NRL Report 8392

Analysis of Conventional and Reflective Butler Matrices with Imperfect Components

J. P. Shelton and J. K. Hsiao

Target Characteristics Branch
Radar Division

March 18, 1980

NAVAL RESEARCH LABORATORY
Washington, D.C.

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Report 8392	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) ANALYSIS OF CONVENTIONAL AND REFLECTIVE BUTLER MATRICES WITH IMPERFECT COMPONENTS		5. TYPE OF REPORT & PERIOD COVERED Interim report on a continuing NRL problem. 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) J.P. Shelton and J.K. Hsiao		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, DC 20375		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61153N; RR021-05-41; 53-0624-0-0
11. CONTROLLING OFFICE NAME AND ADDRESS Department of the Navy Office of Naval Research Arlington, VA 22217		12. REPORT DATE March 18, 1980 13. NUMBER OF PAGES 45
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Multibeam antenna
Butler matrix
Array antenna
Multiple beam form
Network

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In a reflective Butler matrix the input ports are coincident with the output ports. Hence, all error components produced by imperfect hybrid couplers accumulate at the single set of input/output ports, rather than being distributed at the separate sets of input ports and output ports as in the conventional Butler matrix. This report gives a precise scattering analysis of both conventional and reflective Butler matrices made up of imperfect hybrid couplers, and the effect on the performance of such matrices is computed. A computer program for carrying out the analysis is also given.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS

INTRODUCTION 1

SCATTERING MATRIX OF A 3-dB HYBRID COUPLER 1

SCATTERING AND TRANSFER MATRICES OF A
BUTLER NETWORK 3

PATTERNS OF AN ARRAY FED BY
A BUTLER NETWORK 8

CONCLUSIONS 11

REFERENCES 11

APPENDIX — Computer Program for Analysis 17

ANALYSIS OF CONVENTIONAL AND REFLECTIVE BUTLER MATRICES WITH IMPERFECT COMPONENTS

INTRODUCTION

A Butler matrix that forms a cluster of beams evenly distributed in the sin-θ space is not usually symmetric with respect to a plane midway between the input and output ports. However, by properly adjusting the phase shifts and interconnections one may modify a conventional Butler matrix to be symmetric. Such a matrix may also be folded on itself on the line of symmetry, so that the input and output ports are identical. Such a network not only reduces the number of components required; it also becomes a reflection-type system in which the feed positions are in the plane of the aperture. The synthesis of this network was described previously [1,2]. In this report, we analyze the performance of both conventional and reflective Butler matrices. In particular, we investigate the effect of reflected waves on the beam-forming performance. In a conventional Butler matrix, since the input and output ports are separate, the reflected waves emerging from the input ports have no effect on the beam-forming performance. Multiply reflected waves may emerge from output ports; however, their amplitudes are generally small, and their effects are relatively insignificant. In a reflective Butler matrix, the reflected waves accumulate at the input/output ports; hence, the aperture distribution at the antenna array is significantly modified, and this may degrade the beam-forming performance. These effects are investigated, and computer simulated results are presented together with a listing of the computer program.

SCATTERING MATRIX OF A 3-dB HYBRID COUPLER

The basic building block of a Butler matrix is a 3-dB hybrid coupler. For the ideal hybrid coupler, energy fed into any one of the input ports will be split into two equal components, one with a phase shift of 90° relative to the other. However, practical hybrid couplers will in general exhibit amplitude and phase errors in their transfer coefficients. These amplitude and phase errors will affect the transfer coefficients of both reflective and conventional Butler matrices in the same way. That is, the errors in the overall network input/output transfer coefficients will be the same for both conventional and reflective networks. Practical hybrid couplers will also have nonzero reflection and transfer coefficients to the isolated port. For the conventional network, to a first order, the error components due to these effects will appear at the network inputs. For the reflective network, with its inputs and outputs sharing a single set of ports, all error components affect the input/output transfer coefficients.

Thus, the two types of hybrid coupler errors are forward and reverse. Our investigation will be concentrated on the reverse-error components, and we shall assume that there

Manuscript submitted January 4, 1980.

is no amplitude or phase error in the forward-transfer coefficients of the 3-dB coupler. The following analysis is based on the assumption that, when an incident wave of unit amplitude is applied to one of the input ports, two waves of amplitude α will emerge from the two output ports, one with a 90° phase shift and the other with no phase shift. Similarly, waves of amplitude β will be reflected to the two input ports. As shown in Fig. 1(a), when an incident wave of unit amplitude is applied at port 12, reflected waves of $-\beta$ and $-j\beta$ appear at ports 11 and 12 respectively and waves of $-j\alpha$ and α appear at ports 21 and 22. For conservation of energy, one has

$$2\alpha^2 + 2\beta^2 = 1. \tag{1}$$

The isolation factor is defined as the power ratio of the reflected wave to the incident wave. In this case, the isolation is

$$I = \beta^2. \tag{2}$$

Accordingly, in terms of the isolation factor,

$$\alpha = \sqrt{0.5 - I}\ . \tag{3}$$

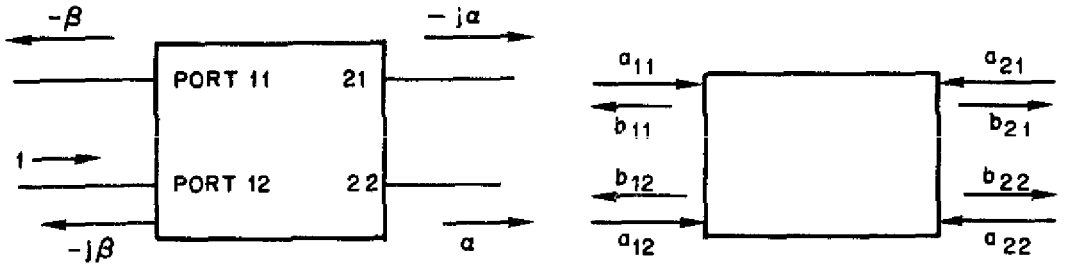
If the parameters in Fig. 1(b) are used, the reflected waves are related to the incident waves by the matrix equation

$$\begin{bmatrix} b_{11} \\ b_{12} \\ \hline b_{21} \\ b_{22} \end{bmatrix} = \left[\begin{array}{cc|cc} -j\beta & -\beta & \alpha & -j\alpha \\ -\beta & -j\beta & -j\alpha & \alpha \\ \hline \alpha & -j\alpha & -j\beta & \beta \\ -j\alpha & \alpha & -\beta & -j\beta \end{array}\right] \begin{bmatrix} a_{11} \\ a_{12} \\ \hline a_{21} \\ a_{22} \end{bmatrix}, \tag{4}$$

where a_{11}, a_{12}, a_{21}, and a_{22} are incident waves and b_{11}, b_{12}, b_{21}, and b_{22} are scattered waves at ports 11, 12, 21, and 22 respectively.

Let

$$\begin{aligned} \mathbf{b}_1 &= \begin{bmatrix} b_{11} \\ b_{12} \end{bmatrix}, \mathbf{b}_2 = \begin{bmatrix} b_{21} \\ b_{22} \end{bmatrix}, \\ \mathbf{a}_1 &= \begin{bmatrix} a_{11} \\ a_{12} \end{bmatrix}, \mathbf{a}_2 = \begin{bmatrix} a_{21} \\ a_{22} \end{bmatrix}, \end{aligned} \tag{5a}$$



(a) Unit-amplitude wave incident at one of the input ports of a 3-dB coupler

(b) Incident and reflected waves on a 3-dB coupler

Fig. 1 — Transfer and reflection in four-port networks

and

$$S_{11} = S_{22} = \begin{bmatrix} -j\beta & -\beta \\ -\beta & -j\beta \end{bmatrix},$$
$$S_{12} = S_{21} = \begin{bmatrix} \alpha & -j\alpha \\ -j\alpha & \alpha \end{bmatrix}. \tag{5b}$$

Matrix Eq. (4) can now be simplified to the form

$$\begin{bmatrix} b_1 \\ \hline b_2 \end{bmatrix} = \left[\begin{array}{c|c} S_{11} & S_{12} \\ \hline S_{21} & S_{22} \end{array}\right] \begin{bmatrix} a_1 \\ \hline a_2 \end{bmatrix}. \tag{6}$$

SCATTERING AND TRANSFER MATRICES OF A BUTLER NETWORK

A Butler network can be represented by a block diagram as shown in Fig. 2.* Blocks in regions 1 and 3 represent the 3-dB couplers described in the previous section, and a phase-shift transfer network is located in region 2. A number of similar networks are

*For the remainder of this report, a network will be considered a physical entity and a matrix a mathematical entity.

connected in cascade to form a complete conventional Butler network. The scattering matrix for regions 1 and 3 is

$$\begin{bmatrix} b_{11} \\ b_{12} \\ \cdot \\ \cdot \\ \cdot \\ b_{1n} \\ b_{21} \\ b_{22} \\ \cdot \\ \cdot \\ \cdot \\ b_{2n} \end{bmatrix} = \begin{bmatrix} -j\beta & -\beta & 0 & 0 & \dots & \alpha & -j\alpha & 0 & 0 & \dots 0 \\ -\beta & -j\beta & 0 & 0 & \dots & -j\alpha & \alpha & 0 & 0 & \dots 0 \\ 0 & 0 & -j\beta & -\beta & 0\ 0 & \dots & & \alpha & -j\alpha & 0 \dots \\ 0 & 0 & -\beta & -j\beta & 0\ 0 & \dots & & -j\alpha & \alpha & 0 \dots \\ \dots & & \dots & & \dots & \dots & & \dots & & \dots \\ \alpha & -j\alpha & 0 & 0 & \dots & -j\beta & -\beta & 0 & 0 & \dots 0 \\ -j\alpha & \alpha & 0 & 0 & \dots & -\beta & -j\beta & 0 & 0 & \dots 0 \\ 0 & 0 & \alpha & -j\alpha & 0\ 0 & \dots & & j\beta & -\beta & \dots 0 \\ 0 & 0 & -j\alpha & \alpha & \dots & \dots & & -\beta & -j\beta & \dots 0 \\ \dots & & \dots & & \dots & \dots & & \dots & & \dots \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ \cdot \\ \cdot \\ \cdot \\ a_{1n} \\ a_{21} \\ a_{22} \\ \cdot \\ \cdot \\ \cdot \\ a_{2n} \end{bmatrix} . \tag{7}$$

Define

$$\mathbf{b}_1 = \begin{bmatrix} b_{11} \\ b_{12} \\ \cdot \\ \cdot \\ \cdot \\ b_{1n} \end{bmatrix}, \mathbf{b}_2 = \begin{bmatrix} b_{21} \\ b_{22} \\ \cdot \\ \cdot \\ \cdot \\ b_{2n} \end{bmatrix}, \tag{8a}$$

$$\mathbf{a}_1 = \begin{bmatrix} a_{11} \\ a_{12} \\ \cdot \\ \cdot \\ \cdot \\ a_{1n} \end{bmatrix}, \mathbf{a}_2 = \begin{bmatrix} a_{21} \\ a_{22} \\ \cdot \\ \cdot \\ \cdot \\ a_{2n} \end{bmatrix}, \tag{8b}$$

$$\mathbf{S}_{11} = \mathbf{S}_{22} = \begin{bmatrix} -j\beta & -\beta & 0 & 0 & \dots & \dots \\ -\beta & -j\beta & 0 & 0 & \dots & \dots \\ 0 & 0 & -j\beta & -\beta & 0\ 0 & \dots \\ 0 & 0 & -\beta & -j\beta & 0\ 0 & \dots \\ \dots & & \dots & & \dots & \dots \\ 0 & 0 & \dots & & \dots & -j\beta \quad -\beta \\ 0 & 0 & \dots & & \dots & -\beta \quad -j\beta \end{bmatrix}, \tag{8c}$$

