“AAAAAA”にストアされる。次に、残差系列に対応した変数の実験データがこの後に全変数で15を越えない範囲でストアされる。Fig. III-13に本サブルーチンのフローチャートを示す。

III-3 解析コード DYSAC2

この解析コードに含まれる主要なサブルーチン及びその機能を以下に要約して述べる。

- メインプログラム (DYSAC2)
- 入力の同定サブルーチン (FPEID2)
 - 入力変数データに対するA-Rモデルの同定
- 入力のスペクトル計算サブルーチン (SINPT2)
 - 上記A-Rモデルを用いた入力変数のパワースペクトル計算
- スペクトルプロットサブルーチン (PLOTS2)
 - 入力変数 y , オープンループ系周波数応答およびフィードバック系周波数応答のディスプレイ面へのプロット。
- フーリエ変換サブルーチン (FTABM2)
 - 係数マトリックス $\{a(m), b(m)\}$ のフーリエ変換
- オープンループ周波数応答計算サブルーチン (FRFOP2)

○出力スペクトル計算サブルーチン (RASPE2)

出力パワースペクトルとそれに対する入力, ノイズの寄与分さらにノイズスペクトルを計算

○出力スペクトルプロットサブルーチン (SPLTO2)

○フィードバックループ周波数応答計算サブルーチン (FRFEE2)

以下に各サブルーチンの機能を詳細に述べる。

Ⅲ-3.1 メインプログラム (DYSAC2)

メインプログラムの機能はDYSAC1と全く同じである。このプログラムのフローチャートをFig.Ⅲ-14に示す。

Ⅲ-3.2 入力の同定サブルーチン (FPEID2)

このサブルーチンと次のスペクトル計算サブルーチンはⅡ章で述べたスペクトル計算式(2)式中 $P_{yy}(f)$ を計算するために用いられる。ここでは入力変数についてのA-Rモデルの同定が行われる。本サブルーチンの実行が開始されると, DYSAC1コードと全く同様に入力条件をキーボードからインプットする。ここで操作変数がない場合はプログラムは自動的にスキップして次のルーチンに移る。本サブルーチンの計算内容はDYSAC1コードのA-Rモデル同定サブルーチンFPEID1と全く同じである。このフローチャートをFig.Ⅲ-15に示す。

Ⅲ-3.3 入力変数のスペクトル計算サブルーチン (SINPT2)

このルーチンでは先に計算された入力変数に関するA-Rモデルの同定結果をフーリエ変換して入力についてのパワースペクトル計算が行われる。この計算式は(2), (3)と全く同じ式が用いられている。ここで操作変数が無い時は自動的にスキップして次のサブルーチンに移る。このサブルーチンの実行に入ると最初にスペクトル計算の周波数分割数LAGHをキーボードからインプットする。与えられたLAGHとデータのサンプル値間隔 Δt によってスペクトル計算の周波数 f が決まる。 $f = \ell / 2 \cdot \Delta t \cdot \text{LAGH}$; ($\ell = 1, 2, \dots, \text{LAGH}$)で

$$\frac{1}{2 \cdot \Delta t \cdot \text{LAGH}} \leq f \leq \frac{1}{2 \cdot \Delta t} \quad (\text{Hz})$$

の区間についてスペクトルが計算される。得られた結果はラインプリンタに出力するか, 次のサブルーチンでディスプレイ面に波形を画かせる。このサブルーチンのフローチャートをFig.Ⅲ-16に示す。

Ⅲ-3.4 スペクトルプロットサブルーチン (PLOTS2)

本サブルーチンは計算されたスペクトル密度関数, 周波数応答関数のゲイン特性をプロットするためのものである。プロットはすべて自動スケールで行われゲインおよび周波数軸については, Linear, Log, DBのうちから希望のスケールを選択できるようにしてある。このサブルーチンのフローチャートをFig.Ⅲ-17に示す。

Ⅲ-3.5 フーリエ変換サブルーチン (FTABM2)

本サブルーチンは同定されたモデルの係数マトリックスをフーリエ変換するためのもので、 ④ , ⑤ 式の計算アルゴリズムが用いられる。ここで計算される周波数点はⅢ-3.3で与えたLAGHで決まる。本サブルーチンのフローチャートをFig.Ⅲ-18に示す。

Ⅲ-3.6 オープンループ周波数応答計算サブルーチン (FRFOP2)

フーリエ変換された係数マトリックスを用いて ④ 式によってオープンループ系周波数応答計算を行なう。計算結果は一旦ディスクにストアされスペクトルプロットサブルーチン(PLOTS 2)によってこのうちゲイン特性がディスプレイ面にプロットされる。また必要に応じて周波数特性はラインプリンタに出力できる。本サブルーチンのフローチャートをFig.Ⅲ-19に示す。

Ⅲ-3.7 出力スペクトル計算サブルーチン (RASPE2)

本サブルーチンでは ④ 式を用いた出力のスペクトル計算、これを入力とノイズの寄与分に分離する作業およびノイズスペクトル推定が行われる。ここではⅢ-3.5と3.6の結果を用いてそれぞれの計算が行われる。本サブルーチンのフローチャートをFig.Ⅲ-20に示す。

Ⅲ-3.8 出力スペクトルプロットサブルーチン (SPLTO2)

本サブルーチンはRASPE2で得られた結果をプロットするためのものである。このジョブの実行に入ると次の様にプロットの内容の選択が行われる。

- 1 ; 出力パワースペクトルのうち入力の寄与する部分のプロット
- 2 ; ノイズの寄与する部分のプロット
- 3 ; ノイズスペクトルのプロット
- 4 ; 正味の出力パワースペクトルのプロット
- 99 ; ジョブの実行終了

このうちのいずれかを選択し、スペクトルプロットが終るとプログラムは再び上記選択の所に戻る。本サブルーチンのフローチャートをFig.Ⅲ-21に示す。

Ⅲ-3.9 フィードバックループ周波数応答計算サブルーチン (FRFEE2)

本サブルーチンでは ④ 式による出力変数間のフィードバックループ周波数応答計算が行われる。応答計算のための周波数点は入力スペクトル計算SINPT2サブルーチンの実行の際与えたLAGHで規定される。得られた結果は一旦ディスクにストアされ、このうちゲイン特性はサブルーチンPLOTS 2を用いてディスプレイ面に表示される。本サブルーチンのフローチャートをFig.Ⅲ-22に示す。

Ⅲ-4 サブルーチンとデータファイルの関係

本解析コードはすべて、各サブルーチンで計算された結果を一旦ディスクメモリにストアし

ジョブの進行につれて必要なものをコアに転送して用いるように作られている。従って各サブルーチンには必ずそのジョブの実行に必要なデータファイルが対応している。

Table III-1に本解析コードで用いているデータファイルの入出力関係を示す。

データファイルは基本的にはオリジナルデータファイルと他のファイルの2つに分けられる。前者はプラッタ1に作成されるもので、永久ファイルの性格を持つものである。後者は解析データのやりとりに使われるものでプラッタ2に作られる。

データファイルはこの計算コードの実行過程で定義されるようになっている。但しオリジナルデータファイル及び“BBBBBB”, “CCCCCC”, “DDDDDD”ファイルはMLCOSPコード内で定義される。データファイルは最初のジョブの実行時に一度定義しておけば良く、再実行の際データファイルを重複して定義するとエラーとなり、実行不可能となるので注意が必要である。

III-5 解析コード使用に当たっての制約条件

主にコアメモリの制約から本解析コードにはインプットパラメータにいくつかの制約がある。以下にジョブの実行手順に従ってまとめておく。

(i) DYSAC1コード

- $N \leq 3000$ データ個数
- $K \leq 10$ 変数の数
- $IR \geq 1$ 被操作変数の数
- $K = IR + IL$ IL ; 操作変数の数
- 同時に使用できるオリジナルデータファイルの数; 2つまで
- ステップ応答計算のステップ数; 400ステップまで
- モデルフィットネスチェックに使われるデータ個数について
- $ISTAT \geq 20$ 実験データ上での計算開始番地
- $1 \leq ISTEP \leq 5$ フィッティングチェックのためのデータ個数
- $LAGH \leq 200$ 周波数領域での解析における周波数分割数

Table III-1 Data file structure of DYSAC in the disc memory

コ ー ド	サ ブ ル ー チ ン	デ ー タ フ ァ イ ル									
		TSDATA	CODATA	IDDATA	オリジナル	AAAAAA	BBBBBB	CCCCCC	DDDDDD	EEEEEE	FFFFFF
	プラグNo.2	2	2	2	1	2	2	2	2	2	2
	123,200語	52800語	52800語	123,200語	123,200語	123,200語	237,600語	237,600語	123,200語	123,200語	123,200語
	定義	定義	定義	○					定義	定義	定義
D Y S A C 1	解析変数選択	●									
	相関計算	○	●	定義							
	同定		○	○							
	オープンループ系 ステップ応答計算			○					●		
	フィードバックループ系 ステップ応答計算			○					●		
	周波数応答計算									○	●
	モデルフィットネス チェック	○		○						●	●
	残差相関 スペクトル計算	○				定義				○	
	入力同定		○								
	入力のスペクトル計算								●		
	スペクトルプロット									○	
	フーリエ変換			○					●		
	オープンループ系 周波数応答計算									○	●
出力スペクトル計算			○						○	○	
出力スペクトルプロット			○						○		
フィードバックループ系 周波数応答計算									○	●	

注) 定義 : ディスクメモリにデータファイルを確認する。定義するか否かはすべて選択可能
 ●印 : データファイルへの書込み
 ○印 : データファイルからの読出し
 ◎印 : 書込み+読出し
 ※印 : MLCOSPコードで定義される

