NTIA Report 87-220

User's Guide for the HF Broadcast Antenna Design and Validation Summary Programs

Eldon J. Haakinson
Susan L. Rothschild
John R. Godwin

U.S. DEPARTMENT OF COMMERCE
Malcolm Baldrige, Secretary

Alfred C. Sikes, Assistant Secretary
for Communications and Information

July 1987

CONTENTS

CONTENTS (continued)

LIST OF FIGURES

User's Guide for the HF Broadcast Antenna Design and Validation Summary Programs

Eldon J. Haakinson, Susan L. Rothschild, and John R. Godwin*

A series of programs have been developed by the Institute for Telecommunication Sciences (ITS) that assist the Voice of America (VOA) to plan and operate high frequency (HF) broadcast stations. The VOA has specific broadcast bands available for use and has particular reception areas to cover. Because of the long distances between the broadcast transmitter and the reception areas, the primary mode of communication is via HF sky wave. Since the ionosphere, which supports HF sky wave, has electrical characteristics that vary with time of day, season of the year, frequency, and sunspot number, the HF broadcast antenna design and validation summary programs must calculate the best frequency and the required antenna power gain and pattern for the various conditions that apply to a particular broadcast scenario. This document will guide a user through the use of the programs, provide samples of the input to each program, and give examples of each program's output.

Key words: curtain array antennas; HF broadcast; ionospheric propagation

1. INTRODUCTION

An international, high frequency (HF) broadcast facility has the function of delivering a specified signal strength with a certain reliability to particular reception areas on a prescribed broadcast schedule. Because of the distances from the facility to the reception areas, the primary mode of transmission is HF sky wave via the ionosphere. Two conditions preclude the use of a single, fixed antenna at the facility: 1) the ionospheric propagation path undergoes diurnal, seasonal, and sunspot cycle changes, and 2) the reception area changes throughout the day depending upon the broadcast schedule. In order to steer the antenna beam in elevation angle to use the most efficient ionospheric propagation mode and to steer the beam in azimuth angle to cover the desired reception area, modern HF broadcast facilities utilize planar arrays of dipoles with electronic beam steering. By controlling the gain and phase of each dipole element of the array, the antenna's main beam can be steered in both elevation and azimuth and the beam can be shaped to cover the reception area.

---

*The authors are with the Institute for Telecommunication Sciences, National Telecommunications and Information Administration, U.S. Department of Commerce, Boulder, CO 80303.

Figure 1 shows a planar array, commonly used for HF broadcast, called a curtain array antenna. The array is composed of horizontally spaced bays, vertically spaced stacks, and a metallic screen behind the array for additional power gain. The CCIR (1984) has published gain patterns for common curtain array configurations, and Kuester (1987) has developed an algorithm for computing the power gain of an idealized curtain array antenna.

Before the system planner can validate an HF broadcast antenna design, the planner must define a location for the transmitter site, define a region (or regions) to receive the broadcast, define a broadcast schedule (such as 4 to 6 p.m. in region A and 6 to 8 p.m. local time in region B), and finally define a candidate antenna design that is to accomplish the tasks. The Institute for Telecommunication Sciences (ITS) of the National Telecommunications and Information Administration (NTIA) has developed a collection of programs to assist the system planner to validate the HF broadcast antenna design. Figure 2 shows how the various programs interface. Note that the user interacts with the programs on the left and right sides of the figure. The program to the left sets up the data file defining the antenna characteristics, the circuits desiring HF ionospheric propagation predictions, and the hours, seasons, and sunspot conditions to make the predictions. The center programs make the antenna gain calculations, perform the ionospheric propagation predictions, and create the data base. The predictions are made using the NTIA/ITS Ionospheric Communication Analysis and Prediction (IONCAP) program (Teters et al., 1983). The program to the right allows the user to select specific broadcast situations and to summarize the antenna performance based upon the criteria set by the U.S. Information Agency/Voice of America (VOA). The VOA has defined the requirements for the design and validation programs in VOA Engineering Standard 16775.01 (VOA, 1985). The summary programs will indicate whether the proposed antenna is adequate for the reception regions to be covered and over the ionospheric conditions selected.

The following sections describe the functions of each of the programs and the user's interaction with the data setup program and the summary program. The user of the programs is assumed to be familiar with program IONCAP and the VOA Engineering Standard 16775.01, since little explanation of either will be given in this document.