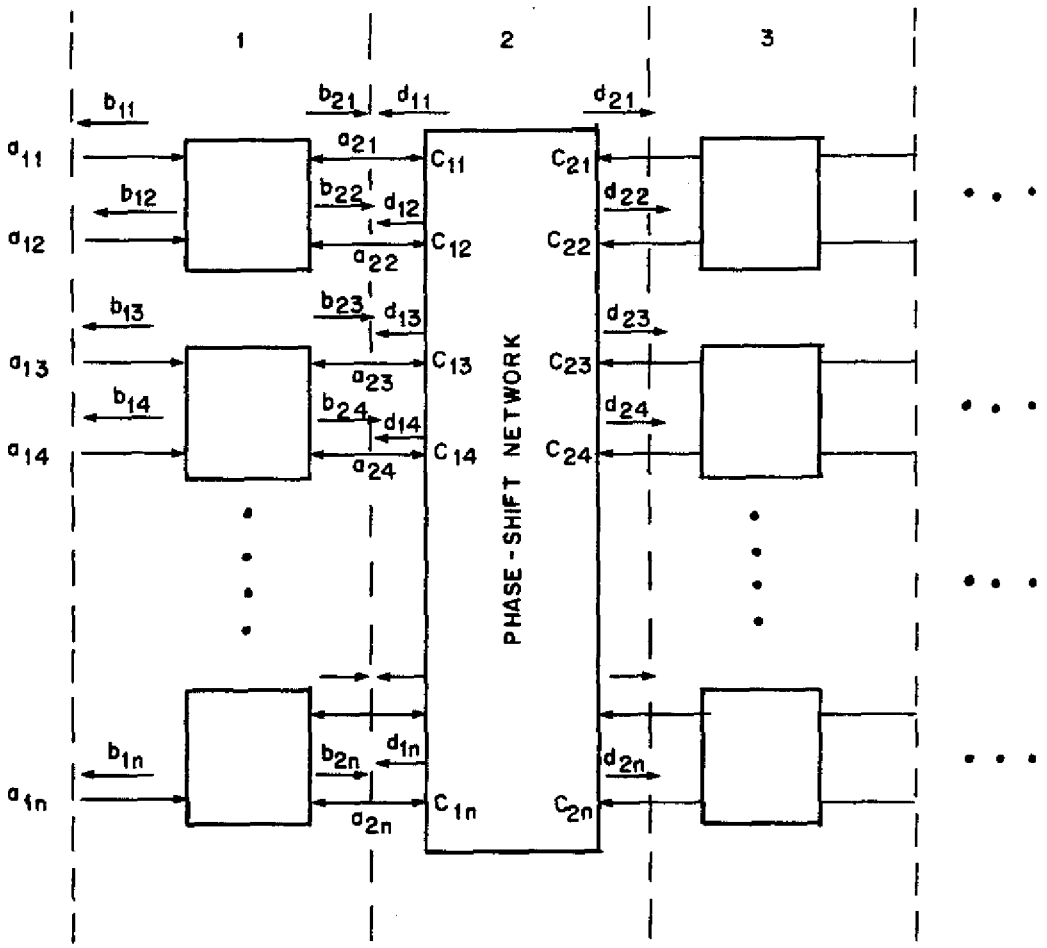


Fig. 2 — Block diagram of a Butler network

and

$$\mathbf{S}_{12} = \mathbf{S}_{21} = \begin{bmatrix} \alpha & -j\alpha & 0 & 0 & \cdots & \cdots \\ -j\alpha & \alpha & 0 & 0 & \cdots & \cdots \\ 0 & 0 & \alpha & -j\alpha & 0\ 0 & \cdots \\ 0 & 0 & -j\alpha & \alpha & 0\ 0 & \cdots \\ \cdots & & \cdots & & \cdots & \cdots \\ 0 & 0 & \cdots & & \cdots & \alpha \quad -j\alpha \\ 0 & 0 & \cdots & & \cdots & -j\alpha \quad \alpha \end{bmatrix} . \tag{8d}$$

Equation (7) can now be simplified to

$$\begin{bmatrix} \mathbf{b}_1 \\ \hline \mathbf{b}_2 \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{S}_{11} & \mathbf{S}_{12} \\ \hline \mathbf{S}_{21} & \mathbf{S}_{22} \end{array}\right] \begin{bmatrix} \mathbf{a}_1 \\ \hline \mathbf{a}_2 \end{bmatrix} . \tag{9}$$

The scattering matrix in region 2, which is a phase-shift and transfer network, can be represented as

$$\begin{bmatrix} \mathbf{d}_1 \\ \hline \mathbf{d}_2 \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{R}_{11} & \mathbf{R}_{12} \\ \hline \mathbf{R}_{21} & \mathbf{R}_{22} \end{array}\right] \begin{bmatrix} \mathbf{c}_1 \\ \hline \mathbf{c}_2 \end{bmatrix} , \tag{10}$$

where $\mathbf{d}_1$, $\mathbf{d}_2$, $\mathbf{c}_1$, and $\mathbf{c}_2$ are vectors such that

$$\mathbf{d}_1 = \begin{bmatrix} d_{11} \\ d_{12} \\ \cdot \\ \cdot \\ \cdot \\ d_{1n} \end{bmatrix} , \mathbf{d}_2 = \begin{bmatrix} d_{21} \\ d_{22} \\ \cdot \\ \cdot \\ \cdot \\ d_{2n} \end{bmatrix} , \tag{11a}$$

$$\mathbf{c}_1 = \begin{bmatrix} c_{11} \\ c_{12} \\ \cdot \\ \cdot \\ \cdot \\ c_{1n} \end{bmatrix} , \mathbf{c}_2 = \begin{bmatrix} c_{21} \\ c_{22} \\ \cdot \\ \cdot \\ \cdot \\ c_{2n} \end{bmatrix} . \tag{11b}$$

Matrices $\mathbf{R}_{11}$ and $\mathbf{R}_{22}$ are zero, and matrices $\mathbf{R}_{12}$ and $\mathbf{R}_{21}$ have identical elements. These matrices describe the phase shifts and interconnections from one row of couplers to the

next. Their elements depend on the configuration of the Butler network. As an example, the R matrix of the 4-port Butler network shown in Fig. 3 is

$$\mathbf{R}_{12} = \mathbf{R}_{21} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & e^{-j\frac{\pi}{4}} & 0 \\ 0 & e^{-j\frac{\pi}{4}} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} . \tag{12}$$

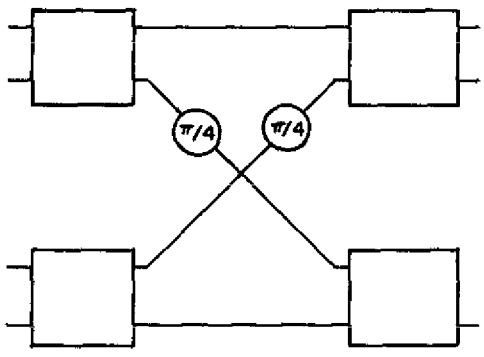


Fig. 3 — Four-port Butler network

Since we are interested in the overall scattering matrix of this network, we must first convert the scattering matrix in each region to a transfer matrix, which in turn can be multiplied to form the overall transfer matrix of the whole network. A transfer matrix can be represented as

$$\begin{bmatrix} \mathbf{b}_2 \\ \hline \mathbf{a}_2 \end{bmatrix} = \left[\begin{array}{c|c} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \hline \mathbf{T}_{21} & \mathbf{T}_{22} \end{array}\right] \begin{bmatrix} \mathbf{a}_1 \\ \hline \mathbf{b}_1 \end{bmatrix} , \tag{13}$$

where $\mathbf{a}_1$ and $\mathbf{b}_1$ are the incident and reflected waves at the left hand ports and $\mathbf{a}_2$ and $\mathbf{b}_2$ are similar waves at the right hand ports.

It can be shown that a matrix T is related to an S matrix by the following relations [3,4]:

$$\mathbf{T}_{11} = \mathbf{S}_{21} - \mathbf{S}_{22}\, \mathbf{S}_{12}^{-1}\, \mathbf{S}_{11}, \tag{14a}$$

$$\mathbf{T}_{12} = \mathbf{S}_{22}\, \mathbf{S}_{12}^{-1}, \tag{14b}$$

$$T_{21} = -S_{12}^{-1} S_{11}, \tag{14c}$$

and

$$T_{22} = S_{12}^{-1}. \tag{14d}$$

The overall transfer matrix is

$$T = \prod_{i=1}^{k} T_i \tag{15}$$

where $T_1, T_2, ..., T_k$ are transfer matrices in regions $1, 2, ..., k$.

The overall transfer matrix can be converted to a scattering matrix by the relations

$$S_{11} = -T_{22}^{-1} T_{21}, \tag{16a}$$

$$S_{12} = T_{22}^{-1}, \tag{16b}$$

$$S_{21} = T_{11} - T_{12} T_{22}^{-1} T_{21}, \tag{16c}$$

and

$$S_{22} = T_{12} T_{22}^{-1}. \tag{16d}$$

Since $S_{12} = S_{21}$, one may use the simpler relation of Eq. (16b) instead of Eq. (16c).

Elements of matrix S_{21} (or S_{12}) represent the transmitted waves at the output ports when a unit incident wave is applied at any one of the input ports. Therefore, matrix S_{21} is the transfer function of a conventional Butler network. Elements of matrix S_{11} (or S_{22}) represent the reflected waves at the input ports when a unit incident wave is applied at any one of the input ports. In a reflective Butler network both the reflected waves and transmitted waves emerge from the same set of ports. Therefore, the scattering matrix of such a network is the sum of matrices S_{12} and S_{11}, or

$$S = S_{11} + S_{12}. \tag{17}$$

In deriving this relation, we have made the assumption that the symmetry plane of a reflective Butler network exhibits an open-circuit unity reflection coefficient.

PATTERNS OF AN ARRAY FED BY A BUTLER NETWORK

Figure 4 shows a schematic diagram of a reflective Butler network, which has half the components of a conventional Butler network. There are n ports, since ports a_{11},

$a_{12}, ..., a_{1n}$ are identical with ports $a_{21}, a_{22}, ..., a_{2n}$. Using previously developed notation and setting $[b_2] = [a_2] = 0$, this can be represented as

$$[b_1] = [S_{11} + S_{12}][a_1]. \tag{18}$$

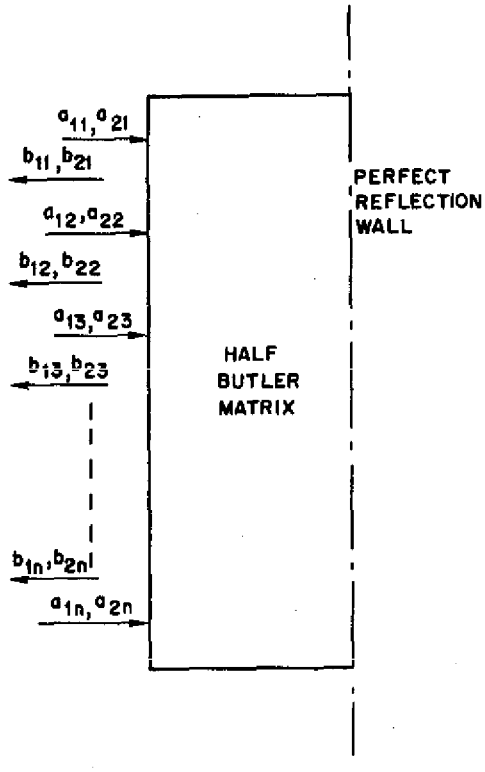


Fig. 4 — Reflective Butler network

The vector input of $[a_1]$ can be represented, for the case of an incident plane wave received by a linear array, by

$$a_{1k} = A_k \exp [j(k-1)u] \tag{19}$$

where $u = 2\pi d \sin \theta / \lambda$,

with λ = wavelength,
θ = angle of incidence from the normal to the array, and
d = element spacing.

In the subsequent discussion, we shall assume that the array has a uniform illumination function, that is, that $A_k = 1$. The scattering matrix $[S_{11}] + [S_{12}]$ is computed as a function of isolation factor I. Radiation patterns of the network-fed array are represented by two types of plot. One shows the main beams formed by several ports of the reflective Butler network, and the other shows the complete array pattern of one port of the network, in the range $0 \leqslant u \leqslant 180°$.

Figure 5 shows the array patterns of an eight-port reflective Butler network. Figure 5a shows four of the main beams for variation of the isolation factor of the 3-dB hybrid from 10 dB to 40 dB. Figure 5b shows the array pattern when the main beam is at $u = 22.5°$ for the same range of isolation factor. Figure 6 shows the corresponding patterns for a 16-port reflective network. From these figures, it can be seen that the null filling level is roughly equal to the isolation factor of the 3-dB couplers. That is, for the case of 10-dB isolation, the pattern is filled to a level of about 10 dB below its peak; and for the case of 40 dB isolation, the pattern is filled to a level of about 40 dB below its peak.

Tables 1 and 2 show computed results for eight-port and 16-port reflective Butler networks, respectively. The isolation factors in dB are listed in the first column. The transmitted power is the percentage of incident power, averaged over all inputs and outputs, that would emerge from the outputs for the conventional Butler-network configuration. The remaining power emerges from the input ports. It is seen that the transmitted power decreases as the isolation decreases and as the number of rows of couplers in the network increases. For the reflective-network configuration, the input and output ports are combined, and the components emerging from these ports are also combined. The RMS amplitude and phase errors describe the effects of these spurious components on the combined outputs and are defined by

$$\Delta b = \left[\frac{\sum_{k=1}^{N} \sum_{\ell=1}^{N} \left(|s_{k\ell}| - \bar{s} \right)^2}{N^2} \right]^{1/2}$$

and

$$\Delta \phi = \left[\frac{\sum_{k=1}^{N} \sum_{\ell=1}^{N} \left(\phi_{k\ell} - \phi'_{k\ell} \right)^2}{N^2} \right]^{1/2},$$

where Δb and $\Delta \phi$ are the RMS amplitude and phase errors, respectively, $s_{k\ell}$ is an element of the scattering matrix S,

$$\bar{s} = \sum_{k=1}^{N} \sum_{\ell=1}^{N} |s_{k\ell}| / N^2,$$

$\phi_{k\ell}$ is the phase of $s_{k\ell}$, and $\phi'_{k\ell}$ is the phase of $s_{k\ell}$ for the ideal network with no errors. The error components increase with the number of rows of couplers and with decreasing isolation.

A computer program for carrying out these calculations is listed in the appendix. In addition to providing for imperfect reverse parameters of the hybrid couplers, the program provides for imperfect forward parameters and for errors in the interconnecting transmission lines.