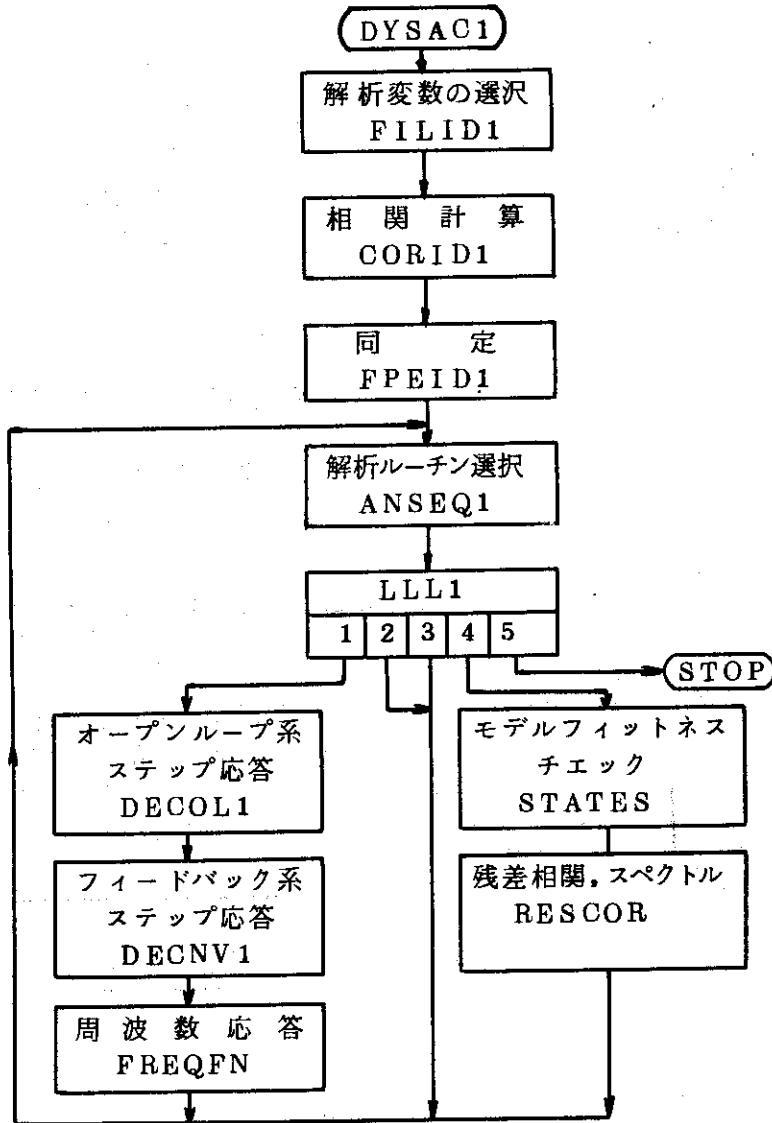


Fig. III-2 Flow Diagram of DYSAC1 Main program

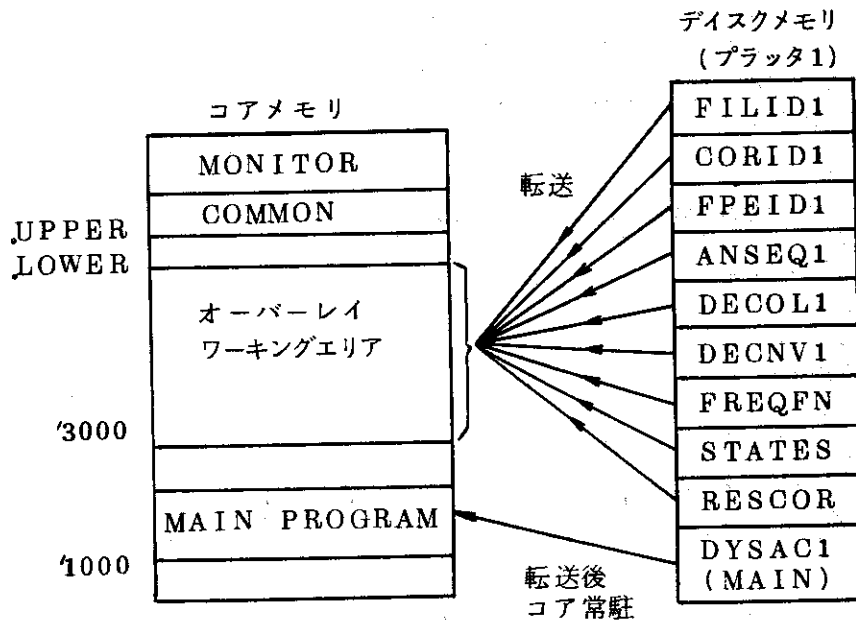


Fig. III-3 Memory Allocation of DYSAC1

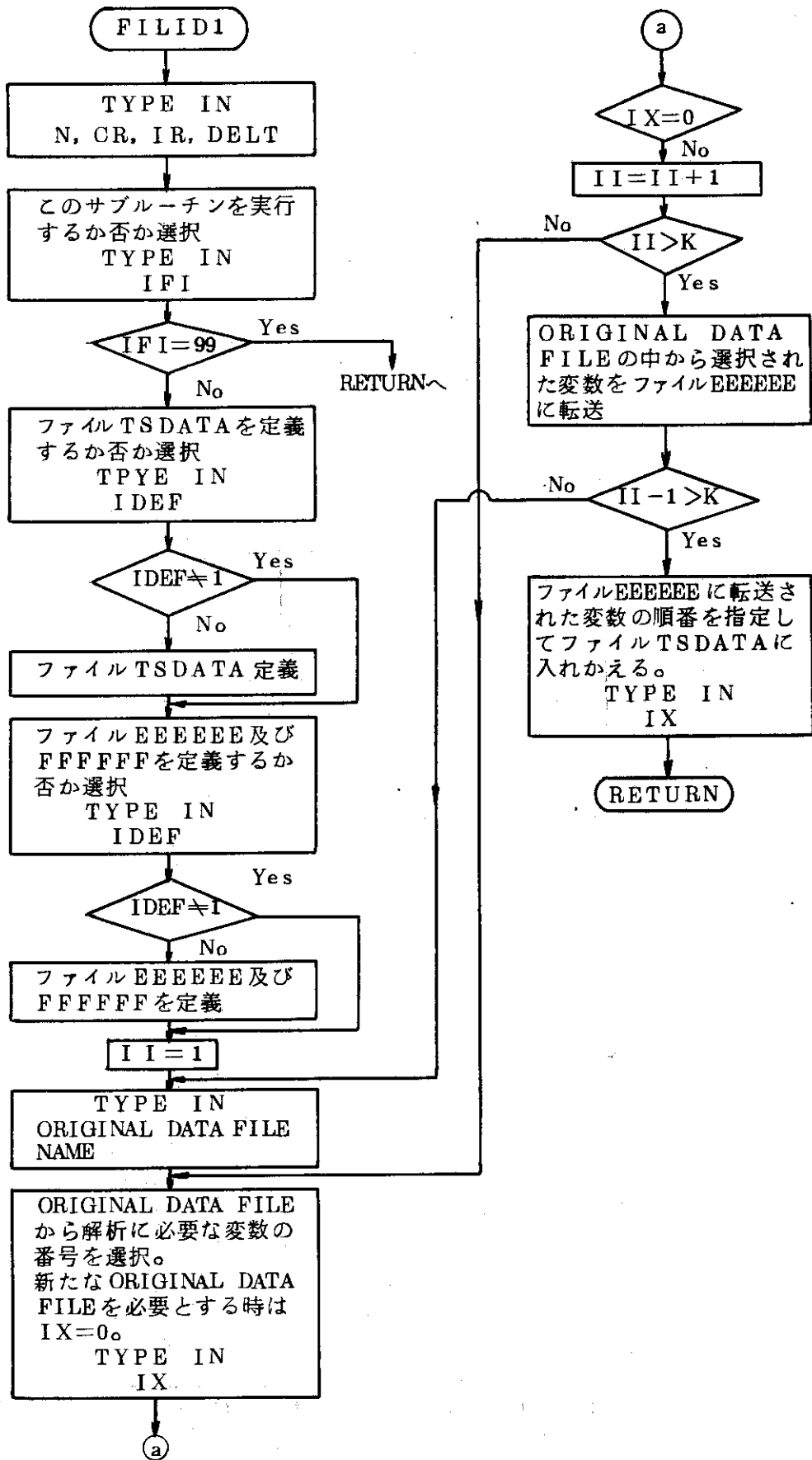


Fig. III-4 Flow Diagram of FILID1 Subroutine

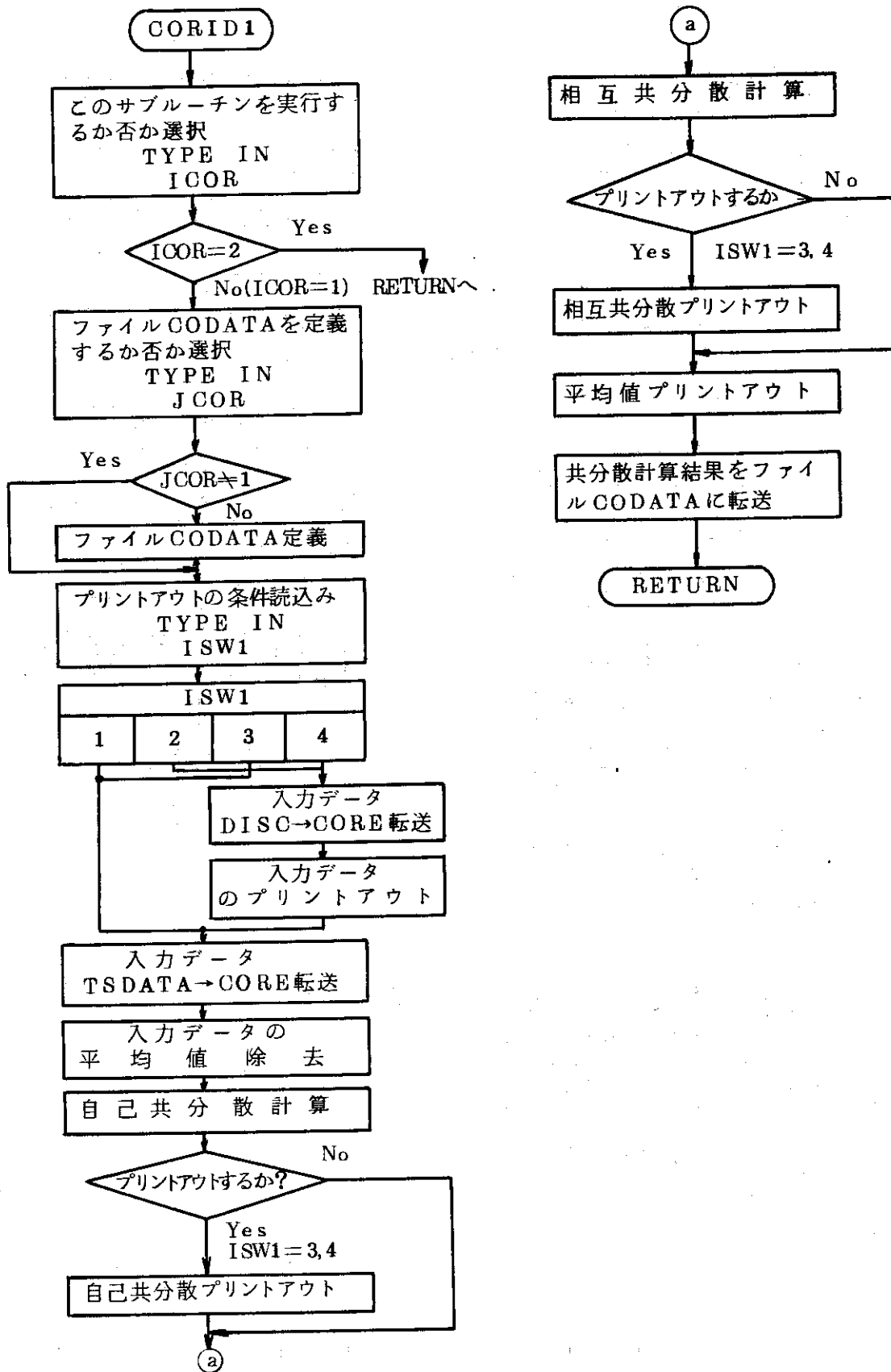


Fig. III-5 Flow Diagram of CORID1 Subroutine

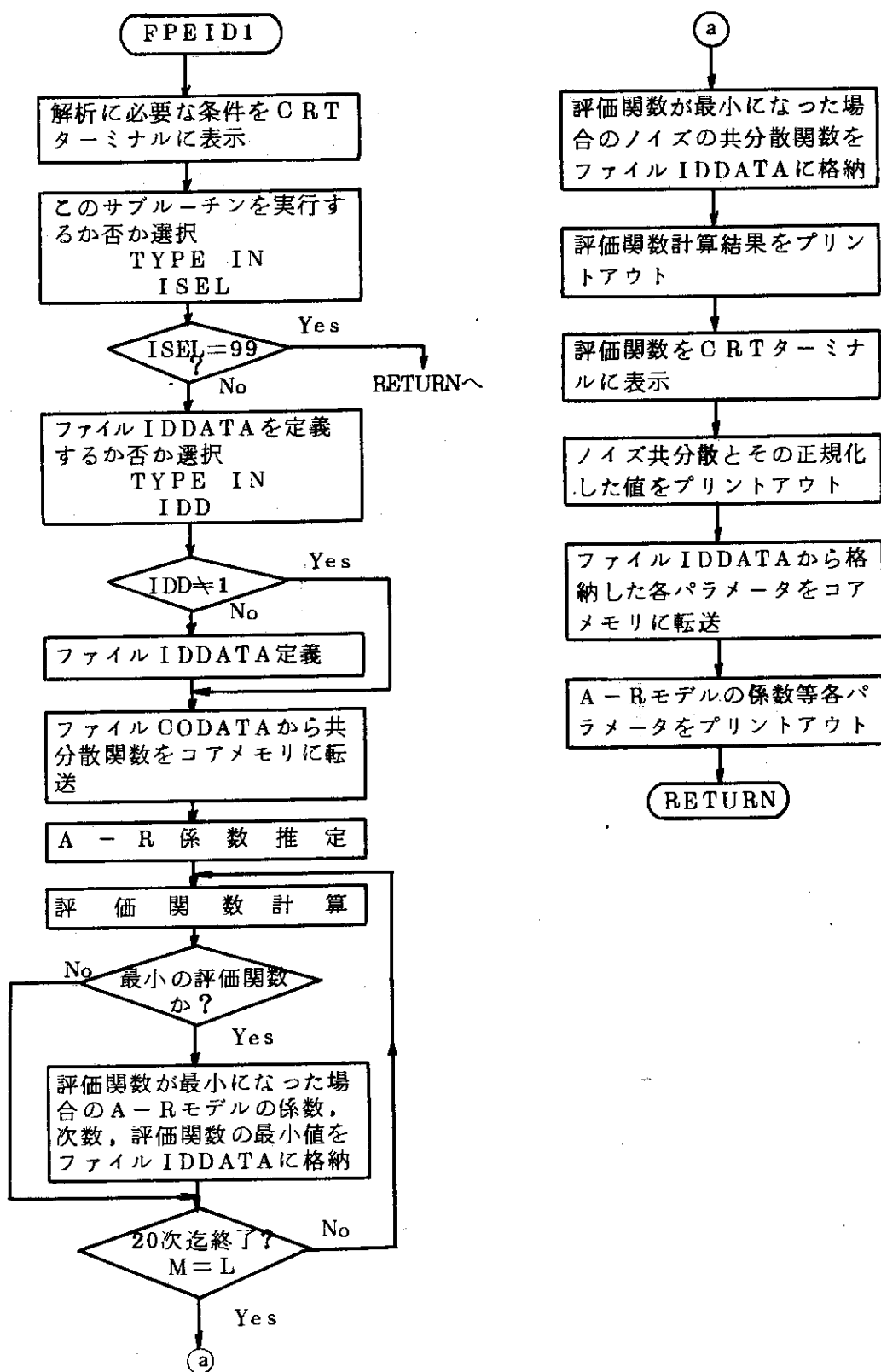


Fig. III-6 Flow Diagram of FPEID1 Subroutine

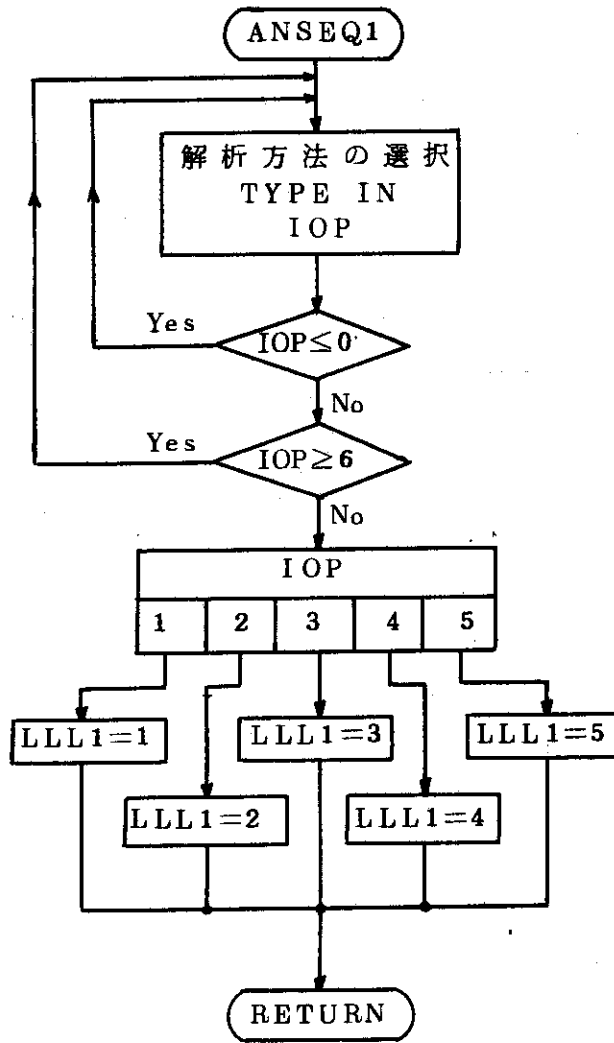


Fig. III-7 Flow Diagram of ANSEQ1 Subroutine

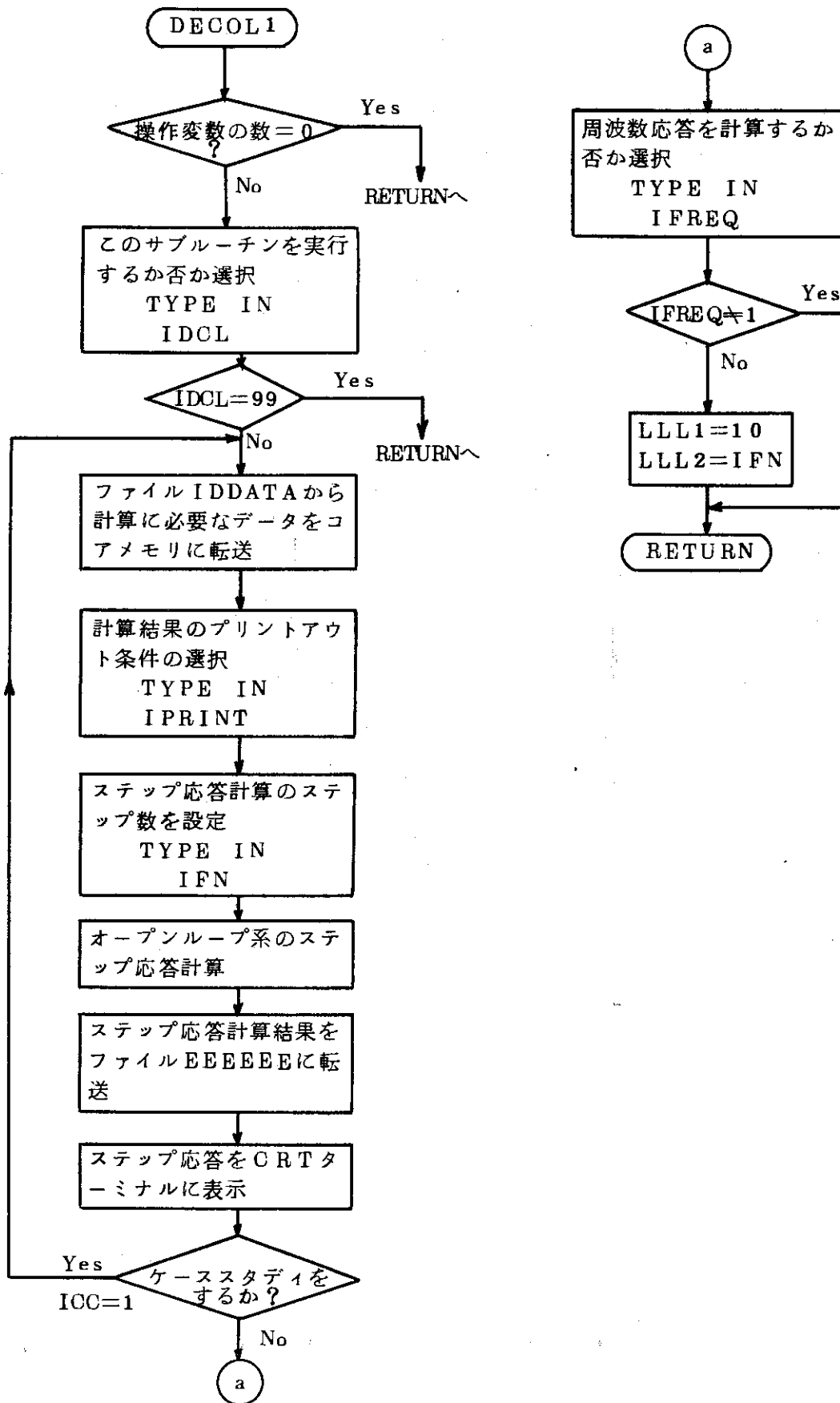


Fig. III-8 Flow Diagram of DECOL1 Subroutine

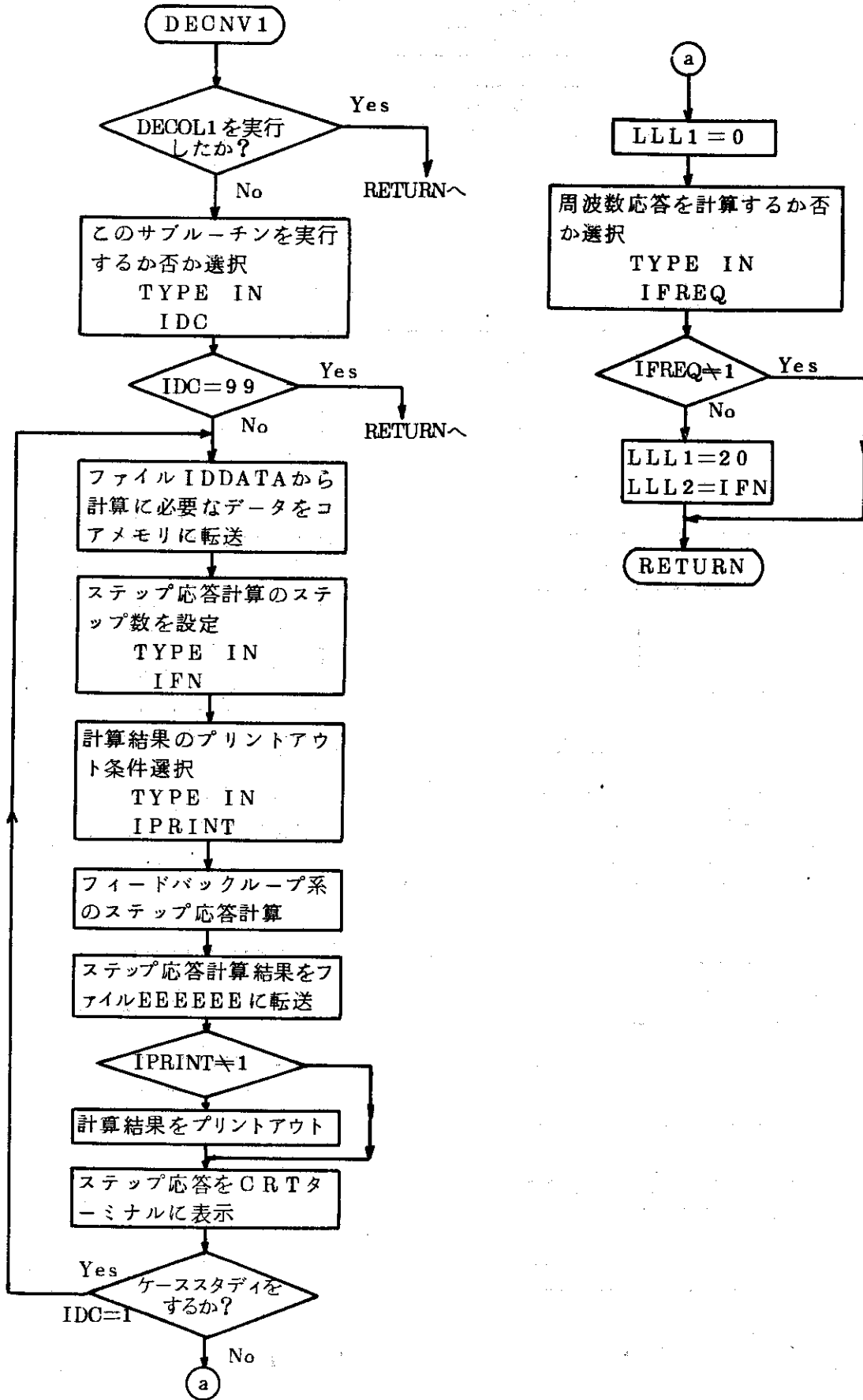


Fig. III-9 Flow Diagram of DECONV1 Subroutine

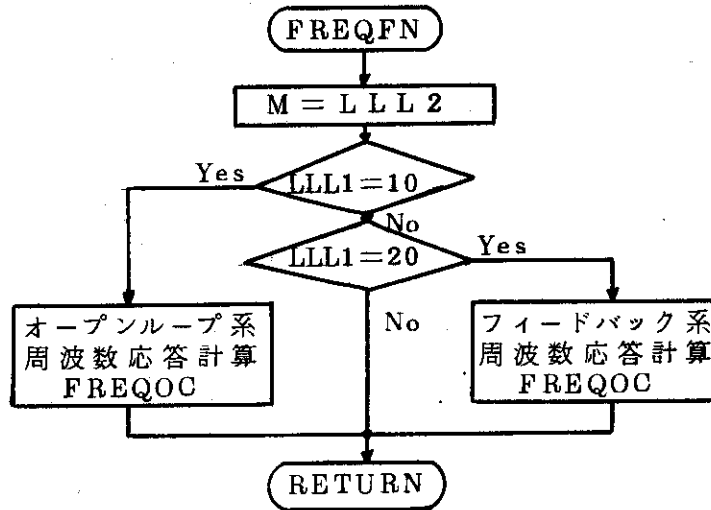


Fig. III -10 (a) Flow Diagram of FREQFN Subroutine

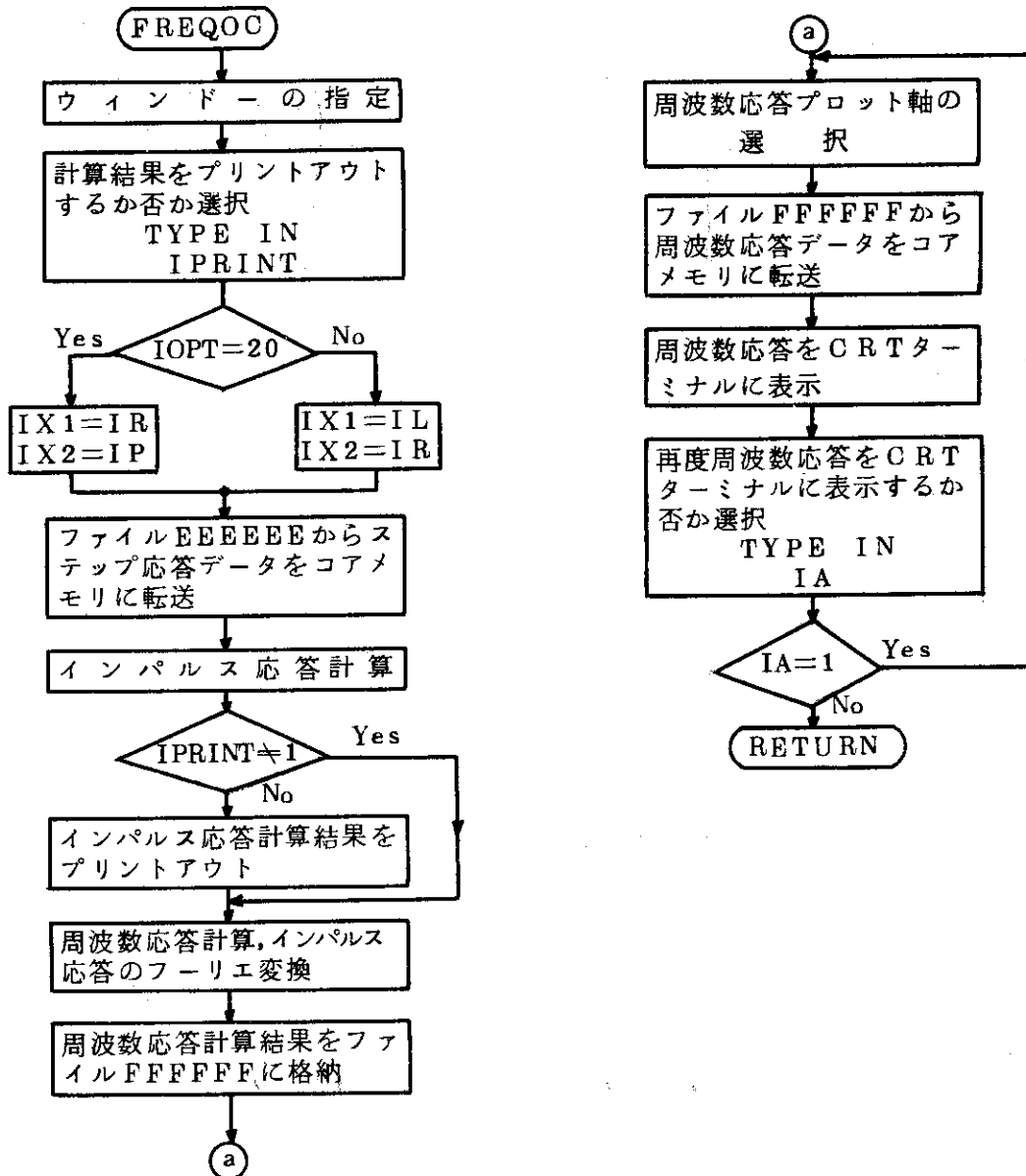


Fig. III-10 (b) Flow Diagram of FREQOC Subroutine

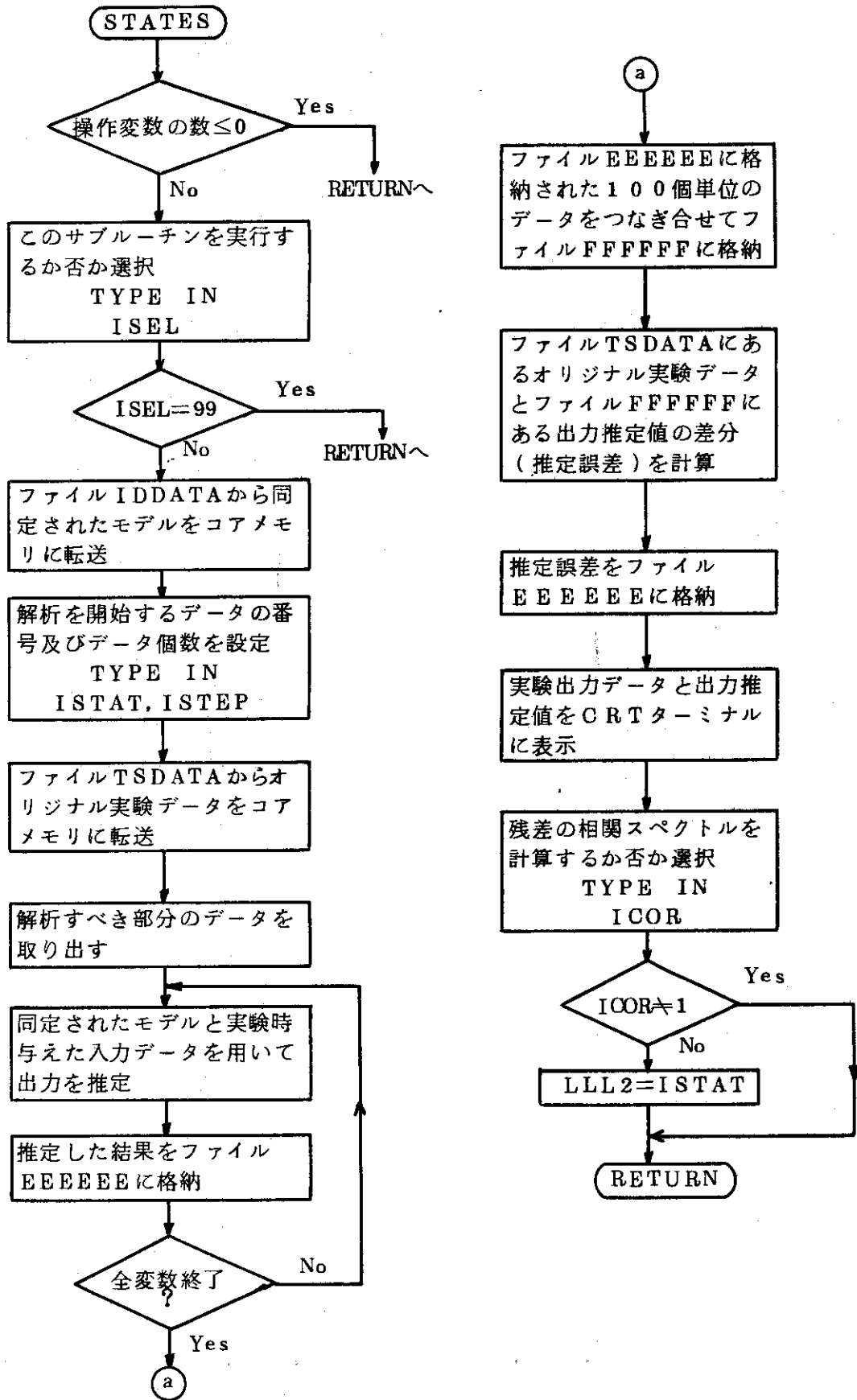


Fig. III-12 Flow Diagram of STATES Subroutine

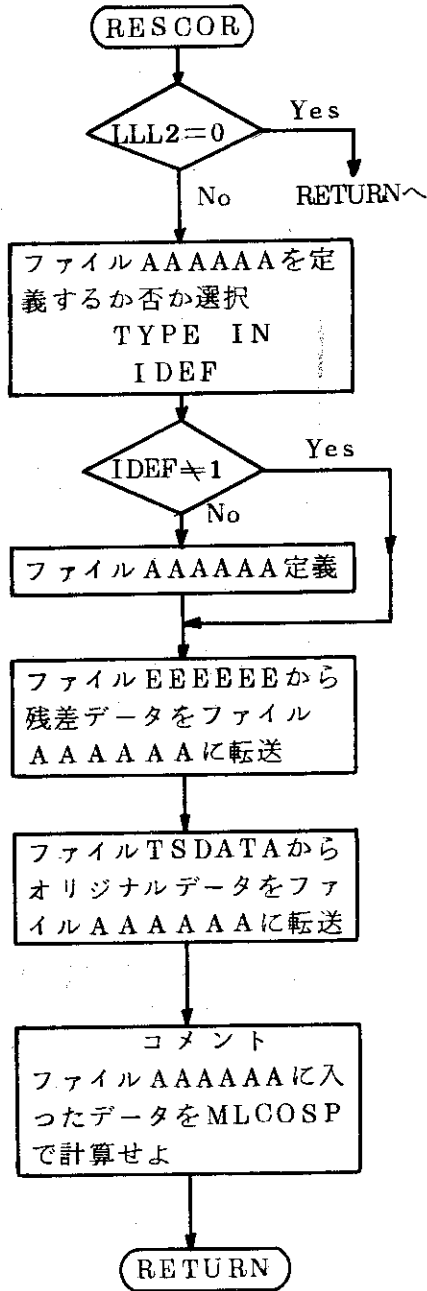


Fig. III-13 Flow Diagram of RESCOR Subroutine

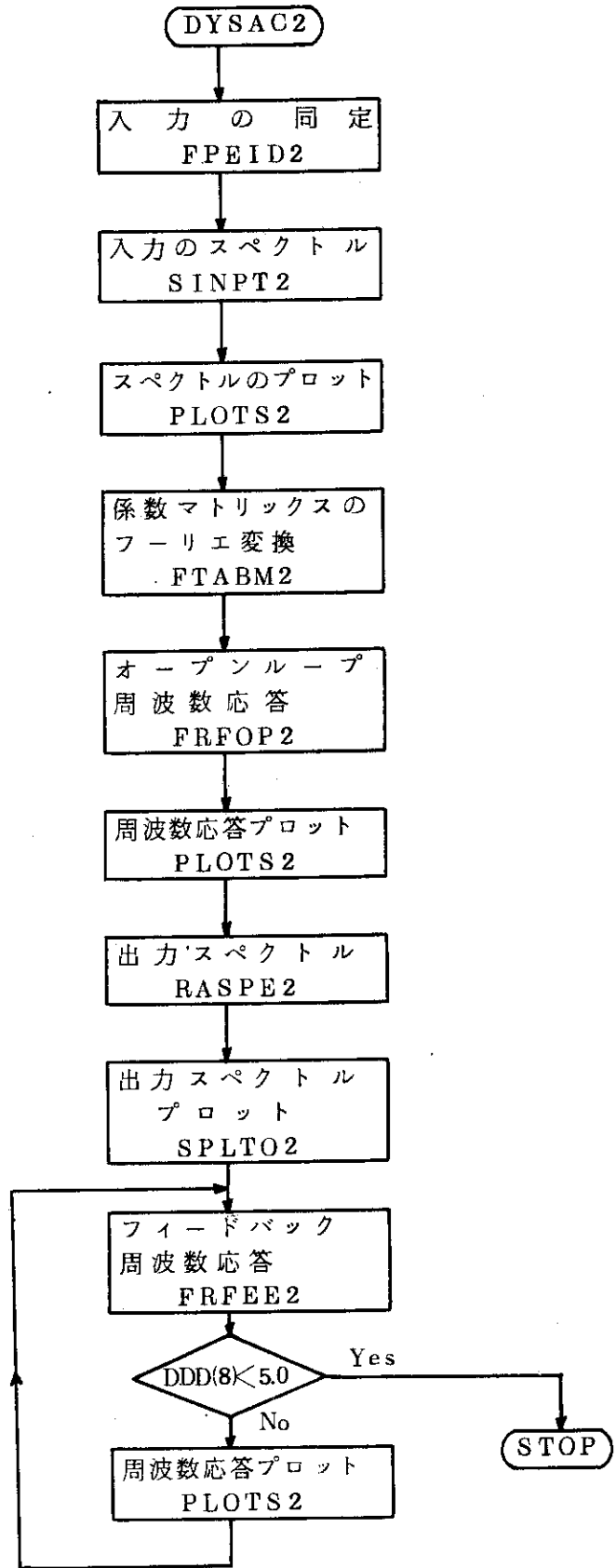


Fig. III-14 Flow Diagram of DYSAC2 Main program

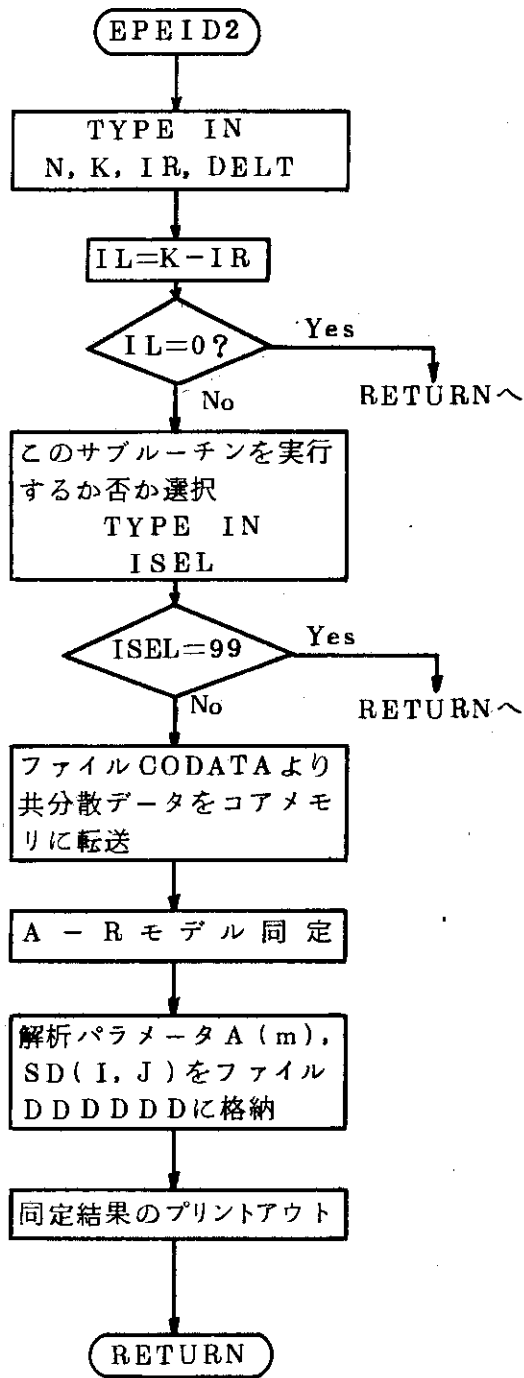


Fig. III-15 Flow diagram of FPEID2 Subroutine

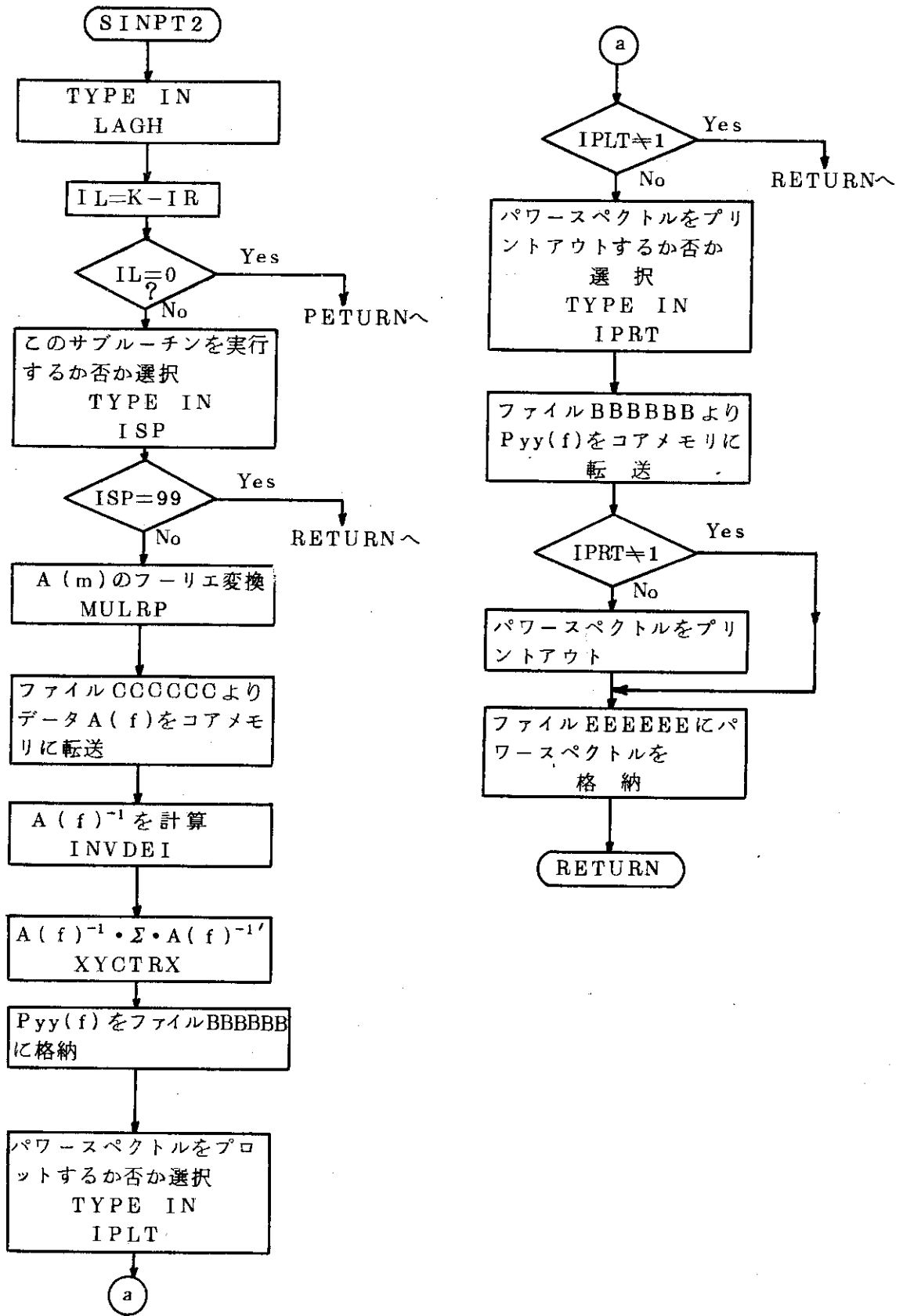


Fig. III-16 Flow Diagram of SINPT2 Subroutine

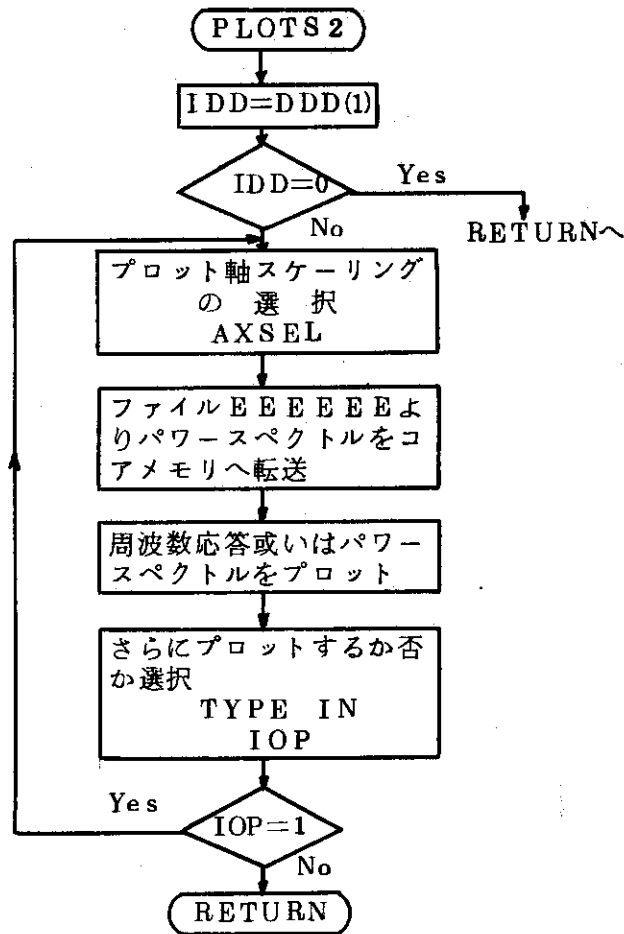


Fig. III-17 Flow Diagram of PLOTS2 Subroutine

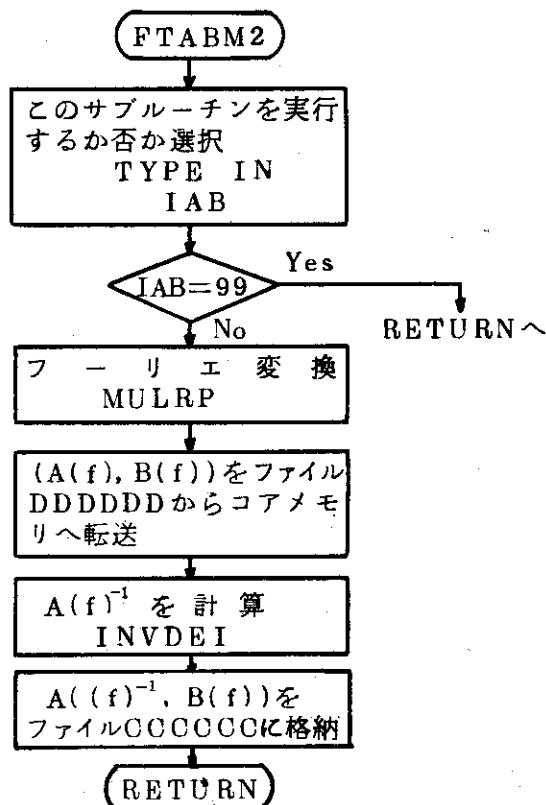


Fig. III-18 Flow Diagram of FTABM2 Subroutine

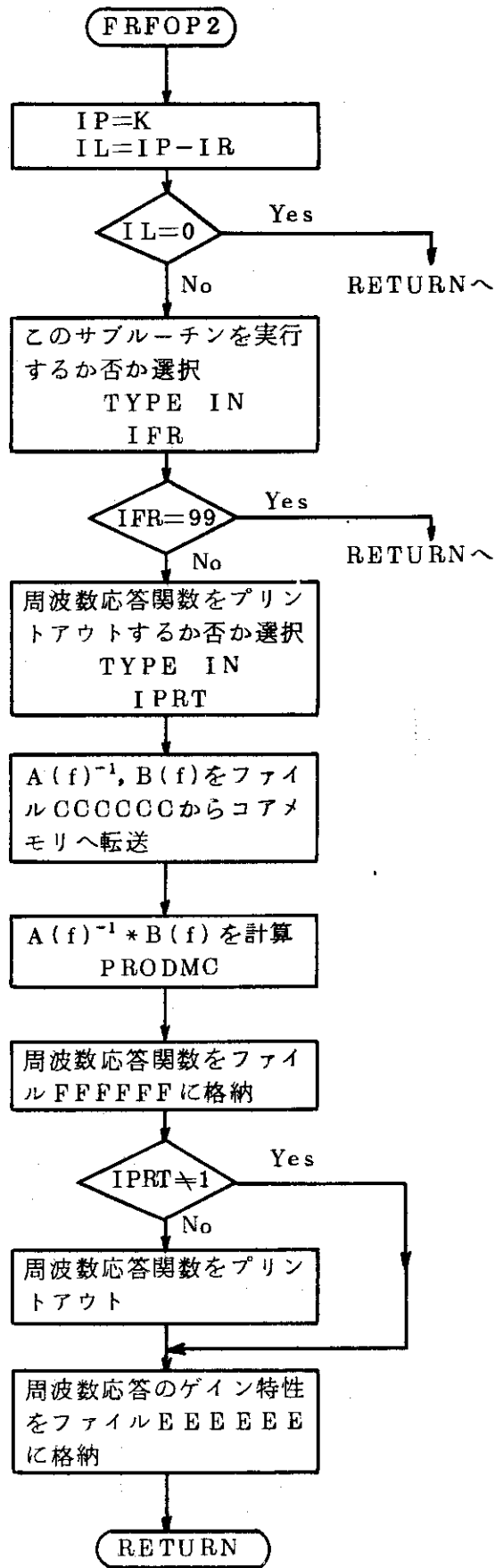


Fig. III-19 Flow Diagram of FRFOP2 Subroutine

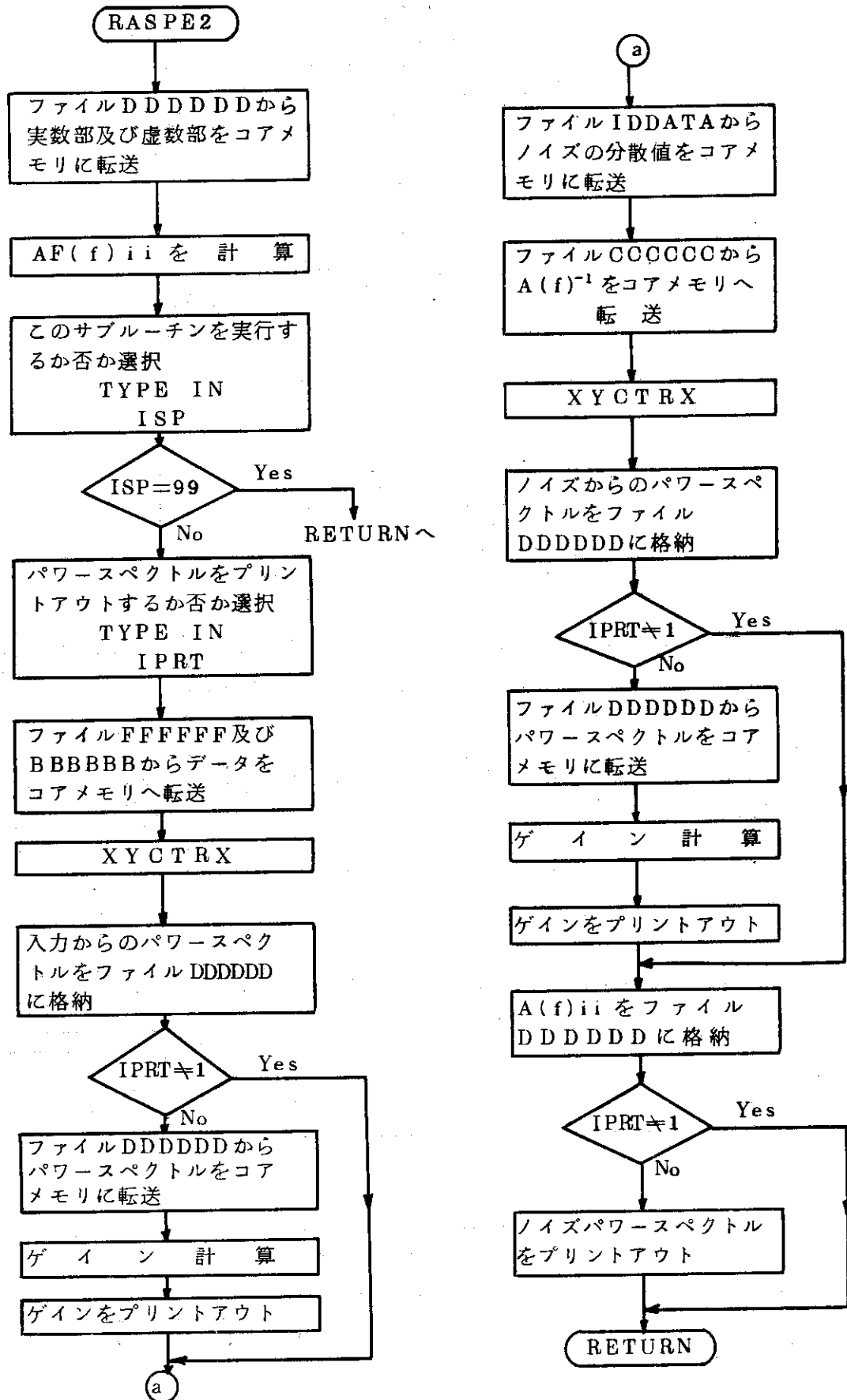


Fig. III-20 Flow Diagram of RASPE2 Subroutine

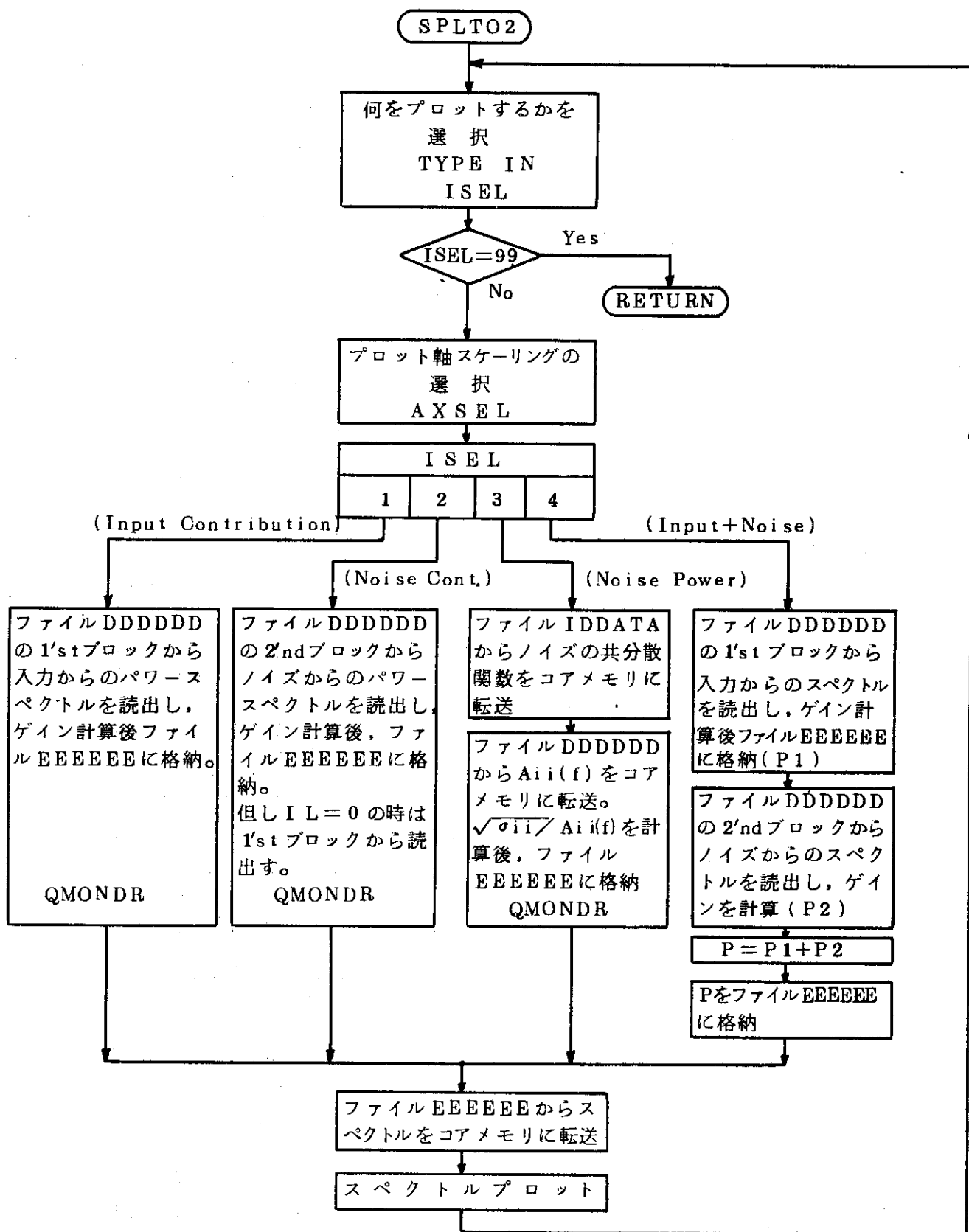


Fig. III-21 Flow Diagram of SPLTO2 Subroutine

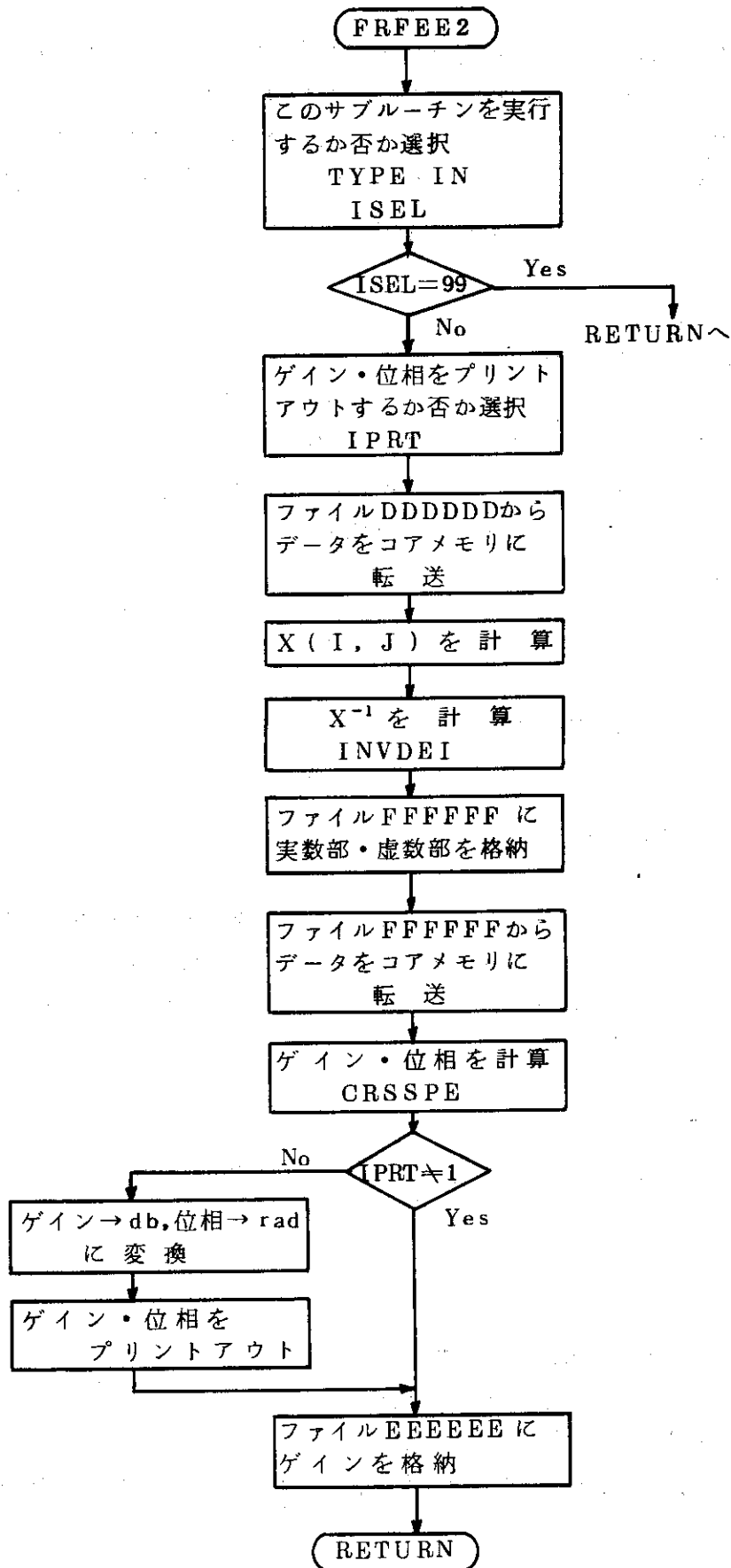


Fig. III-22 Flow Diagram of FRFEE2 Subroutine

IV 解析コードの使用法

この章ではデータ解析を行う場合の解析コードの使用法について述べる。I章でも述べたように本解析コードは計算機ディスプレイターミナルを介して対話形式でジョブが実行されるように作られている。ここではディスプレイ面上に表示される指示に従って計算機操作を進めて行く手順を以下に述べる。手順の中で は使用者がキーボードからタイプインする操作を示す。また \uparrow はキーボードリターン操作を表わす。

IV-1 DYSAC1の操作手順

1) メインプログラムの読出しと実行開始

ディスクに格納されている本コードのメインプログラムをコアに読出し、ジョブの実行を開始する。このジョブ実行に際し、V章応用例でふれるように解析したいデータがオリジナルデータファイルに格納されている必要がある。これを確認して以下の計算機操作に入る。

まず計算機モードをモニターモードにセットし、次の操作によってDYSAC1コードのメインプログラムをコアに転送する。

M- (モニターモード)

#L, DYSAC1, 21 \uparrow

DYSAC1 CI P1 \uparrow } 計算機の応答

LD

#G, 1000 \uparrow

これでプログラムは実行開始され必要なサブルーチンがディスクからコアに転送されて来る。

2) 計算条件の入力

実行が開始されると、まずサブルーチン“FILID1”がコアに読み出され計算に必要な諸条件をディスプレイの指示に従ってタイプインする。ディスプレイの表示を次に示す。

*** KINETICS ANALYSIS CODE BASED ON THE IDENTIFICATION TECHNIQUE ***
PART 1 (A-R MODEL FITTING)

* TYPE IN INPUT DATA
N, K, IR, DELT

データの条件を次の例のようにタイプインする。

2000, 4, 2, 0.5 \uparrow

\uparrow \uparrow \uparrow \uparrow
N K IR DELT

なおサンプリング間隔 DELT は秒単位で入力する。

** DATA SELECTION FOR A-R MODEL FITTING **
** IF UNEXECUTE THIS JOB, TYPE IN 99

このジョブを実行する時は99以外の任意の数字を次のようにタイプインする。FORMATは5桁以内の整数であること。

22 \uparrow

こゝで一度画面が消去され実行が開始される。

3) ファイル名の定義

ディスクに解析用データや編集用データのファイル名を定義する。Ⅲ-22で述べたように予め定義してあればこゝで改めて定義する必要はない。

** IF DEFINE THE FILE, TSDATA, TYPE IN 1

ここで1をタイプインするとディスクプラッタ2にファイル“TSDATA”が自動的に定義される。すでにこのファイルが定義されている場合は次の様に1以外の整数(3ケタ以内)をタイプインする。

3)

** IF DEFINE THE FILE, EEEEE, FFFFF, TYPE IN 1

“TSDATA”と同じように“EEEEEE”, “FFFFFF”のファイル名をディスクに定義するかどうかによって、対応する整数をタイプインする。

これらのファイルを定義する場合は

1)

とする。

** TYPE IN ORIGINAL DATA FILE NAME **

解析したい変数の入っているオリジナルデータファイル名をタイプインする。例えばオリジナルデータファイルを“DATA00”とするとこれを

DATA00)

とタイプインする。このデータファイルをディスクから読み出すと画面が消去され次へ進む。尚オリジナルデータファイル名は、MLCOSPコード或はFILEMKコード³⁾内で定義されるもので、任意の6文字で表わされる。

4) 解析に必要な変数の選択

指定したオリジナルデータファイルから解析に必要な変数を取り出しこれをファイルTSDATAに格納する作業が行われる。この手順は次の様にして行われる。

- i) オリジナルデータファイルから必要な変数を選ぶと、このデータは一旦ファイル“EEEEEE”に格納される。変数選択の順番はオリジナルデータファイルに格納されている順序に従って必要なデータを選択するようになっている。
- ii) ファイル“EEEEEE”に格納されたデータについて変数の順番をここで指定する。これに従ってデータの配列が変わりファイル“TSDATA”に格納される。以後の解析はすべて“TSDATA”に入っているデータをもとに行われる。この様な変数の順番の変更は入力変数、出力変数を各々分けて整理する上で必要であり、このプログラムでは入力変数が順番として後に配列されるように順序付けをしなければならない。

以下に変数選択の手順を述べる。

解析したい変数を選択するに当たって次の2つのケースがある。即ち、変数が1つのデータファイルにすべて収められている場合と、2つのデータファイルにまたがっている場合である。次に2つのケースについて述べる。

- ① 解析したい変数が1つのデータファイルにある場合
3)が終了すると次のコメントが表示される。

```
** TYPE IN VARIABLE NO. IN ORDER (K=4 IR=2)
   (FIRST IR ARE CONTROLLED VARIABLES)
```

```
** TYPE IN (IND(I), I=1, 4)
   IF NECESSARY OTHER DATA FILE, TYPE IN 0
```

このときのKとIRは2)のデータの条件で設定した値が自動的に表示されるもので、その個数Kにもとづいてオリジナルデータの変数の指定を行う。次に示すように“IND”の順に従って変数の番号をタイプインする。ここで“IND”はデータがファイル“EEEE EE”に格納される順番である。

```
IND=1
```

```
1 )
```

```
IND=2
```

```
5 )
```

```
IND=3
```

```
8 )
```

```
⋮
```

このようにしてK個の変数の番号をタイプインすると、オリジナルデータファイルからその番号の変数を読み出し、終了すると画面が消え次のステップ5)へ進む。

- ② 解析したい変数が2つのデータファイルにまたがっている場合

2)の条件でK=5, IR=2として、解析したい変数が2つのデータファイルにまたがっている場合の例を示す。

```
** TYPE IN VARIABLE NO. IN ORDER (K=5 IR=2)
   (FIRST IR ARE CONTROLLED VARIABLES)
```

```
** TYPE IN (IND(I), I=1, 5)
   IF NECESSARY OTHER DATA FILE, TYPE IN 0
```

```
IND=1
```

```
2 )
```

```
IND=2
```

```
5 )
```

```
IND=3
```

```
10 )
```

いま初めに指定したオリジナルデータファイルからは3変数が必要で、残りの2変数は他のデータファイルにある場合は、次に示すように0をタイプインする。

```
IND=4
```

```
0 )
```

すると次のコメントが表示されるので残りのデータが格納されているデータファイル名をタイプインする。

** TYPE IN ORIGINAL DATA FILE NAME **

DATA 11)

すると画面が消され、再び同じようにコメントが表示され、引き続き変数の番号のタイプインが要求される。INDの順番は、前のデータの時に“0”をタイプインしたところから続行される。したがってあとから指定したデータファイル(この場合DATA11)からは、2つの変数を選択することになる。