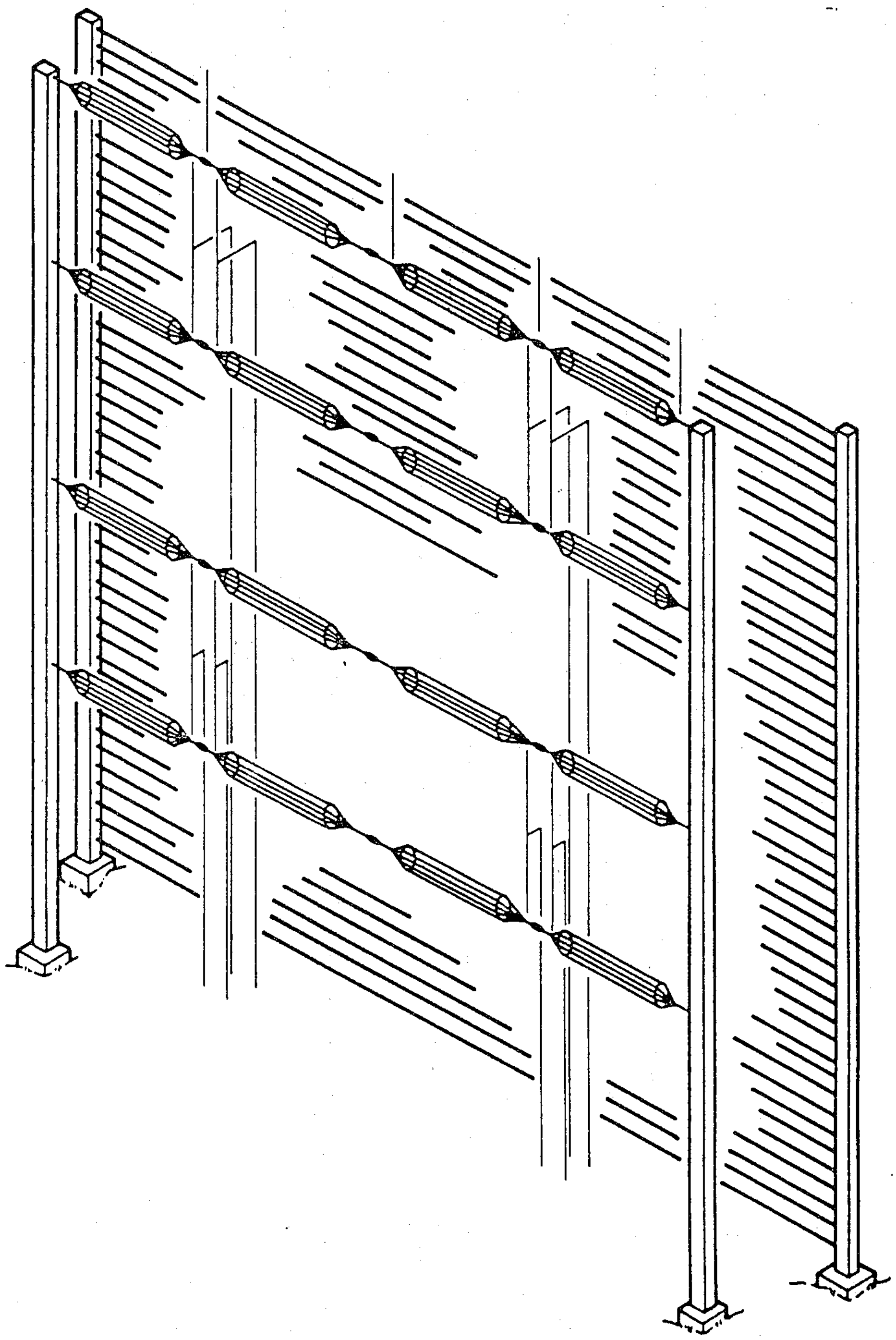

Figure 1. Curtain array with conducting screen (2 bays, 4 stacks).