CONCLUSIONS

An exact analysis procedure has been developed that is applicable to both conventional and reflective Butler networks with imperfect components. The analytical procedure has been programmed for computation of results for conventional and reflective Butler networks of arbitrary size. Results are presented for eight-port and 16-port reflective networks using hybrid couplers with varying degrees of isolation. The results are given in the form of radiation-pattern factors that would be obtained from a linear antenna array fed by the network and also in terms of the RMS phase and amplitude errors of the network transfer coefficients.

REFERENCES

1. J.P. Shelton and J.K. Hsiao, "Reflective Butler Matrices," NRL Report 8188, Mar. 14, 1978.
2. J.P. Shelton and J.K. Hsiao, "Reflective Butler Matrices," IEEE Trans. Antennas Propag. AP-27, 651-659 (Sept. 1979).
3. L.B. Felsen and W.K. Kahn, "Transfer Characteristics of 2N-Port Networks," pp. 477-512 in *Proceedings of the Symposium on Millimeter Waves* (1959), Polytechnic Institute of Brooklyn Microwave Research Institute Symposia Series, Vol. 9, Interscience Publishers, New York, 1960.
4. W.K. Kahn, "Impedance-Match and Element-Pattern Constraints for Finite Arrays," NRL Report 8002, June 1976.

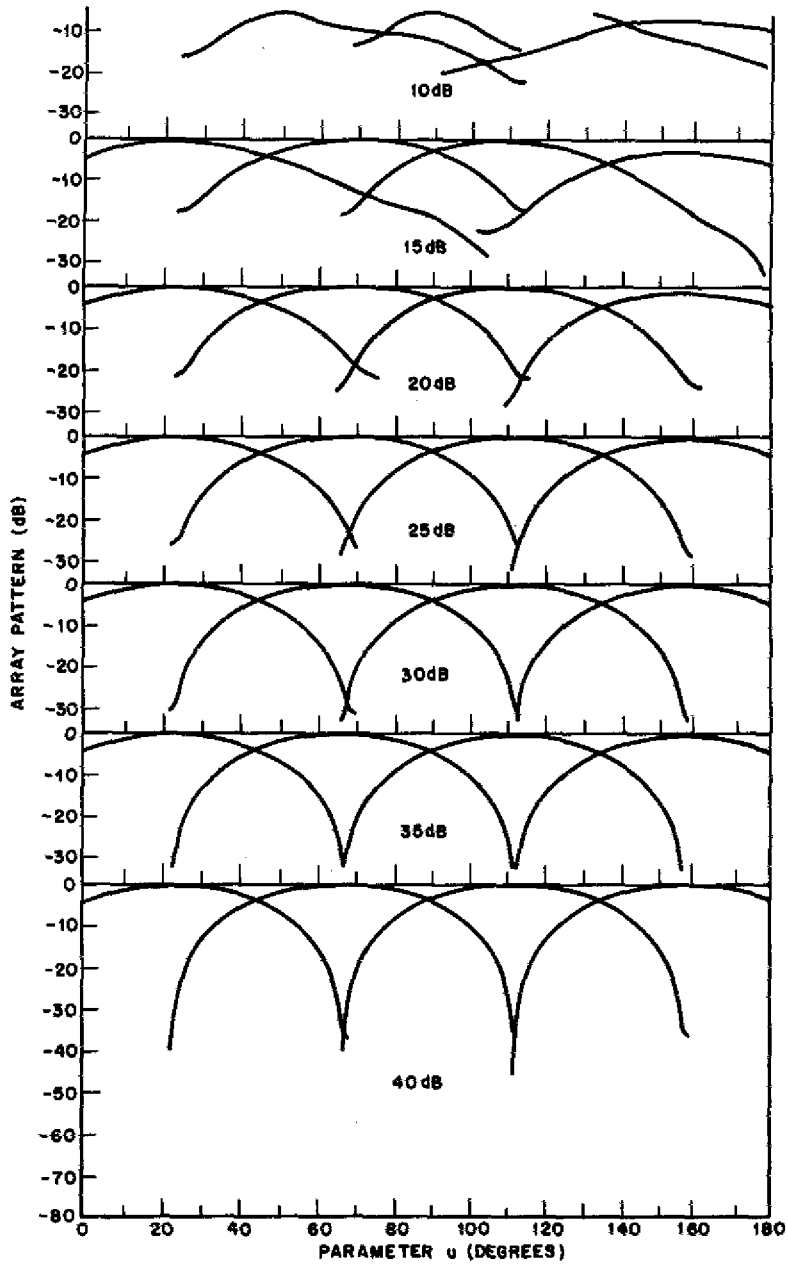


Fig. 5a — Main-beam pattern of an eight-port reflective Butler network; isolation factor varies from 10 dB to 40 dB

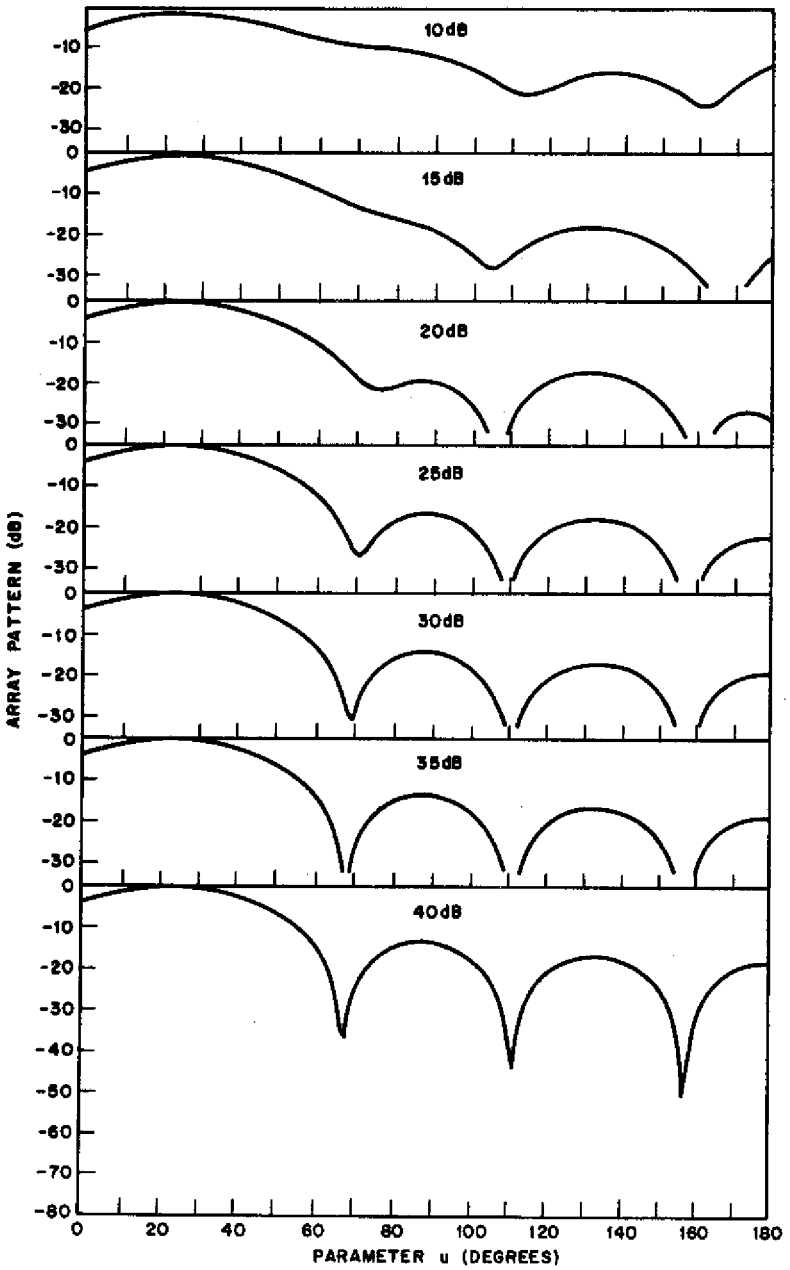


Fig. 5b — Array pattern of an eight-port reflective Butler network; isolation factor varies from 10 dB to 40 dB; main beam at $u = 22.50°$

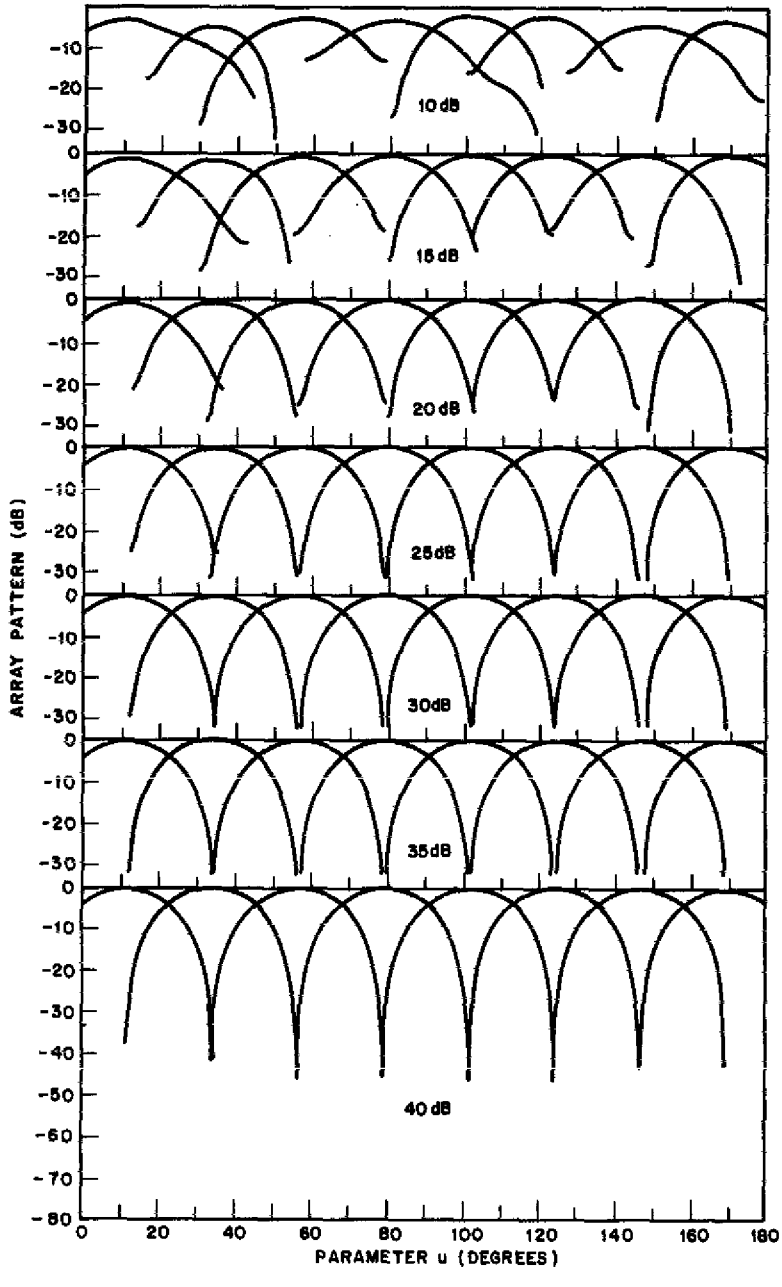


Fig. 6a — Main-beam pattern of a 16-port reflective Butler network; isolation factor varies from 10 dB to 40 dB

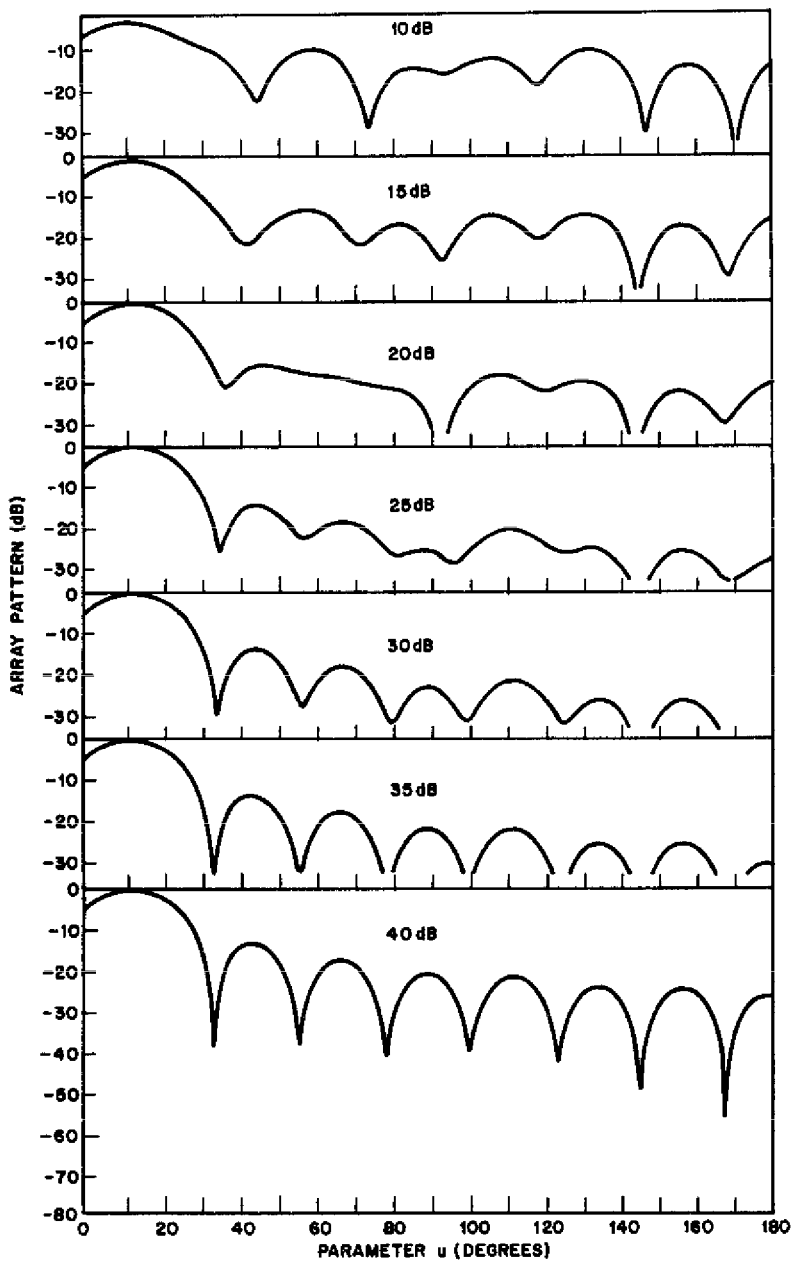


Fig. 6b — Array pattern of a 16-port reflective Butler network; isolation factor varies from 10 to 40 dB; main beam at $u = 11.25^{\circ}$