** TYPE IN VARIABLE NO. IN ORDER (K=5 IR=2)
(FIRST IR ARE CONTROLLED VARIABLES)

** TYPE IN (IND(I), I=1, 5)
IF NECESSARY OTHER DATA FILE, TYPE IN 0

IND=4

2)

IND=5

4)

このように解析したいデータが、2つのデータファイルにまたがっている場合には、最初のデータファイルからの変数選択がすべて終わった後0をタイプインすることによって新たなデータファイルから残りの変数を選択することができる。

5) 変数の順番の変更

オリジナルデータファイルから選択されたデータが、“EEEEEE”のデータファイルに格納された後、前述した様な解析の都合によりこの変数の順番を変更することができる。例えば5変数のデータがあってその順番を変更したいときは次のようにタイプインする。順番の変更をしないときにはINDの番号と同じ番号をタイプインすれば良い。

** TYPE IN THE ORDER ABOUT VARIABLES (I=1, 5)

IND=1

1)

IND=2

5)

IND=3

3)

IND=4

2)

IND=5

4)

このようにして解析のためのデータの順序付けが終了すると、データは解析用データファイル“TSDATA”に格納される。

6) 相関関数の計算

解析用データファイル“TSDATA”が作成されると、CORID1のルーチンが読み出され次のコメントが表示される。そして相関計算を行うか否かをタイプインする。

** MULTIPLE CORRELATION FOR A-R MODEL FITTING **

SELECT FOLLOWING NUMBER

- 1 ; CALCURATE CORRELATION
- 2 ; NOT CALCURATE

** ICOR **

こゝで2をタイプインすれば次のステップ7)へ進む。

 1) とタイプすると次のコメントが表示される。

** IF DEFINE THE FILE CODATA, TYPE IN 1

相関計算用データファイルの定義が要求される。定義してあれば1以外の任意の整数(5桁以内)をタイプインする。

 5 5 2

さらに次のコメントに従ってオリジナルデータ及び相関関数計算結果をラインプリンタに出力するか否かを指定する。

TYPE IN PRINT OUT CONDITION FOR LINE PRINTER

- * ISW1 1 ; NOT PRINT
- 2 ; PRINT OUT ORIGINAL DATA
- 3 ; PRINT OUT CORRELATION
- 4 ; PRINT OUT BOTH DATAS

** ISW1 **

1 2) とタイプインすればラインプリンタの出力は不要となる。この入力が終わると平均値除去、自己・相互共分散計算を開始する。そして計算が終了するとISW1の指定にもとづいて結果が出力される。また平均値は次のようにディスプレイ面に表示される。

** MEAN VALUE OF THE OBSERVED VARIABLES **

1	-.2359E-01
2	-.4761E-01
3	-.2878E-01
4	-.4048E-01

こゝではこの値がスペースバーを押すまで表示されている。得られた相関計算結果はデータファイル“CODATA”に格納される。

7) システムモデルの同定

続いて“CODATA”にファイルされた相関計算の結果を用いて、システムの同定を行うルーチン“FPEID1”が読み出され、まず次のコメントが表示される。

***** A-R MODEL FITTING *****

** RESTRECTED CONDITION

N=2000
L=20
IR=2

```
IL=2
DELT=.5000E+00
```

** IF UNEXECUTE THIS JOB, TYPE IN 99

この条件の中でN, IR, DELTは2)で与えた値が表示される。L, ILはこのルーチンの計算条件である。

このルーチンを実行する時は次のように99以外の任意の整数(5桁以内)をタイプインする。

```
44)
```

本ルーチンのジョブ実行に入ると次のコメントが表示される。

** IF DEFINE THE FILE IDDATA., TYPE IN 1

同定計算結果のデータファイル"IDDATA"を定義するか否かをタイプインする。すでに定義されていれば1以外の整数(5桁以内)をタイプインする。

```
3)
```

ここから同定計算が開始される。A-Rモデルの係数推定, 評価関数の計算を行い結果をファイル"IDDATA"に格納する。計算終了と同時に評価関数計算結果がラインプリンターとディスプレイ面に出力される。さらに, ラインプリンターにノイズの共分散計算結果とその正規化した値およびA-Rモデルの係数推定値が出力される。

このA-Rモデル係数推定の過程ではデータのディスク転送に関するメッセージとしてデータファイル名がディスプレイ面に順次表示される。この場合ディスプレイターミナルの状態を"Auto"にしておくとも自動的に画面のページングが行われる。

8) 解析内容の選択

A-Rモデルで同定された計算結果を用いて, システム解析計算を行うのであるが, はじめに解析ルーチンの選択のためサブルーチン"ANSEQ1"がコアに読み出され, 次のように解析内容の選択が要求される。

*** SYSTEM ANALYSIS BY IDENTIFIED A-R MODEL

*** SELECT THE NUMB. ACCORDING TO THE ANALYSING SEQUENCE

- 1... STEP RESPONSE ANALYSIS
- 2... SPECTRUM ANALYSIS
- 3... NOISE ANALYSIS
- 4... MODEL FITNESS CHECK
- 5... END OF THE JOB

このうち2と3については皿章25で述べたように, こゝで5を選択してジョブを終了し, DYSAC2コードの実行を開始する。また, 選択された解析ルーチンが終了すると, 再びこのルーチンにもどるようになっている。

9) ステップ応答計算

上記で1を選択するとこのルーチンに入る。ここではオープンループ系, フィードバック系ステップ応答計算さらに周波数応答計算がシリーズに行われるようになっている。最初にオープンループ系ステップ応答計算が開始され次のコメントが表示される。

*** STEP RESPONSE CALCULATION WITH OPEN LOOP SYSTEM ***
 MANIPULATED VARIABLE IL=2

*** IF UNEXECUTE THIS JOB, TYPE IN 99

この計算を実行するには99以外の整数(4桁以内)をタイプインする。

333 ↓

“IDDATA”にファイルされた同定計算結果をコアに読み出し、次のコメントが表示される。

** IF PRINT OUT THE RESULT, TYPE IN 1

計算結果をラインプリンタに出力するか否かの選択を行う。出力するときは1 ↓ を、出力不要の時は1以外の整数(4桁以内)をタイプインする。

*** TYPE IN STEP NUMB. FOR CALCULATING STEP RESP. FN. ***

続いてステップ応答計算のステップ数をタイプインする。ステップ数は最大400である。例えばステップ数を100にしたい時は次のようにタイプインする。

100 ↓

ステップ数のタイプインが終了すると、入力変数からみた出力へのステップ応答の計算が開始され、その計算結果はデータファイル“EEEEEE”に格納される。操作変数ILの数に応じて計算結果が順次グラフィック表示される。

このグラフィック表示は、スペースバーを押すまでPAUSEされている。ステップ応答のグラフィック表示がすべて終了すると次のコメントが表示される。

** IF EXECUTE THE CASE STUDY, TYPE IN 1

1 ↓

とタイプインすれば、画面が消去され再び本計算ルーチンの最初にもどりステップ数を変えて計算することが可能となる。

1以外の整数(4桁以内)例えば222 ↓ とタイプインすると次のコメントが表示される。

IF CALCULATE THE FREQ. RESP. FN. TYPE IN 1

周波数応答計算を行う場合は1 ↓ とタイプインする。ここでⅢ-2章で述べた様に、周波数応答計算プログラムはオープンループ系、フィードバック系共通に用いられるためこの場合は次のフィードバック系の解析をスキップして先に周波数応答計算を行う。1以外の整数をタイプインすると次のフィードバック系計算ルーチンに進む。

フィードバック系のステップ応答計算

オープンループ系のステップ応答計算が終了するとフィードバック系のステップ応答計算が開始され次のコメントが表示される。

*** STEP RESPONSE CALCULATION WITH CLOSED LOOP SYSTEM ***
 CONTROLLED VARIABLE IR=2

*** IF UNEXECUTE THIS JOB, TYPE IN 99

ここで、先のジョブでオープンループ系周波数応答計算を選択した場合、又は本計算が不要な場合は99 ↓ とタイプインする。これによってプログラムは10)の周波数応答計算ルーチンにとぶ。

それ以外の場合は本ルーチンのジョブが開始され次のコメントが表示される。

** TYPE IN STEP NUMB. FOR CALCULATING STEP RESP. **

100)

続いてラインプリンタへの出力条件が要求される。

** IF PRINT OUT THE CALCULATED RESULT, TYPE IN 1

計算結果をラインプリンタに出力するときは 1) を、出力不要のときは 1 以外の整数 (4 桁以内) をタイプインする。

そして計算が終了すると、その結果がラインプリンタ (出力要求をしていれば) およびグラフィックに表示され、オープンループ系と同じくステップ数を変えて種々計算することができる。

** IF EXECUTE THE CASE STUDY, TYPE IN 1

1 以外の整数 (4 桁以内) をタイプインすれば次のコメントが表示される。

IF CALCULATE THE FREQ. RESP. FN. TYPE IN 1

こゝで周波数応答計算を実行する時は 1) をタイプインする。1 以外の整数 (4 桁以内) をタイプインすると、10) の周波数応答計算ルーチンは自動的にスキップされ再び 8) の解析内容選択ルーチンにもどる。

10) 周波数応答計算

9) のオープンループ系又はフィードバック系ステップ応答計算においていずれかの場合に周波数応答計算の選択を行った場合、このジョブは実行される。そうでない場合は自動的にスキップして 8) にもどる。このジョブが開始されると次のコメントが表示される。*

** FREQUENCY RESPONSE FN. CALCULATION WITH CLOSED LOOP SYSTEM

* FOURIER TRANSF. OF IMPULSE RESP. FN.

** IF PRINT OUT THE IMPULSE RESPONSE FN. TYPE IN 1

ステップ応答のインパルス応答計算結果をラインプリンタに出力したい時は 1) を、出力を必要としない時は 1 以外の整数 (5 桁以内) をタイプインする。

インパルス応答計算が終了すると、ラインプリンタ出力を必要としていればこゝで計算結果を打出す。

このインパルス応答をフーリエ変換して周波数応答を求める。得られた結果はグラフィック表示される。その時に次のようなコメントが表示され、プロットする際の縦軸、横軸のスケールリングの選択をおこなう。

** AXIS SCALING CONDITION **

X AXIS	1...LINEAR	2...LOG	
Y AXIS	1...LINEAR	2...LOG	3...DB

* TYPE IN THE SCALING NO.

例えば、X軸を対数、Y軸を db (デシベル) でプロットさせたい時は 2 3) という様に、任意に選択してタイプインする。それによって Fig. N-1 のようなグラフィック表示がなされる。またスケールリングの条件を変えてグラフィック表示をしたい時には、次のコメントに

* ここではフィードバック系についてのコメントを記しているがオープンループ系の場合も同様である。

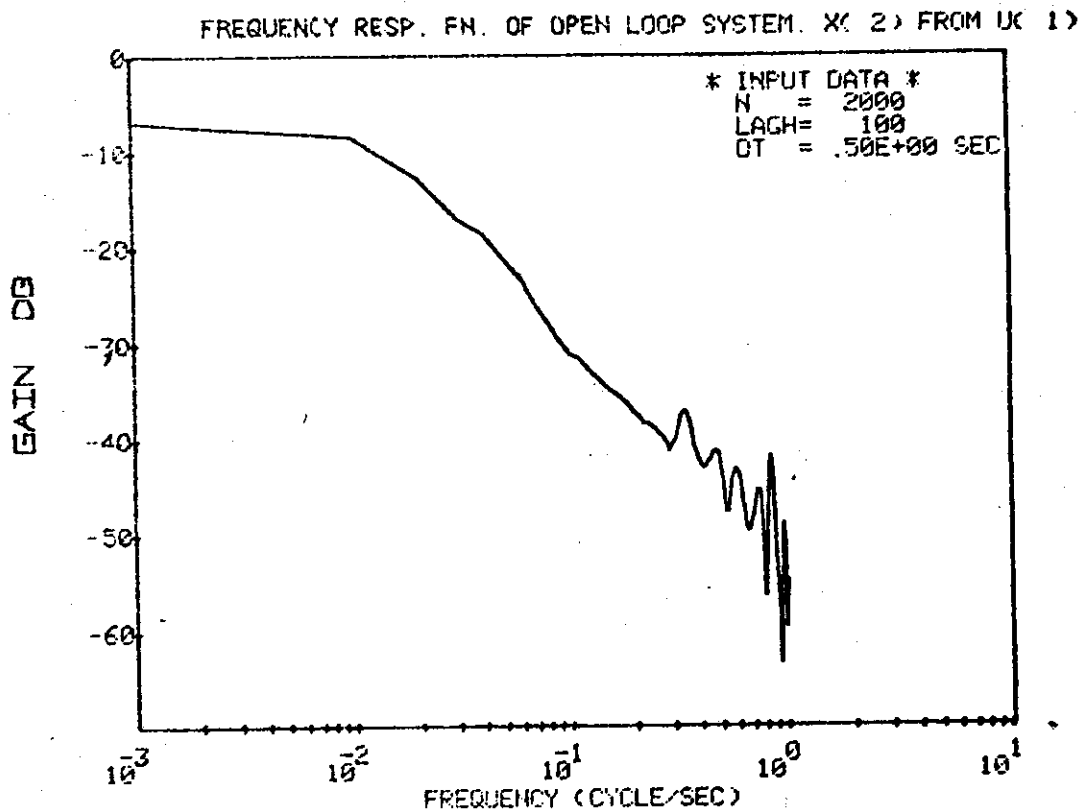


Fig. IV-1 An example of CRT display (Frequency response)

1) をタイプインする。すると再びスケージングの条件選択が可能となる。

IF PLOT THE FREQ. RESP. FN. AGAIN, TYPE IN 1

1 以外の整数 (5 桁以内) をタイプインすると, 8) の解析内容選択ルーチンへもどる。

11) 出力状態の推定

同定計算の結果を用いて出力推定を行うルーチンで, Ⅲ章 2.9 で述べたように 2 つの方法があり, まず次のコメントでその選択が要求される。99 はジョブのスキップである。

** STATE ESTIMATION BY A-R MODEL

IF UNEXECUTE THE JOB, TYPE IN 99

IF ESTIMATE THE INNOVATION PROCESS, TYPE IN NEGATIVE NO.

IF ESTIMATE THE OUTPUT STATE, TYPE IN POSITIVE NO.

こゝで, 出力状態推定を行う場合について以下に述べる。この場合, 例えば **5)** とタイプインすると同定計算結果がデータファイル "IDDATA" からコアに転送される。続いて次の計算条件を指定するためのコメントが表示される。

** TYPE IN...START NO., BLOCK NO...

NUMB. OF DATA=2000

ここで START NO. は出力推定計算を行う初めのデータの番号で, これは 20 以上の値を入れる。BLOCK NO. は計算ステップ数で, 1 ステップが 100 個のデータを表わし, 最大 5 ステップまで与えることができる。

例えば START NO. を 300 番目からとし、ステップ数を 4、すなわち 400 個の計算をさせたいときには次のようにタイプインする。

300, 4)

(なおここで NUMB OF DATA=2000 と表示されているのはデータ総数で、本コードの初めに入力した値 (N) がそのまま表示される。したがって計算ステップはこの N の値を越えてはならない。)

上記のタイプインが終ると、出力状態の推定計算に入り計算結果が Fig. III-11 (III-2-9) のようにグラフィック表示される。

そしてすべての計算結果の表示が完了すると次のコメントが表示される。

IF CALCULATE THE INNOVATION CORRELATION, TYPE IN 1

残差相関の計算をしない時は 1 以外の整数 (5 桁以内) をタイプインし、8) の解析内容選択ルーチンへもどる。

ここで **1**) とタイプインすると 12) 残差相関の計算に入る。

12) 残差相関計算

このルーチンは直接残差相関を計算するのではなく、計算のためのデータファイルを作成するものである。(残差相関計算はこのルーチン実行後、MLCOSP コードを用いて実行する。) まず次のコメントが表示されデータファイル "AAAAAA" の定義の要求がなされる。

** RESIDUAL CORRELATION OF STATE ESTIMATION
IF DEFINE THE FILE AAAAAA, TYPE IN 1

すでに定義してあれば 1 以外の整数 (4 桁以内) をタイプインする。 **1**) とすればデータファイル "AAAAAA" がディスクに定義される。

このデータファイルには、残差相関計算に必要なデータが格納される。これが完了すると次のような解析条件のコメントが表示される。

* RESIDUAL CORRELATION
1ST IR... RESIDUE T.S. DATA
2ND IR... OUTPUT DATA

* CALL THE PROGRAM... MLCOSP...
FILE NAME. AAAAAA
N = 400
K = 4
DELT = .500E+00

この表示はスペースバーを押すまで PAUSE されておりスペースバーを押すと 8) の解析内容選択ルーチンにもどる。そこでジョブ終了の **5**) をタイプインし、改めて多変数相関・スペクトル解析コード "MLCOSP" をコアメモリに読出し残差相関の計算を行う。この計算に際しての入力条件はすべて上記コメントで与えられている。ここで本ルーチンに関するデータは上記のようにファイル "AAAAAA" に格納されるので使用者はオリジナルデータファイルを同一名にしないよう注意する事が必要である。

N-2 DYSAC2の操作手順

1) メインプログラムの読出しと実行開始

DYSAC1と同様にまずメインプログラムをコアに読出す。

```
#L, DYSAC2, 21 )
DYSAC2 CI P1 } 計算機の応答
LD
```

```
#G, 1000 )
```

こゝで実行開始となる。

2) 入力変数の同定計算

実行が開始されると、はじめに入力の同定サブルーチン“FPEID2”がコアに読み出され次のコメントが表示される。

```
** DYNAMIC SYSTEM ANALYSIS CODE FROM A-R MODEL **
** CALCULATION OF DYSAC1 MUST BE TERMINATED BEFOREHAND
DATA FILE BBB. CCC. DDD. EEE. FFF. MUST BE DEFINED
** TYPE IN INPUT DATA .. N, K, IR, DELT..
```

こゝでDYSAC1と同様の入力データをタイプインする。

```
2000, 4, 2, 0.5 )
  ↑   ↑   ↑   ↑
  N   K  IR  DELT
```

ここで操作変数がない場合(即ちIL=0)このルーチンは自動的にスキップされ次のジョブに進む。

また、このルーチンに入る前には必ず“DYSAC1”の計算が終了しており、各データファイルが定義されていることが前提となる。

入力データが与えられると次のコメントが表示される。

```
** A-R MODEL FITTING TO INPUT DATA **
IF UNEXECUTE THE JOB, TYPE IN 99
```

このルーチンを実行するには99以外の整数(5桁以内)を次のようにタイプインする。

```
12 )
```

この入力終了すると、入力のA-Rモデルの同定計算に入る。計算が終ると自動的にその結果がディスプレイ面にプロットされる。又“FPEID1”と同様にラインプリンタに計算結果が打出される。

グラフィック表示は、PAUSEされているので必要があればここでハードコピーを取ることができる。スペースバーを押すと、次のステップ3)へ進む。

3) パワースペクトル計算

2)と同様に入力変数がない場合は、このジョブおよび4)のプロットルーチンは自動的にスキップされ5)の係数フーリエ変換ルーチンに飛ぶ。ここでは2)のA-Rモデルの同定結果をフーリエ変換して、入力変数についてのパワースペクトル計算を行う。

このルーチンに入るとまず次のコメントが表示されそれによって周波数分割数LAGHをタイプインする。

** POWER SPECTRUM OF INPUT VARIABLES **
TYPE IN LAGH **

100)

LAGHは最大200まで与えることができ、この値はDYSAC2解析コード全体を通じて共通に使用される。

次にこのルーチンを実行するか否かの選択をし、実行しない場合は5) フーリエ変換へスキップする。

IF UNEXECUTE THE JOB, TYPE IN 99

99以外の整数(5桁以内)を次のようにタイプインすると入力変数のパワースペクトル計算が行われる。

23)

計算の終了と同時に次のコメントが表示される。

** IF PLOT THE POWER SPECTRUM OF INPUT VARIABLES, TYPE IN 1

計算結果をグラフィック表示する場合は 1) をタイプインする。1以外の整数(5桁以内)をタイプインすると、次の4)はスキップされ5)へ進む。

グラフィック表示を実行する時のみ次のコメントが表示される。

IF PRINT THE CALCULATED P. SPECTRUM, TYPE IN 1

ラインプリンタへ計算結果を出力したい場合は 1) を与える。1以外の整数(5桁以内)をタイプインすれば、ラインプリンタ出力はせずに次のプロットルーチンへ進む。

4) スペクトルプロット表示

スペクトルの計算結果をプロット表示するルーチンであり、その計算を行わない時はこのプロットルーチンは自動的にスキップされる。ここで次のコメントに従って、ゲイン及び周波数軸を選択すれば、スペクトル結果は自動スケージングでプロットされる。

** RATIONAL SPECTRUM DERIVED FROM A-R MODEL

** AXIS SCALING CONDITION **

X AXIS 1 ... LINEAR 2 ... LOG
Y AXIS 1 ... LINEAR 2 ... LOG 3 ... DB

* TYPE IN THE SCALING NO.

例えばX軸、Y軸ともに対数目盛でプロット表示させたい場合は次のようにタイプインする。

2, 2)

Fig. N-2 にその表示例を示す。

プロット表示が終了すると次のコメントが表われる。

** IF PLOT AGAIN, TYPE IN 1

プロット軸を変更して表示したい時は、 1) をタイプインし、上記プロット軸選択のところへもどる。1以外の整数(5桁以内)をタイプインすると5)のフーリエ変換ルーチンへ進む。

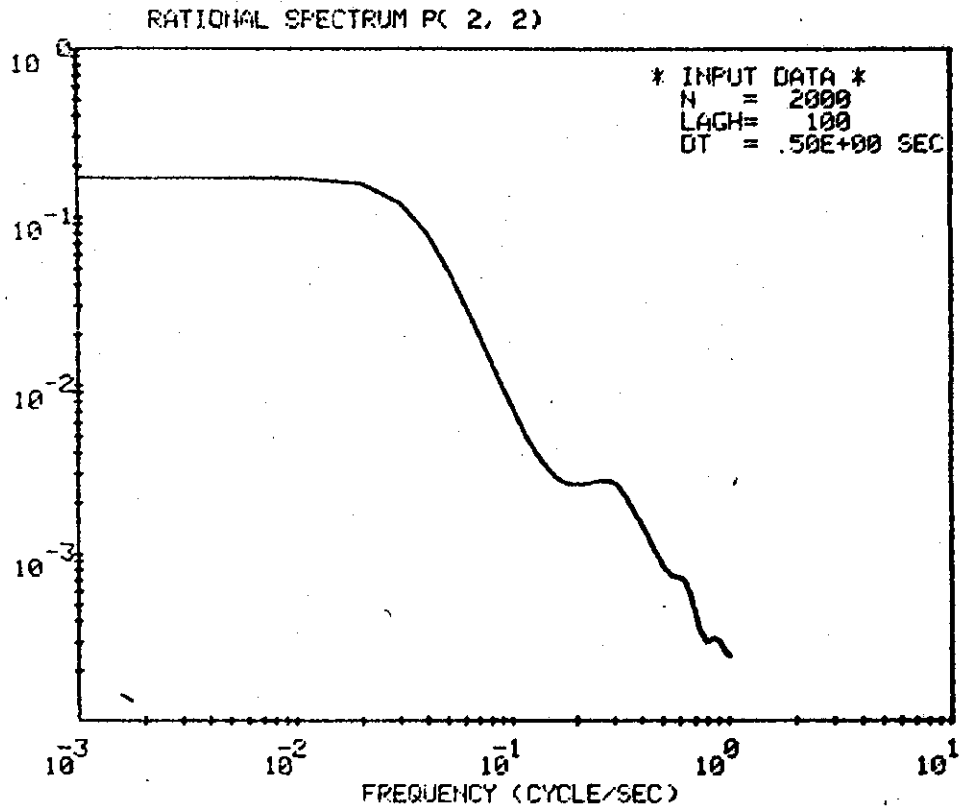


Fig. IV-2 An example of CRT display (Power spectrum)

5) A-Rモデル計算のフーリエ変換

同定されたモデルの係数マトリックスをフーリエ変換するルーチンである。

** FOURIER TRANSFORMATION OF MATRIX (A, B)
IF UNEXECUTE THE JOB, TYPE IN 99

この計算を実行する時は99以外の整数(5桁以内), 例えば 34 のようにタイプインする。こゝでは計算が終了すると自動的に次のステップ6)へ移る。

6) オープンループ周波数応答計算

フーリエ変換された係数からオープンループ系の周波数応答計算を行う。

操作変数がない場合は自動的にこのルーチンはスキップされる。

** FREQ. RESP. FN. OF OPEN LOOP SYSTEM DERIVED FROM A-R MODEL
IF UNEXECUTE THE JOB, TYPE IN 99

この計算を行う時は, 99以外の整数(5桁以内)を 43 のようにタイプインする。次に計算結果をラインプリンタに出力させるか否かの選択を行う。

** IF PRINT OUT THE FREQ. RESP. FN. TYPE IN 1

出力する場合は 1 を, 出力させない時は1以外の整数(5桁以内)をタイプインする。

計算が終了すると計算結果のプロット表示が行われる。もし, ラインプリンタへの出力要求があればその前に実行される。

プロットルーチンは4)で述べた“PLOTS2”ルーチンが利用される。もちろん, このジョ

ブを実行しない時は、自動的にプロットルーチンもスキップされる。コメント表示は次のようになる。

** FREQ. RESP. FN. WITH OPEN LOOP SYSTEM DERIVED FROM A-R MODEL

** AXIS SCALING CONDITION **

X AXIS 1 ... LINEAR 2 ... LOG
Y AXIS 1 ... LINEAR 2 ... LOG 3 ... DB

* TYPE IN THE SCALING NO.

こゝでゲイン特性のプロット軸の選択を行い、自動スケーリングでプロットされる。

** IF PLOT AGAIN, TYPE IN 1

スケールを種々変更してゲイン特性をプロットしたい時は を、このルーチンを終了したい時は 1 以外の整数をタイプインし、次のステップ 7) へ進む。

7) 出力スペクトル計算

出力パワースペクトルとそれに対する入力、ノイズの寄与分、さらにノイズスペクトルを計算するルーチンである。

はじめにこのジョブを実行するか否かをきめるコメントが表示される。

*** R. POWER SPECTRUM OF OUTPUT VARIABLES
DERIVED FROM A-R MODEL. IF UNEXECUTE THE JOB, TYPE IN 99

99 以外の整数 (5桁以内) をタイプインするとこのジョブの実行に入り、計算結果の打出しをするか否かの選択が行われる。

** IF PRINT THE POWER SPECTRUM, TYPE IN 1

ラインプリンタへ出力したい時は を、出力不要の時は 1 以外の整数 (5桁以内) をタイプインする。このタイプインが終るとスペクトル計算の実行が開始され、要求があった場合には計算結果がラインプリンタへ出力される。そして次の 8) へ進む。

8) 出力スペクトルプロット表示

これは上記 7) で得られた結果をプロット表示するルーチンでこのジョブの実行が開始されると次のコメントが表示される。

*** PLOT THE OUTPUT POWER SPECTRUM ***

IR=1

IL=2

** TYPE IN NO.

1... INPUT CONTRIBUTION TO POWER SPECTRUM
2... NOISE CONTRIBUTION
3... NOISE POWER SPECTRUM
4... INPUT+NOISE CONTRIBUTION
99... UNEXECUTE THE JOB

こゝで IR と IL は先に入力したデータ条件の表示である。タイプインされる番号は解析の必要に応じて上記の条件の中から選択する。選択された内容のプロットが終了すると再び本サブルーチンの最初にもどりプロット内容の選択が行われる。 をタイプインするとこのジョブを終了し 9) へ進む。

上記1～4を選択すると続いて次のコメントが表示される。

** AXIS SCALING CONDITION **

X AXIS 1 ... LINEAR 2 ... LOG
Y AXIS 1 ... LINEAR 2 ... LOG 3 ... DB

* TYPE IN THE SCALING NO.

ここでX, Y軸のスケールをタイプインすると, 出力スペクトル計算の中から, 先にタイプインした番号のプロット表示が実行される。

スペクトルプロット表示が終了するとプログラムは自動的に最初のプロット内容選択表示にもどる。

9) フィードバックループ周波数応答計算

本ルーチンの実行が開始されるとまずフィードバックループ周波数応答計算を行うか否か, 次の指示に従ってタイプインする。

** FREQ. RESP. FN. OF FEEDBACK LOOP
IF UNEXECUTE THE JOB, TYPE IN 99

ここで **99** とすれば, "STOP" となり本コードの実行は終了する。

実行する時は99以外の整数をタイプインする。

次にゲインと位相の計算結果を, ラインプリンタに出力するか否かを指示する。

** IF PRINT OUT THE GAIN, PHASE, TYPE IN 1

ラインプリンタへの出力不要の時は1以外の整数(5桁以内)をタイプインする。このタイプインが終了すると, 周波数応答計算が開始される。計算結果は上記出力指定のある場合はラインプリンタへの打出しが行われる。

続いて以下の様にして周波数応答のゲイン特性がグラフィック表示される。その手順は4)と同様で次の様にプロット軸の選択が行われる。

** FREQ. RESP. FN. WITH CLOSED LOOP SYSTEM DERIVED FROM A-R MODEL

** AXIS SCALING CONDITION **

X AXIS 1 ... LINEAR 2 ... LOG
Y AXIS 1 ... LINEAR 2 ... LOG 3 ... DB

* TYPE IN THE SCALING NO.

グラフィック表示に適應するX, Y軸を選んで **2, 1** のようにタイプインする。

一連のグラフィック表示が終了すると次のコメントが表示される。

** IF PLOT AGAIN, TYPE IN 1

1 をタイプインすれば, 再度スケールを変更してプロット表示を行うことができる。1以外の整数(5桁以内)をタイプインすると次のコメントが表示され, 本コードのジョブはすべて完了する。

STOP 00000

V 応用 例

本解析コードを用いて、プラント動特性解析、制御系の評価、ノイズ解析等すでに多くの分野のデータ解析がなされている。ここでは、本コードを使用する場合の参考としてJPDR同定実験データの解析結果を実際のジョブ進行に従って述べて行く。

JPDRは沸騰水型軽水炉でJP-IIに改造されて強制循環形式になり定格出力は90MWeである。プラント全体の概念図をFig.V-1に示す。また、タービン系とダンプ系の2つの負荷系統を持っている。同定実験は定格出力の50%において、圧力制御系と水位制御系を自動からはずし固定した状態で、強制循環流量および主蒸気流量に各操作端から独立なM系列信号を手動で与え全体として定常状態近傍で不規則変動するようにしている。得られた実験データはアナログデータレコーダに一旦収録しこれをハイブリッド計算機に入力し以下のデータ処理と解析を進めて行く。

1. オリジナルデータファイルの作成

アナログデータレコーダに収録された多チャンネル実験データを再生してこれをデジタル計算機のディスクに定義されているオリジナルデータファイルにストアする。ここではデータレコーダの出力をハイブリッド計算機アナログ部に入力し、これをA/D変換してデジタル化したデータをディスクにストアする。データのサンプリング間隔はロジック部タイマーを用いて最高 $10\mu\text{sec}$ から任意に定めることができるが、本例ではサンプリングタイムを0.5secとしている。オリジナルデータファイルの作成は相関、スペクトル解析コード“MLCOSP”に内蔵されているデータ変換サブルーチンを用いて行う。²⁾

またデータ変換専用のプログラムも作られている。³⁾ 得られた実験データの1例をFig.V-2に示す。

2. 同定結果例

オリジナルデータファイルの中から必要な変数のみを選択して同定計算に入る。計算が終るとまずTable.V-1に示すように解析に用いられている入力条件と評価関数の値、その最小値等がラインプリンタに出力される。と同時にグラフィックディスプレイにこれを図形化したものがFig.V-3のように表示される。さらに推定されたノイズの2次モーメントの値、システムパラメータの値がラインプリンターに出力される。(Table.V-2)

3. ステップ応答計算

同定されたモデルを用いてオープンループ系およびフィードバック系ステップ応答計算をした結果の1例をFig.V-4, Fig.V-5に示す。

Fig.V-4はForced Circulation Pump (FCP) Flow (U(1))及び蒸気流量(U(2))をステップ変化させた場合の中性子密度(CIC3)の応答を示し、Fig.V-5は各変数に対する炉圧力のステップ応答を示している。

このディスプレイ面表示は必要な変数の組合せについて順次、自動的に行うようにできている。

4. モデルフィットネスチェック

得られた同定結果の精度評価について、ここではモデルを用いて推定した出力と実験データ

の比較が行われる。Fig. V-6では実験データの入力を同定されたモデルの入力として与えた場合の推定値を点線で示している。さらに推定誤差分についての特性を議論する場合は残差相関スペクトル計算を行う。

5. スペクトル解析

DYSAC2コードのジョブの実行に入ると入力変数についてのスペクトルが計算される。Fig. V-7は、実験で用いられた2つの入力のうち強制循環流量のパワースペクトル計算結果を示している。続いて、出力のパワースペクトル計算、及びその変数に寄与する各成分への分離計算が行われる。

これらの計算結果はすべてディスプレイ面に表示出来るが、ここでは一例として中性子束密度のパワースペクトルについて入力とシステムノイズからの寄与分をFig. V-8に示す。

これらの考察から制御系に関する評価、ノイズ源の推定等が可能となる。スペクトル計算結果は一旦ディスクメモリに格納され、オープンループあるいはフィードバックループの周波数応答計算に用いられる。

Fig. V-9にFCP Flowからの中性子束密度への周波数応答関数(ゲイン特性)の一例を示す。

以上の様な計算を種々の変数の組について実行し、その中からシステムの特性を抽出して行く作業が行われる。

Table V-1 Input parameters for A-R model fitting
and Final Prediction Error

*** A-R MODEL FITTING FOR T. S. DATA ***
** INPUT DATA **

N = 2000
K = 7
IR = 5
IL = 2
DELTA = .5000E+00

*** FINAL PREDICTION ERROR ***

M	FPEC(M)	RFPEC(M)
0	.88957E-09	.10000E+01
1	.13921E-14	.15649E-05
2	.10379E-14	.11668E-05
3	.94431E-15	.10615E-05
4	.91215E-15	.10254E-05
5	.90195E-15	.10139E-05
6	.89235E-15	.10031E-05
7	.88815E-15	.99840E-06
8	.88819E-15	.99845E-06
9	.89269E-15	.10035E-05
10	.87154E-15	.97973E-06
11	.86450E-15	.97182E-06
12	.86383E-15	.97107E-06
13	.86992E-15	.97791E-06
14	.87843E-15	.98748E-06
15	.88247E-15	.99202E-06
16	.89038E-15	.10009E-05
17	.90096E-15	.10128E-05
18	.91004E-15	.10298E-05
19	.91618E-15	.10299E-05
20	.93238E-15	.10461E-05

MINIMUM FPEC = .86383E-15 MINIMUM RFPEC = .97107E-06

ATTAINED AT M = 12

GEOMETRICAL MEAN OF MIN FPEC(M)
.2778E-01

Table V-2 Estimated variables using A-R model

ESTIMATED NOISE 2-ND MOMENT
OSD(I,J)

1	.1330E+00				
2	.2850E-03	.2751E-04			
3	.1026E-02	-.8185E-05	.1301E-02		
4	.5396E-03	-.1609E-04	.7897E-03	.9551E-03	
5	-.4311E-03	.1043E-04	-.3676E-05	-.4174E-04	.2399E-03

***NORMALIZED NOISE COVARIANCE OSDN(I,J)

1	.1000E+01				
2	.1490E+00	.1000E+01			
3	.7570E-01	-.4199E-01	.1000E+01		
4	.4788E-01	-.9927E-01	.6876E+00	.1000E+01	
5	-.7633E-01	.1284E+00	-.6387E-02	-.8721E-01	.1000E+01

ESTIMATED SYSTEM PARAMETERS
(A(I),B(I))
ESTIMATED ORDER M= 12

II= 1

1	.4491E+00 .3681E+00	.1957E+01 -.4331E+00	.2320E+00	.5804E+00	-.1308E+01
2	.6633E-03 .0124E-03	.8296E+00 -.9678E-02	.4934E-02	-.9872E-02	-.1217E-01
3	.4046E-02 .1873E-01	.7438E-01 .2676E-02	.6063E+00	.4087E-01	-.1367E-01
4	.8596E-02 .1988E-01	.1196E-01 .2330E-02	.1903E+00	.5769E+00	.1925E+00
5	-.1658E-02 .8195E-02	.5464E-01 .3243E-03	-.7589E-01	.3452E-01	.6473E+00

II= 2

1	.1220E+00 -.2073E-01	-.1072E+01 .3440E+00	-.3042E+00	.2011E+00	.4177E+00
2	.1047E-02 .2178E-03	.2465E+00 .7559E-02	-.8396E-02	.1423E-01	.9605E-03