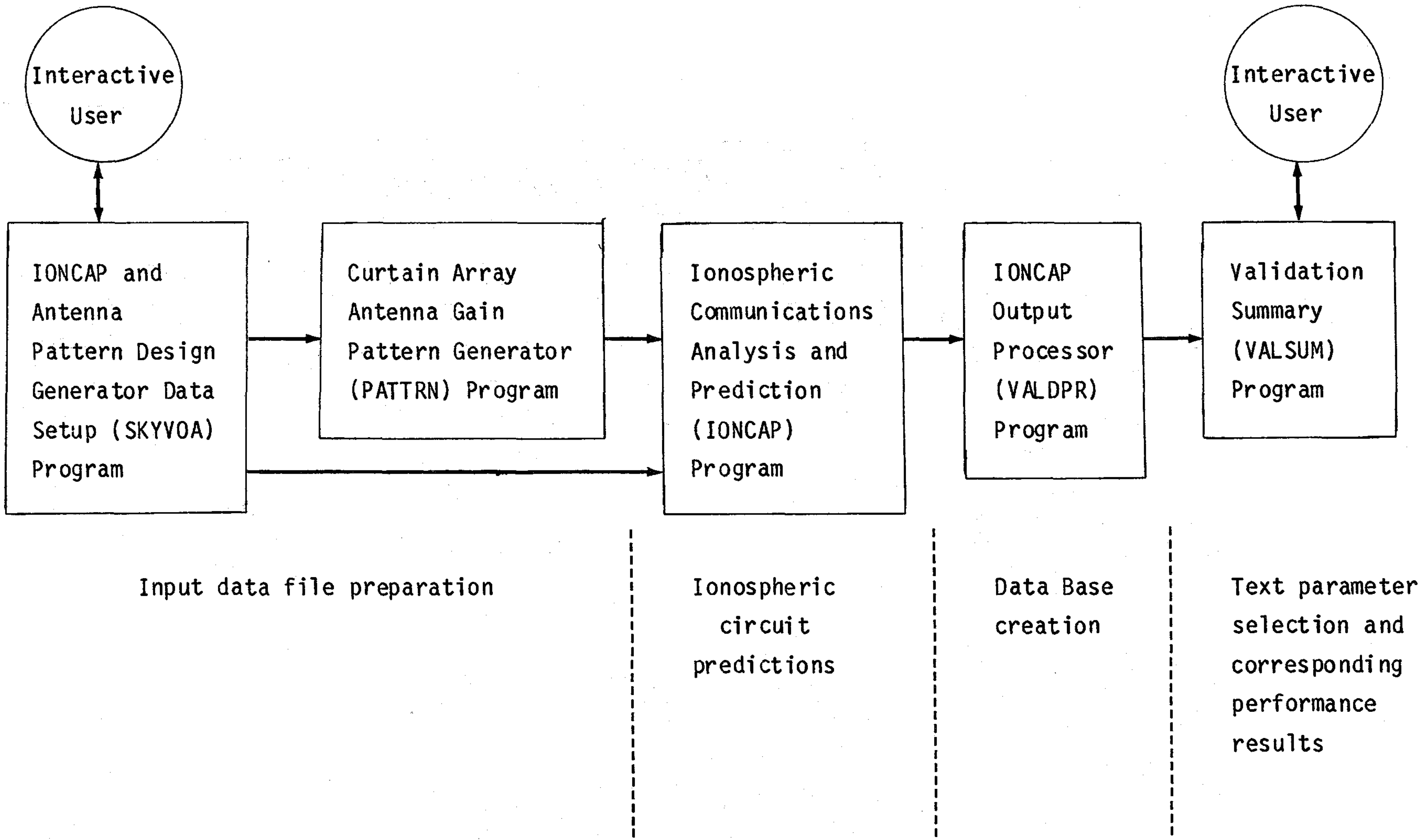

Figure 2. Interrelation between the HF antenna design and validation programs.

2. DATA PREPARATION FOR PROGRAM VALSUM

2.1 General

Programs SKYVOA, PATTRN, IONCAP, and VALDPR (see Figure 2) must be executed before VALSUM can be run. Using the data supplied by the user as defined in Section 3.2 Analysis Input Parameters of VOA (1985) Standard 16675.01, IONCAP will produce the desired output tables for the specified circuits. These output data must be further culled by VALDPR to retain those parameters needed by VALSUM and to create the data base VALSUM uses to produce the summaries.

This portion of the report will give guidelines for the setup and operation of IONCAP, and to the function of VALDPR. Since VALDPR's input and output are machine dependent, this report shows an example of VALDPR's implementation with the warning to the user that other schemes may have to be used with other machines.

Because of their length, the source code listings for programs SKYVOA and VALSUM are not included in this document. Requests for the source code should be made to the Voice of America.