Table 1 — Computed Statistical Parameters for Eight-Port Reflective Network

Isolation (dB)	Transmitted Power (percent of incident)	RMS Amplitude Error (percent)	RMS Phase Error (degrees)
10	54.85	30.25	38.69
15	80.76	19.42	23.53
20	93.20	12.30	12.97
25	97.77	7.30	7.16
30	99.29	4.21	3.99
35	99.77	2.39	2.24
40	99.93	1.35	1.25

Table 2 — Computed Statistical Parameters for 16-Port Reflective Network

Isolation (dB)	Transmitted Power (percent of incident)	RMS Amplitude Error (percent)	RMS Phase Error (degrees)
10	44.92	47.29	50.58
15	76.30	31.96	30.95
20	91.49	20.41	12.66
25	97.19	11.67	6.91
30	99.10	6.61	3.84
35	99.71	3.73	2.16
40	99.91	2.10	1.22

Appendix

COMPUTER PROGRAM FOR ANALYSIS

This computer program computes the coupling coefficients from the input ports to the output ports and the power transmitted and reflected; it also plots the array radiation pattern if it is desired. The type of Butler matrix analyzed by this program can be either a conventional or a reflective type as described in this report. For this program three input data cards are required. The first data card enters the following fixed-point (I5 format) data:

NPT — Number of ports of the Butler matrix to be computed.

NROW — Number of rows of this network.

KLL — Absolute value of KLL represents the beam index whose pattern is to be plotted. If KLL = 0, there is no plot. If KLL is less than 0, the program plots the array pattern and also plots all main beams formed by the Butler matrix network.

LPRINT — Printout control. If LPRINT = 0, the program prints all detailed output at each computation step.

The second data card, which is also in a fixed-point I5 format, specifies the number of ports in each basic coupling network in each row. This implies that identical coupling networks are used in each row. However, coupling networks of different ports may be used in different rows.

The third input data card, which has a F10.6 floating-point format, specifies the coupling coefficients of the 3-dB coupler used as the basic building block of the Butler matrix network. These coefficients are read in the sequence A1, B1, C1, D1. These numbers are related to the coupling coefficient of the 3-dB coupler by the relations (see Fig. 1a)

$$\beta_1 = 10^{-(0.05 \times \mathrm{A1})},$$
$$\beta_2 = 10^{-(0.05 \times \mathrm{B1})},$$
$$\alpha_1 = 10^{-(0.05 \times \mathrm{C1})},$$