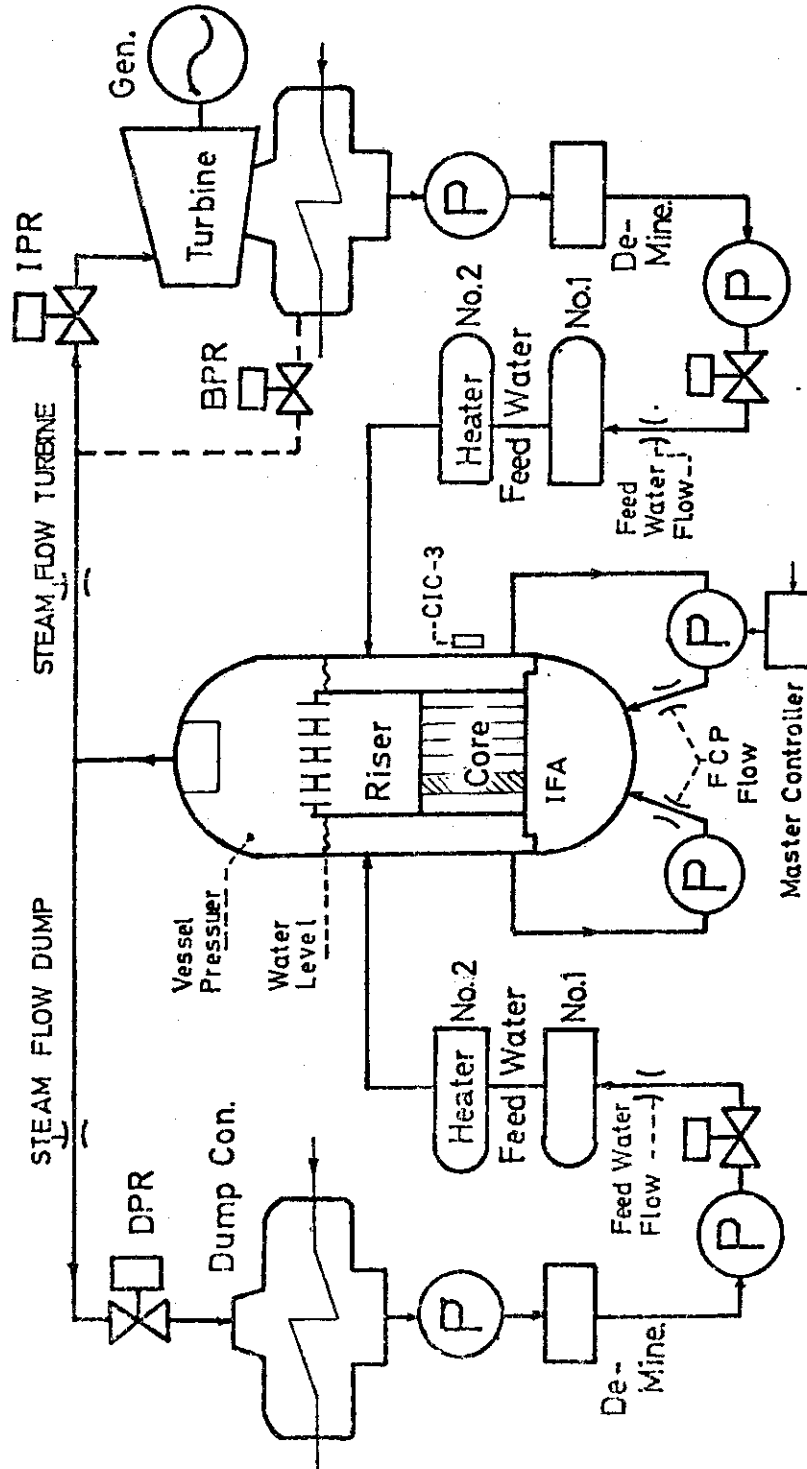


Fig. V-1 Schematic flow diagram of JPDR-II

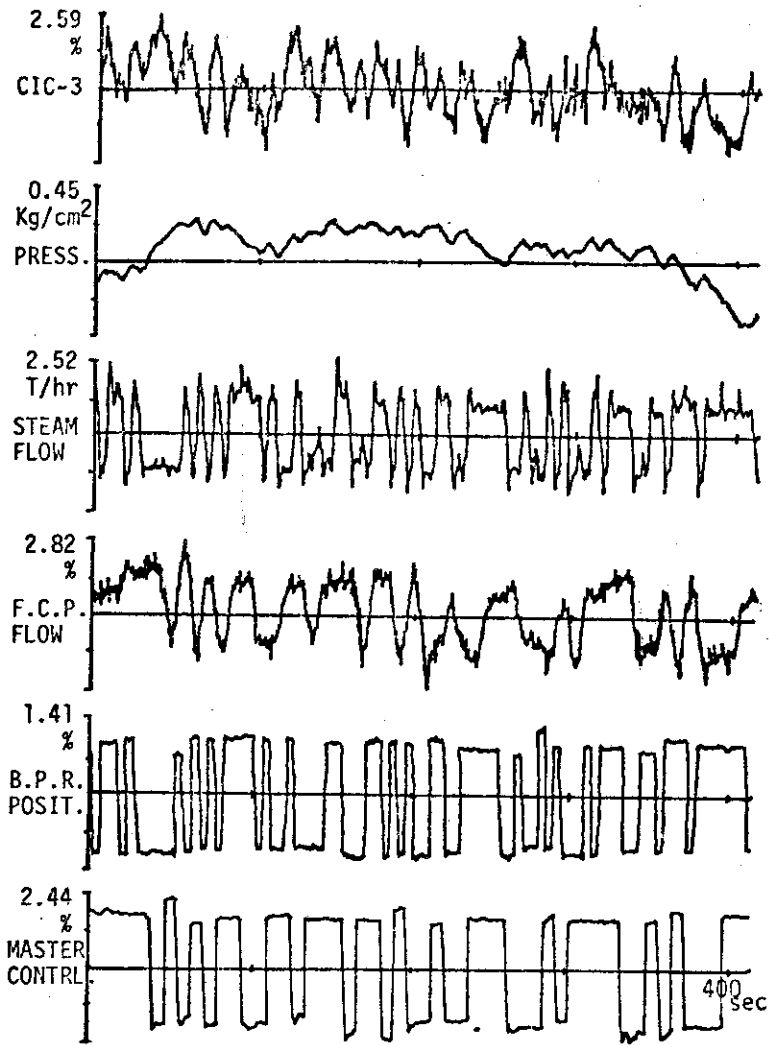


Fig. V-2 Signal wave form obtained from JPDR-II experiment

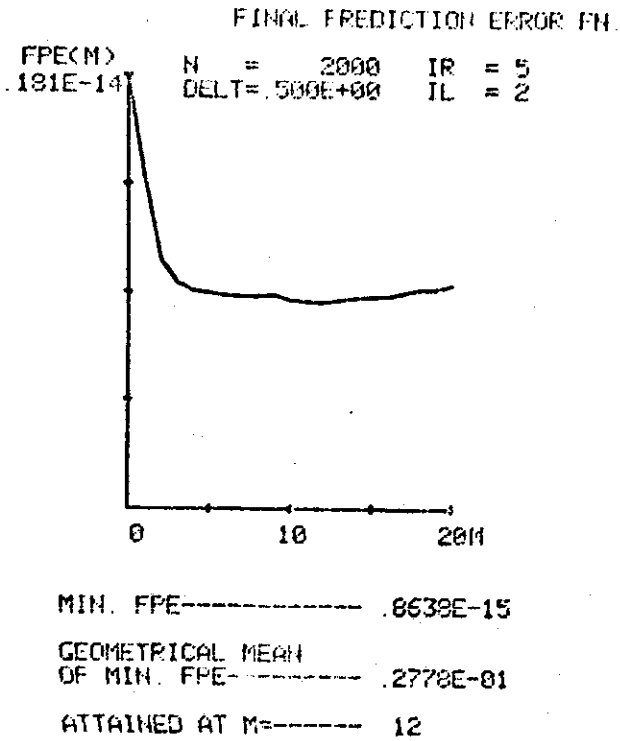


Fig. V-3 Input parameters for A-R model fitting and Final Prediction Error

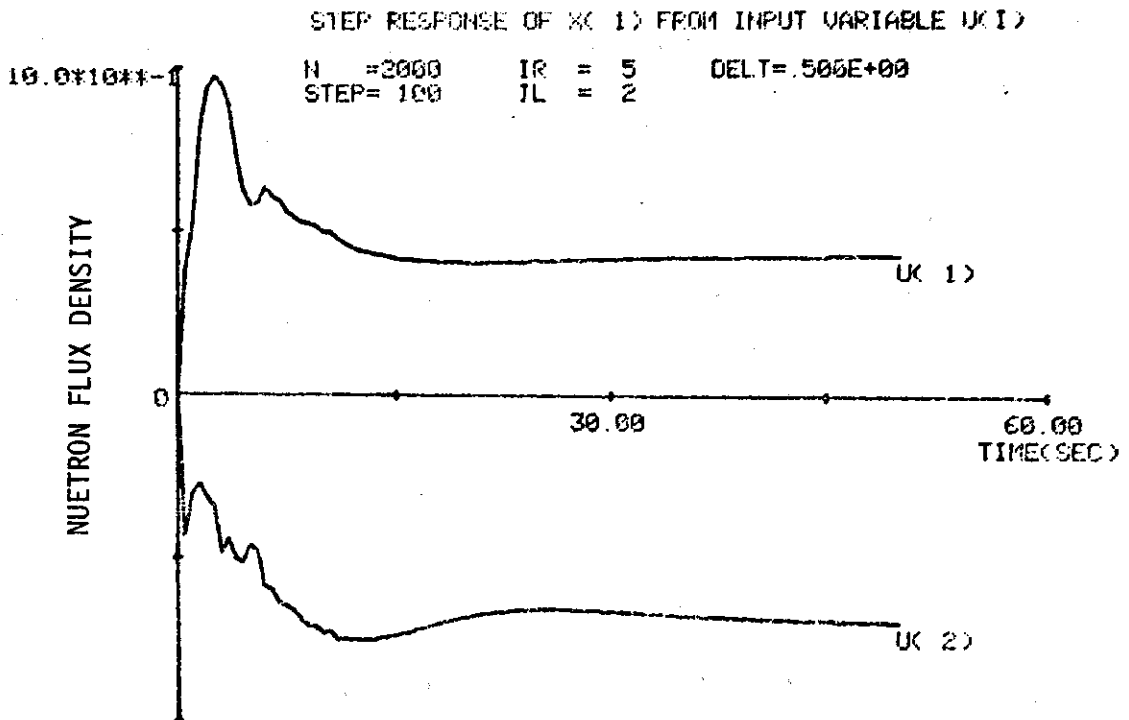
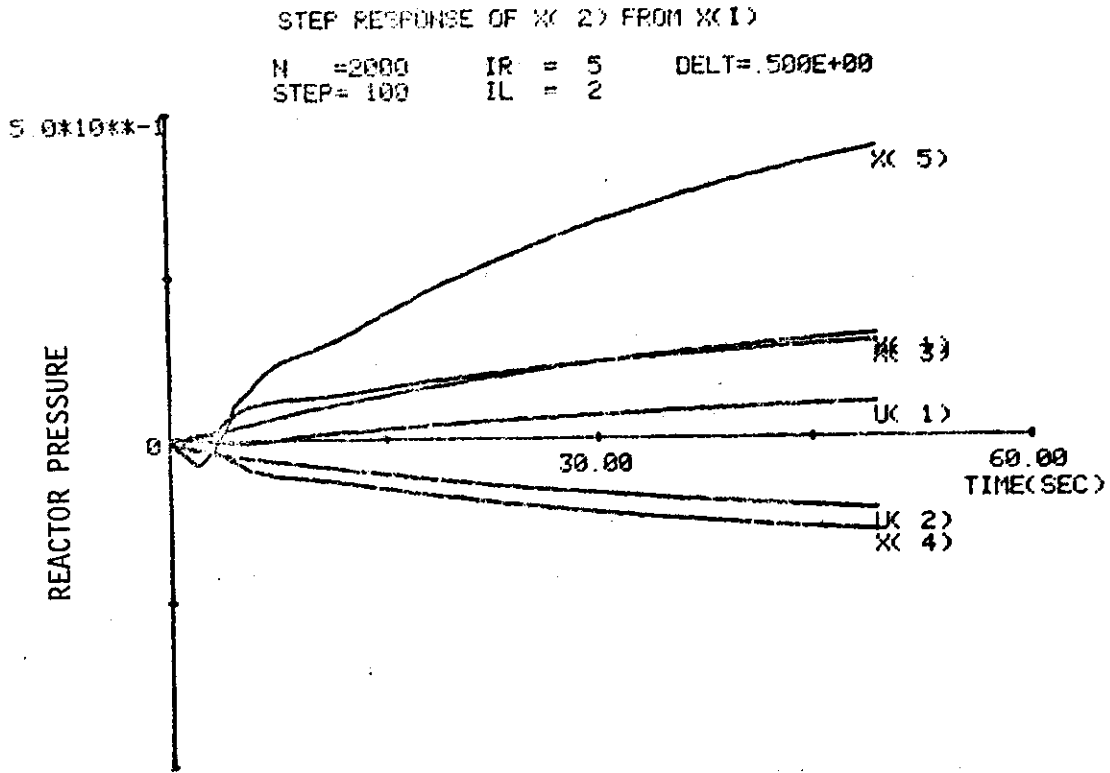


Fig. V-4 Estimated step response of Flux density vs. F.C.P. flow(U1) and Main steam flow(U2)



Disturbances; X(1): Flux density(CIC-3) X(3): Inlet flow(IFA-4)
 X(4): Outlet flow(IFA-4) X(5): Inlet temp.(IFA-4)
 U(1): F.C.P.flow U(2): Main steam flow

Fig. V-5 Estimated step response of Reactor pressure various disturbances

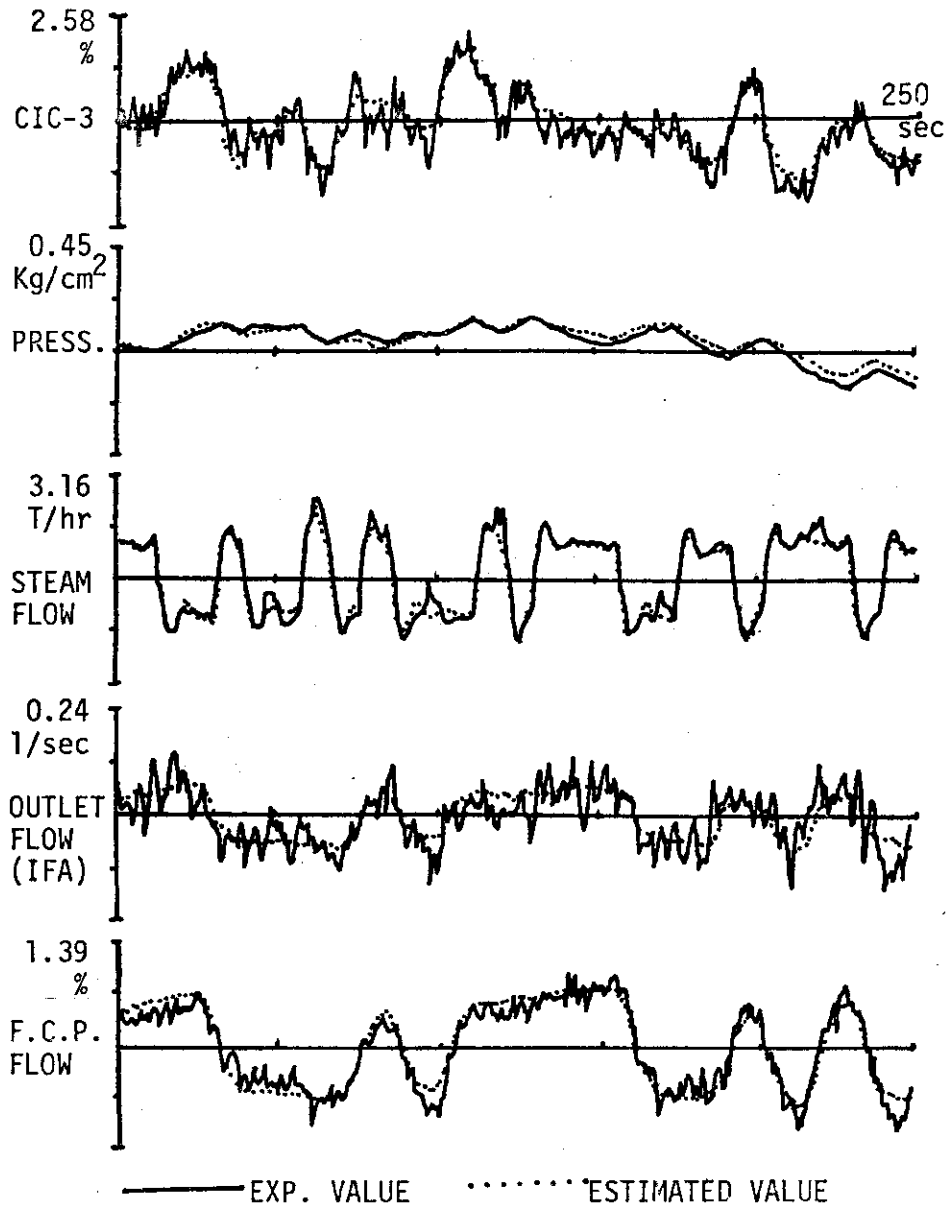


Fig. V-6 Comparison of estimated value with real data using A-R model

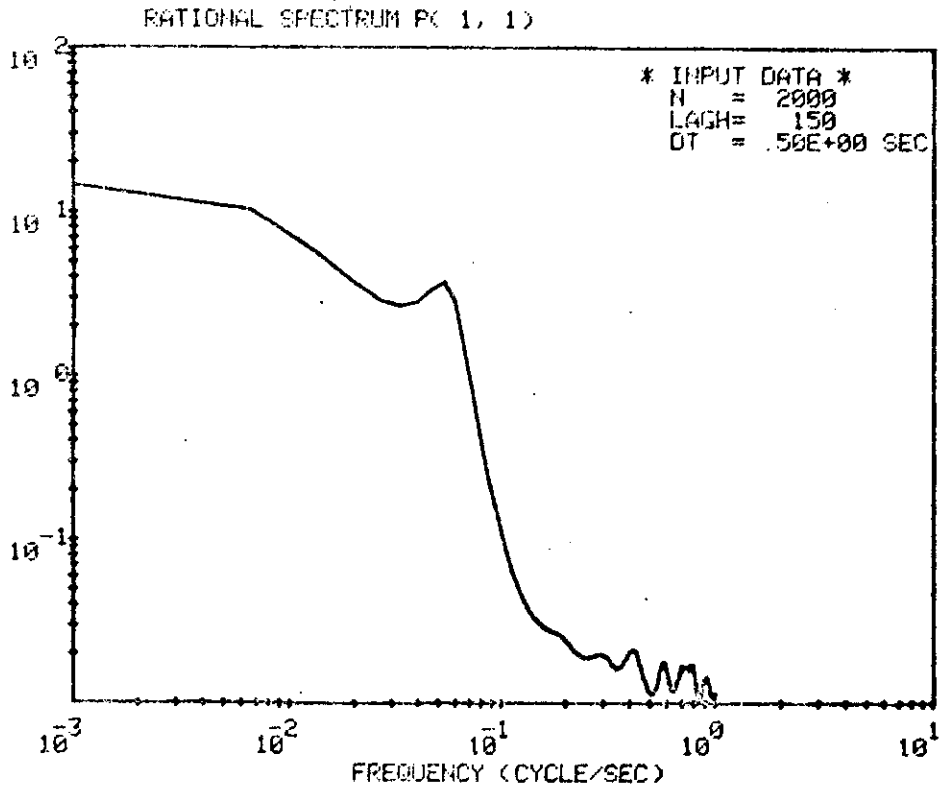


Fig. V-7 Estimated power spectrum of F.C.P. flow

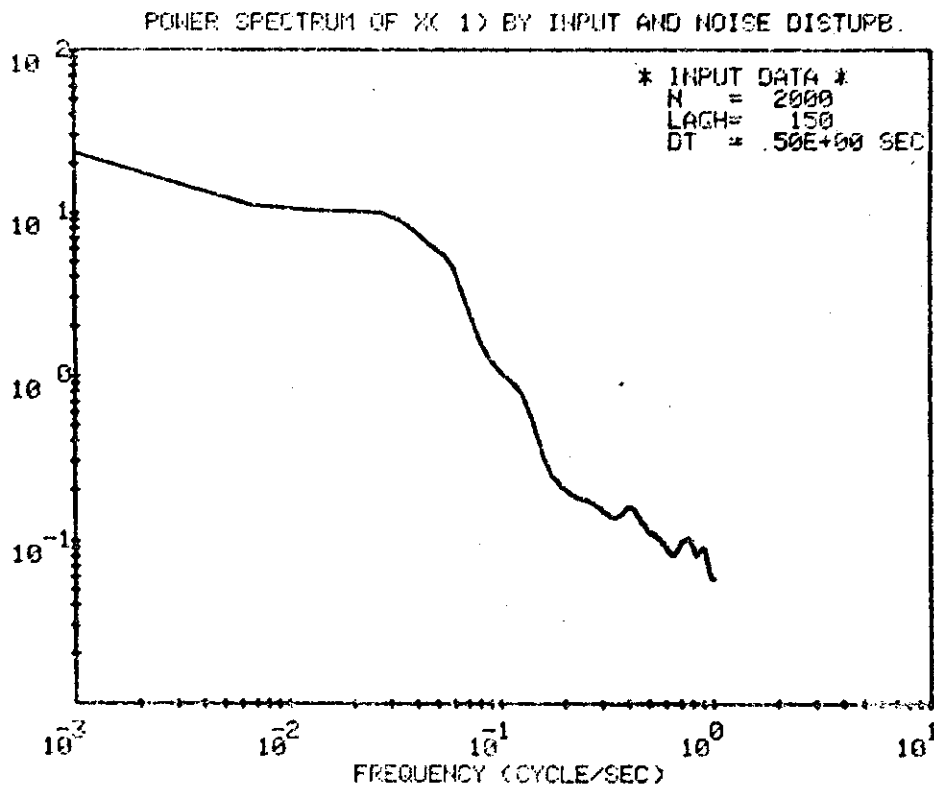


Fig. V-8 Estimated power spectrum of Flux density
 (contribution from input and system noise)

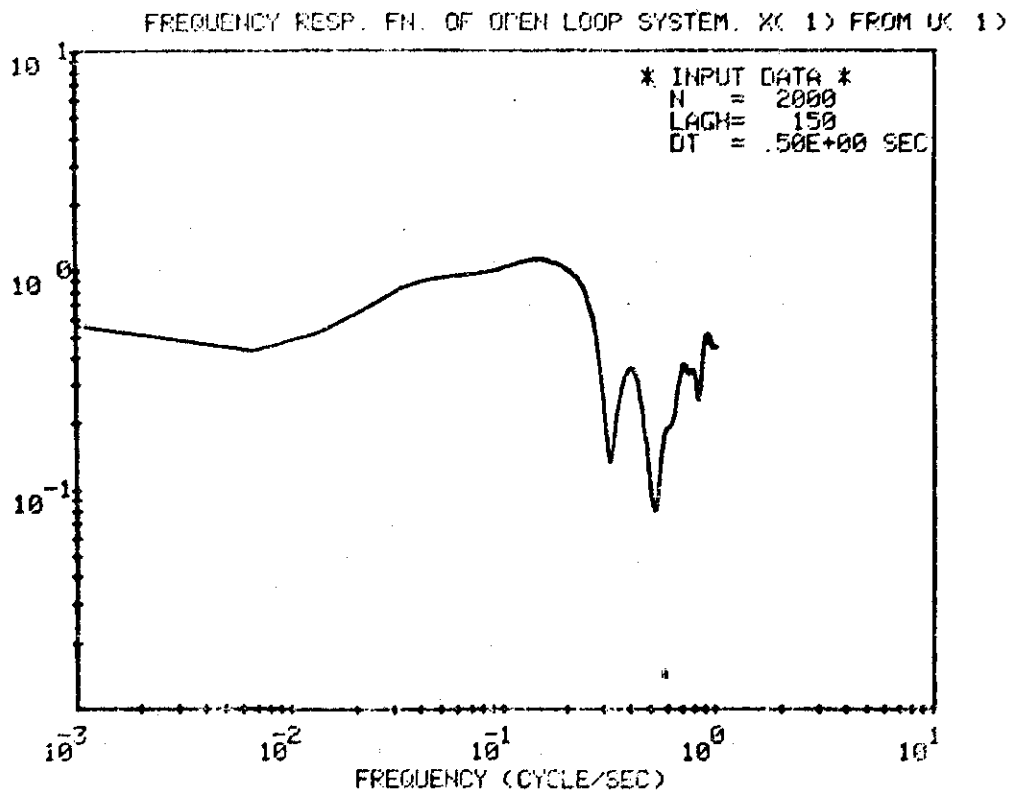


Fig. V-9 Estimated frequency response of Flux density from F.C.P. flow

VI 結 論

実験等で得られた時系列データをもとにシステムの動特性を解析する計算コードは実際的
要求が随所に見られるにもかかわらず、比較的少ない。それは、有効な同定と解析の手法及び計
算アルゴリズムに加えて膨大なデータ処理を効率良く進めて行くための計算機システムの構成
とそれに見合った計算コードの作成等いくつかのハード及びソフトウェアの条件を必要とする
からである。本解析コードはこれらの点を十分考慮して作成されており、この種の解析の経験
の無い利用者にも十分使いこなせるよう注意がはらわれている。

従来、システム同定、システムの特性解析そして制御系等のシンセシスの問題はそれぞれ異
なる領域の問題として取扱われる傾向が強くこれらを一貫した流れの中で処理して行くための
意識的な追求はここ最近その成果を現わし始めたにすぎない。本解析コードの作成に当っては
この種の研究の中ですでにいくつかの応用成果も報告されている赤池の開発した同定手法と解
析アルゴリズムを多く参考にして作成している。この計算コードの中にまだ組込まれてないい
くつかのシステム解析上有意義なアルゴリズムも残されているが、大半の解析内容は本コード
を用いて実行可能である。

VII 参 考 文 献

1. 赤池, 中川 “ダイナミックシステムの統計的解析と制御” S47. 4. サイエンス社
2. 尾熊, 藤井, 臼井, 渡辺 “ハイブリッド計算機による多変数相関及びスペクトル解析コ
ード(MLCOSP)” JAERI-M 6252 1975.10.
3. 尾熊 “データファイル作成プログラム(FILEMK)”
4. 尾熊, 松原, 北村 “原子炉・プラントへの同定手法の応用” 日本原子力学会誌
Vol. 18, No. 7 PP408~414, 1976
5. 尾熊, 藤井, 坪, 未川 “OWL-0プラントの動特性同定と制御系の評価”
JAERI-M 6801 1976.11

VI 結 論

実験等で得られた時系列データをもとにシステムの動特性を解析する計算コードは実際的
要求が随所に見られるにもかかわらず、比較的少ない。それは、有効な同定と解析の手法及び計
算アルゴリズムに加えて膨大なデータ処理を効率良く進めて行くための計算機システムの構成
とそれに見合った計算コードの作成等いくつかのハード及びソフトウェアの条件を必要とする
からである。本解析コードはこれらの点を十分考慮して作成されており、この種の解析の経験
の無い利用者にも十分使いこなせるよう注意がはらわれている。

従来、システム同定、システムの特性解析そして制御系等のシンセシスの問題はそれぞれ異
なる領域の問題として扱われる傾向が強くこれらを一貫した流れの中で処理して行くための
意識的な追求はここ最近その成果を現わし始めたにすぎない。本解析コードの作成に当っては
この種の研究の中ですでにいくつかの応用成果も報告されている赤池の開発した同定手法と解
析アルゴリズムを多く参考にして作成している。この計算コードの中にまだ組込まれてないい
くつかのシステム解析上有意義なアルゴリズムも残されているが、大半の解析内容は本コード
を用いて実行可能である。

VII 参 考 文 献

1. 赤池, 中川 “ダイナミックシステムの統計的解析と制御” S47. 4. サイエンス社
2. 尾熊, 藤井, 臼井, 渡辺 “ハイブリッド計算機による多変数相関及びスペクトル解析コ
ード(MLCOSP)” JAERI-M 6252 1975.10.
3. 尾熊 “データファイル作成プログラム(FILEMK)”
4. 尾熊, 松原, 北村 “原子炉・プラントへの同定手法の応用” 日本原子力学会誌
Vol. 18, No. 7 PP408~414, 1976
5. 尾熊, 藤井, 坪, 未川 “OWL-0プラントの動特性同定と制御系の評価”
JAERI-M 6801 1976.11

PAGE 1 C *** IDENTIFICATION ORIGINAL DATA FILE ***

APPENDIX-1 DYSAC1 CODE LIST

```

SUBROUTINE FILIDI
COMMON K,IR,N,DELTA,LLL1,LLL2
DIMENSION XX(3001),IND(15)
DIMENSION NAME1(3),NAME2(3),NAME3(3),NAME4(3)
DATA NAME2(1),NAME2(2),NAME2(3),NAME3(1),NAME3(2),NAME3(3),NAME4(1),NAME4(2),NAME4(3)
DATA NAME3(1),NAME3(2),NAME3(3),NAME4(1),NAME4(2),NAME4(3)
CALL BEGIN(9600,1)
PAGE 1 C ** KINETICS ANALYSIS CODE BASED ON THE IDENTIFICATION TECHNIQUE * 1000
CALL ERASE
WRITE(1,1)
ACCEPT(1,2)
ACCEPT(3,N,K,IR,DELTA)
IL=K-IR
M1=N+1
WRITE(1,13)
WRITE(1,12)
ACCEPT(1,12)
ACCEPT(1,12)
CALL ERASE
IF(IFI.EQ.99) GO TO 999
WRITE(1,31)
ACCEPT(9,IOEF)
IF(IDEF.NE.1) GO TO 32
CALL QMOND(22,121,2)
CALL QMOND(23,121,NAME2,3,88)
CALL QMOND(24,121,NAME2,88,1400)
WRITE(1,33)
ACCEPT(9,IOEF)
IF(IDEF.NE.1) GO TO 34
CALL QMOND(22,123,2)
CALL QMOND(23,123,NAME3,3,88)
CALL QMOND(24,123,NAME3,88,1400)
CALL QMOND(22,124,2)
CALL QMOND(23,124,NAME4,3,88)
CALL QMOND(24,124,NAME4,88,1400)
CALL QMOND(21,123,NAME3)
IT=1
K0=1
PAGE 1
WRITE(1,4)
READ(2,5) (NAME1(I),I=1,3)
CALL ERASE
WRITE(1,6) K,IR
WRITE(1,7) K
WRITE(1,8) IX
ACCEPT(9,IX)
IF(IX.EQ.0) GO TO 110
IND(11)=IX
II=1+1
IF(II.GT.K) GO TO 110
GO TO 100
K1=1+1
DO 150 I=K0,K1
IX=IND(I)
CALL QMOND(22,122,1)
CALL QMOND(21,122,NAME1)
DO 155 IR=1,15
SUBROUTINE OVRLAY(NAME,LOC)
CALL QMOND(22,123,1)
CALL QMOND(21,123,NAME)
CALL QMOND(20)
LA LOC
STA ,100
L ,100.2
RETURN
ADR 0
END
100

```

```

PAGE 2 C *** IDENTIFICATION ORIGINAL DATA FILE ***
PAGE 1 C ***** SUBROUTINE FOR MULTIPLE CORRELATION *****

CALL QMOND(17,'22,XX(1),XX(N1))
IF(18,EQ,1X) GO TO 160
CONTINUE
WRITE(1,10)
SUBROUTINE CORRID
COMMON K,IR,N,DEL1,LL1,LLL2
DIMENSION X(3001),Y(3001)
DIMENSION C1(21),C2(21),CC(11),R1(21,10,10)
DIMENSION SH(13),CP(13)
DIMENSION NAME1(3),NAME2(3)
DATA NAME1(1),NAME1(2),NAME1(3)/2HCO,2HDA,2HTA/
DATA NAME2(1),NAME2(2),NAME2(3)/2HDA,2HTA,
1900 CALL BEGIN(9600,1)
192 CALL ERASE
LAGH=20
WRITE(1,3)
WRITE(1,4)
WRITE(1,5)
ACCEPT 1,ICOR
ACCEPT 1,ICOR
IF(ICOR,EG,1) GO TO 101
IF(ICOR,EG,2) GO TO 100
GO TO 102
101 WRITE(1,2)
ACCEPT 1,ICOR
IF(JCOR,NE,1) GO TO 193
CALL QMOND(22,'22,2)
CALL QMOND(23,'22,NAME1,3,80)
CALL QMOND(24,'22,NAME1,80,600)
103 CALL QMOND(22,'22,2)
CALL QMOND(21,'22,NAME1)
WRITE(1,8)
WRITE(1,9)
ACCEPT 1,ISW1
N1=N+1
LAGH1=LAGH+1
LAGH2=LAGH+2
GOTO(221,210,221,210,221,210),ISW1
C INITIAL CONDITION INPUT + ORIGINAL DATA PRINT OUT
210 WRITE(16,50)
WRITE(16,51)
WRITE(16,53)
WRITE(16,53)
CALL QMOND(22,'21,2)
CALL QMOND(21,'21,NAME2)
GO 220 II=1,K
CALL QMOND(17,'21,X(1),X(N1))
WRITE(16,54)II
WRITE(6,55)(X(I),I=1,N)
220 CONTINUE
C COVARIANCE COMPUTATION
DO 10 II=1,K
CALL QMOND(22,'21,2)
CALL QMOND(21,'21,NAME2)
DO 20 KI=1,II
READ FROM DISC
CALL QMOND(17,'21,X(1),X(N1))
CONTINUE
20 CONTINUE
C MEAN DELETION
CALL SMDL(K,N,XMEAN)
SH(II)=XMEAN
END
155 CONTINUE
IF(18,EQ,1X) GO TO 160
CALL TPAUSE
160 CALL QMOND(16,'23,XX(1),XX(N1))
150 CONTINUE
IF(K1,GE,K) GO TO 200
KW=K1+1
GO TO 200M
CALL ERASE
WRITE(1,14) K
DO 300 II=1,K
WRITE(1,B) II
ACCEPT 9,IX
IF(IX,GT,K) GO TO 400
IF(IX,LE,0) GO TO 400
IND(II)=IX
CONTINUE
GO TO 450
400 WRITE(1,10)
GO TO 200
450 CALL QMOND(22,'21,2)
CALL QMOND(21,'21,NAME2)
DO 500 I=1,K
CALL QMOND(22,'23,2)
CALL QMOND(21,'23,NAME3)
IX=IND(I)
DO 510 II=1,K
CALL QMOND(17,'23,XX(1),XX(N1))
IF(II,EQ,IX) GO TO 520
CONTINUE
WRITE(1,10)
CALL TPAUSE
520 CALL QMOND(16,'21,XX(1),XX(N1))
500 CONTINUE
999 RETURN
1 FORMAT(70H*** KINETICS ANALYSIS CODE BASED ON THE IDENTIFICATION I
TECHNIQUE ***/4X,70HPART 1 (A-R MODEL FITTING)////)
2 FORMAT(70H* TYPE IN INPUT DATA'/2X,30NN, K, IR, DELT1//)
3 FORMAT(315,F10,5)
4 FORMAT(//,70H** IF DEFINE THE FILE, ISDATA, TYPE IN 11//)
5 FORMAT(3A2)
6 FORMAT(//,70H** TYPE IN VARIABLE NO, IN ORDER(K=1,13,5H IR=,13,1H
1),/70H (FIRST IR ARE CONTROLLED VARIABLES)1//)
7 FORMAT(//,70H** TYPE IN (IND(1), I=1,'13,1H)/70H IF NECESSARY OT
HER DATA FILE, TYPE IN 01//)
8 FORMAT(//,30HIND=1,13//)
9 FORMAT(13)
10 FORMAT(//,70H*** CHECK ERROR ***//)
11 FORMAT(15)
12 FORMAT(//,70H** IF UNEXECUTE THIS JOB, TYPE IN 991//)
13 FORMAT(//,70H** DATA SELECTION FOR A-R MODEL FITTING **1)
14 FORMAT(70H** TYPE IN THE ORDER ABOUT VARIABLES (I=1,'14,1H)/)
33 FORMAT(//,70H** IF DEFINE THE FILE, FEEEEE, FEEEEE, TYPE IN 11//)
END

```

```

PAGE 2 C ***** SUBROUTINE FOR MULTIPLE CORRELATION *****
C AUTO COVARIANCE COMPUTATION
CALL CROSCO(X,X,N,C1,LAGH1)
DO 30 IA=1,LAGH1
30 RI(AA,II,II)=C1(IA)
GOTO(136,136,146,146),ISMI
C AUTO COVARIANCE PRINT OUT
140 WRITE(16,162) II,II,SM(II)
WRITE(16,163)
CALL PCOL2(C1,C2,1,LAGH1,1)
130 IF(II.EQ.1) GO TO 18
IMI=II-1
CALL QNDND(21,121,NAME2)
DO 11 J=1,IMI
C READ FROM DISC.
CALL QNDND(17,121,Y(1),Y(N1))
MEAN DELETION
CALL SNEADL(Y,N,XMEAN)
C CROSS COVARIANCE COMPUTATION
CALL CROSCOX(Y,N,C1,LAGH1)
DO 31 IB=1,LAGH1
R1(10,II,JJ)=C1(18)
R1(10,JJ,II)=C2(18)
31 CONTINUE
DO 10 I1=1,LAGH1
I1I=1
T=0.0
IL=N-1
DO 20 J=1,II
J1=J+1
20 T=T+X(JI)*Y(J)
RETURN
END
CALL ERASE
WRITE(1,33)
DO 34 IJ=1,K
WRITE(1,35) IJ,SM(IJ)
34 CONTINUE
CALL TPAUSE
KPI=N+1
DO 70 IT=1,LAGH1
DO 70 II=1,K
DO 71 JJ=1,K
CC(JJ)=R1(11,II,JJ)
CALL QNDND(16,122,CC(1),CC(KPI))
70 CONTINUE
100 RETURN
C
1 FORMAT(15)
2 FORMAT(,70H** IF DEFINE THE FILE CADATA, TYPE IN 11/)
3 FORMAT(20H** MULTIPLE CORRELATION FOR A-R MODEL FITTING. **//)
4 FORMAT(,60H** SELECT FOLLOWING NUMBER,/,/,60M1 7 CALCULATE CORRELATI
50M1,/,60H2 7 NOT CALCULATE!)
5 FORMAT(,60H** ICOR **//)
8 FORMAT(,/,70H**TYPE IN PRINT OUT CONDITION FOR LINE PRINTER,/,70H*
1 1SM1 1 1 NOT PRINT,/,9X,60H2 1 PRINT OUT ORIGINAL DATA,/,9X,50H
23 1 PRINT OUT CORRELATION,/,9X,50H4 1 PRINT OUT BOTH DATA!)
9 FORMAT(,1H ,70H** ISMI **//)
33 FORMAT(,/,70H** MEAN VALUE OF THE OBSERVED VARIABLES. **//)
PAGE 3 C ***** SUBROUTINE FOR MULTIPLE CORRELATION *****
35 FORMAT(,15,E15,4)
50 FORMAT(1H,1,20H**MULTIPLE CORRELATION)
51 FORMAT(1H,1,17H**INITIAL CONDITION)
52 FORMAT(1H,1,20H** 15,5X,5MLAGN,15,5X,2MK,15)
53 FORMAT(1H,1,13H**ORIGINAL DATA)
54 FORMAT(1H,1,10H**CHANNEL = ,13,/)
55 FORMAT(1H ,9X,6F10,5)
162 FORMAT(,/,1H,1,14H**AUTOCOVARIANCE,5X,6HC1J(L),5X,2HI,15,5X,2HI,15,5)
* X, SM MEAN = F15, 5)
163 FORMAT(1H,1,4X,1HL,8X,6HC1J(L),5X,10H**NORMALIZED)
165 FORMAT(,/,1H,1,10H**CROSS COVARIANCE,5X,6HC1J(L),5X,2HI,15,5X,2HI,15)
* )
166 FORMAT(1H,1,4X,1HL,5X,6HC1J(L),8X,10H**NORMALIZED,4X,6HC1J(L),8X,10H
* )
NORMALIZED)
END
C
C ***** SUBROUTINE FOR COVARIANCE COMPUTATION ***
SUBROUTINE CROSCOX(Y,N,C,LAGH1)
DIMENSION X(N),Y(N),C(LAGH1)
AN=N
RN=1.0/AN
DO 10 I1=1,LAGH1
I1I=1
T=0.0
IL=N-1
DO 20 J=1,II
J1=J+1
20 T=T+X(JI)*Y(J)
RETURN
END
PAGE 1 C ***** SUBROUTINE FOR MEAN DELETION ***
SUBROUTINE SNEADL(X,N,XMEAN)
DIMENSION X(N)
XMEAN=SUMF(X,N)/AN
DO 10 I=1,N
10 X(I)=X(I)-XMEAN
RETURN
END

```

```

FUNCTION SUMF(X,N)
DIMENSION X(N)
SUMF = 0.0
DO 10 I=1,N
10 SUMF=SUMF+X(I)
RETURN
END
SURROUTINE FPEID1
COMMON K,IR,N,DELTA,LL1,LL12
DIMENSION A(21,10,10),G1(21,10,10),PP(21,10,10),PARANT(10),CI(10),
1 SD(10,10),SE(10,10),SF(10,10),XSD(10,10),XSD(10,10),D(10,10),
2 E(10,10),Z1(10,10),XSF(10,10),PFP(21),OSDN(10,10),OPPW(21)
DIMENSION IDDATA(3),NAME1(3)
DIMENSION ORPFW(21)
EQUIVALENCE (Z1(1,1),OSDN(1,1))
DATA IDDATA(1),IDDATA(2),IDDATA(3)/2HID,2HDA,2HTA/
DATA NAME1(1),NAME1(2),NAME1(3)/2HCO,2HDA,2HTA/
CALL BEGIN(9600,1)
CALL ERASE
C ABSOLUTE DIMENSIONS USED FOR SUBROUTINE CALL
L=20
MJ=10
MJ0=21
L1=L+1
L2=L+1
L3=L+1
L4=L+1
L5=L+1
L6=L+1
L7=L+1
L8=L+1
L9=L+1
L10=L+1
L11=L+1
L12=L+1
L13=L+1
L14=L+1
L15=L+1
L16=L+1
L17=L+1
L18=L+1
L19=L+1
L20=L+1
L21=L+1
L22=L+1
L23=L+1
L24=L+1
L25=L+1
L26=L+1
L27=L+1
L28=L+1
L29=L+1
L30=L+1
L31=L+1
L32=L+1
L33=L+1
L34=L+1
L35=L+1
L36=L+1
L37=L+1
L38=L+1
L39=L+1
L40=L+1
L41=L+1
L42=L+1
L43=L+1
L44=L+1
L45=L+1
L46=L+1
L47=L+1
L48=L+1
L49=L+1
L50=L+1
L51=L+1
L52=L+1
L53=L+1
L54=L+1
L55=L+1
L56=L+1
L57=L+1
L58=L+1
L59=L+1
L60=L+1
L61=L+1
L62=L+1
L63=L+1
L64=L+1
L65=L+1
L66=L+1
L67=L+1
L68=L+1
L69=L+1
L70=L+1
L71=L+1
L72=L+1
L73=L+1
L74=L+1
L75=L+1
L76=L+1
L77=L+1
L78=L+1
L79=L+1
L80=L+1
L81=L+1
L82=L+1
L83=L+1
L84=L+1
L85=L+1
L86=L+1
L87=L+1
L88=L+1
L89=L+1
L90=L+1
L91=L+1
L92=L+1
L93=L+1
L94=L+1
L95=L+1
L96=L+1
L97=L+1
L98=L+1
L99=L+1
L100=L+1
L101=L+1
L102=L+1
L103=L+1
L104=L+1
L105=L+1
L106=L+1
L107=L+1
L108=L+1
L109=L+1
L110=L+1
L111=L+1
L112=L+1
L113=L+1
L114=L+1
L115=L+1
L116=L+1
L117=L+1
L118=L+1
L119=L+1
L120=L+1
L121=L+1
L122=L+1
L123=L+1
L124=L+1
L125=L+1
L126=L+1
L127=L+1
L128=L+1
L129=L+1
L130=L+1
L131=L+1
L132=L+1
L133=L+1
L134=L+1
L135=L+1
L136=L+1
L137=L+1
L138=L+1
L139=L+1
L140=L+1
L141=L+1
L142=L+1
L143=L+1
L144=L+1
L145=L+1
L146=L+1
L147=L+1
L148=L+1
L149=L+1
L150=L+1
L151=L+1
L152=L+1
L153=L+1
L154=L+1
L155=L+1
L156=L+1
L157=L+1
L158=L+1
L159=L+1
L160=L+1
L161=L+1
L162=L+1
L163=L+1
L164=L+1
L165=L+1
L166=L+1
L167=L+1
L168=L+1
L169=L+1
L170=L+1
L171=L+1
L172=L+1
L173=L+1
L174=L+1
L175=L+1
L176=L+1
L177=L+1
L178=L+1
L179=L+1
L180=L+1
L181=L+1
L182=L+1
L183=L+1
L184=L+1
L185=L+1
L186=L+1
L187=L+1
L188=L+1
L189=L+1
L190=L+1
L191=L+1
L192=L+1
L193=L+1
L194=L+1
L195=L+1
L196=L+1
L197=L+1
L198=L+1
L199=L+1
L200=L+1
L201=L+1
L202=L+1
L203=L+1
L204=L+1
L205=L+1
L206=L+1
L207=L+1
L208=L+1
L209=L+1
L210=L+1
L211=L+1
L212=L+1
L213=L+1
L214=L+1
L215=L+1
L216=L+1
L217=L+1
L218=L+1
L219=L+1
L220=L+1
L221=L+1
L222=L+1
L223=L+1
L224=L+1
L225=L+1
L226=L+1
L227=L+1
L228=L+1
L229=L+1
L230=L+1
L231=L+1
L232=L+1
L233=L+1
L234=L+1
L235=L+1
L236=L+1
L237=L+1
L238=L+1
L239=L+1
L240=L+1
L241=L+1
L242=L+1
L243=L+1
L244=L+1
L245=L+1
L246=L+1
L247=L+1
L248=L+1
L249=L+1
L250=L+1
L251=L+1
L252=L+1
L253=L+1
L254=L+1
L255=L+1
L256=L+1
L257=L+1
L258=L+1
L259=L+1
L260=L+1
L261=L+1
L262=L+1
L263=L+1
L264=L+1
L265=L+1
L266=L+1
L267=L+1
L268=L+1
L269=L+1
L270=L+1
L271=L+1
L272=L+1
L273=L+1
L274=L+1
L275=L+1
L276=L+1
L277=L+1
L278=L+1
L279=L+1
L280=L+1
L281=L+1
L282=L+1
L283=L+1
L284=L+1
L285=L+1
L286=L+1
L287=L+1
L288=L+1
L289=L+1
L290=L+1
L291=L+1
L292=L+1
L293=L+1
L294=L+1
L295=L+1
L296=L+1
L297=L+1
L298=L+1
L299=L+1
L300=L+1
L301=L+1
L302=L+1
L303=L+1
L304=L+1
L305=L+1
L306=L+1
L307=L+1
L308=L+1
L309=L+1
L310=L+1
L311=L+1
L312=L+1
L313=L+1
L314=L+1
L315=L+1
L316=L+1
L317=L+1
L318=L+1
L319=L+1
L320=L+1
L321=L+1
L322=L+1
L323=L+1
L324=L+1
L325=L+1
L326=L+1
L327=L+1
L328=L+1
L329=L+1
L330=L+1
L331=L+1
L332=L+1
L333=L+1
L334=L+1
L335=L+1
L336=L+1
L337=L+1
L338=L+1
L339=L+1
L340=L+1
L341=L+1
L342=L+1
L343=L+1
L344=L+1
L345=L+1
L346=L+1
L347=L+1
L348=L+1
L349=L+1
L350=L+1
L351=L+1
L352=L+1
L353=L+1
L354=L+1
L355=L+1
L356=L+1
L357=L+1
L358=L+1
L359=L+1
L360=L+1
L361=L+1
L362=L+1
L363=L+1
L364=L+1
L365=L+1
L366=L+1
L367=L+1
L368=L+1
L369=L+1
L370=L+1
L371=L+1
L372=L+1
L373=L+1
L374=L+1
L375=L+1
L376=L+1
L377=L+1
L378=L+1
L379=L+1
L380=L+1
L381=L+1
L382=L+1
L383=L+1
L384=L+1
L385=L+1
L386=L+1
L387=L+1
L388=L+1
L389=L+1
L390=L+1
L391=L+1
L392=L+1
L393=L+1
L394=L+1
L395=L+1
L396=L+1
L397=L+1
L398=L+1
L399=L+1
L400=L+1
L401=L+1
L402=L+1
L403=L+1
L404=L+1
L405=L+1
L406=L+1
L407=L+1
L408=L+1
L409=L+1
L410=L+1
L411=L+1
L412=L+1
L413=L+1
L414=L+1
L415=L+1
L416=L+1
L417=L+1
L418=L+1
L419=L+1
L420=L+1
L421=L+1
L422=L+1
L423=L+1
L424=L+1
L425=L+1
L426=L+1
L427=L+1
L428=L+1
L429=L+1
L430=L+1
L431=L+1
L432=L+1
L433=L+1
L434=L+1
L435=L+1
L436=L+1
L437=L+1
L438=L+1
L439=L+1
L440=L+1
L441=L+1
L442=L+1
L443=L+1
L444=L+1
L445=L+1
L446=L+1
L447=L+1
L448=L+1
L449=L+1
L450=L+1
L451=L+1
L452=L+1
L453=L+1
L454=L+1
L455=L+1
L456=L+1
L457=L+1
L458=L+1
L459=L+1
L460=L+1
L461=L+1
L462=L+1
L463=L+1
L464=L+1
L465=L+1
L466=L+1
L467=L+1
L468=L+1
L469=L+1
L470=L+1
L471=L+1
L472=L+1
L473=L+1
L474=L+1
L475=L+1
L476=L+1
L477=L+1
L478=L+1
L479=L+1
L480=L+1
L481=L+1
L482=L+1
L483=L+1
L484=L+1
L485=L+1
L486=L+1
L487=L+1
L488=L+1
L489=L+1
L490=L+1
L491=L+1
L492=L+1
L493=L+1
L494=L+1
L495=L+1
L496=L+1
L497=L+1
L498=L+1
L499=L+1
L500=L+1
L501=L+1
L502=L+1
L503=L+1
L504=L+1
L505=L+1
L506=L+1
L507=L+1
L508=L+1
L509=L+1
L510=L+1
L511=L+1
L512=L+1
L513=L+1
L514=L+1
L515=L+1
L516=L+1
L517=L+1
L518=L+1
L519=L+1
L520=L+1
L521=L+1
L522=L+1
L523=L+1
L524=L+1
L525=L+1
L526=L+1
L527=L+1
L528=L+1
L529=L+1
L530=L+1
L531=L+1
L532=L+1
L533=L+1
L534=L+1
L535=L+1
L536=L+1
L537=L+1
L538=L+1
L539=L+1
L540=L+1
L541=L+1
L542=L+1
L543=L+1
L544=L+1
L545=L+1
L546=L+1
L547=L+1
L548=L+1
L549=L+1
L550=L+1
L551=L+1
L552=L+1
L553=L+1
L554=L+1
L555=L+1
L556=L+1
L557=L+1
L558=L+1
L559=L+1
L560=L+1
L561=L+1
L562=L+1
L563=L+1
L564=L+1
L565=L+1
L566=L+1
L567=L+1
L568=L+1
L569=L+1
L570=L+1
L571=L+1
L572=L+1
L573=L+1
L574=L+1
L575=L+1
L576=L+1
L577=L+1
L578=L+1
L579=L+1
L580=L+1
L581=L+1
L582=L+1
L583=L+1
L584=L+1
L585=L+1
L586=L+1
L587=L+1
L588=L+1
L589=L+1
L590=L+1
L591=L+1
L592=L+1
L593=L+1
L594=L+1
L595=L+1
L596=L+1
L597=L+1
L598=L+1
L599=L+1
L600=L+1
L601=L+1
L602=L+1
L603=L+1
L604=L+1
L605=L+1
L606=L+1
L607=L+1
L608=L+1
L609=L+1
L610=L+1
L611=L+1
L612=L+1
L613=L+1
L614=L+1
L615=L+1
L616=L+1
L617=L+1
L618=L+1
L619=L+1
L620=L+1
L621=L+1
L622=L+1
L623=L+1
L624=L+1
L625=L+1
L626=L+1
L627=L+1
L628=L+1
L629=L+1
L630=L+1
L631=L+1
L632=L+1
L633=L+1
L634=L+1
L635=L+1
L636=L+1
L637=L+1
L638=L+1
L639=L+1
L640=L+1
L641=L+1
L642=L+1
L643=L+1
L644=L+1
L645=L+1
L646=L+1
L647=L+1
L648=L+1
L649=L+1
L650=L+1
L651=L+1
L652=L+1
L653=L+1
L654=L+1
L655=L+1
L656=L+1
L657=L+1
L658=L+1
L659=L+1
L660=L+1
L661=L+1
L662=L+1
L663=L+1
L664=L+1
L665=L+1
L666=L+1
L667=L+1
L668=L+1
L669=L+1
L670=L+1
L671=L+1
L672=L+1
L673=L+1
L674=L+1
L675=L+1
L676=L+1
L677=L+1
L678=L+1
L679=L+1
L680=L+1
L681=L+1
L682=L+1
L683=L+1
L684=L+1
L685=L+1
L686=L+1
L687=L+1
L688=L+1
L689=L+1
L690=L+1
L691=L+1
L692=L+1
L693=L+1
L694=L+1
L695=L+1
L696=L+1
L697=L+1
L698=L+1
L699=L+1
L700=L+1
L701=L+1
L702=L+1
L703=L+1
L704=L+1
L705=L+1
L706=L+1
L707=L+1
L708=L+1
L709=L+1
L710=L+1
L711=L+1
L712=L+1
L713=L+1
L714=L+1
L715=L+1
L716=L+1
L717=L+1
L718=L+1
L719=L+1
L720=L+1
L721=L+1
L722=L+1
L723=L+1
L724=L+1
L725=L+1
L726=L+1
L727=L+1
L728=L+1
L729=L+1
L730=L+1
L731=L+1
L732=L+1
L733=L+1
L734=L+1
L735=L+1
L736=L+1
L737=L+1
L738=L+1
L739=L+1
L740=L+1
L741=L+1
L742=L+1
L743=L+1
L744=L+1
L745=L+1
L746=L+1
L747=L+1
L748=L+1
L749=L+1
L750=L+1
L751=L+1
L752=L+1
L753=L+1
L754=L+1
L755=L+1
L756=L+1
L757=L+1
L758=L+1
L759=L+1
L760=L+1
L761=L+1
L762=L+1
L763=L+1
L764=L+1
L765=L+1
L766=L+1
L767=L+1
L768=L+1
L769=L+1
L770=L+1
L771=L+1
L772=L+1
L773=L+1
L774=L+1
L775=L+1
L776=L+1
L777=L+1
L778=L+1
L779=L+1
L780=L+1
L781=L+1
L782=L+1
L783=L+1
L784=L+1
L785=L+1
L786=L+1
L787=L+1
L788=L+1
L789=L+1
L790=L+1
L791=L+1
L792=L+1
L793=L+1
L794=L+1
L795=L+1
L796=L+1
L797=L+1
L798=L+1
L799=L+1
L800=L+1
L801=L+1
L802=L+1
L803=L+1
L804=L+1
L805=L+1
L806=L+1
L807=L+1
L808=L+1
L809=L+1
L810=L+1
L811=L+1
L812=L+1
L813=L+1
L814=L+1
L815=L+1
L816=L+1
L817=L+1
L818=L+1
L819=L+1
L820=L+1
L821=L+1
L822=L+1
L823=L+1
L824=L+1
L825=L+1
L826=L+1
L827=L+1
L828=L+1
L829=L+1
L830=L+1
L831=L+1
L832=L+1
L833=L+1
L834=L+1
L835=L+1
L836=L+1
L837=L+1
L838=L+1
L839=L+1
L840=L+1
L841=L+1
L842=L+1
L843=L+1
L844=L+1
L845=L+1
L846=L+1
L847=L+1
L848=L+1
L849=L+1
L850=L+1
L851=L+1
L852=L+1
L853=L+1
L854=L+1
L855=L+1
L856=L+1
L857=L+1
L858=L+1
L859=L+1
L860=L+1
L861=L+1
L862=L+1
L863=L+1
L864=L+1
L865=L+1
L866=L+1
L867=L+1
L868=L+1
L869=L+1
L870=L+1
L871=L+1
L872=L+1
L873=L+1
L874=L+1
L875=L+1
L876=L+1
L877=L+1
L878=L+1
L879=L+1
L880=L+1
L881=L+1
L882=L+1
L883=L+1
L884=L+1
L885=L+1
L886=L+1
L887=L+1
L888=L+1
L889=L+1
L890=L+1
L891=L+1
L892=L+1
L893=L+1
L894=L+1
L895=L+1
L896=L+1
L897=L+1
L898=L+1
L899=L+1
L900=L+1
L901=L+1
L902=L+1
L903=L+1
L904=L+1
L905=L+1
L906=L+1
L907=L+1
L908=L+1
L909=L+1
L910=L+1
L911=L+1
L912=L+1
L913=L+1
L914=L+1
L915=L+1
L916=L+1
L917=L+1
L918=L+1
L919=L+1
L920=L+1
L921=L+1
L922=L+1
L923=L+1
L924=L+1
L925=L+1
L926=L+1
L927=L+1
L928=L+1
L929=L+1
L930=L+1
L931=L+1
L932=L+1
L933=L+1
L934=L+1
L935=L+1
L936=L+1
L937=L+1
L938=L+1
L939=L+1
L940=L+1
L941=L+1
L942=L+1
L943=L+1
L944=L+1
L945=L+1
L946=L+1
L947=L+1
L948=L+1
L949=L+1
L950=L+1
L951=L+1
L952=L+1
L953=L+1
L954=L+1
L955=L+1
L956=L+1
L957=L+1
L958=L+1
L959=L+1
L960=L+1
L961=L+1
L962=L+1
L963=L+1
L964=L+1
L965=L+1
L966=L+1
L967=L+1
L968=L+1
L969=L+1
L970=L+1
L971=L+1
L972=L+1
L973=L+1
L974=L+1
L975=L+1
L976=L+1
L977=L+1
L978=L+1
L979=L+1
L980=L+1
L981=L+1
L982=L+1
L983=L+1
L984=L+1
L985=L+1
L986=L+1
L987=L+1
L988=L+1
L989=L+1
L990=L+1
L991=L+1
L992=L+1
L993=L+1
L994=L+1
L995=L+1
L996=L+1
L997=L+1
L998=L+1
L999=L+1
L1000=L+1

```

PAGE 1 C *** SUBROUTINE TO PRINT OUT 2 VECTORS ***

```

SUBROUTINE PRCOL2(P1,P2,INDI,INDL,ISHIFT)
THIS SUBROUTINE PRINTS VECTORS
(P1(I),P2(I),I=INDI,INDL)
COLUMNWISE IN THE FORMAT
(I=ISHIFT,P1(I),P2(I),I=INDI,INDL)
DIMENSION P1(INDL),P2(INDL)
DO 61 I=INDI,INDL
IM1=ISHIFT
IM2=ISHIFT
61 WRITE(16,62) IM1,P1(I),P2(I)
RETURN
62 FORMAT(1H ,15,2X,2E14.5)
END

```

PAGE 1 C *** SUBROUTINE TO PRINT OUT 4 VECTORS ***

```

SUBROUTINE PRCOL4(P1,P2,P3,P4,INDI,INDL,ISHIFT)
THIS SUBROUTINE PRINTS VECTORS
(P1(I),P2(I),P3(I),P4(I),I=INDI,INDL)
COLUMNWISE IN THE FORMAT
(I=ISHIFT,P1(I),P2(I),P3(I),P4(I),I=INDI,INDL)
DIMENSION P1(INDL),P2(INDL),P3(INDL),P4(INDL)
DO 61 I=INDI,INDL
IM1=ISHIFT
IM2=ISHIFT
IM3=ISHIFT
IM4=ISHIFT
61 WRITE(16,62) IM1,P1(I),P2(I),P3(I),P4(I)
RETURN
62 FORMAT(1H ,15,2X,4E14.5)
END

```


C	OPPEC, ORFPEC, COMPUTATION		WRITE(16,15)
C	CALL SPPEC(SD,N,IP,IR,MS,OFPEC,ORFPEC,OPPEC,MJ)		WRITE(16,16) N,K,IR,IL,DELT
	OPFH(1)=OPPEC		DO 500 M=1,L1
	ORFPM(1)=ORFPEC		M1=1
C	ITERATION M=1 TO L	500	WRITE(16,4) M1,OPFH(M),ORFPM(M)
	DO 400 M=1,L	C	MIN, FPEC, MIN, RFPEC PRINT OUT
C	INVERSE OF SD OF COMPUTATION		WRITE(16,5) OPPEC,ORFPEC,IFPEC
	CALL INVDY(XSD,SOE,IP,MJ)		RT=19
	CALL INVDY(XSF,SPDET,IP,MJ)		RR=1,0/RI
C	D,E,SD,SP, COMPUTATION		GGG=1,
	CALL MULPLY(SE,XSF,D,IP,IP,MJ,MJ,MJ)		DO 51 IG=1,IR
	CALL TRAMD(3E,XSD,E,IP,IP,MJ,MJ,MJ)	51	GGG=GGG+1(IG)
	CALL TRAMD(D,SE,Z1,IP,IP,MJ,MJ,MJ)		GX=OFPEC/GGG
	CALL SUBTAL(SD,Z1,IP,IP,MJ,MJ)		GOFPEC=GX*RR
	CALL MULPLY(E,SE,X1,IP,IP,MJ,MJ,MJ)		WRITE(16,6) GOFPEC
	CALL SUBTAL(SF,Z1,IP,IP,MJ,MJ)		OPF=OPPEC
	M8=M		CALL PLOTFP(OPF,IFPEC,OPFH,GOFPEC,L,N,IR,IL,DELT)
	DO 410 II=1,IR		CALL ERASE
	DO 410 JJ=1,IP		WRITE(16,7)
	XSD(II,JJ)=SD(II,JJ)		DO 503 II=1,IR
410	XSF(II,JJ)=SF(II,JJ)		WRITE(16,9) II,(OSD(II,II),JJ),JJ=1,II)
C	FPEC, RPPEC, COMPUTATION	563	CONTINUE
	CALL SPPEC(SD,N,IP,IR,MS,FPEC,RPPEC,ORFPEC,MJ)		IF(IR,EO,1) GO TO 458
C	FPEC RPPEC PRINT OUT	C	OSD(1,1) NORMALIZATION
	OPFH(M)=OPPEC		DO 450 I=1,IR
	ORFPM(M)=ORFPEC		DO 450 J=1,I
C	FORWARD AND BACKWARD PREDICTOR COMPUTATION		IF(J,EO,1) GO TO 451
	CALL COEFAR(A1,S1,D,E,MS,IP,MJ,MJ)		OIJ=OSD(I,1)*OSD(J,1)
C	MIN, FPEC, MIN, RPPEC, COMPUTATION		OSDN(I,1)=OSD(I,1)/SBRT(OIJ)
	IF(OPPEC,LE,RFPEC) GO TO 448		GO TO 459
	CALL GMOND(22,122,2)	451	OSDN(I,1)=1.0
	CALL GMOND(21,122,10DATA)	450	CONTINUE
	ORFPEC=RFPEC		WRITE(16,8)
	IFPEC=M		DO 453 I=1,IR
	DO 500 II=1,IR		WRITE(16,9) I,(OSDN(I,1),JJ=1,I)
	DO 500 JJ=1,IP		CONTINUE
	OSD(II,1)=SD(II,1)		IF(IFPEC,LE,0) GO TO 699
	PARAM(1)=IFPEC		GO TO 459
	PARAM(2)=RFPEC	458	OSDN(I,1)=OSD(1,1)
	PARAM(3)=RFPEC	459	CALL GMOND(22,122,2)
	CALL GMOND(16,122,PARAM(1),PARAM(10))		CALL GMOND(21,122,10DATA)
	DO 561 I=1,M		IP1=IP+1
	DO 561 J=1,IR		M=PARAM(1)
	DO 562 JJ=1,IP		DO 460 II=1,M
562	PP(JJ)=A1(II,II,II)		CALL GMOND(17,122,PP(1),PP(IP1))
561	CONTINUE		DO 461 JJ=1,IP
448	IF(M,EO,1) GO TO 488	461	A1(II,II,II)=PP(JJ)
C	SE COMPUTATION	460	CONTINUE
	CALL NERSE(A1,SE,MS,IP,MJ,MJ)		WRITE(16,10)
	CONTINUE		M=PARAM(1)
488	DO 110 II=1,IR		DO 462 II=1,M
	DO 111 JJ=1,IP		WRITE(16,20) M
111	PP(JJ)=OSD(II,II)		WRITE(16,21) IY
	CALL GMOND(16,122,PP(1),PP(IP1))		DO 462 II=1,IR
110	CONTINUE	462	WRITE(16,9) II,(A1(II,II,II),JJ=1,IP)

```

699 RETURN
1  FORMAT(/,70H **** A-R MODEL FITTING *****//,70H ** RESTRICTED C
   10NDITION//)
2  FORMAT(5X,2HN#,14/,5X,2HL#,14/,5X,3HIR#,13/,5X,3HIL#,13/,5X,5HDEL
   1#E10.4)
3  FORMAT(1H0,/,70H *** FINAL PREDICTION ERROR **//,1H ,4X,1HM,6X,
   1#E10.4)
4  FORMAT(1H ,15,2X,1E14.3)
5  FORMAT(/,1H ,6X,13MINIMUM FPEC,3X,14MINIMUM RPEC,/,10X,E12.5,2
   1X,E12.5//,17X,40MATTAINED AT M=1,15)
6  FORMAT(1H0,6X,70HGEOMETRICAL MEAN OF MTN FPEC(M)1/,8X,E12.4)
7  FORMAT(1H1,70H**ESTIMATED NOISE 2-ND MOMENT**1/10X,10HSD(11,1J)1)
8  FORMAT(/,70H ***NORMALIZED NOISE COVARIANCE OSDN(T,J)1/1)
9  FORMAT(1H0,15,3X,5E12.4//,6X,5E12.4)
10 FORMAT(/,70H ***ESTIMATED SYSTEM PARAMETERS**1/10X,11HCA(1),
   18(1))
11 FORMAT(/,70H** IF DEFINE THE FILE,DDATA., TYPE IN 1'//)
12 FORMAT(15)
13 FORMAT(/,70H** IF UNEXECEUTE THIS JOB, TYPE IN 99'//)
14 FORMAT(1H1,/,70H *** A-R MODEL FITTING FOR T., S., DATA ***1/30H **
   1 INPUT DATA **1/1)
15 FORMAT(1H ,3X,5MN #,15//,4X,5HK #,15//,4X,5HIR #,15/4X,5HIL #
   1,15//,4X,5HDEL1#,E12.4/)
16 FORMAT(1H ,70H ESTIMATED ORDER M=1,13)
20 FORMAT(1H0,20HIT=1,13)
21 END

SUBROUTINE COEFARAI,B1,D,E,MS,K,MJ0,MJ)
THIS SUBROUTINE COMPUTS FORWARDIAL AND BACKWARD(B).PREDICTOR
COEFFICIENTS.
DIMENSION AI(MJ0,MJ,MJ),BI(MJ0,MJ,MJ),D(MJ,MJ),E(MJ,MJ),A(10,10),
1 B(10,10),Z1(10,10),Z2(10,10)
IF(MS.EQ.1) GO TO 40
MSI=MS-1
DO 10 I=1,MSH
MHI=MS-I
DO 20 J=1,K
DO 20 JJ=1,K
A(I,J)=AI(I,J,JJ)
B(I,J,JJ)=BI(MHI,I,JJ)
CALL MULTIPLY(D,B,Z1,K,K,K,K,MJ,MJ,MJ)
CALL MULTIPLY(E,A,Z2,K,K,K,K,MJ,MJ,MJ)
CALL SUBTAL(A,Z1,K,K,MJ,MJ)
CALL SUBTAL(B,Z2,K,K,MJ,MJ)
DO 21 I=1,K
DO 21 JJ=1,K
A(I,I,JJ)=A(I,I,JJ)
B1(MHI,I,JJ)=B(I,J,J)
CONTINUE
DO 30 I=1,K
DO 30 JJ=1,K
AI(MS,I,JJ)=D(I,I,JJ)
B1(MS,I,JJ)=E(I,I,JJ)
RETURN
END
10
END

SUBROUTINE NENSE(A1,SE,MS,K,MJ0,MJ)
SE COMPUTATION
DIMENSION A1(MJ0,MJ,MJ),SE(MJ,MJ),A(10,10),R(10,10),Z(10,10)
DO 10 I=1,K
DO 10 J=1,K
Z(I,I,JJ)=0.,0
MSP2=MS+2
DO 11 I=1,MS
MHI=MSP2-I
DO 12 I=1,K
DO 12 JJ=1,K
A(I,I,JJ)=A1(I,JJ)
CALL REDATA(R,MHI,K)
CALL MULTIPLY(A,R,SE,K,K,K,MJ,MJ,MJ)
CALL MATADL(Z,SE,K,K,MJ,MJ)
CALL REDATA(R,MSP2,K)
CALL SUBTAC(R,Z,SE,K,K,MJ,MJ)
RETURN
END
11
END

SUBROUTINE REDATA(R,MT,K)
DIMENSION R(10,10),RR(11)
DATA NAME1(1),NAME1(2),NAME1(3)/2HCO,2HDA,2HTA/
K1=K+1
CALL QMOND(Z2,121,2)
CALL QMOND(21,121,NAME1)
DO 10 I=1,MT
DO 10 II=1,K
DO 11 JJ=1,K
CALL QMOND(17,121,RR(1),RR(K1))
DO 11 JJ=1,K
R(II,JJ)=RR(JJ)
CONTINUE
RETURN
END
10
END

SUBROUTINE SFPEC(SD,N,K,IR,MS,Z,RZ,00Z,MJ)
FPEC COMPUTATION
DIMENSION SD(MJ,MJ),SD1(10,10)
ANEN
KMSK*MS
ANEN+1+KM
ANEN-1-KM
ANEN-1-KM
APR=AP/ANM
DO 9 I=1,IR
DO 9 J=1,IR
SD1(I,J)=SD(I,J)
CALL SUBDET(SD1,SORM,IR,MJ)
Z=APR*SORM
IF(MS.NE.0) GO TO 10
00Z=1.0/Z
RZ=Z*00Z
RETURN
END
10
END

```

```

SUBROUTINE SUBDEY(X,XDETHI,MM,MJ)
C THIS SUBROUTINE COMPUTES THE DETERMINANT OF UPPER LEFT MM X MM
C OF X. FOR GENERAL USE STATEMENTS 20-21 SHOULD BE RESTORED.
C X=ORIGINAL MATRIX
C XDETHI=DETERMINANT OF UPPER LEFT MM X MM OF X
C MJ=ABSOLUTE DIMENSION OF X IN THE MAIN ROUTINE
C DIMENSION X(MJ,MJ)
XDETHI=1.0
IF(MM.EQ.1) GO TO 18
MMI=MM-1
DO 10 I=1,MMI
XDETHI=XDETHI*X(I,I)
XC=1.0/X(I,I)
II=I+1
DO 15 J=II,MM
XC=X(J,II)*XC
DO 16 K=II,MM
X(J,K)=X(J,K)-X(I,K)*XC
15 CONTINUE
16 XDETHI=XDETHI*X(MM,MM)
17 RETURN
END

```

```

SUBROUTINE SUBTAC(X,Y,Z,MM,NN,MN,MJ1,MJ2)
C MATRIX SUBTRACTION
C Z=X-Y
C (UPPER LEFT MM X NN OF X)=(UPPER LEFT MM X NN OF Y)
C (MJ1,MJ2) ABSOLUTE DIMENSION OF X, Y AND Z IN THE MAIN ROUTINE
C DIMENSION X(MJ1,MJ2),Y(MJ1,MJ2),Z(MJ1,MJ2)
DO 10 I=1,MM
DO 16 J=1,NN
Z(I,J)=X(I,J)-Y(I,J)
RETURN
END

```

```

SUBROUTINE SUBTAD(X,Y,MM,NN,MN,MJ1,MJ2)
C MATRIX ADDITION
C Z=X+Y
C (UPPER LEFT MM X NN OF X)=(UPPER LEFT MM X NN OF Y)
C (MJ1,MJ2)=ABSOLUTE DIMENSION OF X AND Y IN THE MAIN ROUTINE
C DIMENSION X(MJ1,MJ2),Y(MJ1,MJ2)
DO 10 I=1,MM
DO 16 J=1,NN
Z(I,J)=X(I,J)+Y(I,J)
RETURN
END

```

```

SUBROUTINE SUBTAL(X,Y,MM,NN,MN,MJ1,MJ2)
C MATRIX SUBTRACTION
C Z=X-Y
C (UPPER LEFT MM X NN OF X)=(UPPER LEFT MM X NN OF Y)
C (MJ1,MJ2)=ABSOLUTE DIMENSION OF X AND Y IN THE MAIN ROUTINE
C DIMENSION X(MJ1,MJ2),Y(MJ1,MJ2)
DO 10 I=1,MM
DO 16 J=1,NN
Z(I,J)=X(I,J)-Y(I,J)
RETURN
END

```

PAGE 1 C *** SUBROUTINE FOR MATRIX MULTIPLICATION *** PAGE 1 C *** SUBROUTINE TO COMPUTE THE INVERSE AND DETERMINANT OF X **

```

C SUBROUTINE MULTPLY(X,Y,Z,MM,NN,NC,MJ1,MJ2,MJ3)
C MATRIX MULTIPLICATION
C Z=XY
C (UPPER LEFT MM X NC OF Z)=(UPPER LEFT MM X NN OF Y)*(UPPER LEFT
C NN X NC OF Y)
C (MJ1,MJ2)= ABSOLUTE DIMENSION OF X IN THE MAIN ROUTINE
C (MJ2,MJ3)= ABSOLUTE DIMENSION OF Y IN THE MAIN ROUTINE
C (MJ1,MJ3)= ABSOLUTE DIMENSION OF Z IN THE MAIN ROUTINE
C DIMENSION X(MJ1,MJ2),Y(MJ2,MJ3),Z(MJ1,MJ3)
C DO 10 I=1,MM
C SUM=0
C DO 12 K=1,NN
C 12 SUM=SUM+X(I,K)*Y(K,J)
C Z(I,J)=SUM
11 CONTINUE
10 CONTINUE
RETURN
END
C DO 110 I=L,MM
C PIVOTING AT L-TH STAGE
C XMAXP=0,10000E=10
C MAXI=0
C DO 110 I=L,MM
C FOR COMPLEX VERSION NEXT STATEMENT SHOULD BE REPLACED BY
C IF(CDABS(XMAXP),GE,CDABS(X(I,L))) GO TO 110
C XX=X(I,L)
C IF(ABS(XMAXP),GE,ABS(XXX)) GO TO 110
C XMAXP=X(I,L)
C MAXI=I
110 CONTINUE

```

```

PAGE 1 C *** SUBROUTINE FOR TRANSPOSE MULTIPLY ***
C SUBROUTINE TRANSL(Y,Z,MM,NN,NC,MJ1,MJ2,MJ3)
C TRANSPOSE MULTIPLY (LEFT)
C Z=XI*Y
C (UPPER LEFT NN X NC OF Z)=(UPPER LEFT MM X NN OF X)*(UPPER LEFT
C MM X NC OF Y)
C (MJ1,MJ2)= ABSOLUTE DIMENSION OF X IN THE MAIN ROUTINE
C (MJ1,MJ3)= ABSOLUTE DIMENSION OF Y IN THE MAIN ROUTINE
C (MJ2,MJ3)= ABSOLUTE DIMENSION OF Z IN THE MAIN ROUTINE
C DIMENSION X(MJ1,MJ2),Y(MJ1,MJ3),Z(MJ2,MJ3)
C DO 10 I=1,NN
C SUM=0
C DO 12 K=1,MM
C 12 SUM=SUM+X(K,I)*Y(K,J)
C Z(I,J)=SUM
11 CONTINUE
10 CONTINUE
RETURN
END
C DO 11 J=1,MM
C SUM=0
C DO 12 K=1,MM
C 12 SUM=SUM+X(K,I)*Y(K,J)
C Z(I,J)=SUM
11 CONTINUE
10 CONTINUE
RETURN
END
C DO 13 J=1,MM
C 13 X(I,J)=X(I,J)+XC*X(L,J)
12 CONTINUE
10 CONTINUE
C IF(MM.GT.1) GO TO 123
C GO TO 140
C COLUMN INTERCHANGE
C DO 130 J=1,MM
C MMJ=MM-J

```

```

PAGE 2 C *** SUBROUTINE TO COMPUTE THE INVERSE AND DETERMINANT OF X ***
PAGE 1 C ** PLOT SUBROUTINE OF FPE **

JJ=IDS(MMJ)
IF (JJ.EQ.MMJ) GO TO 130
DO 131 I=1,MM
  XC=X(I,JJ)
  X(I,JJ)=X(I,MMJ)
  X(I,MMJ)=XC
131 CONTINUE
130 RETURN
140 END

SUBROUTINE PLOTFP (OFF,TFP,X,GOPP,L,N,IR,L,DELT)
DIMENSION X(21)
CALL BEGIN(9600,1)
CALL ERASE
L1=L+1
CALL SCALE(1,1,1,200.,300.)
CALL AXIS(0.,0.,300.,400.,75.,100.,1,1)
XMAX=X(1)
DO 10 I=2,L1
  YI=X(I)
  IF (X(I).LE.XMAX) GO TO 10
  XMAX=YI
10 CONTINUE
XMAX=XMAX*1.3
YI=XMAX
XS=400./XMAX
XS=15.
YY=X(1)*YS
XX=XI
CALL VECTOR
CALL TPLOT(XX,YY,0,0)
DO 20 I=2,L1
  XX=XX+XS
  YY=YY+YS
  CALL TPLOT(XX,YY,1,0)
20 CONTINUE
CALL ALPHA
CALL TPLOT(0.,470.,0,0)
WRITE(1,1)
CALL TPLOT(95.,0.,430.,0,0)
WRITE(1,4)
CALL TPLOT(115.,405.,0,0)
WRITE(1,5) XMAX
J1=10
X1=100.
CALL TPLOT(X1,10.,0,0)
DO 30 I=1,3
  J1=J1+10
  X1=X1+150.
  CALL TPLOT(X1,-10.,0,0)
  WRITE(1,6) J1
30 CONTINUE
CALL TPLOT(320.,-10.,0,0)
WRITE(1,7)
CALL TPAUSE
RETURN
1 FORMAT(70HF1G. FINAL PREDICTION ERROR FN.1/)
2 FORMAT(10X,5NN *10,3X,5MIR *12,/,10X,5HDELT,*,E0,3,3X,5HIL *,
J12)
4 FORMAT(6HFRE(H))
5 FORMAT(E8.3)
6 FORMAT(T2)
7 FORMAT(1HM/)

```

```

PAGE 2 C ** PLOT SUBROUTINE OF FPE ** PAGE 1 C ** STEP RESPONSE COMPUTING SUBROUTINE FOR OPEN LOOP SYSTEM *
8 FORMAT(/,10X,20HMIN. FPE-----,E10.4//10X,10HGEOMETRICAL MEA
1N//10X,20HOF MIN. FPE-----,E10.4//10X,20HATTAINED AT M-----
214)
END

SUBROUTINE DECOLI
COMMON K,IR,N,DELTA,LLL1,LLL2
DIMENSION G(501,10),A(21,10),C(11),CC(501),NAME1(3),IDDATA(3)
DATA NAME1(1),NAME1(2),NAME1(3)/2HEE,2HEE,2HEE/
DATA IDDATA(1),IDDATA(2),IDDATA(3)/2MID,2MID,2MID/
CALL BEGIN(9600,1)
CALL ERASE
IL=K+IR
LLL1=0
IF(IL.EQ.0) GO TO 999
WRITE(1,1) IL
ACCEPT 2, IDCL
1000 IF(IDCL.EQ.99) GO TO 999
CALL GMOND(22,122,2)
CALL GMOND(21,122,IDDATA)
CALL GMOND(17,122,C(1),C(10))
M=C(1)
K1=K+1
DO 50 IY=1,M
DO 50 II=1,IR
CALL GMOND(17,122,C(1),C(K1))
DO 51 JY=1,N
A(IY,II,JY)=C(JY)
CONTINUE
50 WRITE(1,6)
ACCEPT 2, IPRINT
WRITE(1,4)
ACCEPT 2, IFN
IFN=IFN+1
CALL GMOND(22,121,2)
DO 300 JJ=1,IL
CALL STEPR(C,G,M,IR,JJ,IFN,C)
DO 310 IY=1,IFN
CC(IY)=G(IY,1)
310 CALL GMOND(16,121,CC(1),CC(IFN1))
CONTINUE
320
300 CONTINUE
CALL PLOTSP(G,CC,NAME1,N,IR,IL,DELTA,IFN,IPRINT)
WRITE(1,600)
ACCEPT 2, ICC
IF(ICC.EQ.1) GO TO 1000
WRITE(1,610)
ACCEPT 2, IFREQ
IF(IFREQ.NE.1) GO TO 999
LLL1=0
LLL2=IFN
999 RETURN
1 FORMAT(70H*** STEP RESPONSE CALCULATION WITH OPEN LOOP SYSTEM ***!
2 1//70H MANIPULATED VARIABLE IL=13//70H*** IF UNEXECUTE THIS JOB
3 2//TYPE IN 991//)
2 FORMAT(14)
4 FORMAT(/,20H*** TYPE IN STEP NUMB. FOR CALCULATING STEP RESP. FN.
1 ***!)
5 FORMAT(/,20H*** IL=0. THIS ROUTINE IS NOT NECESSARY ***1//)