and

$$\alpha_2 = 10^{-(0.05 \times \mathrm{D1})}.$$

```
      PROGRAM RFBMTX
C     THIS PROGRAM FIRST FIGURES OUT BUTLER MATRIX CONNECTION AND PHASE
C     ANGLE , COMPUTES THE TRANSFER FUNCTION AND THEN PLOT THE PATTERN
C     MATRIX LIMIT TO THE SIZE OF 64
C     COMPILED ON JULY 13,1976 BY J. K. HSIAO
C     REVISED ON AUGUST 18,1976 BY J. K. HSIAO
C     ABSOLUTE VALUE OF KLL REPRESENTS THE BEAM INDEX WHOSE PATTERN IS
C     TO BE PLOTTED
C     KLL=0  NO PLOT
C     KLL GRATER THAN 0 PLOT PATTERN ONLY
C     KLL LESS THAN 0 PLOT BOTH PATTERN AND MAIN BEAMS
C     LLL=1, FULL MATRIX, LLL=0 REFLECTIVE MATRIX
C     LPRINT =0, PRINT ALL DETAILED OUTPUTS
C     IF LPRINT NOT EQUAL 0 NO MATRIX MULTIPLICATION RESULT IS PRINTED
C     IF LPRINT LT 0 PRINT ONLY THE TRANSFER FUNCTION
      COMMON/C$1/PLTAY(500)
      COMMON/C$4/A1,A2,B1,B2
      DIMENSION NBP(16),NBK(16)
      DIMENSION MC(8,64),PHA(8,64)
      DIMENSION S11(32,32),S12(32,32),S21(32,32),S22(32,32)
      COMPLEX S11,S12,S21,S22
      CALL PLOTS(PLTAY,500,0.)
      NMAX=32
      KC=0
    1 READ 100,NTP,NROW,KLL,LPRINT,LLL
      IF(NTP.EQ.0)GO TO 2
    3 READ 100,(NBP(I),I=1,NROW)
  100 FORMAT(16I5)
      READ 101, A1,A2,B1,B2
  101 FORMAT(8F10.6)
      IF(KC.GT.0)CALL ORIGIN(14.,0.)
      KC=KC+1
      NR1=NROW+1
      CALL NTWK(NTP,NR1,NBP,NBK,MC,PHA)
      IF(LLL.GT.0)GO TO 4
      CALL HLFMTX(NTP,NR1, NBP,NBK,MC,PHA)
    4 CALL TRFMTX(NMAX,NTP,NR1,NBP,NBK,MC,PHA,S11,S12,S21,S22)
      LL=0
      CALL PRTOUT(NTP,S21,S11,LL,NMAX,LTFP,LPRINT)
      LTFP=1
      IF(KLL.EQ.0)GO TO 1
      NPAV=1
      CALL PATERN (NTP,S21,S11,KLL,NPAV,NMAX)
      GO TO 1
    2 CALL ENDPLT
      END
```

```
0001          SUBROUTINE PRTOUT(NTP,TRFF,TRFB,LL,NMX,LTFP,LPRINT)
        C     LLGT.0 FOR BLOCK, AND LLIS THE BLOCK NUMBER
        C     LL=0 FOR OVERAL TRANSFER FUNCTION
0002          DIMENSION TRFF(NMX,NMX),TRFB(NMX,NMX)
0003          COMMON/C$4/A1,A2,B1,B2
0004          COMMON/C$6/AMPT(32,32),ANGL(32,32),ANGT(32),TRFF2(32,32), TR(32,32
             C),AMPAV(32),ANGAV(32),AMX(32),ANX(32),AMPRMS(32),ANGRMS(32),
             C  SUMR(1824)
0005          COMPLEX TRFF,TRFB,TRFF2,TR,SR
0006          KC=0
0007          PI=3.1415926536
0008          RAC=180./PI
0009          K6=6
0010          LLL=0
0011          IF(A1.LE.0..OR.LL.GT.0)LLL=1
0012          IF(LL.LE.0)GO TO 1
0013          PRINT 101,LL
0014     101  FORMAT(//,20X,'THIS IS THE TRANSFER FUNCTION OF BLOCK',I5)
0015          GO TO 4
0016     1    PRINT 111
0017          PRINT 106
0018     106  FORMAT(//,20X,'OVERAL TRANSFER FUNCTION')
0019     2    IF(A1.GT.0.)GO TO 3
0020          PRINT 119
0021     119  FORMAT(//,10X,'ZERO REFLECTION')
        C     GENERATE TRANSFER FUNCTION FOR AN IDEAL BUTLER MATRIX
0022    3     PRINT 124,NTP,A1
0023          PRINT 117
0024     124  FORMAT(/,20X,'NUMBER OF PORTS',I5,5X,'ISOLATION(DB)',F10.4,//)
0025          CALL TRFIDL(NTP)
0026          IF(LTFP.GT.0)GO TO 4
0027          IF(LPRINT .GT.0)GO TO 4
0028          PRINT 107,((AMPT(I,J),J=1,NTP),I=1,NTP)
0029          PRINT 117
0030          PRINT 107,((ANGL(I,J),J=1,NTP),I=1,NTP)
0031          PRINT 117
0032     4    IF(A1.LE.0.)K6=2
0033          DO 60 K=1,K6
0034          SUM=0.
0035          DO 15 I=1,NTP
0036     15   SUMR(I)=0.
0037          IF(LPRINT .GT.0)GO TO 75
0038          GO TO (71,72,73,74,76,77)K
0039     71   PRINT 102
0040     102  FORMAT(//,20X,'AMPLITUDE OF FORWARD TRANSFER FUNCTION')
0041          PRINT 117
0042     117  FORMAT(/)
0043          GO TO 75
0044     72   PRINT 103
0045     103  FORMAT(//,20X,'PHASE ANGLE OF FORWARD TRANSFER FUNCTION')
```

```
      PRINT 117
      GO TO 75
73    PRINT 104
104   FORMAT(//,20X,'AMPLITUDE OF REFLECTIVE TRANSFER FUNCTION')
      PRINT 117
      GO TO 75
74    PRINT 105
105   FORMAT(//,20X,'PHASE ANGLE OF REFLECTIVE TRANSFER FUNCTION')
      PRINT 117
      GO TO 75
76    PRINT 109
109   FORMAT(//,20X,'AMPLITUDE OF THE RESULTANT TRANSFER FUNCTION')
      PRINT 117
      GO TO 75
77    PRINT 110
110   FORMAT(//,20X,'PHASE ANGLE OF THE RESULTANT TRANSFER FUNCTION')
      PRINT 117
75    DO 67 I=1,NTP
      DO 70 J=1,NTP
      GO TO(61,62,63,64,65,66)K
61    ANGT(J)=CABS(TRFF(I,J))
      ANGT2=ANGT(J)**2
      SUM=SUM+ANGT2
      SUMR(J)=SUMR(J)+ANGT2
      GO TO 70
62    IF(LPRINT .GT.0)GO TO 70
      ANGT(J)=CANG(TRFF(I,J))*RAC
      GO TO 70
63    ANGT(J)=CABS(TRFB(I,J))
      ANGT2=ANGT(J)**2
      SUM=SUM+ANGT2
      SUMR(J)=SUMR(J)+ANGT2
      GO TO 70
64    IF(LPRINT .GT.0)GO TO 70
      ANGT(J)=CANG(TRFB(I,J))*RAC
      GO TO 70
65    TR(I,J)    =TRFF(I,J)+TRFB(I,J)
      ANGT(J)=CABS(TR(I,J))
      ANGT2=ANGT(J)**2
      SUMR(J)=SUMR(J)+ANGT2
      SUM=SUM+ANGT2
      IF(LLL.GT.0)GO TO 70
      AMPT(I,J)=(ANGT(J)-AMPT(I,J))/AMPT(I,J)
      GO TO 70
66    ANGT(J)=CANG(TR(I,J))*RAC
      IF(LLL.GT.0)GO TO 70
      AG        =ANGT(J)-ANGL(I,J)
      ANGL(J,I)=AG
      IF(ABS(AG).LE.180.)GO TO 70
      NSIGN=1
```

```
      IF(AG.GT.0.)NSIGN=-1
      ANGL(I,J)=NSIGN*(360.-ABS(AG))
70    CONTINUE
      IF(LPRINT .GT.0)GO TO 67
      PRINT 107,(ANGT(J),J=1,NTP)
107   FORMAT(10X, 8F10.4)
67    CONTINUE
      KMOD=MOD(K,2)
      IF(KMOD.LE.0)GO TO 60
      PRINT 122,SUM
122   FORMAT(//,10X,'TOTAL POWER OUTPUT',F10.4)
      PRINT 123,(SUMR(I),I=1,NTP)
123   FORMAT(//,10X,'POWER FROM EACH PORT',/,(10X,10F10.4))
60    CONTINUE
      IF(LLL.GT.0)GO TO 7
      IF(LPRINT .GT.0)GO TO 8
      DO 50 L=1,2
      GO TO (51,52)L
51    PRINT 120
120   FORMAT(//,20X,'ERROR FUNCTION',//,20X,'AMPLITUDE',/)
      GO TO 53
52    PRINT 121
121   FORMAT(//,20X,'PHASE ANGLE',/)
53    DO 50 I=1,NTP
      GO TO (54,55)L
54    PRINT 107, (AMPT(J,I),J=1,NTP)
      GO TO 50
55    PRINT 107, (ANGL(J,I),J=1,NTP)
50    CONTINUE
      IF(LPRINT .LT.0)RETURN
      PRINT 111
8     DO 59 L=1,2
      DO 57 I=1,NTP
      IF(L.GT.1.AND.I.GT.1)GO TO 58
      ANGS=0.
      AMPS=0.
      ANGX=0.
      AMPX=0.
58    DO 56 J=1,NTP
      AMPS=AMPS+AMPT(J,I)
      ANGS=ANGS+ANGL(J,I)
      IF(AMPT(J,I).GT.AMPX)AMPX=AMPT(J,I)
56    IF(ABS(ANGL(J,I)).GT.ABS(ANGX))ANGX=ANGL(J,I)
      IF(L.GT.1)GO TO 57
      AMPAV(I)=AMPS/NTP
      ANGAV(I)=ANGS/NTP
      AMX(I)=AMPX
      ANX(I)=ANGX
57    CONTINUE
      IF(L.GT.1)GO TO 59
```

```
      PRINT 117
      PRINT 107,(AMPAV(K),K=1,NTP)
      PRINT 117
      PRINT 107,(ANGAV(K),K=1,NTP)
      PRINT 117
      PRINT 107,(AMX(K),K=1,NTP)
      PRINT 117
      PRINT 107,(ANX(K),K=1,NTP)
      PRINT 117
59    CONTINUE
      AMPS=AMPS/NTP**2
      ANGS=ANGS/NTP**2
      ANGSST=0.
      AMPSST=0.
      DO 80 I=1,NTP
      ANGSS=0.
      AMPSS=0.
      DO 81 J=1,NTP
      AMPSS=AMPSS+(AMPT(J,I)-AMPAV(I))**2
      ANGSS=ANGSS+(ANGL(J,I)-ANGAV(I))**2
      AMPSST=AMPSST+(AMPT(J,I)-AMPS)**2
81    ANGSST=ANGSST+(ANGL(J,I)-ANGS)**2
      AMPRMS(I)=SQRT(AMPSS /NTP)
      ANGRMS(I)=SQRT(ANGSS /NTP)
80    CONTINUE
      PRINT 107,(AMPRMS(K),K=1,NTP)
      PRINT 117
      PRINT 107,(ANGRMS(K),K=1,NTP)
      AMPSS=SQRT(AMPSST/NTP**2)
      ANGSS=SQRT(ANGSST/NTP**2)
      PRINT 117
      PRINT 107,AMPS,ANGS,AMPX,ANGX,AMPSS,ANGSS
7     IF(LL.GT.0)RETURN
      IF(LPRINT .NE.0)RETURN
      PRINT 111
111   FORMAT(1H1)
      L3=K6/2
6     DO 10 L=1,L3
      GO TO (11,12,13)L
11    PRINT 112
112   FORMAT(//,20X,'IDEAL CASE',//)
      GO TO 14
12    PRINT 113
113   FORMAT(//,20X,'ACTUAL CASE',//)
      GO TO 14
13    PRINT 114
114   FORMAT(//,20X,'DIFFERENCE',//)
14    DO 30 I=1,NTP
      DO 30 J=1,NTP
      SR=CMPLX(0.,0.)
```

```
      DO 40 K=1,NTP
      GO TO (41,42,43)L
41    SR=SR+TRFF(I,K)*CONJG(TRFF2(J,K))
      GO TO 40
42    SR=SR+TR(I,K)*CONJG(TRFF2(J,K))
      GO TO 40
43    SR=SR+TRFB(I,K)*CONJG(TRFF2(J,K))
40    CONTINUE
      ANGL(I,J)=CANG(SR)*RAC
30    AMPT(I,J)=CABS(SR)
      DO 20 K=1,2
      GO TO (21,22)K
21    PRINT 115
115   FORMAT(20X,"AMPLITUDE",/)
      GO TO 23
22    PRINT 116
116   FORMAT(//,20X,"PHASE ANGLE",/)
23    DO 20 I=1,NTP
      GO TO (24,25)K
24    PRINT 107, (AMPT(I,J),J=1,NTP)
      GO TO 20
25    PRINT 107, (ANGL(I,J),J=1,NTP)
20    CONTINUE
10    CONTINUE
      IF( KC.GT.0)RETURN
      IF(A1.LE.0.)RETURN
      PRINT 118
118   FORMAT(1H1,10X,"REFLECTION MATRIX IS USED",//)
      DO 5 I=1,NTP
      DO 5 J=1,NTP
5     TRFF2(I,J)=TRFF(I,J)
      KC=KC+1
      GO TO 6
      END
```

```
0001          SUBROUTINE TRFMTX(NM,NN,NR1,NBP,NBK,MC,PHA,S11,S12,S21,S22)
0002          DIMENSION NBP(16),NBK(16)
0003          DIMENSION MC(NR1,NN),PHA(NR1,NN)
0004          DIMENSION S11(NM,NM),S12(NM,NM),S21(NM,NM),S22(NM,NM)
0005          COMMON/C$5/T11(32,32),T12(32,32),T21(32,32),T22(32,32)
0006          COMMON/C$6/R11(32,32),R12(32,32),R21(32,32),R22(32,32)
0007          DIMENSION SS11(8,8),SS12(8,8),SS21(8,8),SS22(8,8)
0008          DIMENSION MCT(32)
0009          COMPLEX S11,S12,S21,S22,T11,T12,T21,T22,R11,R12,R21,R22,AR,SS11,
             CSS12,SS21,SS22
        C     FIRST INDEX ROW
        C     SECOND INDEX COLUMN
0010          DO 10 I=1,NR1
        C     TRANSFER MATRIX IN CONNECTION REGION
0011          DO 11 L=1,NN
0012          LL=MC(I,L)
0013     11   MCT(LL)=L
0014          PRINT 102,(MC(I,L),L=1,NN)
0015          PRINT 102,(MCT(L),  L=1,NN)
0016          PRINT 101,(PHA(I,L),L=1,NN)
0017     102  FORMAT(//,(10X,8I5))
0018     101  FORMAT(//,(10X,8F10.4))
0019          DO 20 J=1,NN
0020          DO 20 K=1,NN
0021          T11(J,K)=CMPLX(0.,0.)
0022          T12(J,K)=CMPLX(0.,0.)
0023          T21(J,K)=CMPLX(0.,0.)
0024          T22(J,K)=CMPLX(0.,0.)
0025          IF(MCT(J).NE.K)GO TO 20
0026          T11(J,K)=AR(PHA(I,J))
0027          T22(J,K)=CONJG(T11(J,K))
0028     20   CONTINUE
0029          PRINT 100,((T11(M,N),N=1,NN),M=1,NN)
0030          PRINT 100,((T22(M,N),N=1,NN),M=1,NN)
0031          IF(I.GT.1)GO TO 21
0032          DO 22 J=1,NN
0033          DO 22 K=1,NN
0034          R11(J,K)=T11(J,K)
0035          R12(J,K)=CMPLX(0.,0.)
0036          R21(J,K)=CMPLX(0.,0.)
0037     22   R22(J,K)=T22(J,K)
0038          GO TO 23
0039     21   CALL MTXMLT(NM,NN,    T11,T12,T21,T22,R11,R12,R21,R22)