```

```

PAGE 2 C ** STEP RESPONSE COMPUTING SUBROUTINE FOR OPEN LOOP SYSTEM ** PAGE 1
6 FORMAT(/,70H** IF PRINT OUT THE RESULT, TYPE IN 1/)
600 FORMAT(70H** IF EXECUTE THE CASE STUDY, TYPE IN 1/)
610 FORMAT(/,70H IF CALCULATE THE FREQ. RESP. FN, TYPE IN 1/)
END
SUBROUTINE PLOTSP(G,CC,NAME1,N,IR,IL,DELT,IFN,IPRINT)
DIMENSION G(501,10),CC(501),NAME1(3)
CALL REGIN(9600,1)
CALL ERASE
IFN1=IFN*1
DO 100 IX=1,IR
CALL GOND(22,'21,2)
CALL GOND(21,'21,NAME1)
DO 110 JJ=1,IL
DO 110 II=1,IR
CALL GOND(17,'21,CC(1),CC(IFN1))
IF(IX,NE,II) GO TO 110
DO 120 IT=1,IFN
G(IT,II)=CC(IT)
CONTINUE
IF(IPRINT,NE,1) GO TO 300
WRITE(16,3) N,IR,IL,DELT,IFN
WRITE(16,4) IX
WRITE(16,5)
DO 200 IT=1,IFN
WRITE(16,6) IT,(G(IT,I),I=1,IL)
CONTINUE
CALL MAXABS(G,IFN,IL,501,10,XMAX)
CALL ETYPE(XMAX,R,EE,ZZ)
IRRR=1.
RI=IRR
KNORM=RI*10.**EE
DO 320 IT=1,IFN
DO 320 I=1,IL
G(IT,J)=G(IT,J)/KNORM
FIN=IFN
STEP=FIN*1.2
IT=DELT*STEP
YL=800.
YL=300.
DX=XL/STEP
CALL ERASE
CALL SCALE(1,1,150.,320.)
CALL AXIS(0.,-300.,600.,600.,200.,150.,1.1)
DO 500 I=1,IL
XX=0.
YY=0.
CALL TPLOT(XX,YY,0,0)
CALL VECTOR
DO 510 IT=1,IFN
XX=XX+DX
YY=YY+DY
CALL TPLOT(XX,YY,1,0)
CONTINUE
CALL ALPHA
WRITE(1,51) II
CONTINUE
500 CALL TPLOT(10.,400.,0,0)
WRITE(1,520) IX
WRITE(1,525) N,IR,DELT,IFN,IL
CALL TPLOT(-150.,300.,0,0)

```



```

PAGE 1 C ** STEP RESP. CALCULATING WITH CLOSED LOOP SYSTEM **
SUBROUTINE DECNV1
COMMON K,IR,N,DELTA,LLLL,LLLLL2
DIMENSION A1(21,10),A(21),C1(501),HS(401,10)
DIMENSION NAME1(3),IDDATA(3)
DATA NAME1(1),NAME1(2),NAME1(3)/2HEE,2HEE,2HEE/
DATA IDDATA(1),IDDATA(2),IDDATA(3)/2HD10,2HD10,2HD10/
CALL BEGIN(9698,1)
CALL ERASE
IF(LLL1.EQ.10) GO TO 999
I=K+IR
IPI=IP+1
WRITE(1,1) IR
ACCEPT 2, IDC
IF(IDC.EQ.99) GO TO 999
CALL GMOND(22,12,2)
CALL GMOND(21,12,10DATA)
M=0
DO 100 IY=1,MH
DO 100 IY=1,IR
CALL GMOND(17,12,A(I),A(10))
DO 100 IY=1,MH
DO 100 IY=1,IR
CALL GMOND(17,12,A(I),A(10))
DO 100 IY=1,IR
DO 100 IY=1,IP
DO 110 JJ=1,JP
A(I,I,1,1)=A(I,J)
CONTINUE
CALL ERASE
ACCEPT 2, IPN
IF(NIPFN+1)
WRITE(1,40)
ACCEPT 2, IPRNT
IPX=IP
CALL GMOND(22,12,1,2)
CALL GMOND(21,12,1,NAME1)
DO 300 IY=1,IR
CALL DECONV(A1,A,C1,HS,IPN,MH,IP,IPX,11)
DO 320 JJ=1,JP
DO 310 IY=1,IPN
C1(IY)=HS(IY,JP)
310 CONTINUE
CALL GMOND(16,12,C1(1),C1(IPN))
320 CONTINUE
IP(IPRNT,ME,1) GO TO 300
WRITE(16,6)
WRITE(16,7) N,K,IR,IL,IPN,DELTA
WRITE(16,8) II
WRITE(16,9)
DO 210 IY=1,IPN
WRITE(16,10) IY,(HS(IY,JP),JJ=1,JPX)
210 CONTINUE
300 CALL PLOTS(HS,C1,NAME1,N,IR,IL,DELTA,IPN,IPX)
WRITE(1,200)
ACCEPT 2, IDC
IF(IDC.EQ.1) GO TO 1000
LLL=0
WRITE(1,220)

```

```

PAGE 2 C ** STEP RESP. CALCULATING WITH CLOSED LOOP SYSTEM **
ACCEPT 2,IFREQ
IF(IFREQ.NE.1) GO TO 999
LLL=20
LLL2=JFN
999 RETURN
1 FORMAT(70H*** STEP RESPONSE CALCULATION WITH CLOSED LOOP SYSTEM **
1*/70H CONTROLLED VARIABLE IR=1,13,170H*** IF UNEXECUTE THIS J
200, TYPE IN 991/)
2 FORMAT(14)
4 FORMAT(17,70H** TYPE IN STEP NUMB, FOR CALCULATING STEP RESP. **1/
1)
40 FORMAT(17,70H** IF PRINT OUT THE CALCULATED RESULT, TYPE IN 11/)
6 FORMAT(161,170H *** STEP RESPONSE CALCULATION WITH CLOSED LOOP SY
1STEP **1/)
7 FORMAT(140,70H ** INPUT DATA1/5X,5MH *18/5X,5HK *18/5X,5H
1IR *18/5X,5MIL *18/5X,5HSTEP,18/5X,5HDELTA,18/5X,5H
6 FORMAT(11,1H,70H** OUTPUT VARIABLE X(1,12,14)/)
9 FORMAT(14,18X,1H,13X,1H2,13X,1H3,13X,1H4,13X,1H5)
10 FORMAT(11,15,5E15,4)
200 FORMAT(17,70H** IF EXECUTE THE CASE STUDY, TYPE IN 11/)
220 FORMAT(17,70H IF CALCULATE THE FREQ, RESP, FN, TYPE IN 11/)
END

```

```

SUBROUTINE DECONV(A1,A,C1,HS,ML,MH,IP,IPX,11)
DIMENSION A1(21,10),A(21),C1(21),SA(501),C1(501),HS(401,10)
IPI=IP+1
MLP1=ML+1
DO 10 IY=1,401
DO 10 IY=1,IPX
DO 10 IY=1,ML
HS(IY)=0
DO 11 IY=1,M0
C1(IY)=A1(IY,11)
DO 20 JJ=1,JPX
IF(IY.EQ.JJ) GO TO 20
DO 22 IY=1,M0
A(IY)=A1(IY,11,JPX)
DO 30 IY=1,MLP1
C1(IY)=0
30 SA(IY)=0
SA(2)=A(1)
DO 40 IY=2,ML
IY=I-1
IF(IY.GT.M0) GO TO 42
C1(IY)=C1(IY,1)
IY=I+1
42 CALL CONVOL(C1,SA,SA2,IPI)
IF(IY.GT.M0) GO TO 41
SA2=A(IY)+SA2
41 SA(IY)=SA2
40 CONTINUE
HSS=0
DO 43 IY=1,MLP1
HSS=HSS+SA(IY)
HS(IY,JPX)=HSS
43 CONTINUE
20 RETURN
END

```

```

SUBROUTINE CONVOL(A,B,SUM,K1)
THIS SUBROUTINE COMPUTES CONVOLUTION
C C(K)=A(K)*B(K)+A(1)*B(K-1)+...+A(K-1)*B(1)+A(K)*B(0)
C K1..K PLUS 1
DIMENSION A(K1),B(K1)
K2=K1+1
SUM=0.0
DO 10 I=1,K1
K1K2=I
SUM=SUM+A(I)*B(K1)
10 RETURN
END

SUBROUTINE PLOTSP(G,CC,NAME1,N,IR,IL,DELT,IFN,IPX)
DIMENSION CC(501),NAME1(3),G(401,10)
CALL BEGIN(9600,1)
CALL ERASE
IFN1=IFN+1
IFN2=IFN+2
CALL GROUND(22,121,2)
CALL GROUND(21,121,NAME1)
DO 100 IX=1,IR
DO 110 JJ=1,IPX
CALL GROUND(17,121,CC(1),CC(IFN2))
DO 120 IT=1,IFN
G(IT,JJ)=CC(IT)
120 CONTINUE
110 IF(IPX.GE.2) GO TO 300
WRITE(1,261)
GO TO 100
300 CALL MAXABS(G,IFN,IPX,401,10,XMAX)
CALL ETYPE(XMAX,R,EE,ZZ)
IRR=IR+1
RT=IRR
XNORM=RI+10,EE
DO 320 IY=1,IFN
DO 320 J=1,IPX
G(IY,J)=G(IY,J)/XNORM
320 FIN=IFN
STEP=FIN+.2
IT=DELT*STEP
XL=800.
YL=300.
DX=XL/STEP
CALL ERASE
CALL SCALE(1,1,150,320)
CALL AXIS(0,300,800,800,200,150,1,1)
DO 500 IY=1,IPX
IF(IY.EQ.IX) GO TO 500
XX=0.
YY=0.
CALL TPLOT(XX,YY,0,0)
CALL VECTOR
DO 510 IY=2,IFN
XX=XX+DX
YY=YY+DY
CALL TPLOT(XX,YY,1,0)
510 CONTINUE
CALL ALPHA
IF(IY.LE.IR) GO TO 400
IJ=IJ+IR
WRITE(1,512) IJ
GO TO 500
400 WRITE(1,511) IY
500 CONTINUE
CALL TPLOT(10,400,0,0)
WRITE(1,520) IX
CALL TPLOT(-150,305,0,0)

```

```

IEESEE
WRITE(1,526) RI,IEE
ISE=0
IF(TI.LE.300.) GO TO 530
TY=TY/60.
ISE=1
530 TY=TY/2.0
CALL TPLOT(-20.,10.,0.)
I20=0
WRITE(1,535) I20
CALL TPLOT(50.,10.,0.0)
WRITE(1,536) I11
CALL TPLOT(750.,10.,0.0)
WRITE(1,536) I1
CALL TPLOT(740.,35.,0.0)
IF(ISE.EQ.0) GO TO 540
WRITE(1,541)
GO TO 550
540 WRITE(1,542)
550 CALL TPAUSE
CALL ERASE
100 CONTINUE
RETURN
2 FORMAT(70H** IF PRINT OUT THE CALCULATED STEP RESP., TYPE IN 11/1)
3 FORMAT(1H,/,70H*** STEP RESPONSE WITH OPEN LOOP SYSTEM DERIVED F
10X A=0 MODEL **/,5X,SHN =,10/,5X,SHIR =,10/,5X,SHIL =,10/,
25X,SHDELTA,EB,S/,5X,SHSTEP,18)
4 FORMAT(/,/,1H,70H** OUTPUT VARIABLE X(1,12,1H))
5 FORMAT(/,1H0.30H** IL=,10X,1H1.10X,1H2.10X,1H3.10X,1H4.10X,1H5/)
6 FORMAT(1H,15,5E15,4)
511 FORMAT(2X(,12,1H))
520 FORMAT(70HFIG. STEP RESPONSE OF X(1,12,70H) FROM X(I)/)
201 FORMAT(/,70H**** IR=1 ****/)
512 FORMAT(2HU(,12,1H))
525 FORMAT(10X,5HN =,14,5X,5HIR =,13,5X,5HDELTA,EB,S/,10X,5HSTEP,1)
526 FORMAT(F4,1,5H*10**12)
535 FORMAT(I1)
530 FORMAT(F6.2)
541 FORMAT(9HTIME(MIN))
542 FORMAT(9HTIME(SEC))
END

SUBROUTINE ETYPE(X,Y,Z,ZZ)
ZZ=1.0
IF(X.EQ.0) GO TO 100
IF(X.LT.0) ZZ=-1.0
Y=ABS(Y)
X=ABS(X)
XLAB=LOG10(XA)
XL=ABS(XL)
IF(XL.GT.15.0) GO TO 101
DO 10 I=1,11
FI=I
IF(XL.AGE.FI) GO TO 10
E1=FI-1.0
GO TO 20
10 CONTINUE
20 IF=FI
IF(XL.GE.0.0) GO TO 30
IF=-IE-1
Y=X/(10.0**IE)
Z=IE
GO TO 110
30 CONTINUE
C ZZ=1 X IS NEGATIVE
C ZZ=1 X IS POSITIVE
C ZZ=0 X IS UNREASONABLE
100 Y=0.
Z=0.
ZZ=1.0
GO TO 110
110 ZZ=0.0
GO TO 110
101 ZZ=0.0
RETURN
END

SUBROUTINE MAXABS(X,M,N,MJ,MJI,XM)
DIMENSION X(MJ,MJI)
XM=0.
DO 10 I=1,M
DO 10 J=1,N
XT=X(I,J)
XABS=ABS(XT)
IF(XABS.LT.XM) GO TO 10
XM=XABS
CONTINUE
10 RETURN
END
    
```

```

PAGE 1 C *** FREQUENCY RESPONSE FUNCTION CALCULATE FROM IMPULSE RESP. SERIE PAGE 1
SUBROUTINE FREQFN
COMMON K,IR,N,DELT,LLL1,LLL2
CALL BEGIN(9680,1)
IL*IR
N*LLL2
1000 CALL ERASE
IF(LLL1.EQ.10) GO TO 10
IF(LLL1.EQ.20) GO TO 20
GO TO 999
10 WRITE(I,1)
CALL FREQOC(M,N,IR,N,DELT,LLL1)
GO TO 999
20 WRITE(I,11)
CALL FREQOC(M,K,IR,N,DELT,LLL1)
999 RETURN
2 FORMAT(I5)
1 FORMAT(70H** FREQUENCY RESPONSE FN. CALCULATION WITH OPEN LOOP SYS
ITEM1//,70H * FOURIER TRANSF. OF IMPULSE RESP. FN//)
11 FORMAT(70H** FREQUENCY RESPONSE FN. CALCULATION WITH CLOSED LOOP S
YSTEM1//,70H * FOURIER TRANSF. OF IMPULSE RESP. FN//)
END
SUBROUTINE FREQOC(N,IP,IR,NN,DELT,IOP1)
DIMENSION X(402),Y(402),FC(402),FS(402),A1(2),NAME1(3),NAME2(3)
DATA NAME1(1),NAME1(2),NAME1(3),NAME2(1),NAME2(2),NAME2(3),ZHEE,2HEE,
DATA NAME2(1),NAME2(2),NAME2(3)/2HEE,2HEE,2HEE/
IP1=IP-1
N1=NN+1
N2=NN+2
IL*IP=IR
A1(1)=0.5
A1(2)=0.25
MLA1=2
WRITE(I,1,1)
ACCEPT 2,IPRINT
CALL OMOND(22,121,2)
CALL OMOND(21,121,NAME1)
CALL OMOND(22,122,2)
CALL OMOND(21,122,NAME2)
IF(IOP1.EQ.10) GO TO 30
IF(I.EQ.J) GO TO 40
CALL IMPLS(X,Y,N)
IF(IPRINT.NE.1) GO TO 35
WRITE(16,3)I,J
DO 36 II=1,N1
WRITE(16,4) II,Y(II),X(II)
36 CONTINUE
Y(1)=0.5*Y(1)
Y(N1)=0.5*Y(N1)
35 CALL FGEROC(Y,N1,FC,N1)
DO 28 II=1,N1
Y(II)=Y(II)
CALL FGERSI(Y,N1,FS,N1)
CALL CROSSP(FC,FS,X,Y,N1,A1,MLA1)
CALL CRSSPE(X,Y,FC,FS,N)
40 CALL OMOND(16,122,FC(1),FC(N2))
10 CONTINUE
1000 CALL AXSEL(NOP1,N,NOPTV)
CALL OMOND(22,122,2)
CALL OMOND(21,122,NAME2)
DO 50 I=1,IY1
DO 50 J=1,IX2
CALL OMOND(17,122,FC(1),FC(N2))
IF(IOP1.EQ.10) GO TO 60
IF(I.EQ.10) GO TO 50
60 CALL PLAUT(NOP1,N,NOPTV,N,DELT,FC,I,J,NN,IOP1)
50 CONTINUE
WRITE(17)
ACCEPT 2,IA
    
```

PAGE 1

PAGE 2

```

IF(I1,EO,1) GO TO 1000
RETURN
1  FORMAT(/,70H** IF PRINT OUT THE IMPULSE RESPONSE FN. TYPE IN 11/)
2  FORMAT(I3)
3  FORMAT(I11,70H *** IMPULSE RESPONSE FN. BY A-R MODEL 1/,30H U(
1,12,7H) TO X(,12,1H)/)
4  FORMAT(1H ,15,2E16.5)
7  FORMAT(/,70H IF PLOT THE FREQ. RESP. FN. AGAIN, TYPE IN 11/)
END

SUBROUTINE PLAUT(NOPTV,NOPTV,LAGH,DT,H,HS,M,MS)
DIMENSION P(402),H(402),HS(10),YS(20)
DIMENSION DUMY(402)
LAGH1=LAGH+1
MX=800.0
MY=600.0
MY0=100.0
MY1=100.0
MXN=MX+MX0
MYN=MY+MY0
MXN=MX
MYN=MY
CALL HCORD(NORTH,LAGH,DT,H,HS,M,MS)
XL=(MX*FLOAT(NOPTV))/FLOAT(H)
CALL VCORD(NOPTV,LAGH,P,VS,MW,ML,MN,II,JJ,DUMY)
COEF=1.0
IF(NOPTV.EQ.0) COEF=1.0
YL=(MY*COEF)/FLOAT(MH)
CALL ERASE
CALL SCALE(1.0,1.0,MX0,MY0)
CALL AXIS(0.0,0.0,MX,MY,XL,YL,1,1)
CALL VECTOR
CALL TPLOT(0,MY,0,0)
CALL TPLOT(MX,0,1,0)
DO 20 I=1,LAGH1
H(I)=H(I)*MX
P(I)=P(I)*MY
20 CONTINUE
CALL TPLOT(0.0,0.0,0)
MARK=0
Y1=P(1)
CALL TPLOT(0.,Y1,0,MARK)
DO 30 I=1,LAGH
I1=I+1
Y1=P(I1)
Y2=P(I)
CALL TPLOT(Y2,Y1,1,MARK)
30 CONTINUE
CALL ALPHA
MYC=MY+40.0
CALL TPLOT(=30.0,MYC,0,0)
IF(IOPT.EQ.20) GO TO 7
WRITE(I,1) JJ,II
1  FORMAT(70HF16.  FREQUENCY RESP. FN. OF OPEN LOOP SYSTEM. X(1,12,1
19H) FROM U(1,12,1H))
GO TO 8
7  WRITE(I,9) JJ,II
9  FORMAT(70HF16.  FREQUENCY RESP. FN. OF CLOSED LOOP SYSTEM. X(1,12
1,20H) FROM X(1,12,1H))
8  M1=M+1
MXD=ABS(XL)-16.0
DO 40 I=1,M1
MXD=MXD+ABS(XL)
MYD=10.0
CALL TPLOT(MXD,MYD,0,0)

```

```

11#MS(I)
IF(NORTH.NE.1) GO TO 98
WRITE(1,2) MS(I)
GO TO 100
98 WRITE(1,4) IH
2 FORMAT(F5.1)
100 IF(NORTH.EQ.1) GO TO 40
MX#MXD=13.0
MY#MYD=15.0
CALL TPLOT(MXX,MYY,0,0)
110#10
WRITE(1,4) I10
4 FORMAT(I2)
40 CONTINUE
CALL TPLOT(200.,-50.,0,0)
WRITE(1,3)
3 FORMAT(78#FREQUENCY (CYCLE/SEC))
IF(NORTH.NE.1) GO TO 50
CALL TPLOT(750.,-50.,0,0)
WRITE(1,5) MS
5 FORMAT(78H(=10**1,12,1H))
50 CALL TPLOT(-50.,0.,0,0)
MYD=15.0
DO 60 I=1,MN
MYD#MYD=ABS(YL)
CALL TPLOT(-97.,MYD,0,0)
IPI=1
IVS=VS(IPI)
IF(NOPTV.LE.0) GO TO 110
WRITE(1,2) VS(IPI)
GO TO 120
110 WRITE(1,8) IVS
6 FORMAT(I4)
120 IF(NOPTV) 60,70,60
70 MX#=57.0
MY#MYD=14.0
CALL TPLOT(MXX,MYY,0,0)
WRITE(1,4) I10
60 CONTINUE
81 MYC#MY=20.0
MCC=130.
CALL TPLOT(MXC,MYC,0,0)
WRITE(1,5) MN
GO TO 80
82 CALL TPLOT(-98.0,250.0,0,0)
CALL VECTOR
CALL WRITEV(2.0,90.0,0.0,0.0,0.0)
80 CALL ALPHA
MY#MY=10.0
CALL TPLOT(532.0,MYY,0,0)
WRITE(1,80)
MY#MY=20.0
CALL TPLOT(566.0,MYY,0,0)
WRITE(1,83) MN
MY#MY=20.0
CALL TPLOT(560.0,MYY,0,0)
WRITE(1,84) LAGH
SUBROUTINE MAXMIN(X,N,XMAX,XMIN)
DIMENSION X(402)
XMAX=X(1)
XMIN=X(1)
DO 1 I=1,N
X=X(I)
IF(X.LE.XMAX) GO TO 1
XMAX=X(I)
1 CONTINUE
XMIN=X(1)
DO 2 I=1,N
X=X(I)
IF(X.GE.XMIN) GO TO 2
XMIN=X(I)
2 CONTINUE
RETURN
END

```

PAGE 1

PAGE 2

```

DO 40 I=1,MM1
VX(I)=(Z+MINX+1)*1.0
40 CONTINUE
FMM=MM*10
FMIN=MINX*10
DO 50 I=1,LAGH1
XLX(I)
X(I)=(CDB*ALOG10(XL)-FMIN)/FMM
ML=0
50 CONTINUE
MN=0
60 CONTINUE
RETURN
END

SUBROUTINE HCORD(NOPT,LACH,DT,H,MS,MH,MS)
DIMENSION H(402),HS(10)
FLAG=LACH
DO 10 I=1,LACH
FI=1
H(I)=FI/(2.0*DT*FLAG)
10 CONTINUE
IF(NOPT.LT.0) GO TO 30
ALPH(LACH)
CALL ETYP(A1,COEF,ZE,IZE)
MSIZE
IF(IZE.EQ.0) GO TO 200
H(3)=COEF
HS(2)=COEF/2.0
HS(1)=0.
MH=2
DO 20 I=1,LACH
H(I)=H(I)/H(LACH)
20 CONTINUE
30 XM=H(LACH)
MAXH=XL*1.0
MH=H(1)
XL=ALOG10(XM)
MINH=XL-1.0
MH=MAX-MINH
MH1=MH+1
DO 40 I=1,MH1
MS(I)=I-MINH+1
40 CONTINUE
DO 50 I=1,LACH
XL=H(I)
H(I)=(ALOG10(XL)-FMIN)/FMM)/FLOAT(MH)
50 CONTINUE
MH=MH
MS=0
GO TO 60
200 WRITE(16,100)
60 RETURN
100 FORMAT(1H ,70HITS UNREASONABLE JOB. /)
END
    
```

```

PAGE 1                                PAGE 1
SUBROUTINE ETYPE(X,Y,Z,MZ)
M2=1
IF(X.EQ.0.) GO TO 100
IF(X.LT.0.) ZZ=1.0
X=ABS(X)
X=ALOG10(X)
X=ABS(X)
IF(XLAGY,15.0) GO TO 101
DO 10 I=1,11
PI=1
IF(XLAG,GE,FI) GO TO 10
GO TO 20
10 CONTINUE
20 IF=1
IF(XLAG,GE,0.0) GO TO 30
IF=1.0
30 Y=Z/(10.0+IE)
Z=IE
GO TO 110
C Z2=1 X IS NEGATIVE
C Z2=1 X IS POSITIVE
C Z2=0 X IS UNREASONABLE
100 Y=0.
Z=0.
M2=1
GO TO 110
101 M2=0
110 CONTINUE
RETURN
END

SUBROUTINE SGLPAC(ARC,PH,LAGH1)
DIMENSION ARC(LAGH1),PH(LAGH1)
PI=3.1415926536
PI2=PI*2.0
PH(1)=ARC(1)
DO 10 I=2,LAGH1
DK=ARC(I)-ARC(I-1)
IF(DK.GT.PI) GO TO 11
IF(DK.LT.-PI) GO TO 12
PH(I)=PH(I-1)+DK
GO TO 10
11 PH(I)=PH(I-1)+DK-PI2
GO TO 10
12 PH(I)=PH(I-1)+DK+PI2
10 CONTINUE
RETURN
END

SUBROUTINE SPHASE(C,S,PH,LAGH1)
DIMENSION C(LAGH1),S(LAGH1),PH(LAGH1),ARC(501)
CALL SGLARC(C,S,ARC,LAGH1)
CALL SGLPAC(ARC,PH,LAGH1)
RETURN
END

SUBROUTINE SGLPAC(ARC,PH,LAGH1)
DIMENSION ARC(LAGH1),PH(LAGH1)
PI=3.1415926536
PI2=PI*2.0
PH(1)=ARC(1)
DO 10 I=2,LAGH1
DK=ARC(I)-ARC(I-1)
IF(DK.GT.PI) GO TO 11
IF(DK.LT.-PI) GO TO 12
PH(I)=PH(I-1)+DK
GO TO 10
11 PH(I)=PH(I-1)+DK-PI2
GO TO 10
12 PH(I)=PH(I-1)+DK+PI2
10 CONTINUE
RETURN
END

1 FORMAT(70H** AXIS SCALING CONDITION **//70H X AXIS 1...LINEAR
2...LOG//70H Y AXIS 1...LINEAR 2...LOG 3...081//70Hs TY
20E IN THE SCALING NO.1/)
2 FORMAT(2I3)
END
    
```



```

SUBROUTINE SCLARC(C,S,ARC,LAGH1)
DIMENSION C(LAGH1),S(LAGH1),ARC(LAGH1)
PI=3.1415926536
DO 10 I=1,LAGH1
  IF(C(I)) 11,12,13
  11 IF(S(I)) 14,15,16
  12 IF(S(I)) 17,18,19
  13 ARC(I)=ATAN(S(I)/C(I))
  60 TO 10
  14 ARC(I)=ATAN(S(I)/C(I))-PI
  60 TO 10
  15 ARC(I)=PI
  60 TO 10
  16 ARC(I)=ATAN(S(I)/C(I))+PI
  60 TO 10
  17 ARC(I)=-PI/2.0
  60 TO 10
  18 ARC(I)=0.
  60 TO 10
  19 ARC(I)=PI/2.0
  60 TO 10
  20 CONTINUE
  RETURN
  END

SUBROUTINE ECORCO(FC,LAGH1,FC1,LAGSHF,LA1)
DIMENSION FC(LAGH1),FC1(LAGSHF)
LAGH2=LAGH1+1
LA=LA1-1
DO 100 I=1,LAGH1
  I1=LAGH2-I
  I2=I+LA
  100 FC1(I2)=FC(I1)
  LA2=LAGH1+LA
  DO 110 I=1,LA
    I1=LA1-I
    I2=LA1+I
    I3=LA2-I
    I4=LA2+I
    FC1(I1)=FC1(I2)
    FC1(I3)=FC1(I4)
  110 FC1(I4)=FC1(I3)
  RETURN
  END

SUBROUTINE ECORSI(FS,LAGH1,FS1,LAGSHF,LA1)
DIMENSION FS(LAGH1),FS1(LAGSHF)
LAGH2=LAGH1+1
LA=LA1-1
DO 100 I=1,LAGH1
  I1=LAGH2-I
  I2=I+LA
  100 FS1(I2)=FS(I1)
  LA2=LAGH1+LA
  DO 110 I=1,LA
    I1=LA1-I
    I2=LA1+I
    I3=LA2-I
    I4=LA2+I
    FS1(I1)=FS1(I2)
    FS1(I3)=FS1(I4)
  110 FS1(I4)=FS1(I3)
  RETURN
  END

SUBROUTINE CROSSP(FC,FS,P1,P2,LAGH1,A,LA1)
DIMENSION FC(LAGH1),FS(LAGH1),P1(LAGH1),P2(LAGH1)
DIMENSION A(LA1)
DIMENSION FC1(521),FS1(521)
LA=LA1-1
LAGSHF=LAGH1+2*LA
CALL ECORCO(FC,LAGH1,FC1,LAGSHF,LA1)
CALL SHOSPE(FC1,LAGSHF,A,LA1,P1,LAGH1)
CALL ECORSI(FS,LAGH1,FS1,LAGSHF,LA1)
CALL SHOSPE(FS1,LAGSHF,A,LA1,P2,LAGH1)
RETURN
END

SUBROUTINE IMPLS(X,Y,N)
DIMENSION X(4021),Y(402)
IMPULSE RESPONSE FN, DERIVED FROM STEP RESP. FN.
Y(1)=0.
DO 10 I=2,N
  Y(I)=Y(I-1)
  I=I+1
  Y(I)=X(I)*Y(I-1)
  RETURN
  END

SUBROUTINE ECORSI(FS,LAGH1,FS1,LAGSHF,LA1)
DIMENSION FS(LAGH1),FS1(LAGSHF)
LAGH2=LAGH1+1
LA=LA1-1
DO 100 I=1,LAGH1
  I1=LAGH2-I
  I2=I+LA
  100 FS1(I2)=FS(I1)
  LA2=LAGH1+LA
  DO 110 I=1,LA
    I1=LA1-I
    I2=LA1+I
    I3=LA2-I
    I4=LA2+I
    FS1(I1)=FS1(I2)
    FS1(I3)=FS1(I4)
  110 FS1(I4)=FS1(I3)
  RETURN
  END

SUBROUTINE SMOSEX(LAGSHF,A,LA1,Z,LAGH1)
DIMENSION X(LAGSHF),A(LA1),Z(LAGH1)
LA=LA1-1
DO 10 I=1,LAGH1
  I0=I+LA
  SUM1=0.0
  DO 11 I=1,LA
    J1=I0-I
    J2=I0+J
    11 SUM1=SUM1+A(J0-I)*X(J1)+X(J2)
  10 Z(I)=A(I)*X(I0)+SUM1
  RETURN
  END

```

```

SUBROUTINE PGER3I(G,LGPI,FS,LF1)
DIMENSION G(LGPI),FS(LF1)
LG=LGP1+1
LF=LF1+1
IF(LGPI.LE.1) GO TO 110
LG3=LGP1/2
LG4=LGP1/2
DO 100 I=1,LG4
  I2=LG3-I
  I3=LG3+I
  Y3(I)=G(I)*G(I2)
  Y3(I3)=G(I)*G(I2)
100 G(I2)=Y
110 PI=3.1415926536
ALF=LF
T=PI/ALF
AK=K+1
TK=T*AK
CK=COS(TK)
SK=SIN(TK)
CR2=CK*CK
UM2=0.0
UM1=0.0
IF(LG.EQ.0) GO TO 12
DO 11 I=1,LG
  UM0=CR2*UM1+UM2*G(I)
  UM2=UM1
11 UM1=UM0
12 FC(K)=CK*UM1+UM2*G(LGPI)
10 CONTINUE
RETURN
END

```

```

SUBROUTINE STATES
COMMON X,IR,N,DELT,LLL1,LLL2
DIMENSION NAME1(3),NAME2(3),NAME3(3),C(20),A(21,10,10),Z(3001),
1 X(12,10),EX(12,8),Z(225),IDDATA(3)
DATA NAME1(1),NAME1(2),NAME1(3)/2HEE,2HEE,2HEE/
DATA NAME2(1),NAME2(2),NAME2(3)/2HTS,2HDA,2HTA/
DATA NAME3(1),NAME3(2),NAME3(3)/2HFF,2HFF,2HFF/
DATA IDDATA(1),IDDATA(2),IDDATA(3)/2HID,2HDA,2HTA/
CALL BEGIN(9000,1)
CALL ERASE
LLL2=0
NLEN=1
IL=0;IR=0
IF(IL,LE,0) GO TO 999
WRITE(1,1)
ACCEPT 2,ISEL
IF(ISEL.EQ.99) GO TO 999
CALL GMOND(22,122,2)
1000 CALL GMOND(21,122,10DATA)
CALL GMOND(17,122,C(1),C(10))
MFC(1)
MIR=1
MIR=1
DO 50 I=1,IR
CALL GMOND(17,122,C(1),C(K1))
DO 51 JJ=1,K
50 CONTINUE
A(X(I),JJ)=C(JJ)
1001 WRITE(1,5) N
ACCEPT 6,ISTAT,ISTEP
ISLEN=ISTAT
IF(ISTAT.LE.20) GO TO 1001
IF(ISTEP.EQ.0) GO TO 1001
IF(I3,LY,101) GO TO 1001
IS=ISTAT+ISTEP*100+2
IF(I33,GT,N) GO TO 1001
INITIAL SET
C
CALL GMOND(22,123,2)
CALL GMOND(21,123,NAME2)
DO 200 I=1,K
CALL GMOND(17,123,Z(1),Z(N1))
DO 201 IM=1,M
IF(ISTAT=IM)
X(IM,1)=Z(IY)
200 EX(IM,1)=Z(IY)
CONTINUE
CALL GMOND(22,121,2)
CALL GMOND(21,121,NAME1)
DO 100 I=1,K
CALL GMOND(22,123,2)
CALL GMOND(21,123,NAME2)
DO 101 I=1,K
CALL GMOND(17,123,Z(1),Z(N1))
J1=ISTAT+(IY-1)*100+1
J2=J1+99

```

```

999 RETURN
1  FORMAT(//,70H** STATE ESTIMATION BY AER MODEL//,70H IF UNEXECU
   1TE THE JOB,TYPE IN 991//,70H IF ESTIMATE THE INNOVATION PROCESS,
   2 TYPE IN NEGATIVE NO.,1//,70H IF ESTIMATE THE OUTPUT STATE,TYPE
   3IN POSITIVE NO.,1//)
2  FORMAT(I5)
5  FORMAT(//,70H** TYPE IN **START NO.,BLOCK NO. **//,70H NUMB. 0
   6 IF DATA=1,15//)
6  FORMAT(2I5)
7  FORMAT(//,70HIF CALCULATE THE INNOVATION CORRELATION,TYPE IN 1//)
   END

10 WRITE(1,1) N,DELTA
   GO TO 30
20 WRITE(1,2) N,DELTA
30 X0=140.
   V0=600.
   FN=N
   DX=800./FN
   DD=DX*50.
   ISN1=ISN1
   ISN1=ISN1
   DO 100 I=1,IR
   CALL GOMND(17,123,Z(1),Z(ISN1))
   YN=0.
   DO 40 J=1,ISN
   X8=ABS(X)
   IF(XABS.LT.XM) GO TO 40
   YN=YABS
   CONTINUE
   40 CALL SCALE(1,1,X0,Y0)
   CALL AXIS(0,1,100,500,200,DDX,50,1,1,1)
   CALL TPLOT(0,0,0,0)
   CALL VECTOR
   XS=0.
   DO 200 IT=1,N
   IV=IT*IS1
   XS=X8*DX
   VS=100.*Z(IV)/XM
   CALL TPLOT(XS,YS,1,0)
   CONTINUE
   200 CALL GOMND(17,124,Z(1),Z(N1))
   CALL TPLOT(0,0,0,0)
   CALL POINT
   XS=0.
   DO 300 IT=1,N
   VS=X8*DX
   YS=100.*Z(IT)/XM
   CALL TPLOT(XS,YS,1,0)
   CONTINUE
   300 CALL ALPHA
   CALL TPLOT(=125,110,0,0)
   IAN=0
   IAN=0
   CALL TPLOT(=35,15,0,0)
   WRITE(1,7) IAN
   IF(I.EQ.IR) GO TO 400
   IF(I.EQ.3) GO TO 400

```

```

PAGE 2                                PAGE 1    C ** RESIDUAL CORRELATION

IF(1,EG,6) GO TO 400
Y0=Y0+200.
CALL SCALE(1,1,1,10,10)
GO TO 100
400  CALL VECTOR
    VCR=130.
    CALL TPLOT(0, YC, 0, 0)
    CALL TPLOT(00, YC1, 0)
    YC1=YC*15.0
    CALL TPLOT(00, YC1, 0, 0)
    CALL ALPHA
    WRITE(1,4)
    CALL TPAUSE
    Y0=000.
    CALL ERASE
    CONTINUE
100  RETURN
1   FORMAT(70H FIG. ESTIMATED OUTPUT BY A-R MODEL WHEN OBSERVED INP
2   IUTS/OUTPUTS//,40X,5H** N=,I4,7H DELT*,E6,3,3H **)
3   IUTS//,40X,5H** N=,I4,7H DELT*,E6,3,3H **)
4   FORMAT(70H FIG. ESTIMATED OUTPUT BY A-R MODEL WHEN OBSERVED INP
5   IUTS//,40X,5H** N=,I4,7H DELT*,E6,3,3H **)
6   FORMAT(70H EXP. VALUE .....ESTIMATED VALUE)
7   FORMAT(12)
END

SUBROUTINE RESCOR
COMMON K,IR,N,DELTA,LLLL1,LLLL2
DIMENSION X(400),Y(1000)
DIMENSION NAME1(3),NAME2(3),NAME3(3)
DATA NAME1(1),NAME1(2),NAME1(3),NAME2(3),NAME3(3)
DATA NAME2(1),NAME2(2),NAME2(3),NAME3(1),NAME3(2),NAME3(3)
DATA NAME3(1),NAME3(2),NAME3(3),NAME3(3)/2HAA,2HAA,2HAA,2HAA/
DATA NAME3(1),NAME3(2),NAME3(3),NAME3(3)/2HEE,2HEE,2HEE,2HEE/
CALL BEGIN(900,1)
CALL ERASE
IF(LLLL2,EG,0) GO TO 999
MULL1
ISTAT=LLLL2
N1=N+1
WRITE(1,1)
ACCEPT 2,1DEF
IF(IDEF.NE.1) GO TO 100
CALL GMOND(22,121,1)
CALL GMOND(23,121,NAME1,3,88)
CALL GMOND(24,121,NAME1,66,1400)
100  CALL GMOND(22,121,1)
    CALL GMOND(21,121,NAME1)
    CALL GMOND(22,122,2)
    CALL GMOND(21,122,NAME2)
    CALL GMOND(22,123,2)
    CALL GMOND(21,123,NAME3)
    M1=M+1
    DO 10 I=1,IR
    CALL GMOND(17,122,X(1),X(M1))
    CALL GMOND(16,121,X(1),X(M1))
10   CONTINUE
    DO 20 I=1,IR
    CALL GMOND(17,123,X(1),X(N1))
    DO 30 J=1,M
    J=J+1STAT
    Y(J)=X(IJ)
30   CONTINUE
    IT=I+IR
    IF(IT=9T,15) GO TO 21
    CALL GMOND(16,121,Y(1),Y(M1))
20   CONTINUE
    IRR=IR+IR
    GO TO 22
21   IRR=15
22   WRITE(1,3)
    WRITE(1,5)
    WRITE(1,4) M,IRR,DELTA
    CALL TPAUSE
    CALL ERASE
999  RETURN
1   FORMAT(70H** RESIDUAL CORRELATION OF STATE ESTIMATION//,70H IF 0
2   IFFINE THE FILE AAAAA, TYPE IN 1//)
3   FORMAT(14)
4   FORMAT(///,70H*****//)
5   FORMAT(70H* CALL THE PROGRAM ...MLCOSP...//,70H FILE NAME. AAAAA
6   I//,70H N =,I5,30H K =,I5,30H DELT*,E10,3//)
7   FORMAT(//,70H* RESIDUAL CORRELATION //,70H 1ST IR**RESIDUE Y, 3,
8   1DATA//,70H 2ND IR**OUTPUT DATA//)
END

```

APPENDIX - 2 DYSAC2 CODE LIST

PAGE 1

```

SUBROUTINE FPEID2
COMMON K,IR,N,DELT,ODD(10)
DIMENSION A1(21,10,10),B1(21,10,10),PP(21),PARAMT(10),CI(10),
1 SD(10,10),SE(10,10),SF(10,10),OSD(10,10),XSD(10,10),D(10,10),
2 F(10,10),ZI(10,10),XSF(10,10),PPF(21),OSDN(10,10),OPPM(21)
DIMENSION NAME1(3),NAME2(3)
EQUIVALENCE (ZI(1,1),OSDN(1,1))
DATA NAME1(1),NAME1(2),NAME1(3),NAME2(1),NAME2(2),NAME2(3)/2HCO,2HDA,2HYA/
DATA NAME2(1),NAME2(2),NAME2(3)/2HDD,2HDD,2HDD/
CALL BEGIN(9000,1)
CALL ERASE
WRITE(1,11)
WRITE(1,12)
READ(2,1) N,K,IR,DELT
L=20
NJ=10
NJB=1
L1=L+1
IL=K+IR
IP=K
IP1=IP+1
IR1=IR+1
IP(1,10,0) GO TO 699
WRITE(1,14)
ACCEPT 12, ISEL
IF(ISEL.EQ.99) GO TO 699
CALL ERASE
C INITIAL SD,SP,SE COMPUTATION
C BATH STEP COMPUTATION
13 CALL OMOND(21,121,2)
DO 330 I1=1,IP
CALL OMOND(17,121,PPF(I1),PPF(IP1))
IF(I1.LE.IR) GO TO 330
DO 331 JJ=IR1,IP
IN=I1+IR
JN=JJ+IR
DO 332 I1=1,IP
80(I1,JN)=PPF(JJ)
330 CONTINUE
DO 332 I1=1,IP
CALL OMOND(17,121,PPF(I1),PPF(IP1))
IF(I1.LE.IR) GO TO 332
DO 333 JJ=IR1,IP
IN=I1+IR
JN=JJ+IR
8K(I1,JN)=PPF(JJ)
333 CONTINUE
K=K+K
IR=IR+IR
IL=IL+IL
K=K+IL
IP=K
IR=K
IP1=IP+1
DO 334 I1=1,IP
    
```

PAGE 1 C ** DYNAMIC SYSTEM ANALYSIS CODE BASED ON IO, TECHNIQ, **

```

C ** CODED BY R. OGUMA 1976 2*8
C ** FILE NAME ...DYSAC2...
COMMON K,IR,N,DELT,ODD(10)
CALL OVRLAY(OMHPEID2,127250)
CALL OVRLAY(OMHINPT2,125120)
CALL OVRLAY(OMHPLT2,14447)
CALL OVRLAY(OMHFBM2,14533)
CALL OVRLAY(OMHFRFP2,130151)
CALL OVRLAY(OMHPLT2,14447)
CALL OVRLAY(OMHRASPE2,120563)
CALL OVRLAY(OMHSPLOT2,124217)
CALL OVRLAY(OMHFRFE2,134725)
CALL OVRLAY(OMHPLT2,14447)
STOP
END
    
```

PAGE 1

```

SUBROUTINE OVRLAY(NAME,LOC)
CALL OMOND(22,123,1)
CALL OMOND(21,123,NAME)
CALL OMOND(20)
LA LOC
STA 100
L 100,2
RETURN
ADR 0
END
    
```

```

DO 334 JJ=1,IP
SF(I1,J1)SD(I1,J1)
XSD(I1,J1)SD(I1,J1)
XSF(I1,J1)SF(I1,J1)
334 CONTINUE
DO 335 II=1,IP
C(I11)SD(I11,11)
IFPEC=8
MS99
OFPEC,ORFPEC,COMPUTATION
CALL SPPEC(SD,N,IP,IR,MS,OFPEC,ORFPEC,ORFPEC,MJ)
OFPEC,ORFPEC,PRINT OUT
OPFH(I1)ORFPEC
ORFPM(I1)ORFPEC
C ITERATION N=1 TO L
DO 400 M=1,L
C INVERSE OF SD SP COMPUTATION
CALL INVDET(XSD,SDET,IP,MJ)
CALL INVDET(XSF,SDET,IP,MJ)
D,E,SD,SP,COMPUTATION
CALL MULTIPLY(SE,XSF,D,IP,IP,IP,MJ,MJ,MJ)
CALL TRANDL(SE,XSD,E,IP,IP,IP,MJ,MJ,MJ)
CALL TRANDR(D,SE,ZI,IP,IP,IP,MJ,MJ,MJ)
CALL SUBTAL(SD,ZI,IP,IP,MJ,MJ)
CALL MULTIPLY(E,SE,ZI,IP,IP,IP,MJ,MJ,MJ)
CALL SUBTAL(SF,ZI,IP,IP,MJ,MJ)
MS99
DO 410 I1=1,IP
DO 410 J1=1,IP
XSD(I1,J1)SD(I1,J1)
XSF(I1,J1)SF(I1,J1)
FPEC,RPPEC,COMPUTATION
CALL SPPEC(SD,N,IP,IR,MS,FPEC,RPPEC,ORFPEC,MJ)
FPEC,RPPEC,PRINT OUT
OPFH(M1)RPPEC
ORFPM(M1)RPPEC
C FORWARD AND BACKWARD PREDICTOR COMPUTATION
CALL COEFAR(A1,E1,D,E,M3,IP,MJ,MJ)
C MIN, FPEC MIN, RPPEC COMPUTATION
CALL QMOND(22,122,2)
IF(OFPEC,LE,RPPEC) GO TO 448
CALL QMOND(18,122,NAME2)
OFPEC,RPPEC
ORFPEC,RPPEC
IFPEC=8
DO 450 I1=1,IR
DO 450 J1=1,IR
OSD(I1,J1)OSD(I1,J1)
OISD(I1,J1)OSD(I1,J1)
OSD(I1,J1)OSD(I1,J1)/SORI(OIJJ)
GO TO 450
OSD(I1,J1)E1,E
CONTINUE
WRITE(I10,8)
DO 453 I1=1,IR
WRITE(I16,9) I1,(OSD(I1,J1),J1=1,JI)
CONTINUE
IF(IFPEC,LE,8) GO TO 699
GO TO 459
OSD(I11)OSD(I1,11)
CALL QMOND(22,122,2)
CALL QMOND(21,122,NAME2)
CALL QMOND(17,122,PARAMT(1),PARAMT(10))
IPI=IP+1
M=PARAMT(1)
DO 460 I1=1,M
DO 460 J1=1,IR
PP(JJ)=AI(I1,J1)
CALL QMOND(18,122,PP(1),PP(1),PP(1),PP(1))

```

```

PAGE 4
451 DO 461 JJ=1,JP
452 A(I,II,III,IIII)=PP(JJ)
460 CONTINUE
WRITE(10,20) N
DO 462 II=1,M
WRITE(10,21) II
DO 462 III=1,IR
WRITE(10,9) II,(A(II,II,II,II),JJ=1,JP)
CONTINUE
K=K+1
IRERR
IL=ILL
RETURN
11 FORMAT(70H** DYNAMIC SYSTEM ANALYSIS CODE FROM A-R MODEL**I//,70H*
1* CALCULATION OF DYSAC1 MUST BE TERMINATED BEFOREHANDI//,70H DAT
2A FILE 888, CCC, DDD, EEE, FFF, MUST BE DEFINED//)
2 FORMAT(/,70H** TYPE IN INPUT DATA **N,X,IG,DELT,/)
3 FORMAT(100//,70H ** FINAL PREDICTION ERROR **I//,1H ,4X,1NH,6X,
17HPREC(N),7X,BHPREC(N))
4 FORMAT(1H ,15,2X,2E14,5)
5 FORMAT(/,1H ,6X,15HMINIMUM FREQ=3X,14HMINIMUM REPEC=,10X,E12,5,2
1X,E12,5//,7X,40HATTAINED AT #,15)
6 FORMAT(100,6X,70HGEOMETRICAL MEAN OF MIN FREQ(1)I//,8X,E12,4)
7 FORMAT(101,70H**ESTIMATED NOISE 2-ND MOMENT**I//10X,10HOSD(II,II))
8 FORMAT(/,70H **NORMALIZED NOISE COVARIANCE OSDN(II,II))
9 FORMAT(100,15,3X,5E12,4//,8X,5E12,4)
10 FORMAT(/,70H **ESTIMATED SYSTEM PARAMETERS**I//10X,11H(A(II,
18(1)))
11 FORMAT(315,F10.5)
12 FORMAT(15)
14 FORMAT(/,70H** A-R MODEL FITTING TO INPUT DATA **I//,70H IF UNEX
15 ECUTE THE JOB, TYPE IN 99//)
15 FORMAT(101//,70H ** A-R MODEL FITTING FOR I, S, DATA **I//30H **
1 INPUT DATA **I//)
16 FORMAT(1H ,3X,5HN =,15//,4X,5HK =,15//,4X,5HR =,15//,4X,5HL =
1,15//,4X,5HOLT*,E12,4//)
20 FORMAT(1H ,70H ESTIMATED ORDER M=,1,13)
21 FORMAT(100,20HIT=,1,13)
END

```

PAGE 1

PAGE 4

```

SUBROUTINE CONFAC(A1,B1,D,E,MS,K,MJ0,MJ)
THIS SUBROUTINE COMPUTES FORWARD(A) AND BACKWARD(B) PREDICTOR
COEFFICIENTS
DIMENSION A1(MJ0,MJ,MJ),B1(MJ0,MJ,MJ),D(CHJ,MJ),E(MJ,MJ),A(10,10),
1 B(10,10),Z(10,10),Z2(10,10)
IF(MS.EQ.1) GO TO 40
MS1=MS-1
DO 10 I=1,MSM1
M1=MS-I
DO 20 II=1,K
DO 20 JJ=1,K
A(II,II)=A1(II,II,II)
CALL MULTIPLY(D,B,Z1,K,K,K,MJ,MJ,MJ)
CALL MULTIPLY(E,A,Z2,K,K,K,MJ,MJ,MJ)
CALL SUBTAL(A,Z1,K,K,MJ,MJ)
CALL SUBTAL(B,Z2,K,K,MJ,MJ)
DO 21 III=1,K
A(II,II,III)=A1(II,III,III)
B(III,II,II)=B(II,II,III)
DO 30 III=1,K
DO 30 JJ=1,K
A(MS,II,III)=D(III,III)
B(III,II,III)=E(III,III)
RETURN
END

```

PAGE 1

PAGE 1

```

SUBROUTINE NEMSE(A1,SE,MS,K,MJ0,MJ,KK,IR)
SE COMPUTATION
DIMENSION A1(MJ0,MJ,MJ),SE(MJ,MJ),A(10,10),R(10,10),Z(10,10)
DO 10 II=1,K
DO 10 JJ=1,K
Z(II,II)=0.0
MSP2=MS*2
DO 11 I=1,MS
M1=MSP2-I
DO 12 II=1,K
DO 12 JJ=1,K
A(II,II)=A1(II,II,II)
CALL REDATA(R,M1,KK,IR)
CALL MULTIPLY(A,R,SE,K,K,K,MJ,MJ,MJ)
CALL MATADL(Z,SE,K,K,MJ,MJ)
CALL REDATA(R,MSP2,KK,IR)
CALL SUBTAC(R,Z,SE,K,K,MJ,MJ)
RETURN
END

```


PAGE 1

PAGE 1

```

SUBROUTINE REDATA(R,M7,K,IR)
DIMENSION R(10,10),RR(11)
DIMENSION NAME(13)
DATA NAME(1),NAME(2),NAME(3),2HCO,2HDA,2HTA/
KI,K*1
IR1=IR*1
CALL GMOND(22,121,2)
CALL GMOND(21,121,NAME1)
DO 10 I=1,MY
DO 10 J=1,K
CALL GMOND(17,121,RR(K1))
IF(I1,LE,IR) GO TO 10
DO 11 J=IR1,K
IN=J*IR
JM=J*IR
R(IM,JM)=RR(IJ)
CONTINUE
RETURN
END
11
10
16
15
10
18
17
END
    
```

PAGE 1

PAGE 1

```

SUBROUTINE SFPEC(SD,N,K,IR,MS,Z,RZ,OOZ,MJ)
SPEC COMPUTATION
DIMENSION SD(MJ,MJ),SD1(10,10)
AN=SD
K=K*MS
ANP=1*KK
ANM=1*KK
APZ=AP/AN
APW=AP*IR
DO 9 I=1,IR
DO 9 J=1,IR
SD(I,J)=SD(I,J)
CALL SUBDET(SD1,SDRM,IR,MJ)
IF(MS,NE,0) GO TO 10
OOZ=1.0/Z
RZ=Z*OOZ
RETURN
END
    
```

PAGE 1

PAGE 1

```

SUBROUTINE SUBDET(X,XOETM,MM,MJ)
THIS SUBROUTINE COMPUTES THE DETERMINANT OF UPPER LEFT MM X MM
OF X, FOR GENERAL USE STATEMENTS 20-21 SHOULD BE RESTORED.
X, ORIGINAL MATRIX
XOETM, DETERMINANT OF UPPER LEFT MM X MM OF X
MJ, ABSOLUTE DIMENSION OF X IN THE MAIN ROUTINE
DIMENSION X(MJ,MJ)
XOETM=1.0
IF(MM,EQ,1) GO TO 10
MM1=MM-1
DO 10 I=1,MM1
XC=1.0/X(I,I)
XOETM=XOETM*X(I,I)
I=I+1
DO 10 J=I+1,MM
XC=X(J,I)*X(I,K)-X(I,I)*X(J,K)
XOETM=XOETM*XC
DO 10 K=I+1,MM
X(J,K)=X(J,K)-X(I,I)*X(J,K)
CONTINUE
CONTINUE
XOETM=XOETM*X(MM,MM)
RETURN
END
    
```

PAGE 1

PAGE 1

```

SUBROUTINE SUBTAC(X,Y,Z,MM,NN,MJ1,MJ2)
MATRIX SUBTRACTION
Z=X-Y
(UPPER LEFT MM X NN OF Z)=(UPPER LEFT MM X NN OF X)-(UPPER LEFT
MM X NN OF Y)
(MJ1,MJ2) ABSOLUTE DIMENSION OF X, Y AND Z IN THE MAIN ROUTINE
(MJ1,MJ2) ABSOLUTE DIMENSION OF X, Y AND Z IN THE MAIN ROUTINE
DIMENSION X(MJ1,MJ2),Y(MJ1,MJ2),Z(MJ1,MJ2)
DO 10 I=1,MM
DO 10 J=1,NN
Z(I,J)=X(I,J)-Y(I,J)
RETURN
END
    
```

PAGE 1 C *** SUBROUTINE FOR MATRIX MULTIPLICATION ***

```

SUBROUTINE SUBTAL(X,Y,MM,NN,NC,MJ1,MJ2)
MATRIX SUBTRACTION
C
X=X-Y
C
C (UPPER LEFT MM X NN OF X)=(UPPER LEFT
MM X NN OF Y)
C (MJ1,MJ2)=ABSOLUTE DIMENSION OF X AND Y IN THE MAIN ROUTINE
DIMENSION X(MJ1,MJ2),Y(MJ1,MJ2)
DO 10 I=1,MM
DO 10 J=1,NN
X(I,J)=X(I,J)-Y(I,J)
RETURN
END
SUM=0
DO 12 K=1,NN
12 SUM=SUM+X(I,K)*Y(K,J)
Z(I,J)=SUM
11 CONTINUE
10 CONTINUE
RETURN
END
    
```

PAGE 1 C *** SUBROUTINE FOR TRANSPOSE MULTIPLY ***

```

SUBROUTINE TRANDL(X,Y,Z,MM,NN,NC,MJ1,MJ2,MJ3)
TRANSPOSE MULTIPLY (RIGHT)
C
Z=X*Y
C
C (UPPER LEFT MM X NC OF Z)=(UPPER LEFT MM X NN OF X)*(UPPER LEFT
NC X NN OF Y)
C (MJ1,MJ2)=ABSOLUTE DIMENSION OF Y IN THE MAIN ROUTINE
C (MJ1,MJ3)=ABSOLUTE DIMENSION OF Z IN THE MAIN ROUTINE
DIMENSION X(MJ1,MJ2),Y(MJ2,MJ3),Z(MJ1,MJ3)
DO 10 I=1,MM
DO 11 J=1,NC
SUM=0
DO 12 K=1,NN
12 SUM=SUM+X(I,K)*Y(K,J)
Z(I,J)=SUM
11 CONTINUE
10 CONTINUE
RETURN
END
    
```

PAGE 1 C *** SUBROUTINE FOR MATRIX ADDITION ***

```

SUBROUTINE MATADD(X,Y,MM,NN,NC,MJ1,MJ2)
MATRIX ADDITION
C
X=X+Y
C
C (UPPER LEFT MM X NN OF X)=(UPPER LEFT MM X NN OF X)+(UPPER LEFT
MM X NN OF Y)
C (MJ1,MJ2)=ABSOLUTE DIMENSION OF X AND Y IN THE MAIN ROUTINE
DIMENSION X(MJ1,MJ2),Y(MJ1,MJ2)
DO 10 I=1,MM
DO 10 J=1,NN
10 X(I,J)=X(I,J)+Y(I,J)
RETURN
END
    
```

PAGE 1 C *** SUBROUTINE TO COMPUTE THE INVERSE AND DETERMINANT OF X *** PAGE 2 C *** SUBROUTINE TO COMPUTE THE INVERSE AND DETERMINANT OF X ***

```

SUBROUTINE INVDET(X,XDET,MM,MJ)
THIS SUBROUTINE COMPUTES THE INVERSE AND DETERMINANT OF
UPPER LEFT MM X MM OF X.
X IS ORIGINAL MATRIX
MM DIMENSION OF UPPER LEFT OF X (SHOULD BE LESS THAN 11)
XDETB DETERMINANT OF UPPER LEFT MM X MM OF X
MJS ABSOLUTE DIMENSION OF X IN THE MAIN ROUTINE
C THE INVERSE MATRIX IS OVERRITTEN ON THE ORIGINAL.
C NEXT STATEMENT SHOULD BE REPLACED BY
C IMPLICIT COMPLEX*(8)
C FOR COMPLEX VERSION, ALSO STATEMENT NO.1 NEEDS MODIFICATION.
DIMENSION X(MJ,MJ)
DIMENSION I03(10)
XDETB=0
DO 10 J=1,MM
PIVOTING AT LETH STAGE
XMAXP0=10000E=10
MAXI=0
DO 110 I=L,MM
FOR COMPLEX VERSION NEXT STATEMENT SHOULD BE REPLACED BY
C IF(CDABS(XMAXP),GE,CDABS(X(L,L))) GO TO 110
1 XMAXX(I,L)
IF(CDABS(XMAXP),GE,ABS(XXX)) GO TO 110
XMAXP=X(I,L)
MAXI=I
110 CONTINUE
I03(L)=MAXI
IF(MAXI.EQ.L) GO TO 120
IF(MAXI.GT.0) GO TO 121
XDETB=0
GO TO 140
ROW INTERCHANGE
121 DO 14 J=1,MM
XC=X(MAXI,J)
X(MAXI,J)=X(L,J)
14 X(L,J)=XC
XDETB=XDET
120 XDETB=XDET*XMAXP
XC=1.0/XMAXP
X(L,L)=0
DO 11 J=1,MM
11 X(L,J)=X(L,J)*XC
GO 12 J=1,MM
IF(L.EQ.L) GO TO 12
XC=X(I,L)
X(I,L)=0
DO 13 J=1,MM
13 X(I,J)=X(I,J)-XC*X(L,J)
12 CONTINUE
10 CONTINUE
IF(MJ.GT.1) GO TO 123
GO TO 140
C COLUMN INTERCHANGE
123 MH1=MM-1
DO 130 J=1,MH1
MHJ=MM-J

```

```

JJ=I03(MHJ)
IF(JJ.EQ.MHJ) GO TO 130
DO 131 I=1,MM
XC=X(I,JJ)
X(I,JJ)=X(I,MHJ)
131 X(I,MHJ)=XC
130 CONTINUE
140 RETURN
END

```

```

PAGE 1 C ** PLOT SUBROUTINE OF FPE **
PAGE 2 C ** PLOT SUBROUTINE OF FPE **
      8 FORMAT(/,18X,20HMIN FPE,,,,,E10.4//18X,16HGEOMETRICAL MEA
1N/,18X,20HOP MIN. FPE,,,,,E10.4//18X,20HATTAINED AT H-----,
2I6)
      END
SUBROUTINE PLOTTP(IPP, IFF, X, GOPP, L, N, IR, IL, DELT)
DIMENSION X(21)
CALL BEGIN(9899, 1)
CALL ERASE
L1P1+1
CALL SCALE(1, 1, 1, 200, 1, 300, )
CALL AXIS(6, 6, 300, 400, 70, 100, 1, 1)
X(1)=X(2)
XMAX=X(1)
DO 10 I=2, L1
  YI=X(I)
  IF(XI.LE.XMAX) GO TO 10
  XMAX=XI
CONTINUE
XMAX=XI
XMAX=XMAX+1.3
X(1)=XMAX
Y8=400./XMAX
X8=15
Y8X(1)=Y8
X8=0.
CALL VECTOR
CALL TPLOT(XX, YY, 0, 0)
DO 20 I=2, L1
  X8X=X8
  Y8X(1)=Y8
  CALL TPLOT(XX, YY, 1, 0)
CONTINUE
CALL ALPHA
CALL TPLOT(0., 470., 0, 0)
WRITE(1, 1)
WRITE(1, 2) N, IR, DELT, IL
CALL TPLOT(95, 0, 430, 0, 0)
WRITE(1, 4)
CALL TPLOT(=115., 405., 0, 0)
WRITE(1, 5) XMAX
J1=10
X1=100.
CALL TPLOT(X1, =10., 0, 0)
DO 30 I=1, 3
  J1=J1+10
  X1X1=150
  CALL TPLOT(X1, =10., 0, 0)
  WRITE(1, 6) J1
CONTINUE
CALL TPLOT(320., =10., 0, 0)
WRITE(1, 7)
WRITE(1, 8) IPP, GOPP, IFF
CALL TPAUSE
RETURN
1  FORMAT(20HPIC. FINAL PREDICTION ERROR FN,1/)
2  FORMAT(18X,5HN  #,16,3X,5MIR  #,12/,18X,5HOELT=#8.3,3X,5HIL #,
1I2)
4  FORMAT(6HPPE(N))
5  FORMAT(E8.3)
6  FORMAT(I2)
7  FORMAT(1HM/)

```

```

SUBROUTINE SINPT2
COMMON K,X,R,N,DELTA,DDD(10)
COMPLEX X,Y,I,XDET
DIMENSION NAME1(3),NAME2(3),X(10,10),Z(10,10),Y(10,10),C(11,1)
1 NAME3(3),P(402,9),PS(402),NAME4(3)
DATA NAME1(1),NAME1(2),NAME1(3)/2HBB,2HBB,2HBB/
DATA NAME2(1),NAME2(2),NAME2(3)/2HCC,2HCC,2HCC/
DATA NAME3(1),NAME3(2),NAME3(3)/2HEE,2HEE,2HEE/
DATA NAME4(1),NAME4(2),NAME4(3)/2HDD,2HDD,2HDD/
CALL BEGIN(9600,1)
CALL ERASE
WRITE(1,1)
ACCEPT 2,1PRT
CALL QMOND(22,121,2)
DO 50 I=1,LAGH1
DO 60 J=1,IL
CALL QMOND(17,121,C(1),C(11,1))
JPR=J
IF(I,NE,J) GO TO 60
P=C(J)**2+C(JP)**2
P=SQRT(P)
CONTINUE
50 CONTINUE
IF(I,PT,NE,1) GO TO 99
WRITE(16,6)
DO 80 IT=1,LAGH1
IT=IT*1
WRITE(16,7) IT,(P(IT),I),I,1,IL)
CONTINUE
80 CONTINUE
LAGH2=LAGH1+1
CALL QMOND(22,123,2)
CALL QMOND(21,123,NAME3)
DO 70 I=1,IL
DO 71 IT=1,LAGH1
PS(IT)=P(IT,I)
71 CONTINUE
CALL QMOND(16,123,PS(1),PS(LAGH2))
CONTINUE
70 CONTINUE
DDD(1)=38
DDD(5)=11
DDD(6)=11
RETURN
999 RETURN
1 LAGH=1
2 FORMAT(17)
3 FORMAT(17,70H IF UNEXECUTE THE JOB, TYPE IN 991/)
4 FORMAT(17,70H IF PLOT THE POWER SPECTRUM OF INPUT VARIABLES, TYPE IN 11/)
5 FORMAT(17,70H IF PRINT THE CALCULATED P. SPECTRUM, TYPE IN 11/)
6 FORMAT(1H1,70H RATIONAL SPECTRUM OF INPUT VARIABLES1/)
7 FORMAT(1H ,110,5E15,4)
END
30 X(11,11)=CHPLX(GR,GI)
20 CONTINUE
CALL INVDI(X,DET,IL,MJ1)
CALL XCTRX(X,Y,IL,IL,MJ1,MJ1)
DO 40 I=1,IL
DO 41 JJ=1,IL
JPR=J+JJ
C(JJ)=REAL(Y(I,JJ))
C(JP)=AIMAG(Y(I,JJ))
41 CALL QMOND(16,121,C(1),C(11,1))
40 CONTINUE
10 CONTINUE
WRITE(1,4)
ACCEPT 2,1PLT
IF(I,PT,NE,1) GO TO 999
WRITE(1,5)

```

PAGE 1 C ** SUBROUTINE FOR FOURIER TRANSF. OF MATRIX (A,B) PAGE 1 C *** SUBROUTINE FOR FOURIER TRANSFORM(GOERTZEL METHOD) ***

```

SUBROUTINE HULRPN(IR,IP,NAME1,NAME2,X,ZD,C,H)
  COMPLEX X,ZD
  DIMENSION A(21,10,10),X(10,10),G(21),C(25),
  1 NAME(13),NAME2(3),ZD(10,10)
  INTEGER H,M
  MJB=21
  HJIS=10
  HJMS=1
  CALL GMOND(22,124,2)
  CALL GMOND(21,124,NAME1)
  CALL GMOND(17,124,C(1),C(10))
  MSC(1)
  IP=IP+1
  IP2=IP+IP
  IP1=IP2+1
  DO 10 II=1,M
    DO 10 I=1,IR
      CALL GMOND(17,124,C(1),C(IP1))
      DO 11 JJ=1,IP
        A(II,II,JJ)=C(JJ)
      CONTINUE
    CALL GMOND(22,122,2)
    CALL GMOND(21,122,NAME2)
    DO 20 JF=1,HI
      DO 30 I=1,HI
        DO 31 JJ=1,IP
          IP(II,NE,JJ) GO TO 32
        G(1)=0
        GO TO 33
      G(1)=0
    GO 35 I=1,M
  I=101
  G(1)=A(1,II,JJ)
  LGM
  CALL FGERI(G,LG,GR,G1,H,JF)
  X(II,JJ)=CMPLX(GR,G1)
31 CONTINUE
30 CONTINUE
  DO 40 II=1,IR
    DO 41 JJ=1,IP
      JP=JP+1
      C(JJ)=REAL(X(II,JJ))
41 C(JJ)=AIMAG(X(II,JJ))
  CALL GMOND(16,122,C(1),C(IP1))
40 CONTINUE
20 CONTINUE
  IRI=IR+1
  DO 50 I=1,IR
    CALL GMOND(17,124,C(1),C(IR1))
    DO 51 J=1,IR
      ZD(I,J)=C(J)
51 CONTINUE
50 RETURN
  END
  
```

```

PAGE 1 C *** SUBROUTINE TO COMPUTE THE INVERSE AND DETERMINANT OF X *** PAGE 1

SUBROUTINE INVDEI(X,XDET,MM,MJ)
COMPLEX X,XDET,XHARP,XC
DIMENSION X(MJ,MJ)
DIMENSION IOS(10)
XDET=1.0
DO 10 I=1,MM
DO 10 J=1,MM
XSUM=0.0
C PIVOTING AT L-TH STAGE
XHARP=0.10000E+8
MAXI=0
DO 110 I=1,MM
FOR COMPLEX VERSION NEXT STATEMENT SHOULD BE REPLACED BY
1 IF(CABS(XHARP).GE.CABS(X(I,I))) GO TO 110
XHARP=X(I,I)
MAXI=I
110 CONTINUE
IOS(L)=MAXI
IF(MAXI.EQ.L) GO TO 120
IF(MAXI.GT.0) GO TO 121
XDET=X(0)
GO TO 140
C ROW INTERCHANGE
DO 121 J=1,MM
XC=X(MAXI,J)
X(MAXI,J)=X(L,J)
14 X(L,J)=XC
XDET=XDET
120 XDET=XDET*XHARP
XC=1.0/XHARP
X(L,L)=1.0
DO 11 J=1,MM
11 X(L,J)=X(L,J)*XC
IF(I.EQ.L) GO TO 12
XC=X(I,L)
X(I,L)=0.0
DO 13 J=1,MM
13 X(I,J)=X(I,J)-XC*X(L,J)
12 CONTINUE
16 CONTINUE
GO TO 140
IF(MM.GT.1) GO TO 123
C COLUMN INTERCHANGE
DO 130 J=1,MM
MMJ=MM-J
JJ=IOS(MMJ)
IF(JJ.EQ.MMJ) GO TO 130
DO 131 I=1,MM
XC=X(I,JJ)
X(I,JJ)=X(I,MMJ)
131 X(I,MMJ)=XC
130 CONTINUE
140 RETURN
END
    
```

```

SUBROUTINE PLOTS2
COMMON K,IR,N,DELTA,DDD(10)
DIMENSION X(402),H(402),HS(10),YS(20)
DATA NAME1(1),NAME1(2),NAME1(3)/ZHEE,2HEE,2HEE/
CALL BEGIN(9600,1)
IDD=DDD(1)
IF(1DD.EQ.0) GO TO 999
1000 CALL ERASE
IL=K-IR
IX1=DDD(5)
IX2=DDD(6)
IF(1DD=20) 10,20,30
GO TO 40
WRITE(1,1)
20 WRITE(1,2)
GO TO 40
30 WRITE(1,3)
40 CALL AXSEL(NOPHT,NOPTV)
LAGH1=DDD(2)
LAGH2=LAGH1+1
LAGH=LAGH1+1
CALL GROND(22,121,2)
CALL GROND(21,121,NAME1)
DO 50 J=1,IX1
DO 50 JJ=1,IX2
IDD=DDD(1)
IF(1DD.NE.30) GO TO 60
IF(11.NE.JJ) GO TO 50
60 CALL GROND(17,121,X(1),X(LAGH2))
CALL PLAUT(NOPHT,NOPTV,N,DELTA,X,JJ,II,LAGH1,IDD)
50 CONTINUE
WRITE(1,4)
ACCEPT 9,10P
IF(1OP.EQ.1) GO TO 1000
RETURN
1 FORMAT(70H** FREQ. RESP. FN. WITH OPEN LOOP SYSTEM DERIVED FROM A*)
2 FORMAT(70H** FREQ. RESP. FN. WITH CLOSED LOOP SYSTEM DERIVED FROM
1A=R MODEL1/)
3 FORMAT(70H** RATIONAL SPECTRUM DERIVED FROM A=R MODEL1/)
4 FORMAT(//,70H** IF PLOT AGAIN, TYPE IN 1/)
5 FORMAT(13)
END

SUBROUTINE PLAUT(NOPHT,NOPTV,N,DT,P,II,JJ,LAGH1,10PT)
DIMENSION P(402),H(402),HS(10),YS(20)
DIMENSION DUMY(402)
LAGH=LAGH1+1
MX=600,0
MY=600,0
MXR=150,0
MYR=100,0
MXN=MX+MX0
MYN=MY+MY0
MXM=MX
MYM=MY
CALL HCORD(NOPHT,LAGH,DT,M,MS,M,MS)
XL=(MX+FLOAT(NOPHT))/FLOAT(N)
CALL VCORD(NOPHT,LAGH,P,VS,MM,ML,MN,II,JJ,DUMY)
COEF=1,0
IF(NOPTV.EQ.0) COEF=-1,0
YL=(MY+COEF)/FLOAT(M)
CALL ERASE
CALL SCALE(1,0,1,0,NX0,NY0)
CALL AXIS(0,0,0,NX,NY,XL,YL,1,1)
CALL VECTOR
CALL TPLOT(0,NY,0,0)
CALL TPLOT(NX,0,1,0)
DO 20 I=1,LAGH1
M1)=0(II)+NX
P(I)=P(II)+MY
20 CONTINUE
CALL TPLOT(0,0,0,0,0)
MARK=0
V1=P(1)
CALL TPLOT(0,V1,0,MARK)
DO 30 I=1,LAGH
I1=I+1
V1=P(I1)
V2=P(I)
30 CONTINUE
CALL TPLOT(Y2,V1,I1,MARK)
CALL ALPHA
MYC=MY+40,0
CALL TPLOT(-30,0,NYC,0,0)
IF(1OPT=20) 200,210,220
WRITE(1,1) JJ,II
FORMAT(70HFIG. FREQUENCY RESP. FN. OF OPEN LOOP SYSTEM, X(1,12,1
19H) FROM U(1,12,1H))
GO TO 8
210 WRITE(1,9) JJ,II
FORMAT(70HFIG. FREQUENCY RESP. FN. OF CLOSED LOOP SYSTEM, X(1,12,1
1,20H) FROM X(1,12,1H))
GO TO 8
220 WRITE(1,7) II,JJ
FORMAT(70HFIG. RATIONAL SPECTRUM P(1,12,1H,I2,1H))
8 M1=M+1
MXD=ARS(XL)=10,0
DO 40 I=1,M1
    
```



```

PAGE 2
XHD=HXD+ABS(XL)
HYD=18.0
CALL TPLOT(WXD,WYD,0.0)
IMHS(I)
IF(NOPHY,NE,1) GO TO 90
WRITE(1,2) HS(I)
GO TO 100
90 WRITE(1,4) IH
2 FORMAT(F3.1)
100 IF(NOPHY,EO,1) GO TO 40
WXX=XD-13.0
WYY=WD-15.0
CALL TPLOT(WXX,WYY,0.0)
IIB=10
WRITE(1,4) I10
4 FORMAT(I2)
40 CONTINUE
CALL TPLOT(260.0,50.0,0.0)
WRITE(1,3)
3 FORMAT(20HFREQUENCY (CYCLE/SEC))
CALL TPLOT(750.0,50.0,0.0)
WRITE(1,5) HS
5 FORMAT(20H(510**1,12,1H))
50 CALL TPLOT(50.0,0.0,0.0)
W0=181.7H
WYD=HYD+ABS(YL)
CALL TPLOT(57.0,WYD,0.0)
IPI=1
IVS=VS(IP1)
IF(NOPHY,LE,0) GO TO 110
WRITE(1,2) VS(IP1)
GO TO 120
110 WRITE(1,0) IVS
6 FORMAT(I4)
120 WXX=52.0
WYY=WD-14.0
CALL TPLOT(WXX,WYY,0.0)
WRITE(1,4) I10
60 CONTINUE
IF(NOPHY) 82,80,81
81 WYD=HYD+20.0
WXC=130.
CALL TPLOT(WXC,WYC,0.0)
WRITE(1,5) WYD
GO TO 80
82 CALL TPLOT(-90.0,250.0,0.0)
CALL VECTOR
80 CALL ALPHA
WYD=WD-10.0
CALL TPLOT(532.0,WYD,0.0)
WRITE(1,80)
WYD=WD+20.0
CALL TPLOT(560.0,WYD,0.0)
WRITE(1,83) N

```

PAGE 3

```

WYD=WD+20.0
CALL TPLOT(560.0,WYD,0.0)
WRITE(1,84) LAGH
WYD=WD-20.0
CALL TPLOT(560.0,WYD,0.0)
WRITE(1,85) OT
400 CALL TPAUSE
CALL ERASE
RETURN
83 FORMAT(5HN *I6)
84 FORMAT(5HLAGH,I6)
85 FORMAT(5HOT *E8,2,4H SEC)
86 FORMAT(70H* INPUT DATA *)
END

```

PAGE 1

```

SUBROUTINE MAXMIN(X,N,XMAX,XMIN)
DIMENSION X(402)
XMAX=X(1)
DO 1 I=1,N
X=X(I)
IF(X,LE,XMAX) GO TO 1
XMAX=X(I)
1 CONTINUE
XMIN=X(1)
DO 2 I=1,N
X=X(I)
IF(X,GE,XMIN) GO TO 2
XMIN=X(I)
2 CONTINUE
RETURN
END

```

```

PAGE 1
SUBROUTINE VCORD(NOPT,LAGH,Y,VX,MM,ML,MN,III,JJJ,YX)
DIMENSION X(482),YX(20),YX(482)
LAGH=LAGH+1
JJ=0
DO 4 I=1,LAGH
  XLX(I)
  IF(XL,LE,0.) GO TO 4
  J=JJ+1
  YX(JJ)=YX(I)
  CALL MAXMIN(YX,JJ,XMAX,XMIN)
  DO 5 I=1,LAGH
    XLX(I)
    IF(XL,GT,0.) GO TO 5
    X(I)=XMIN
  5 CONTINUE
  XL=ALOG10(XMAX)
  IF(NOPT) 3,2,1
  1 H=XL
  IF(XL,GE,0.0) GO TO 70
  H=XL-1.0
  70 H=HMAX/(10.0+H)+1.0
  H=HMAX/(10.0+H)+1.0
  VX(2)=VX(3)/2.0
  VX(1)=0.
  XNORM=FLOAT(HS)+10.0+H
  DO 10 I=1,LAGH
    X(I)=X(I)/XNORM
  10 CONTINUE
  MM=0
  ML=0
  MN=0
  2 MAXX=XL+1.0
  XL=ALOG10(XMIN)
  MINX=XL-1.0
  MM=MAXX-MINX
  MM1=MM+1
  DO 20 I=1,MM1
    VX(I)=FLOAT(MINX)+FLOAT(I)-1.0
  20 CONTINUE
  DO 30 I=1,LAGH
    XLX(I)
    X(I)=(ALOG10(XL)-FLOAT(MINX))/FLOAT(MH)
  30 CONTINUE
  ML=0
  MN=0
  DO 30 I=1,LAGH
    XL=CDR+ALOG10(XMAX)/10.0
    MAXX=XL+1.0
    XL=CDR+ALOG10(XMIN)/10.0
    MINX=XL-1.0
    MM=MAXX-MINX
    MM1=MM+1
  40 I=1,MM1
    VX(I)=(I-MINX+1)*10
  40 CONTINUE
  FMM=MM+10
  FMIN=MINX+10
  DO 50 I=1,LAGH
    XL=X(I)
    X(I)=(CDR+ALOG10(XL)-FMIN)/FMM
  50 CONTINUE
  ML=0
  MN=0
  60 CONTINUE
  RETURN
END
PAGE 2