0040     23   PRINT 100,((R11(M,N),N=1,NN),M=1,NN)
0041          PRINT 100,((R12(M,N),N=1,NN),M=1,NN)
0042          PRINT 100,((R21(M,N),N=1,NN),M=1,NN)
0043          PRINT 100,((R22(M,N),N=1,NN),M=1,NN)
0044     100  FORMAT(//,(10X,8F10.4))
0045          IF(I.EQ.NR1)GO TO 10
        C     TRANFER MATRIX IN BLOCK REGION
```

```
      NP=NBP(I)
      IF(I.LE.1)GO TO 26
      IF(NP.EQ.NBP(I-1))GO TO 27
   26 CALL BLK(8,NP,SS11,SS12,SS21,SS22)
C     RESET S MATRIX
      DO 24 J=1,NN
      DO 24 K=1,NN
   24 S11(J,K)=CMPLX(0.,0.)
      S12(J,K)=CMPLX(0.,0.)
      S21(J,K)=CMPLX(0.,0.)
      S22(J,K)=CMPLX(0.,0.)
      DO 25 J=1,NN,NP
      DO 25 JJ=1,NP
      J1=JJ-1
      DO 25 KK=1,NP
      K1=KK-1
      S11(J+J1,J+K1)=SS11(JJ,KK)
      S12(J+J1,J+K1)=SS12(JJ,KK)
      S21(J+J1,J+K1)=SS21(JJ,KK)
      S22(J+J1,J+K1)=SS22(JJ,KK)
   25 CONTINUE
      PRINT 100,((S11(M,N),N=1,NN),M=1,NN)
      PRINT 100,((S12(M,N),N=1,NN),M=1,NN)
      PRINT 100,((S21(M,N),N=1,NN),M=1,NN)
      PRINT 100,((S22(M,N),N=1,NN),M=1,NN)
C     INVERSE S-MATRIX
      CALL INVS1(NM,NN,S12)
      CALL STTRF(NM,NN,S11,S12,S21,S22,T11,T12,T21,T22)
      PRINT 100,((S11(M,N),N=1,NN),M=1,NN)
      PRINT 100,((S12(M,N),N=1,NN),M=1,NN)
      PRINT 100,((S21(M,N),N=1,NN),M=1,NN)
      PRINT 100,((S22(M,N),N=1,NN),M=1,NN)
   27 DO 50 J=1,NN
      DO 50 K=1,NN
      T11(J,K)=S11(J,K)
      T12(J,K)=S12(J,K)
      T21(J,K)=S21(J,K)
   50 T22(J,K)=S22(J,K)
      CALL MTXMLT(NM,NN,T11,T12,T21,T22,R11,R12,R21,R22)
      PRINT 100,((R11(M,N),N=1,NN),M=1,NN)
      PRINT 100,((R12(M,N),N=1,NN),M=1,NN)
      PRINT 100,((R21(M,N),N=1,NN),M=1,NN)
      PRINT 100,((R22(M,N),N=1,NN),M=1,NN)
   10 CONTINUE
      CALL INVS1(NM,NN,R22)
      DO 40 J=1,NN
      DO 40 K=1,NN
      S12(J,K)=R22(J,K)
      S21(J,K)=R22(J,K)
      S11(J,K)=CMPLX(0.,0.)
      S22(J,K)=CMPLX(0.,0.)
      DO 40 L=1,NN
      S11(J,K)=S11(J,K)-R22(J,L)*R21(L,K)
      S22(J,K)=S22(J,K)+R12(J,L)*R22(L,K)
   40 CONTINUE
      PRINT 100,((S11(M,N),N=1,NN),M=1,NN)
      PRINT 100,((S12(M,N),N=1,NN),M=1,NN)
      PRINT 100,((S21(M,N),N=1,NN),M=1,NN)
      PRINT 100,((S22(M,N),N=1,NN),M=1,NN)
      RETURN
      END
```

```
0001          SUBROUTINE STRF(NM,NN,NR1,NBP,NBK,MC,PHA,S11,S12,S21,S22)
0002          DIMENSION NBP(16),NBK(16)
0003          DIMENSION MC(NR1,NN),PHA(NR1,NN)
0004          DIMENSION S11(NM,NM),S12(NM,NM),S21(NM,NM),S22(NM,NM)
0005          COMMON/C$5/T11(8,8),T12(8,8),T21(8,8),T22(8,8),R11(8,8),R12(8,8),
             C  R21(8,8),R22(8,8),SPACE(7168)
0006          DIMENSION MCT(32)
0007          DIMENSION SS11(2,2),SS12(2,2),SS21(2,2),SS22(2,2)
0008          COMPLEX S11,S12,S21,S22,T11,T12,T21,T22,R11,R12,R21,R22,AR,SS11,
             CSS12,SS21,SS22
0009          CALL TWOPT(SS11,SS12,SS21,SS22,2)
        C     1ST INDEX,COLUMN
        C     2ND INDEX,ROW
0010          DO 10 I=1,NR1
        C     TRANSFER MATRIX IN CONNECTION REGION
0011          DO 11 L=1,NN
0012          LL=MC(I,L)
0013     11   MCT(LL)=L
0014          DO 20 J=1,NN
0015          DO 20 K=1,NN
0016          T11(J,K)=CMPLX(0.,0.)
0017          T12(J,K)=CMPLX(0.,0.)
0018          T21(J,K)=CMPLX(0.,0.)
0019          T22(J,K)=CMPLX(0.,0.)
0020          IF(MCT(J).NE.K)GO TO 20
0021          T11(J,K)=AR(PHA(I,J))
0022          T22(J,K)=CONJG(T11(J,K))
0023     20   CONTINUE
0024          IF(I.GT.1)GO TO 21
0025          DO 22 J=1,NN
0026          DO 22 K=1,NN
0027          R11(J,K)=T11(J,K)
0028          R12(J,K)=T12(J,K)
0029          R21(J,K)=T21(J,K)
0030     22   R22(J,K)=T22(J,K)
0031          GO TO 23
0032     21   CALL MTXMLT(NM,NN,     T11,T12,T21,T22,R11,R12,R21,R22)
0033          IF(I.EQ.NR1)GO TO 10
        C     TRANFER MATRIX IN BLOCK REGION
        C     RESET S MATRIX
0034     23   NP=NBP(1)
0035          IF(I.LE.1)GO TO 26
0036          IF(NP.EQ.NBP(I-1))GO TO 27
0037     26   DO 24 J=1,NN
0038          DO 24 K=1,NN
0039     24   S11(J,K)=CMPLX(0.,0.)
0040          S12(J,K)=CMPLX(0.,0.)
0041          S21(J,K)=CMPLX(0.,0.)
0042          S22(J,K)=CMPLX(0.,0.)
0043          DO 25 J=1,NN,NP
```

```
0044          DO 25 JJ=1,NP
0045          J1=JJ-1
0046          DO 25 KK=1,NP
0047          K1=KK-I
0048          S11(J+J1,J+K1)=SS11(JJ,KK)
0049          S12(J+J1,J+K1)=SS12(JJ,KK)
0050          S21(J+J1,J+K1)=SS21(JJ,KK)
0051          S22(J+J1,J+K1)=SS22(JJ,KK)
0052     25   CONTINUE
        C     INVERSE S-MATRIX
0053          CALL INVS2(NM,NN,S12)
0054          CALL STTRF(NM,NN,S11,S12,S21,S22,T11,T12,T21,T22)
0055     27   DO 50 J=1,NN
0056          DO 50 L=1,NN
0057          T11(J,K)=S11(J,K)
0058          T12(J,K)=S12(J,K)
0059          T21(J,K)=S21(J,K)
0060     50   T22(J,K)=S22(J,K)
0061          CALL MTXMLT(NM,NN,T11,T12,T21,T22,R11,R12,R21,R22)
0062     10   CONTINUE
0063          CALL INVS2(NM,NN,R22)
0064          DO 40 J=1,NN
0065          DO 40 K=1,NN
0066          S12(J,K)=R22(J,K)
0067          S21(J,K)=R22(J,K)
0068          S11(J,K)=CMPLX(0.,0.)
0069          S22(J,K)=CMPLX(0.,0.)
0070          DO 40 L=1,NN
0071          S11(J,K)=S11(J,K)-R22(J,L)*R21(L,K)
0072          S22(J,K)=S22(J,K)+R12(J,L)*R22(L,K)
0073     40   CONTINUE
0074          RETURN
0075          END
```

```
0001          SUBROUTINE STTRF(NM,NN,S11,S12,S21,S22,T11,T12,T21,T22)
        C     THIS SUBROUTINE INVERSES S MATRIX AND STORES IN T
        C
0002          DIMENSION S11(NM,NM),S12(NM,NM),S21(NM,NM),S22(NM,NM)
0003          DIMENSION T11(NM,NM),T12(NM,NM),T21(NM,NM),T22(NM,NM)
0004          COMPLEX S11,S12,S21,S22,T11,T12,T21,T22
0005          DO 30 J=1,NN
0006          DO 30 K=1,NN
0007          T22(J,K)=S12(J,K)
0008          T21(J,K)=CMPLX(0.,0.)
0009          DO 30 L=1,NN
0010     30   T21(J,K)=T21(J,K)-S12(J,L)*S11(L,K)
0011          DO 31 J=1,NN
0012          DO 31 K=1,NN
0013          T12(J,K)=CMPLX(0.,0.)
0014          T11(J,K)=S21(J,K)
0015          DO 31 L=1,NN
0016          T12(J,K)=T12(J,K)+S22(J,L)*S12(L,K)
0017     31   T11(J,K)=T11(J,K) +S22(J,L)*T21(L,K)
0018          DO 20 K=1,NN
0019          DO 20 J=1,NN
0020          S11(J,K)=T11(J,K)
0021          S12(J,K)=T12(J,K)
0022          S21(J,K)=T21(J,K)
0023     20   S22(J,K)=T22(J,K)
0024          RETURN
0025          END
```

```
      SUBROUTINE MTXMLT(NM,NN,R11,R12,R21,R22,T11,T12,T21,T22)
C     THIS SUBROUTINE MULTIPLE  SUBMATRICES R*T THEN STORE THE RESULT
C     IN R
C     S=R*T
C     S11=R11*T11+R12*T21
C     S12=R11*T12+R12*T22
C     S21=R21*T11+R22*T21
C     S22=R21*T12+R22*T22
      DIMENSION TT1(32,32),TT2(32,32)
      DIMENSION T11(NM,NM),T12(NM,NM),T21(NM,NM),T22(NM,NM)
      DIMENSION R11(NM,NM),R12(NM,NM),R21(NM,NM),R22(NM,NM)
      COMPLEX T11,T12,T21,T22,R11,R12,R21,R22,TT1,TT2
      PRINT 101
      PRINT 100,((R11(M,N),N=1,NN),M=1,NN)
      PRINT 100,((R12(M,N),N=1,NN),M=1,NN)
      PRINT 100,((R21(M,N),N=1,NN),M=1,NN)
      PRINT 100,((R22(M,N),N=1,NN),M=1,NN)
      PRINT 100,((T11(M,N),N=1,NN),M=1,NN)
      PRINT 100,((T12(M,N),N=1,NN),M=1,NN)
      PRINT 100,((T21(M,N),N=1,NN),M=1,NN)
      PRINT 100,((T22(M,N),N=1,NN),M=1,NN)
      PRINT 101
  100 FORMAT(//,(10X,8F10.4))
  101 FORMAT(//,"......................................",//)
      DO 10 J=1,NN
      DO 10 K=1,NN
      TT1(J,K)=CMPLX(0.,0.)
      TT2(J,K)=CMPLX(0.,0.)
      DO 10 L=1,NN
      TT1(J,K)=TT1(J,K)+R11(J,L)*T11(L,K)+R12(J,L)*T21(L,K)
      TT2(J,K)=TT2(J,K)+R11(J,L)*T12(L,K)+R12(J,L)*T22(L,K)
   10 CONTINUE
      DO 20 J=1,NN
      DO 20 K=1,NN
      R11(J,K)=TT1(J,K)
   20 R12(J,K)=TT2(J,K)
      DO 30 J=1,NN
      DO 30 K=1,NN
      TT1(J,K)=CMPLX(0.,0.)
      TT2(J,K)=CMPLX(0.,0.)
      DO 30 L=1,NN
      TT1(J,K)=TT1(J,K)+R21(J,L)*T11(L,K)+R22(J,L)*T21(L,K)
      TT2(J,K)=TT2(J,K)+R21(J,L)*T12(L,K)+R22(J,L)*T22(L,K)
   30 CONTINUE
      DO 40 J=1,NN
      DO 40 K=1,NN
      R21(J,K)=TT1(J,K)
   40 R22(J,K)=TT2(J,K)
      DO 50 J=1,NN
      DO 50 K=1,NN
      T11(J,K)=R11(J,K)
      T12(J,K)=R12(J,K)
      T21(J,K)=R21(J,K)
   50 T22(J,K)=R22(J,K)
      RETURN
      END
```

```
0001          SUBROUTINE PATERN (NTP,TRFF,TRFB,KLL,NPAV,NMX)
        C     ABSOLUTE VALUE OF KLL REPRESENTS THE BEAM INDEX WHOSE PATTERN IS
        C     TO BE PLOTTED
        C     KLL=0  NO PLOT
        C     KLL GRATER THAN 0 PLOT PATTERN ONLY
        C     KLL LESS THAN 0 PLOT BOTH PATTERN AND MAIN BEAMS
0002          COMMON/C$1/PLTAY(500)
0003          DIMENSION TRFF(NMX,NMX),TRFB(NMX,NMX)
0004          COMMON/C$5/PEAK(64,100),PMAX(64),PAV(64),PMAV(64),KIND1(64),
            C  KIND2(64),PEAKDB(100),SPACE(1372)
0005          COMMON/C$6/CONTA(4096),SINTA(4096)
0006          COMPLEX TRFF,TRFB,S
0007          Z(X)=10.*ALOG10(X)
0008          PRINT 104
0009     104  FORMAT(1H1)
0010          PI=3.1415926536
0011          ATR=PI/180.
0012          KPLOT=IABS(KLL)
0013          NTP2=NTP/2
        C     PLOT FRAME
0014          YSL=80.
0015          NY=YSL
0016          XSL=180.
0017          NX=XSL
0018          HN=5.
0019          SY=2.
0020          XM=10.
0021          YM=5.
0022          YS=2.
0023          YSM=YS+YM
0024          NTA=20*NTP
0025          NTA1=NTA+1
0026          TAINC=PI/NTA
0027          PNOR=NTP
0028          CALL PHASAN(TAINC,IKA)
0029          KL=1
0030          IF(KLL.LT.0)KL=2
0031          NTA1N=2*NTA+1
0032          DO 1 I=1,NTP
0033          DO 1 J=1,NTP
0034      1   TRFF(I,J)=TRFF(I,J)+TRFB(I,J)
0035          DO 20 IL=1,KL
0036          IF(IL.LT.2)GO TO 25
0037          CALL PLOT(XM+4.,0.,-3)
0038          NTA1N=NTA1
0039          NSIGN=1
0040     25   CALL FRAME(XM,YM,XSL,YSL, SY,HN,NX,NY)
0041          DO 20 K=1,NTP2
0042          KK=0
0043          KFLAG=0
```

```
      KMI=1
      KPCONT=1
      KMIND=1
      KMA=0
      LEDGE=0
      DO 30 IJ=1,NTAIN
      IF(IL.GT.1)GO TO 23
      NSIGN=-1
      IF(IJ.GE.NTA1)NSIGN=1
      I=IJ-NTA1+NSIGN
      GO TO 24
   23 I=IJ
   24 II=IABS(I)
      I1=II-1
      PAR=0.
      PAI=0.
      IF(IL.LT.2)GO TO 21
      IF(I.LT.KIND1(K).OR.I.GT.KIND2(K))GO TO 30
      IF(I.EQ.KIND1(K))II=1
   21 DO 40 J=1,NTP