```

```

SUBROUTINE HCORD(NOPT,LAGH,OT,H,HS,MH,MS)
DIMENSION H(482),HS(10)
FLAG=LAGH
DO 10 I=1,LAGH
  F(I)=1
  H(I)=FI/(C.0+DI*FLAG)
10 CONTINUE
IF(NOPT.LT.0) GO TO 30
A1=H(LAGH)
CALL ETYPE(A1,COEF,ZE,IZE)
MSIZE
IF(IZE.EQ.0) GO TO 200
HS(3)=COEF
HS(2)=COEF/2.0
MS(1)=0.
MS2
DO 20 I=1,LAGH
  H(I)=H(I)/H(LAGH)
20 CONTINUE
GO TO 60
30 X=H(LAGH)
XL=ALOG10(X)
MAXX=XL+1.0
XMH(1)
XL=ALOG10(XM)
MINX=XL-1.0
MH=MAX(MINX)
MH=MAX(MINH)
MH=MAX(MH)
DO 40 I=1,MH
  HS(I)=I*MINH+1
40 CONTINUE
DO 50 I=1,LAGH
  XL=H(I)
  H(I)=(ALOG10(XL)+FLOAT(MINH))/FLOAT(MH)
50 CONTINUE
MH=MH
MS=0
GO TO 60
60 WRITE(10,100)
60 RETURN
100 FORMAT(1H ,70HTS UNREASONABLE JOB. 1/)
END

SUBROUTINE ETYPE(X,Y,Z,MZ)
MZ=1
IF(X.EQ.0.) GO TO 100
IF(X.LT.0.) ZZ=1.0
X=ABS(X)
XL=ALOG10(X)
XL=ABS(XL)
IF(XL.GT.15.0) GO TO 101
DO 10 I=1,11
  FI=1
  IF(XL+.6E.FI) GO TO 10
  FI=FI-1.0
  EI=FI-1.0
  GO TO 20
10 CONTINUE
20 IE=1
  IF(XL+.6E.0.P) GO TO 30
  IE=IE+1
  GO Y=Z/(10.0**IE)
  Z=IE
  GO TO 110
30 ZZ=1 X IS NEGATIVE
  ZZ=1 X IS POSITIVE
  C
  ZZ=0 X IS UNREASONABLE
  GO TO 100
  Z=0
  MZ=1
  GO TO 110
101 MZ=0
110 CONTINUE
  RETURN
  END

SUBROUTINE AXSEL(NOPTH,NOPTV)
WRITE(1,1)
ACCEPT 2,NI,N2
NOPTH=NI
IF(NI.EQ.2) NOPTH=1
IF(N2=2) 24,25,26
24 NOPTV=1
  GO TO 22
25 NOPTV=0
  GO TO 22
26 NOPTV=1
22 RETURN
1 FORMAT(70H** AXIS SCALING CONDITION **1//70H X AXIS 1...LINEAR
  1 2...LOG1/70H Y AXIS 1...LINEAR 2...LOG 3...DB1//70H Y
  2PE IN THE SCALING NO. 7/)
2 FORMAT(213)
  END
    
```

```

SUBROUTINE FTABR2
COMMON K,IR,N,DELTA,ODD(10)
COMPLEX X,Z,XDEF
DIMENSION X(10,10),Z(10,10),C(25)
DIMENSION NAME1(3),NAME2(3),NAME3(3)
DATA NAME1(1),NAME1(2),NAME1(3)/2HID,2HDA,2HTA/
DATA NAME2(1),NAME2(2),NAME2(3)/2HOD,2HOO,2HOO/
DATA NAME3(1),NAME3(2),NAME3(3)/2HCC,2HCC,2HCC/
CALL BEGIN(0600,1)
CALL ERASE
LAGH1=ODD(2)
LAGH2=LAGH1-1
IP=K
IP1=IP+1
IP2=IP+2
IP3=IP+3
WRITE(1,1)
FORMAT(20H ** FOURIER TRANSFORMATION OF MATRIX (A,B) / 4ZBH IF U
INEXECUTE THE JOB, TYPE IN 991/)
ACCEPT 2,IAB
FORMAT(15)
IF(LAB,EG,99) GO TO 999
CALL MULRP(N,IR,IP,NAME1,NAME2,X,Z,C,LAGH)
C ** MATRIX (A,B) STORED TO DDDDD
NJ1=10
CALL OMOND(22,122,2)
CALL OMOND(21,123,2)
CALL OMOND(21,123,NAME3)
DO 10 I=1,LAGH1
DO 20 II=1,IR
CALL OMOND(17,121,C(1),C(10))
DO 11 JJ=1,IP
A(I,II,JJ)=C(JJ)
CONTINUE
CALL OMOND(22,122,2)
CALL OMOND(21,122,NAME2)
DO 20 JF=1,JI1
DO 30 IF=1,IR
DO 31 JF=1,IP
IF(II,NE,JI) GO TO 32
G(I)=1.0
GO TO 33
G(I)=0.
DO 35 I=1,M
I1=I+1
G(I1)=A(I,II,JJ)
LG=M
CALL FGERI(G,LC,GR,GI,M,JF)
X(II,JJ)=CMPLX(GR,GI)
CONTINUE
DO 40 II=1,IR
DO 41 JJ=1,IP
JP=JJ+IP
C(JJ)=ATMAG(X(II,JJ))
C(JJ)=REAL(X(II,JJ))
CALL OMOND(16,122,C(1),C(10))
CONTINUE
CONTINUE
IR1=IR+1
DO 50 I=1,IR
JP=IP+JJ
C(JJ)=REAL(Z(I,II))
C(JJ)=ATMAG(Z(I,II))
CALL OMOND(17,121,C(1),C(10))
DO 51 J=1,IR
ZD(I,J)=C(J)
CONTINUE
RETURN
999 ** MATRIX ELEMENT (A=1,B) STORED TO CCCCC
END
    
```

```

PAGE 1 C *** SUBROUTINE TO COMPUTE THE INVERSE AND DETERMINANT OF X *** PAGE 1 C *** SUBROUTINE FOR FOURIER TRANSFORM(GOERTZEL METHOD) ***

SUBROUTINE INVDEIX(X, XDET, MM, MJ)
  COMPLEX X, XDET, XMAXP, XC
  DIMENSION X(MJ, MJ)
  DIMENSION JDS(10)
  XDET=1.0
  DO 10 I=1, MM
    PIVOTING AT L-TH STAGE
  XMAXP=0.10000E+8
  MAXI=0
  DO 110 I=1, MM
    FOR COMPLEX VERSION NEXT STATEMENT SHOULD BE REPLACED BY
  1 IF(CABS(XMAXP).GE.CABS(X(I, L))) GO TO 110
  XMAXP=X(I, L)
  MAXI=I
  110 CONTINUE
  JDS(L)=MAXI
  IF(MAXI.EQ.L) GO TO 120
  IF(MAXI.GT.0) GO TO 121
  XDET=0.0
  GO TO 140
  ROW INTERCHANGE
  121 DO 14 J=1, MM
    XC=X(MAXI, J)
    X(MAXI, J)=X(L, J)
    X(L, J)=XC
  XDET=XDET*XMAXP
  120 XDET=XDET*XMAXP
  X(L, L)=1.0
  DO 11 J=1, MM
    X(L, J)=X(L, J)*XC
  IF(I.EQ.L) GO TO 12
  XC=X(I, L)
  X(I, L)=0.0
  DO 13 J=1, MM
    X(I, J)=X(I, J)-XC*X(L, J)
  13 X(I, J)=X(I, J)-XC*X(L, J)
  12 CONTINUE
  10 CONTINUE
  IF(MM.GT.1) GO TO 123
  GO TO 140
  COLUMN INTERCHANGE
  123 MM1=MM-1
  MJ1=MM-1
  DO 130 J=1, MM1
    J1=J+1
    IF(J1.GE.MJ1) GO TO 130
    DO 131 I=1, MM
      XC=X(I, J)
      X(I, J)=X(I, MJ1)
      X(I, MJ1)=XC
  131 X(I, MJ1)=X(I, MJ1)
  130 CONTINUE
  140 RETURN
  END
  SUBROUTINE FGERI(G, LG, GR, GI, M, JGF)
  INTEGER H
  DIMENSION G(21)
  LGP1=LG+1
  C REVERSAL OF G(I), I=1, ..., LGP1 INTO G(LG3=I) LG3=LGP1+1
  IF(LGP1.LE.1) GO TO 110
  LG3=LGP1+1
  LG4=LG3/2
  DO 100 I=1, LG4
    I2=LG3-I
    T=G(I)
    G(I)=G(I2)
    G(I2)=T
  100 G(I2)=T
  110 PI=3.1415926536
  AM=H
  T=PI/AM
  AK=JGF-1
  TK=T*AK
  CK=COS(TK)
  SK=SIN(TK)
  CK2=CK*CK
  UM2=0.0
  UM1=0.0
  IF(LG.EQ.0) GO TO 12
  DO 11 I=1, LG
    UM0=CK2*UM1+UM2+G(I)
    UM2=UM1
    UM1=UM0
  11 UM1=UM0
  12 G=CK-UM1-UM2+G(LGP1)
  RETURN
  END

```

```

PAGE 1 C ** CALCULATION OF OPEN LOOP SYSTEM FREQ. RESP. FN. PAGE 2 C ** CALCULATION OF OPEN LOOP SYSTEM FREQ. RESP. FN.
10 CONTINUE
CALL QMOND(16,122,C(1),C(IRL1))
CONTINUE
CALL QMOND(22,123,2)
CALL QMOND(21,123,NAME3)
DO 50 IS=1,IR
CALL QMOND(22,122,2)
CALL QMOND(21,122,NAME2)
DO 60 IT=1,LAGH1
CALL QMOND(17,122,C(1),C(IRL1))
DO 61 IT=1,IR
IX=(IT-1)*IL2
DO 61 JJ=1,IL
J1=IX+JJ
J2=IX+IL+JJ
GR=C(J1)
GI=C(J2)
IF(II.NE.IS) GO TO 61
PP=GR+2*GI+2
P(IT,JJ)=SORT(PP)
61 CONTINUE
CONTINUE
IF(IPRT.NE.1) GO TO 63
WRITE(16,4) IS
DO 62 IT=1,LAGH1
WRITE(16,5) IT,(P(IT,JJ),JJ=1,IL)
62 CONTINUE
DO 70 JJ=1,IL
DO 71 IT=1,LAGH1
G(IT)=P(IT,JJ)
CALL QMOND(16,123,G(1),G(LAGH2))
70 CONTINUE
70 CONTINUE
50 CONTINUE
000(1)=10
000(5)=IR
000(6)=1L
RETURN
999
1 FORMAT(70H** FREQ. RESP. FN. OF OPEN LOOP SYSTEM DERIVED FROM A=R
MODEL//,70H IF UNEXECUTE THE JOB, TYPE IN.99//)
2 FORMAT(15)
3 FORMAT(//,70H** IF PRINT OUT THE FREQ. RESP. FN. TYPE IN.1//)
4 FORMAT(5H1,70H FREQUENCY RESPONSE FN. WITH OPEN LOOP SYSTEM',5X,
170HOUTPUT VARIABLE X(1,12,1H))
5 FORMAT(1H ,110,2E16,4)
END
35
PAGE 1
SUBROUTINE PRODMC(X,Y,Z,L,M,N)
COMPLEX X,Y,Z,XSUM
DIMENSION X(10,10),Y(10,10),Z(10,10),Z(10,10)
DO 10 I=1,IR
DO 10 J=1,N
XSUM=0
DO 12 K=1,M
XSUM=XSUM+X(I,K)*Y(K,J)
12 CONTINUE
10 Z(I,J)=XSUM
RETURN
END
41 C(J2)=AIMAG(Z(11,11))
40 CONTINUE

```

```

PAGE 1 C ** POWER SPECTRUM OF OUTPUT VARIABLES DERIVED FROM A-R MODEL ** PAGE 2 C ** POWER SPECTRUM OF OUTPUT VARIABLES DERIVED FROM A-R MODEL
SUBROUTINE RASPE2
COMMON K,IR,N,DELT,DDD(10)
COMPLEX X,Y,Z
DIMENSION AF(402,10),C(402),X(10,10),Y(10,10),Z(10,10),P(31)
DATA NAME1(1),NAME1(2),NAME1(3)/2HEF,2HEF,2HEF/
DATA NAME2(1),NAME2(2),NAME2(3)/2H8B,2H8B,2H8B/
DATA NAME3(1),NAME3(2),NAME3(3)/2HDD,2HDD,2HDD/
DATA NAME4(1),NAME4(2),NAME4(3)/2HD,2HD,2HD/
DATA NAME5(1),NAME5(2),NAME5(3)/2HCC,2HCC,2HCC/
CALL BEGIN(9000,1)
RJ=10
IP=K
IP1=IP+1
IP2=IP+IP
IP3=IP2+1
IR1=IR+1
IR2=IR+IR
IRR1=IRR+1
IL1=IL+1
IL2=IL+IL
ILL1=ILL+1
IRL1=IRL+IR+1
LAGH1=DDD(2)
LAGH=LAGH1+1
LAGM2=LAGH1+1
CALL QMOND(22,123,2)
CALL QMOND(21,123,NAME3)
DO 15 I=1,LAGH1
DO 15 I=1,IR
CALL QMOND(17,123,C(1),C(IPR1))
DO 16 J=1,IR
IF(JJ.NE.1) GO TO 16
JP=JP+J
GR=C(JP)
GG=GR**2+GI**2
AF(1,I)=SOR((GB)
WRITE(1,1)
CALL ERASE
WRITE(1,1)
FORMAT(70H*** R. POWER SPECTRUM OF OUTPUT VARIABLES'//70H DERIV
LED FROM A-R MODEL. IF UNEXECUTE THE JOB. TYPE IN 99'//)
ACCEPT 2,ISP
FORMAT(15)
IF(ISP.EQ.99) GO TO 999
WRITE(1,3)
FORMAT(//70H** IF PRINT THE POWER SPECTRUM, TYPE IN 1'//)
ACCEPT 2,IPR
CALL QMOND(22,123,2)
CALL QMOND(21,123,NAME3)
IF(LEQ.R) GO TO 888
CALL QMOND(22,121,2)
CALL QMOND(21,121,NAME1)
CALL QMOND(22,122,2)
SUBROUTINE RASPE2
DO 100 IT=1,LAGH1
CALL QMOND(17,121,C(1),C(IRR1))
DO 10 I=1,IR
YI=(I-1)*IL+2
DO 10 J=1,IL
JI=JI+J
J2=JI+J+IL
GR=C(J2)
GI=C(J2)
X(I,J)=CMPLX(GR,GI)
CONTINUE
DO 20 I=1,IL
CALL QMOND(17,122,C(1),C(ILL1))
DO 21 J=1,IL
JP=JP+J
GR=C(J)
GI=C(JP)
Y(I,J)=CMPLX(GR,GI)
CONTINUE
CALL XYCTRX(X,Y,Z,IR,IL,MJ,MJ)
I2R=IR*IR+2*1
DO 30 I=1,IR
YI=(I-1)*IR2
DO 30 J=1,IR
GR=REAL(Z(I,J))
GI=AIMAG(Z(I,J))
JI=JI+1
J2=JI+IR+J
C(J1)=GR
C(J2)=GI
CONTINUE
CALL QMOND(16,123,C(1),C(IRR))
CONTINUE
IF(IPRT.NE.1) GO TO 888
WRITE(16,4)
FORMAT(16H,70H*** RATIONAL POWER SPECTRUM OF OUTPUT VARIABLES'//7
10H CAUSED BY INPUT DISTURBANCE ***'//)
CALL QMOND(22,123,2)
CALL QMOND(21,123,NAME3)
DO 40 IT=1,LAGH1
CALL QMOND(17,123,C(1),C(IRR))
DO 41 I=1,IR
YI=(I-1)*IR2
DO 41 J=1,IR
JI=JI+J
J2=JI+IR+J
GR=C(J2)
GI=C(J2)
IF(I.NE.J) GO TO 41
PP=GR**2+GI**2
P(I)=SORT(PP)
CONTINUE
WRITE(16,5) YI,(P(I),I=1,IR)
FORMAT(16H,15.10E10,3)
CONTINUE
888 CALL QMOND(22,124,2)

```

```

PAGE 3 C ** POWER SPECTRUM OF OUTPUT VARIABLES DERIVED FROM A-R MODEL ** PAGE 4 C ** POWER SPECTRUM OF OUTPUT VARIABLES DERIVED FROM A-R MODEL
CALL QMND(21,124,NAME4)
CALL QMND(17,124,C(1),C(10))
M=C(1)
DO 110 I=1,M
CONTINUE
DO 110 II=1,IR
WRITE(16,7) II,(P(II),II=1,IR)
CALL QMND(17,124,C(1),C(IP1))
CONTINUE
DO 111 I=1,IR
DO 300 II=1,IR
C(II)=AF(II,II)
CALL QMND(16,123,C(1),C(LAGH2))
CONTINUE
IF(IPRY.NE.1) GO TO 999
DO 311 II=1,IR
Z1=ZD(II)
CALL QMND(22,122,2)
DO 200 IT=1,LAGH1
DO 210 IJ=1,IR
CALL QMND(17,123,C(1),C(IPP1))
JP=IP+JJ
GR=C(JJ)
GI=C(JJ)
X(I,I,JJ)=CMPLX(GR,GI)
CONTINUE
CALL XYTRX(Z1,Y,IR,MJ,MJ)
IX(I,IR)=IR*2+1
DO 230 II=1,IR
IX=(II+1)*IR2
DO 230 JJ=1,IR
JI=IX+JJ
J2=IX+IR+JJ
GR=REAL(Y(II,JJ))
GI=AIMAG(Y(II,JJ))
C(J1)=GR
C(J2)=GI
CONTINUE
CALL QMND(16,123,C(1),C(IXX1))
CONTINUE
IF(IPRY.NE.1) GO TO 777
WRITE(16,6)
FORMAT(1H1,70H*** RATIONAL POWER SPECTRUM OF OUTPUT VARIABLES /,7
10H CAUSED BY NOISE DISTURBANCE ***//)
CALL QMND(22,123,2)
DO 50 IT=1,LAGH1
CALL QMND(17,123,C(1),C(IXX1))
CONTINUE
DO 60 IY=1,LAGH1
CALL QMND(17,123,C(1),C(IXX1))
DO 61 II=1,IR
IX=(II+1)*IR2
DO 61 JJ=1,IR
J1=IX+JJ
J2=IX+IR+JJ
GR=C(J1)
GI=C(J2)
IF(II.NE.JJ) GO TO 61

```



```

SUBROUTINE SPLT02
COMMON K,IR,N,DELT,DDO(10)
DIMENSION P(402,10),C(402)
DIMENSION NAME(3)
DATA NAME(1),NAME(2),NAME(3)/2HEE,2HEE,2HEE/
CALL BEGIN(900,1)
CALL ERASE
LAGH1=DDO(2)
LAGH2=LAGH1
IP=K
IL=IP-IR
WRITE(1,1) IR,IL
WRITE(1,2)
ACCEPT 3,ISEL
IF(ISEL.EQ.99) GO TO 999
CALL AXSEL(NOPTM,NOPTV)
CALL GMONDR(IR,IL,LAGH1,P,C,ISEL)
CALL GMOND(22,121,2)
CALL GMOND(21,121,NAME1)
DO 300 II=1,LAGH1
DO 200 IT=1,LAGH1
CALL GMOND(17,121,C(1),C(IR21))
CALL PLAUT(NOPTM,NOPTV,N,DELT,C,IT,LAGH1,ISEL)
CONTINUE
300 GO TO 500
999 RETURN
1 FORMAT(70H*** PLOT THE OUTPUT POWER SPECTRUM **/,4X,3HTR=,I4/,4X,
1,3HIL=,I4/)
2 FORMAT(/,70H** TYPE IN NO./,3X,70H1...INPUT CONTRIBUTION TO POW
1ER SPECTRUM/,3X,70H2...NOISE CONTRIBUTION/,3X,70H3...NOISE POW
3EXECUTE THE JOB//)
3 FORMAT(16)
END
110 CONTINUE
100 CONTINUE
CALL GMOND(22,122,2)
CALL GMOND(21,122,NAME2)
DO 120 II=1,IR
DO 121 IT=1,LAGH1
C(IT)*P(IT,IT)
CALL GMOND(16,122,C(1),C(LAGH2))
CONTINUE
GO TO 999
200 CALL GMOND(22,122,2)
CALL GMOND(21,122,NAME2)
CALL GMOND(21,123,NAME3)
CALL GMOND(17,123,C(1),C(10))
M=C(1)
DO 260 IM=1,M
DO 260 IS=1,IR
CALL GMOND(17,123,C(1),C(IP1))
DO 270 II=1,IR

```

PAGE 2 PAGE 3

```

CALL QMOND(17,'23,C(1),C(IR1))
DO 270 JJ=1,IR
  IF(II.NE.JJ) GO TO 270
  ZZ=C(JJ)
  Z(II)=SORT(ZZ)
270 CONTINUE
DO 280 II=1,IR
  CALL QMOND(17,'21,C(1),C(LAGH2))
  DO 290 II=1,LAGH1
    CI=C(II)
    C(II)=Z(II)/CI
290 CONTINUE
280 CONTINUE
GO TO 999
DO 400 II=1,LAGH1
  CALL QMOND(17,'21,C(1),C(IR21))
DO 410 II=1,IR
  IK=(II-1)*IR2
DO 410 JJ=1,IR
  IF(II.NE.JJ) GO TO 410
  J1=IK*JJ
  J2=IK*IR+JJ
  GR=C(J1)
  GI=C(J2)
  GG=GR**2+GI**2
  P(II,II)=SORT(GG)
410 CONTINUE
400 CONTINUE
CALL QMOND(22,122,2)
DO 415 II=1,IR
  DO 416 II=1,LAGH1
    C(II)=P(II,II)
415 CONTINUE
DO 420 II=1,LAGH1
  CALL QMOND(17,'21,C(1),C(IR21))
DO 430 II=1,IR
  IK=(II-1)*IR2
DO 430 JJ=1,IR
  IF(II.NE.JJ) GO TO 430
  J1=IK*JJ
  J2=IK*IR+JJ
  GR=C(J1)
  GI=C(J2)
  GG=GR**2+GI**2
  P(II,II)=SORT(GG)
430 CONTINUE
420 CONTINUE
CALL QMOND(22,'22,2)
DO 440 II=1,IR
  CALL QMOND(17,'22,C(1),C(LAGH2))
DO 440 II=1,LAGH2
  P(II,II)=P(II,II)+C(II)
440 CONTINUE
CALL QMOND(22,'22,2)

```

PAGE 1

PAGE 2

```

WD=ABS(XL)-18.0
DO 40 I=1,M
  WD=WD+ABS(XL)
  WYD=-10.0
  CALL TPLDT(WXD,WYD,0,0)
  I=HSH(I)
  IF(NOPTR,NE.1) GO TO 90
  WRITE(1,2) HSK(I)
  GO TO 100
90 WRITE(1,4) IH
  2 FORMAT(F3.1)
  100 IF(NOPTR,EG.1) GO TO 40
  WXX=WD-13.0
  WYY=WYD-15.0
  CALL TPLDT(WXX,WYY,0,0)
  I=I+10
  WRITE(1,4) I+10
  4 FORMAT(I2)
  40 CONTINUE
  CALL TPLDT(260.,-50.,0,0)
  WRITE(1,3)
  3 FORMAT(20HFREQUENCY (CYCLE/SEC))
  IF(NOPTR,NE.1) GO TO 50
  CALL TPLDT(750.,-50.,0,0)
  WRITE(1,5) MS
  5 FORMAT(20H(10**),I2,1H))
  50 CALL TPLDT(-50.,0,0,0)
  WYD=19.0
  DO 60 I=1,MM
  WYD=WYD+ABS(YL)
  CALL TPLDT(-57.,WYD,0,0)
  I=I+1
  IVS=VS(TPI)
  IF(NOPTR,LE.0) GO TO 110
  WRITE(1,2) VS(TPI)
  GO TO 120
  110 WRITE(1,6) IVS
  120 IF(NOPTR) 60,70,60
  70 WXX=-57.0
  WYY=WYD-14.0
  CALL TPLDT(WXX,WYY,0,0)
  WRITE(1,4) I+10
  60 CONTINUE
  IF(NOPTR) 82,80,81
  81 WXX=WY-20.0
  WXC=-130.0
  CALL TPLDT(WXC,WYC,0,0)
  WRITE(1,5) MN
  GO TO 82
  82 CALL TPLDT(-90.,0,250.,0,0,0)
  CALL VECTOR
  CALL WRITEY(2,0,90.,0,0,8HGAIN DB)
  80 CALL ALPHA
  WYY=WY-10.0
  CALL TPLDT(532.,0,WYY,0,0)
  WRITE(1,86)
  WYY=WY-20.0
  
```

```

PAGE 3                                PAGE 1

CALL TPLOT(500.0, WY, 0.0)
WRITE(1,83) N
WY= WY-20.0
CALL TPLOT(500.0, WY, 0.0)
WRITE(1,84) LAGH
WY= WY-20.0
CALL TPLOT(500.0, WY, 0.0)
WRITE(1,85) DT
400 CALL PAUSE
CALL ERASE

999 RETURN
83 FORMAT(5H ' ', I6)
84 FORMAT(5H LAGH= ', I6)
85 FORMAT(5HDT = ', E8.2, 4H SEC)
86 FORMAT(70H ' INPUT DATA *')
END

PAGE 1
IF(XL.GE.0.0) GO TO 70
M=XL-1.0
70 M=MAX/(10.0+M)+1.0
VX(3)=M
VX(2)=VX(3)/2.0
VX(1)=0.
XNDR=FLOAT(NS)+10.0+M
DO 10 I=1, LAGH
X(I)=X(I)/XNDR
10 CONTINUE
M=2
ML=MX
M=MX
GO TO 60
2 MAX=XL+1.0
XL=ALOG10(XMIN)
MINX=XL-1.0
M=MAX-MIN
M=M+1
DO 20 I=1, M+1
VX(I)=FLOAT(MINX)+FLOAT(I)*1.0
20 CONTINUE
DO 30 I=1, LAGH
XL=X(I)
X(I)=(ALOG10(XL)-FLOAT(MINX))/FLOAT(M)
30 CONTINUE
ML=0
M=0
GO TO 60
3 CDR=20.0
XL=CDR*ALOG10(XMAX)/10.0
M=XL+1.0
XL=CDR*ALOG10(XMIN)/10.0
MINX=XL-1.0
M=MAX-MIN
M=M+1
DO 40 I=1, M+1

```

```

PAGE 2                                PAGE 1
40 CONTINUE
VK(I)=(I+MINX-1)*10
FMHMK=10
FMH=MINX*10
DO 50 I=1,LASH1
XLX(I)
X(I)=(DOB*ALOG10(XL)-FMH)/FMH
50 CONTINUE
HL=0
MH=0
60 CONTINUE
RETURN
END

SUBROUTINE HCORD(NOPT,LAGH,DT,H,MS,MH,MS)
DIMENSION H(402),HS(10)
FLAG=LAGH
DO 10 I=1,LAGH
FI(I)
H(I)=FI/(2.0*DT*FLAG)
10 CONTINUE
IF(NOPT.LI.0) GO TO 30
A1M(LAGH)
CALL ETYP(A1,COEF,ZE,IZE)
M*ZE
HS(3)=COEF
IF(IZE.EQ.0) GO TO 200
HS(2)=COEF/2.0
HS(1)=0.
MHZ
DO 20 I=1,LAGH
H(I)=H(I)/M(LAGH)
20 CONTINUE
GO TO 60
30 XMH(LAGH)
XL=ALOG10(XM)
MAXH=XL+1.0
XMH(1)
XL=ALOG10(XM)
MINH=XL-1.0
MH=MAXH-MINH
MH1=MH+1
DO 40 I=1,MH1
HS(I)=I+MINH+1
40 CONTINUE
DO 50 I=1,LAGH
XLH(I)
H(I)=(ALOG10(XL)-FLOAT(MINH))/FLOAT(MH)
50 CONTINUE
MH=MH
MS=0
GO TO 60
200 WRITE(10,100)
60 RETURN
100 FORMAT(1H ,70HITS UNREASONABLE JOB. /)
END
    
```

```

SUBROUTINE ETYPE(X,Y,Z,MZ)
  MZ=1
  IF(X.EQ.0.) GO TO 100
  IF(X.LT.0.) ZZ=-1.0
  X=ABS(X)
  XL=ALOG10(XA)
  XL=ABS(XL)
  IF(XLA.GT.15.0) GO TO 101
  DO 10 I=1,11
    FI=I
    IF(XLA.GE.FI) GO TO 10
    FI=FI-1.0
  GO TO 20
  10 CONTINUE
  20 IE=1
  IF(XLA.GE.0.0) GO TO 30
  IE=IE-1
  30 Y=X/(10.0**IE)
  Z=IE
  GO TO 110
  ZZ=-1 X IS NEGATIVE
  ZZ=1 X IS POSITIVE
  ZZ=0 X IS UNREASONABLE
  100 Y=0.
  Z=0.
  MZ=1
  GO TO 110
  101 MZ=0
  110 CONTINUE
  RETURN
  END

PAGE 1
SUBROUTINE AXSEL(NOPTH,NOPTV)
  WRITE(I,J)
  ACCEPT 2,N1,N2
  NOPTH=N1
  IF(N1.EQ.2) NOPTH=-1
  IF(N2=2) 24,25,26
  24 NOPTV=1
  GO TO 22
  25 NOPTV=0
  GO TO 22
  26 NOPTV=-1
  22 RETURN
  1 FORMAT(70H** AXIS SCALING CONDITION **//70H X AXIS 1...LINEAR
  2PE IN THE SCALING NO.//70H Y AXIS 1...LINEAR 2...LOG//70H* TY
  2 FORMAT(213)
  END

SUBROUTINE FRFEE2
  COMMON K,IR,N,DELT,DDD(10)
  COMPLEX X,Y,XDET
  DIMENSION X(10,10),Y(10,10),ZD(10,10),C(402),AF(402,10),
  1 C1(402),C2(402),C3(402),C4(402)
  DIMENSION NAME1(3),NAME2(3),NAME3(3)
  DATA NAME1(1),NAME1(2),NAME1(3)/2HDD,2HDD,2HDD/
  DATA NAME2(1),NAME2(2),NAME2(3)/2HEE,2HEE,2HEE/
  DATA NAME3(1),NAME3(2),NAME3(3)/2HEE,2HEE,2HEE/
  CALL BEGIN(9999,1)
  CALL ERASE
  MJ=10
  IP=K
  IL=IP-IR
  IP1=IP+1
  IR1=IR+1
  IR2=IR+IR
  IRR1=IR2+1
  IRR2=IR+IR+1
  IRS1=IR+IR2+1
  DDD(8)=10
  LAGH1=DDD(2)
  LAGH2=LAGH1+1
  LAGH=LAGH1-1
  WRITE(I,J)
  1 FORMAT(70H** FREQ. RESP. FN. OF FEEDBACK LOOP//70H IF UNEXECUTE
  2 THE JOB, TYPE IN 991//)
  ACCEPT 2,ISEL
  FORMAT(15)
  IF(ISEL.EQ.99) GO TO 999
  WRITE(I,J)
  3 FORMAT(//70H** IF PRINT OUT THE GAIN, PHASE, TYPE IN 1//)
  ACCEPT 2,IPR
  CALL GMOND(22,121,2)
  CALL GMOND(21,121,NAME1)
  CALL GMOND(22,122,2)
  CALL GMOND(21,122,NAME2)
  IF(IL.EQ.0) GO TO 200
  DO 100 I=1,LAGH1
    CALL GMOND(17,121,C(I),C(I,RS1))
  200 DO 300 I=1,LAGH1
    CALL GMOND(17,121,C(I),C(I,RS1))
  DO 310 II=1,IR
    IX=(II+1)*IR2
  310 JJ=1,IR
    JJ=IX+JJ
    J2=IX+IR+JJ
    GR=C(J1)
    GT=C(J2)
    X(II,JJ)=CMPLX(GR,GI)
  CONTINUE
  CALL INVDEI(X,XDET,IR,MJ)
  DO 320 II=1,IR
    XDET=X(II,II)
  320 JJ=1,IR
    Y(II,JJ)=X(II,JJ)/XDET
  END
  
```

```

PAGE 2 C ** FREQ. RESP. FN. OF FEEDBACK SYSTEMS DERIVED FROM A-R MODEL PAGE 1 C *** SUBROUTINE TO COMPUTE THE INVERSE AND DETERMINANT OF X ***
320 CONTINUE
DO 340 I=1,IR
IA=(I-1)*2+IR*1
DO 340 JJ=1,IR
IJ=(JJ-1)*2+IA
IJ=IJ*1
C(IJ)=REAL(Y(I,JJ))
C(IJ1)=AIMAG(Y(I1,JJ))
340 CONTINUE
IRR2=2*IR*IR*1
CALL GMOND(16,'22,C(1),C(IRR2))
CONTINUE
CALL GMOND(22,'23,NAME3)
DO 400 I=1,IR
DO 400 JJ=1,IR
CALL GMOND(22,'22,2)
CALL GMOND(21,'22,NAME2)
DO 410 IT=1,LAGH1
CALL GMOND(17,'22,C(1),C(IRR2))
DO 420 I=1,IR
IA=(I-1)*2+IR*1
DO 420 J=1,IR
IJ=(JJ-1)*2+IA
IJ=IJ*1
IF(J,NE,IJ) GO TO 420
C(IJ)=C(IJ1)
C2(IJ)=C(IJ1)
CONTINUE
420 CONTINUE
410 CONTINUE
CALL CRSSPEC(1,C2,C3,C4,LAGH)
IF(IPRT,NE,1) GO TO 440
WRITE(16,4) I1,IJ
4 FORMAT(1H1,/,7H ** FREQ. RESP. FN. OF FEEDBACK SYSTEM F(1,12,1H,
1,12,1H)/,3X,1H1,12X,2HHz,11X,3HRAD,10X,4HGAIN,6X,8HGAIN(DB),9X,5HP
2HASE/)
DO 430 IT=1,LAGH1
FIT=1
FLAG=LAGH
HZ=FIT/2,*FLAG*DELTA
RAD=*.3,141593*HZ
CC3=C3(IT)
GAINDB=20,+ALOG10(CC3)
I=IT*1
WRITE(16,5) I1,Hz,RAD,C3(IT),GAINDB,C4(IT)
5 FORMAT(1H ,14,5E14,4)
CONTINUE
440 CALL GMOND(16,'23,C3(1),C3(LAGH2))
400 CONTINUE
DOO(1)=20
DOO(2)=IR
DOO(3)=IR
999 RETURN
END
SUBROUTINE INVDEI(X,XDET,MM,MJ)
COMPLEX X,XDET,XMAXP,XC
DIMENSION X(MJ,MJ)
DIMENSION IDS(10)
XDET=1.0
DO 10 L=1,MM
PIVOTING AT L-TH STAGE
XMAXP=0.10000E+8
MAXI=0
DO 110 I=L,MM
FOR COMPLEX VERSION NEXT STATEMENT SHOULD BE REPLACED BY
1 IF(CABS(XMAXP).GE.CABS(X(I,L))) GO TO 110
XMAXP=X(I,L)
MAXI=I
110 CONTINUE
IDS(L)=MAXI
IF(MAXI.EQ.L) GO TO 120
IF(MAXI.GT.0) GO TO 121
XDET=X(0)
GO TO 140
C ROW INTERCHANGE
121 DO 14 J=L,MM
XC=X(MAXI,J)
X(MAXI,J)=X(L,J)
14 X(L,J)=XC
XDET=XDET
120 XDET=XDET*XMAXP
XC=1.0/XMAXP
X(L,L)=1.0
DO 11 J=L,MM
DO 12 I=L,MM
IF(I.EQ,L) GO TO 12
XC=X(I,L)
X(I,L)=0
DO 13 J=L,MM
13 X(I,J)=X(I,J)+XC*X(L,J)
12 CONTINUE
10 CONTINUE
10 CONTINUE
IF(MH.GT.1) GO TO 123
GO TO 140
C COLUMN INTERCHANGE
123 MM1=MM-1
DO 130 J=L,MM1
MHJ=MM-J
JJ=IDS(MHJ)
IF(JJ.EQ,MMJ) GO TO 130
DO 131 I=L,MM
XC=X(I,JJ)
X(I,JJ)=X(I,MHJ)
131 X(I,MHJ)=XC
130 CONTINUE
140 RETURN
END

```

PAGE 1

```

SUBROUTINE CRSSPE(P1,P2,P3,P4,LAGH)
DIMENSION P1(502),P2(402),P3(402),P4(402)
LAGH=LAGH*1
DO 1 I=1,LAGH
  XPI(I)=2+P2(I)**2
  P3(I)=SQRT(X)
1 CONTINUE
CALL SPHASE(P1,P2,P4,LAGH)
DO 2 I=1,LAGH
  P4(I)=P4(I)+100./3./3.,1415926536
2 CONTINUE
RETURN
END

```

PAGE 1

```

SUBROUTINE SPHASE(C,S,PH,LAGH)
DIMENSION C(LAGH),S(LAGH),PH(LAGH),ARC(501)
CALL SGLARC(C,S,ARC,LAGH)
CALL SGLPAC(ARC,PH,LAGH)
RETURN
END

```

PAGE 1

```

SUBROUTINE SGLPAC(ARC,PH,LAGH)
DIMENSION ARC(LAGH),PH(LAGH)
PI=3.1415926536
PI2=PI*2.0
PH(I)=ARC(I)
DO 10 I=2,LAGH
  OK=ARC(I)-ARC(I-1)
  IF(OK.GT.PI) GO TO 11
  IF(OK.LT.-PI) GO TO 12
  PH(I)=PH(I-1)+OK
GO TO 10
11 PH(I)=PH(I-1)+OK+PI2
GO TO 10
12 PH(I)=PH(I-1)+OK+PI2
10 CONTINUE
RETURN
END

```

```

SUBROUTINE SGLARC(C,S,ARC,LAGH)
DIMENSION C(LAGH),S(LAGH),ARC(LAGH)
PI=3.1415926536
DO 10 I=1,LAGH
  IF(C(I)) 11,12,13
11 IF(S(I)) 14,15,16
12 IF(S(I)) 17,18,19
13 ARC(I)=ATAN(S(I))/C(I)
GO TO 10
14 ARC(I)=ATAN(S(I))/C(I)+PI
GO TO 10
15 ARC(I)=PI
GO TO 10
16 ARC(I)=ATAN(S(I))/C(I)+PI
GO TO 10
17 ARC(I)=PI/2.0
GO TO 10
18 ARC(I)=0.
GO TO 10
19 ARC(I)=PI/2.0
10 CONTINUE
RETURN
END

```