      S=TREF(J,K)
      JI=(J-1)*I1+1
      JMOD=MOD(JI,IKA)
      IF(JMOD.EQ.0)JMOD=IKA
      PAR=CONTA(JMOD)*REAL(S)-SINTA(JMOD)*AIMAG(S)*NSIGN+PAR
      PAI=CONTA(JMOD)*AIMAG(S)+SINTA(JMOD)*REAL(S)*NSIGN+PAI
   40 CONTINUE
      PAT=PAR**2+PAI**2
      PAT=PAT/PNOR
      IF(IL.EQ.2)GO TO 22
      IF(IJ.LE.1)GO TO 31
      IF(PAT-PAT1)32,31,33
C     EXAMINE IF A MAXIMUM IS PASSED
   32 KMI=1
      IF(IJ.EQ.2)LEDGE=1
      IF(KMA.LE.0)GO TO 31
      IF(PAT1.LE.PEAK(K,KPCONT )) GO TO 34
      KIND1(K)=KMIND
      KFLAG=1
      KPCONT =KK+1
   34 KK=KK+1
      PEAK(K,KK)=PAT1
      KMA=0
      GO TO 31
C     EXAMINE IF A MINIMUM IS PASSED
   33 KMA=1
      IF(KMI.LE.0)GO TO 31
   35 KMIND=I-1
      IF(KFLAG.GT.0)KIND2(K)=I-1
      KFLAG=0
```

```
0092          KMI=0
0093     31   PAT1=PAT
         C    PLOT PATTERN FOR A GIVEN BEAM
0094          IF(K.NE.KPLOT)GO TO 30
0095          IF(IJ.LT.NTA1)GO TO 30
0096     22   DB=Z(PAT)
0097          Y=(1.+DB/YSL)*YM+SY
0098          IF(Y.GT.YSM)Y=YSM
0099          IF(Y.LT.SY)Y=SY
0100          P=I-1
0101          X=P*XM/NTA
0102          IF(II.EQ.1)GO TO 3
0103          CALL PLOT(X,Y,2)
0104          GO TO 30
0105     3    CALL PLOT(X,Y,3)
0106     30   CONTINUE
0107          IF(IL.GE.2)GO TO 20
0108          IF(KMA.GT.0)GO TO 42
0109          IF(KPCONT .EQ.KK)KIND2(K)=NTA1
0110          IF(KIND2(K).LE.0)KIND2(K)=NTA1
         C    DELETE THE MAIN LOBE
0111          GO TO 43
0112     42   IF(LEDGE.LE.0)GO TO 43
0113          KK=KK+1
0114          PEAK(K,KK)=PAT1
0115     43   DO 44 I=1,KK
0116     44   PEAKDB(I)=10.*ALOG10(PEAK(K,I))
0117          PRINT 102,K
0118     102  FORMAT(//,20X,'BEAM INDEX',I5)
0119          PRINT 101,(PEAKDB(I),I=1,KK)
0120          KK=KK-1
0121          DO 53 L=1,KK
0122          IF(L.LT.KPCONT )GO TO 53
0123          PEAK(K,L)=PEAK(K,L+1)
0124     53   CONTINUE
0125          PMAX(K)=0
0126          PSUM=0.
0127          PRINT 101,(PEAK(K,I),I=1,KK)
0128     101  FORMAT(//,(10X,10E12.4))
0129          DO 50 L=1,KK
0130          IF(PEAK(K,L).GT.PMAX(K))PMAX(K)=PEAK(K,L)
0131          PSUM=PSUM+PEAK(K,L)
0132     50   CONTINUE
0133          PAV(K)=PSUM/KK
0134          PMAX(K)=Z(PMAX(K))
0135          PAV(K)=Z(PAV(K))
0136          PRINT 103,PMAX(K),PAV(K)
0137     103  FORMAT(//,10X,'PEAK',F10.4,5X,'AVERAGE',F10.4)
0138     20   CONTINUE
0139          RETURN
0140          END
```

```
0001          SUBROUTINE HLFMTX(NTP,NR1,NBP,NBK,MC,PHA)
0002          DIMENSION NBP(16),NBK(16)
0003          DIMENSION MC(NR1,NTP),PHA(NR1,NTP)
0004          COMMON/C$3/MCT(64)
0005          DIMENSION ANG(64) ,ATEMP(64)
0006          NN=NR1/2
0007          LL=(NR1+1)/2-NN
        C     LL=1 NUMBER OF ROWS IS EVEN
        C     LL=0 NUMBER OF ROWS IS ODD
0008          CALL PHASUM(NR1,NTP,NBP,MC,PHA,ANG)
0009          DO 10 I=1,NTP
0010          II=I
0011          DO 11 J=1,NR1
0012          KK=MC(J,II)
0013          IF(J.EQ.NN)KKP=KK
0014          II=KK
0015     11   CONTINUE
        C     FIND THE JOINT POINT THEN STORE IN MCT ARRAY
0016          DO 12 J=1,NN
0017          KKS=MC(J,KK)
0018     12   KK=KKS
0019          MCT(KKP)=KKS
        C     AVERAGE THE PHASE ANGLES FOR SYMMETRICAL MATRIX
0020          DO 13 J=1,NN
0021          JJ=NR1-J+1
0022          IMC=MC(J,I)
0023          AVG=(PHA(J,IMC)+PHA(JJ,I))/2.
0024     13   PHA(J,IMC)=AVG
0025          PHA(JJ,I)=AVG
0026     10   CONTINUE
        C     CORRECT PHASE ANGLE OF THE MIDDLE ROW WHEN THE NUMBER OF ROWS IS
        C     EVEN
0027          IF(LL.LE.0)GO TO 1
0028          N1=NN+1
0029          DO 20 I=1,NTP
0030          II=MC(N1,I)
0031          IN=MC(N1,MCT(I))
0032          ATEMP(II)=PHA(N1,II)
0033          IF(PHA(N1,IN).GT.ATEMP(II))ATEMP(II)=PHA(N1,IN)
0034     20   CONTINUE
        C     CORRECT THE PHASE ANGLE BY ADDING THE SAME EXTRA PHASE TO EACH
        C     PORT IN A BLOCK
0035          NMP=NBP(N1)
0036          NMB=NBK(N1)
0037          DO 21 I=1,NMB
0038          IMB=(I-1)*NMP
0039          AA=0.
0040          DO 22 J=1,NMP
0041          KK=IMB+J
0042          A=ATEMP(KK)-PHA(N1,KK)
0043          IF(A.GT.AA)AA=A
0044     22   CONTINUE
0045          IF(AA.LE.0.) GO TO 21
0046          DO 23 J=1,NMP
0047          KK=IMB+J
0048     23   PHA(N1,KK)=PHA(N1,KK)+AA
0049     21   CONTINUE
0050          RETURN
        C     CORRECT THE PHASE ANGLES FOR THE CASE WHEN THE NUMBER OF ROWS IS
        C     ODD
```

```
      1     CALL PHASUM(NR1,NTP,NBP,MC,PHA,ATEMP)
            DO 30 I=1,NTP
            AA=ANG(I)-ATEMP(I)
            JJ=MC (1,I)
      30    PHA(1,JJ)=AA
            PHA(NR1,I)=AA
            RETURN
            END
```

```
            SUBROUTINE PHASUM(NR1,NTP,NBP,MC,PHA,AS)
            DIMENSION NBP(16)
            DIMENSION MC(NR1,NTP),PHA(NR1,NTP),AS(NTP)
            DIMENSION LAP(2,64) ,A(64)
      C     SET THE PHASE SHIFT OF THE BOTTOM ROW
            NROW=NR1-1
            NN=NBP(NROW)
            DO 1 J=1,NN
            LAP(2,J)=J
      1     AS(J)=PHA(1,1)
            KK=NN
            DO 10 I=1,NROW
            II=NROW-I
            I1=II+1
            NN=NBP(II)
            IF(II.LE.0)NN=1
            DO 12 J=1,NTP
            LAP(1,J)=LAP(2,J)
      12    A(J)=AS(J)
            KN=0
            DO 20 L=1,KK
            LL=LAP(1,L)
            DO 21 N=1,NTP
            IF(MC(I1,N).NE.LL)GO TO 21
            JJ=N
            GO TO 22
      21    CONTINUE
      22    NMD=MOD(JJ,NN)
            IF(NMD.EQ.0)NMD=NN
            DO 30 K=1,NN
            IND=JJ-NMD+K
            IF(NN.EQ.1)IND=JJ
            KN=KN+1
            LAP(2,KN)=IND
      30    AS(IND)=A(LL)+PHA(I1,LL)
      20    CONTINUE
            KK=KN
      10    CONTINUE
            RETURN
            END
```

```
0001          SUBROUTINE NTWK(NTP,NR1,NBP,NBK,MC,PHA)
        C*****THIS SUBROUTINE FINDS THE CONNECTION OF A BUTLER MATRIX OR FFT
        C     GIVEN THE NUMBER OF ROWS AND THE NUMBER OF PORTS IN EACH BLOCK IN
        C     EACH ROW
        C*****COMPILED BY J. K. HSIAO
        C*****FIRST VERSION IS COMPILED ON MAY 3,1976
        C*****NTP,  NUMBER OF TOTAL INPUT PORTS OR SAMPLES
        C*****NROW, NUMBER OF ROWS REQUIRED TO PERFORM THE TRANSFORMATION
        C*****NBP, AN ARRAY STORES THE NUMBER OF PORTS IN EACH BLOCK AT EACH
        C     ROW. EACH BLOCK IN A ROW HAS THE SAME NUMBER OF PORTS
        C*****NBK, AN ARRAY STORES THE NUMBER OF BLOCKS IN EACH ROW.
        C*****MC, A TWO DIMENSIONAL ARRAY STORES THE CONNCTIONS OF THE NETWORK.
        C     FIRST INDEX OF THE ARRAY REPRESENTS THE NUMBER OF CURRENT ROW. THE
        C     LOCATION OF THE SECOND INDEX REPRESENTS THE PHYSICAL LOCATION OF
        C     THE PREVIOUS ROW WHILE THE CONTENTS OF IT IS THE CONNECTION TO THE
        C     CURRENT ROW
0002          DIMENSION MC(NR1,NTP),PHA(NR1,NTP)
0003          DIMENSION NFTS(64),NBK(16),NBP(16)
        C     COMPUTES THE NUMBER OF PORTS IN EACH BLOCK
0004          NROW=NR1-1
0005          PI=3.1415926536
0006          PI2=PI*2.
0007          NBP(NR1)=1
0008          NTP2=NTP/2
0009          DO 10 I=1,NR1
0010     10   NBK(I)=NTP/NBP(I)
        C**** NFTS ARRAY STORES THE LOCATION OF THE SAMPLES IN EACH BEAM(OR
        C     FREQUENCY SAMPLE). THE STRUCTURE IS CHARACTERIZED BY TWO NUMBERS,
        C     NTS,NUMBER OF TIME SAMPLES(OR INPUT PORTS) AND NFS, NUMBER OF
        C     FREQUENCY SAMPLES(OR NUMBER OF BEAMS).  FOR EXAMPLE, NFTS((3-1)*
        C     NTS+1) IS THE PHYSICAL LOCATION OF THE FIRST TIME SAMPLE IN THE
        C     THIRD FREQUENCY GROUP( OR OF THE THIRD BEAM),THIS IS REPRESENTED
        C     BY LMC
        C
        C     SET THE INITIAL NFTS ARRAY
0011          DO 11 I=1,NTP
0012     11   NFTS(I)=I
        C**** NTS1 IS THE PREVIOUS VALUES OF THE NUMBER OF TIME SAMPLES(OR INPUT
        C     PORTS)
        C**** NTS2 IS THE CURRENT VALUE
        C**** NFS1 IS THE PREVIOUS VALUE OF THE NUMBER OF FREQUENCY SAMPLES(OR
        C     BEAMS)
        C*****NFS2 IS THE CURRFNT VALUE
        C
        C
        C     SET THE INIAL VALUES OF NTS AND NFS
0013          NTS1=NTP
0014          NFS1=1
0015          DO 20 I=1,NR1
        C     MM THE NUMBER OF BLOCKS OF THE CURRENT ROW
```

```
C       NN,  THE NUMBER OF PORTS IN EACH BLOCK OF THE CURRENT ROW
        MM=NBK(I)
        NN=NBP(I)
C       SET NTS2 AND NFS2
        NTS2=NTS1/NN
        NFS2=NTP/NTS2
C**** THE ACTUAL REQUIRED PHASE GRADIENT BETWEEN SUCCESSIVE ELEMENT FOR
C       THE FIRST BEAM IS
        PAG=PI/NFS2
C***    THAVAILABLE PHASE GRADIENT FOR THE FIRST BEAM IN EACH BLOCK IS
        PSG=PI/NN
        KK=0
        DO 30 J=1,MM
        MODJ=MOD(J,NFS1)
        IF(MODJ.EQ.0)MODJ=NFS1
        JJ=(J-1)/NFS1+1
        PAGG=PAG*(MODJ*2-1)
        PASGD=PSG-PAGG
        DO 30 K=1,NN
        K1=K-1
        KK=KK+1
        LMC=(MODJ-1)*NTS1+(K-1)*NTS2+JJ
        MCLOC=NFTS(LMC)
        MC(I,MCLOC)=KK
        IF(KK.LE.NTP2)GO TO 31
        KKI=NTP-KK+1
        PHA(I,KK)=PHA(I,KKI)
        GO TO 30
   31   IF(PASGD.GT.0.)GO TO 32
        PHA(I,KK)=ABS(PASGD)*(NN-K)
        GO TO 30
   32   PHA(I,KK)=PASGD*K1
   30   CONTINUE
C       RECORDING  THE RREQUENCY SAMPLE OR BEAM POSITION INTO NFTS ARRAY
        NTS1=NTS2
        NFS1=NFS2
        KK=0
C       MNS IS THE NUMBER OF BLOCKS WITHIN EACH GROUP OF FREQUENCY SAMPLES
        MNS=MM/NTS1
        DO 40 J=1,NFS1
        JMOD=MOD(J,MNS)
        IF(JMOD.EQ.0)JMOD=MNS
        JJ=(J-1)/MNS+1
        DO 40 K=1,NTS1
        KK=KK+1
   40   NFTS(KK)=(K-1)*NFS1+(JMOD-1)*NN+JJ
   20   CONTINUE
        RETURN
        END
```

```
      SUBROUTINE FRAME(XM,YM,XSL,YSL,SY,HN,NX,NY)
      COMMON/C$1/PLTAY(500)
      YMSY=YM+SY
      HLAB=HN*.035
      HLAS=HLAB+.035
      WLAB=4.*HLAB/7.
      XSCL=XSL/NX
      YSCL=YSL/NY
      DY=YM/NY
      Y=SY
      NNY=NY+1
      CALL PLOT(0.,SY,3)
      CALL PLOT(XM,SY,2)
      CALL PLOT(XM,YMSY,2)
      CALL PLOT(0.,YMSY,2)
      CALL PLOT(0.,SY,2)
      DO 10 I=1,2
      Y=SY
      IF(I.GT.1)GO TO 12
      X1=0.
      X2=-.2
      X3=-.1
      GO TO 13
   12 X1=XM
      X2=XM+.2
      X3=XM+.1
   13 DO 10 J=1,NNY
      CALL PLOT(X1,Y,3)
      MODY=MOD(J-1,10)
      IF(MODY.NE.0)GO TO 11
      CALL PLOT(X2,Y,2)
      IF(I.GT.1)GO TO 10
      A=YSCL*(J-1-NY)
      CALL NUMBER(-6.5*WLAB,Y-HLAB/2.,HLAB,A,0.,4HF3.0)
      GO TO 10
   11 CALL PLOT(X3,Y,2)
   10 Y=Y+DY
      DX=XM/NX
      NXX=NX+1
      DO 20 I=1,2
      X=0.
      IF(I.GT.1)GO TO 22
      Y1=SY
      Y2=Y1-.2
      Y3=Y1-.1
      GO TO 23
   22 Y1=YMSY
      Y2=Y1+.2
      Y3=Y1+.1
   23 DO 20 K=1,NXX
      KK=K-1
      CALL PLOT(X,Y1,3)
      MODX=MOD(KK,10)
      IF(MODX.NE.0)GO TO 21
      CALL PLOT(X,Y2,2)
      IF(I.GT.1)GO TO 20
      A=KK*XSCL
      CALL NUMBER(X-2.5*WLAB,SY-HLAB*3.0,HLAB,A,0.,4HF3.0)
      GO TO 20
   21 CALL PLOT(X,Y3,2)
   20 X=X+DX
```

```
0062          CALL SYMBOL(.5*XM-17.5*WLAB,-5.*HLAB+SY,HLAS,22HPARAMETER U IN DEG
             CREES,0.,22)
0063       35 CALL SYMBOL(-7.*WLAB,YM/2.+SY-15.*WLAB,HLAS,18HARRAY PATTERN (DB),
             *90.,18)
0064     32   CALL PLOT(0.,0.,3)
0065          END
```

```
0001          SUBROUTINE SIMCX(IS,ORIG,NN,MAT,MCT,ANS,LK)
        C     IDENT NUMBER - F1002R00
        C     TITLE - COMPLEX MATRIX INVERSION, SOLUTION OF LINEAR EQUATIONS
        C     IDENT NAME - F1-NRL-SIMCX
        C     LANGUAGE - FORTRAN
        C     COMPUTER - CDC-3800
        C     CONTRIBUTOR - JANET P. MASON, CODE 7813, RESEARCH COMPUTATION
        C                   CENTER, MIS DIVISION
        C     ORGANIZATION - NRL - NAVAL RESEARCH LABORATORY - WASHINGTON, D.C.
        C                                                              20390
        C     DATE - 16 DECEMBER 1970
        C     PURPOSE - TO SOLVE THE COMPLEX MATRIX EQUATION AX=B WHERE A IS A
        C               SQUARE COEFFICIENT MATRIX AND B IS A MATRIX OF CONSTANT
        C               VECTORS.  THE DETERMINANT AND INVERSE OF A ARE ALSO
        C               OBTAINED.
0002          COMPLEX SUM, MAT,ORIG,ANS,B0,B2,B4,B6,B8,B10,B11,B13,B15,CC,CC2,B3
0003          EQUIVALENCE(B2,C),(CC,CX),(CC2,CX2)
0004          DIMENSION MAT(MCT,1),ORIG(NN,1),ANS(MCT),C(2),CX(2),CX2(2)
0005       10 FORMAT(1X, 2E12.6)
0006       15 FORMAT(25H  THIS MATRIX IS SINGULAR/)
0007       18 FORMAT(1H0," VALUE OF DETERMINANT IS ",2E12.6,//)
0008     21   FORMAT(1X,2E12.6,5X,2E12.6)
0009       23 FORMAT(8X,"ORIGINAL CONSTANTS",21X,"DERIVED CONSTANTS"/)
0010       26 FORMAT(1H1,6X,"THE INVERSE (BY COLUMNS)")
0011       27 FORMAT(1H0)
0012       28 FORMAT(1H1,6X,"VALUES OF THE UNKNOWNS")
0013       35 FORMAT(9X,"IDENTITY MATRIX")
0014          B3=(-1.0,0.0)
0015          B4=(0.0,0.0)
0016          B11=(1.0,0.0)
0017          ICT=MCT
0018          JSING=MCT
0019          MT=MCT+1
0020          NCT=MCT+MCT
        C     PUT ORIGINAL MATRIX INTO MAT
0021          IF(IS.EQ.0)GO TO 39
0022          ICT=MCT+IS
0023          NCT=ICT
0024     39   DO 2 J=1,ICT
0025          DO 2 I=1,MCT
0026          MAT(I,J)=ORIG(I,J)
0027        2 CONTINUE
0028          IF(IS.NE.0)GO TO 30
        C     PUT IDENTITY MATRIX INTO RIGHT HALF OF MAT
0029       31 DO 32 J=MT,NCT
0030          DO 32 I=1,MCT
0031       32 MAT(I,J)=B4
0032          DO 33 J=1,MCT
0033       33 MAT(J,J+MCT)=B3
        C     FORM TRIANGULARIZED MATRIX
```

```
0034        30 JCT=MCT-1
0035           DO 3 J=1,JCT
0036           KK=J+1
0037           GOTO 25
0038        24 DO 4 K=KK,MCT
0039           B8=MAT(K,J)/MAT(J,J)
0040           DO 5 L=J,NCT
0041           B10=B8*MAT(J,L)
0042         5 MAT(K,L)=MAT(K,L)-B10
0043         4 CONTINUE
        C      VALUE OF DETERMINANT
0044         3 B11=B11*MAT(J,J)
0045           B11=B11*MAT(MCT,MCT)
0046           LOW=-MCT
0047           MO=-1
        C      TO DO ONE OR MORE BACK SOLUTIONS
0048           DO 6 MINC=MT,NCT
0049           JFIN=MCT
0050           IX=0
        C      BACK SOLUTION
0051           DO 6 INM=LOW,MO
0052           M=IABS(INM)
0053           B0=-MAT(M,MINC)
0054           B2=MAT(M,M)
0055           B4=(0.0,0.0)
0056           IF(IX) 7,22,7
0057        22 IX=IX+1
0058           GOTO 8
0059         7 MO2=-JFIN
0060           DO 9 INN=LOW,MO2
0061           N=IABS(INN)
0062         9 B4=B4+MAT(M,N)*MAT(N,MINC)
0063           B0=B0-B4
0064           JFIN=JFIN-1
0065         8 IF(C(1).EQ.0.0.AND.C(2).EQ.0.0)GO TO 13
0066        29 MAT(M,MINC)=B0/B2
0067           ANS(M)=B0/B2
0068         6 CONTINUE
0069           DO 40 J=MT,NCT
0070           JJ=J-MCT
0071           DO 40 I=1,MCT
0072       40  ORIG(I,JJ)=MAT(I,J)
0073           IF(LK.GT.0)RETURN
0074           IF(IS.EQ.0)GO TO 34
0075           GO TO 41
        C      CHECK FOR SINGULARITY AND TO SEE IF FIRST TERM = 0
0076        25 JV=J
0077           CC=MAT(J,J)
0078           IF(CX(1).NE.0.0.OR. CX(2).NE.0.0)GO TO 12
0079        11 IF(JV.NE.JSING)GO TO 14
```

```
0080        13 PRINT15
0081           PRINT 100,J,(MAT(K,J),K=1,MCT)
0082       100 FORMAT(10X,I5,8F10.4)
0083           PRINT 21,((MAT(I,J),I=1,MCT),J=1,MCT)
0084           RETURN
0085        14 JV=JV+1
0086           CC2=MAT(JV,J)
0087           IF(CX2(1).EQ.0.0.AND.CX2(2).EQ.0.0)GO TO 11
0088        16 DO 17 JJ=J,NCT
0089           B6=MAT(J,JJ)
0090           MAT(J,JJ)=MAT(JV,JJ)
0091        17 MAT(JV,JJ)=B6
0092           B11=-B11
0093        12 CONTINUE
0094           GOTO 24
         C     PRINT SUBSTITUTIONS BACK INTO ORIGINAL MATRIX
0095        45 DO 20 NNV=1,IS
0096           PRINT 27
0097        44 PRINT23
0098           DO 20 LL=1,MCT
0099           B13=(0.0,0.0)
0100           DO 19 MM=1,MCT
0101        19 B13=ORIG(LL,MM)*MAT(MM,MCT+NNV)+B13
0102           B15=-ORIG(LL,MCT+NNV)
0103           PRINT21,B15,B13
0104        20 CONTINUE
0105           RETURN
         C     PRINT TITLE - THE INVERSE
0106        34 PRINT 26
0107           GO TO 43
         C     PRINT TITLE - VALUES OF UNKNOWNS
0108        41 PRINT 28
0109        43 DO 38 JJ=MT,NCT
0110           PRINT 27
0111           DO 38 II=1,MCT
0112        38 PRINT 10, MAT(II,JJ)
         C     PRINT VALUE OF DETERMINANT
0113           PRINT 18,B11
0114           IF(IS.NE.0)GO TO 45
         C     PRINT IDENTITY MATRIX
0115           PRINT 35
0116           DO 36 K=1,MCT
0117           PRINT 27
0118           DO 36 I=1,MCT
0119           SUM=(0.0,0.0)
0120           DO 37 J=1,MCT
0121        37 SUM=ORIG(K,J)*MAT(J,MCT+I)+SUM
0122        36 PRINT 10,SUM
0123           RETURN
0124           END
```

```
      SUBROUTINE BLK(NM,NN,S11,S12,S21,S22)
      DIMENSION NBP(16),NBK(16)
      DIMENSION MC(8,16),PHA(8,16)
      DIMENSION S11(NM,NM),S12(NM,NM),S21(NM,NM),S22(NM, NM)
      COMMON/C$5/T11(8,8),T12(8,8),T21(8,8),T22(8,8),R11(8,8),R12(8,8),
     C  R21(8,8),R22(8,8),SPACE(7168)
      COMPLEX S11,S12,S21,S22
      COMPLEX T11,T12,T21,T22,R11,R12,R21,R22
      IF(NN.GT.2)GO TO 1
      CALL TWOPT(S11,S12,S21,S22,NM)
      RETURN
1     II=0
      N2=NN
3     N2=N2/2
      IF(N2.LE.0)GO TO 2
      II=II+1
      GO TO 3
2     DO 10 I=1,II
10    NBP(I)=2
      II=II+1
      CALL NTWK(NN,II,NBP,NBK,MC,PHA)
      CALL STRF(NM,NN,II,NBP,NBK,MC,PHA,S11,S12,S21,S22)
      RETURN
      END
```

```
      SUBROUTINE INVS1(NM,NN,S12)
      COMMON/C$5/A1(32),T(32,64),SPACE(4032)
      DIMENSION S12(NM,NM)
      COMPLEX A1,T,S12
      CALL SIMCX(0,S12,NM,T,NN,A1,1)
      RETURN
      END
```

```
      SUBROUTINE INVS2(NM,NN,S12)
      COMMON/C$5/A1(8),T(8,16),SPACE(7920)
      DIMENSION S12(NM,NM)
      COMPLEX A1,T,S12
      CALL SIMCX(0,S12,NM,T,NN,A1,1)
      RETURN
      END
```

```
      SUBROUTINE TWOPT (S11,S12,S21,S22,M)
      DIMENSION S11(M,M),S12(M,M),S21(M,M),S22(M,M)
      COMPLEX S11,S12,S21,S22
      COMMON/C$4/A,B,C,D
      BCD=B+C+D
      IF(BCD.GT.0.)GO TO 1
      AR=10.**(-A*.05)
      IF(A.LE.0.)AR=0.
      A1=SQRT(.5-AR*AR)
      A2=SQRT(.5-AR*AR)
      B1=AR
      B2=AR
      GO TO 2
    1 B1=10.**(-A*.05)
      IF(A.LE.0.)B1=0.
      B2=10.**(-B*.05)
      IF(B.LE.0.)B2=0.
      A1=10.**(-C*.05)
      A2=10.**(-D*.05)
    2 S11(1,1)=B1*CMPLX(0.,-1.)
      S11(2,2)=B1*CMPLX(0.,-1.)
      S22(1,1)=B1*CMPLX(0.,-1.)
      S22(2,2)=B1*CMPLX(0.,-1.)
      S11(1,2)=B2*CMPLX(-1.,0.)
      S11(2,1)=B2*CMPLX(-1.,0.)
      S22(1,2)=B2*CMPLX(-1.,0.)
      S22(2,1)=B2*CMPLX(-1.,0.)
      S12(1,1)=A1*CMPLX(1.,0.)
      S12(2,2)=A1*CMPLX(1.,0.)
      S21(1,1)=A1*CMPLX(1.,0.)
      S21(2,2)=A1*CMPLX(1.,0.)
      S12(1,2)=A2*CMPLX(0.,-1.)
      S12(2,1)=A2*CMPLX(0.,-1.)
      S21(1,2)=A2*CMPLX(0.,-1.)
      S21(2,1)=A2*CMPLX(0.,-1.)
      RETURN
      END
```

```
      SUBROUTINE TRT(NTP,TRFF,LL,NMX)
      COMMON/C$6/APMT(32,32),ANGL(32,32),ANGT(32),TRID(32,32),SPACE(4064
     *)
      DIMENSION TRFF(NMX,NMX)
      COMPLEX TRFF,TRID
      IF(LL.GT.0)GO TO 1
      DO 10 I=1,NTP
      DO 10 J=1,NTP
   10 TRID(I,J)=TRFF(I,J)
      RETURN
    1 DO 20 I=1,NTP
      DO 20 J=1,NTP
   20 TRFF(I,J)=TRID(I,J)
      RETURN
      END
```

```
      SUBROUTINE TRFIDL(NTP)
      COMMON/C$6/AMPT(32,32),ANGL(32,32),ANGT(32),TRFF2(32,32), TR(32,32
     C),SUMR(2016)
      COMPLEX TRFF2,TR
      PI=3.1415926536
      PI2=PI*2.
      RTA=180./PI
      A=SQRT(1./NTP)
      P=-PI/NTP
      DO 10 I=1,NTP
      PP=(I-1)*P
      PR=P*(I-.5)*2.
      DO 10 J=1,NTP
      PP=AMOD(PP,PI2)
      RE=A*COS(PP)
      RI=A*SIN(PP)
      TRFF2(I,J)=CMPLX(RE,RI)
      AMPT(I,J)=A
      ANGL(I,J)=PP*RTA
   10 PP=PP+PR
      RETURN
      END
```

```
      SUBROUTINE PHASAN (TAINC,I)
      COMMON/C$6/CONTA(4096),SINTA(4096)
      PI=3.1415926536
      PI2=PI*2.
      TA=0.
      I=0
    1 I=I+1
      CONTA(I)=COS(TA)
      SINTA(I)=SIN(TA)
      TA=TA+TAINC
      IF(TA.GE.PI2)RETURN
      GO TO 1
      END
```

```
0001          COMPLEX FUNCTION AR(AUG)
0002          AMP=1.
0003          AG=AUG
0004          RE=AMP*COS(AG)
0005          RI=AMP*SIN(AG)
0006          AR=CMPLX(RE,RI)
0007          RETURN
0008          END
```

```
0001          FUNCTION CANG(SR)
0002          COMPLEX SR
0003          A1=REAL(SR)
0004          A2=AIMAG(SR)
0005          CANG=ATAN2(A2,A1)
0006          RETURN
0007          END
```