2.2 Program SKYVOA

2.2.1 General

The data and control information for program IONCAP must be created by program SKYVOA. SKYVOA is an interactive program that allows the user to select various parameters used for IONCAP input and the antenna parameters used to create the antenna pattern. The user will be prompted for all the inputs necessary to run IONCAP. SKYVOA then creates an input file for IONCAP in the format described in VOA (1985) Standard 16775.02 which contains the IONCAP User's Manual (Teters, et al., 1983). It also creates a data file for Program PATTRN, the antenna pattern generator.

The IONCAP runs make use of the antenna patterns created by the antenna pattern generator prior to the execution of IONCAP. If the validation level is 0 (the boresight bearing is set to point the antenna main beam directly at the receiver), there is only one antenna pattern used for each transmitter site. If the validation level is 1 (the boresight bearing is selected by the user and off-boresight azimuth to each receiver is calculated), the antenna pattern generator creates a different antenna pattern for each transmitter-receiver combination. In this case, there must be antenna information associated with each circuit.

The program will start after the user properly logs on to the system and types SKYVOA. An example of the dialog between the user and the program is given in Appendix A. The following sections describe the various parts of the dialog with the text that the program prints at the beginning of each section followed by an explanation.

2.2.2 Menu

Program Prints:

```
    Choose from the menu:
           V = Verbose dialog
           C = Concise dialog
           E = Edit data
           S = Summary of data
           P = Process last data set entered
           Q = Quit
    Menu (Verbose)?
```

Explanation:

The user is presented with a menu that allows for either verbose or concise dialog screen presentations for entering the data. Another item on the menu is an "edit data" mode which allows the user to return to a user specified question number and to change the appropriate values. The "summary of data" option provides a list of all the input questions and the user responses. The "process last data set entered" permits the user to execute the data stream that has been entered, reviewed, or edited. The "quit" option allows the user to exit from the SKYVOA program.

To return to the menu from any question, the user should type two colons, i.e., "::".

2.2.3 Input Data Questions

Selecting either CONCISE or VERBOSE from the menu selection causes the program to begin asking the user for responses to the input data questions. In this manual, the "verbose" mode contains the text following the "program prints" statement up to and including the line with the question number followed by text. When the user selects the "concise" mode, only the question number followed by text is printed. If the user is running the concise mode and wishes a verbose explanation, he can type "??" and the verbose explanation will be printed. The following parameters require values selected by the user in order to carry out the calculations and produce the output tables and plots.

2.2.4 Transmitter Site Name

Program Prints:

```
    Transmitter site name
           TNGR = Tangier
           UDRN = Udorn
           PTRC = Puerto Rico
           CRSN = Sri Lanka
           BOTS = Botswana
    1)  Tranmitter site name (TNGR)?
```

Type site lat (followed by carriage return) and site lon (return) for each of the sites. Enter the reference site location first.

```
    Limits are -  0N <= lat <=  90N
                  0S <= lat <=  90S
                  0W <= lon <= 180W
                  0E <= lon <= 180E
```

The default hemispheres are N and E. The S and W locations can be specified by adding an S to the latitude value or adding a W to the longitude value.

Inputs of the form X,Y,Z, imply degrees, minutes and seconds.

Inputs of the form X.Y imply decimal degrees.

```
          1) Transmitter site lat (43.0000 deg N or 43 0 0 dms N)?
          1) Transmitter site lon (26.0000 deg E or 26 0 0 dms E)?
```

Explanation:

Transmitter site names are permitted to have four alpha-numeric characters. These characters will be entered on the label card image of IONCAP. SKYVOA searches a file for the coordinates of the transmitter site. If the coordinates are found, they will be displayed on the screen; otherwise, the user is asked to input the transmitter site latitude and longitude.

Note that for this and all other questions, a default value is entered in parentheses for each question. By typing a carriage return to the question, the program understands that the default value is to be used as the response to the question. Otherwise, the user enters the desired value and a carriage return.

2.2.5 Power

Program Prints:

```
    Power delivered to transmitting antenna (between    .10 and 1000.0 kW)
    2) Power delivered to transmitting antenna (   1.00 kW)?
```

Explanation:

Power delivered to the transmitting antenna is the radio frequency power applied to the input terminals of the transmitting antenna in kilowatts.

2.2.6 Main Beam Direction

Program Prints:

```
Main beam direction toward receiver
       Y = Yes
       N = No
3)  Main beam direction toward receiver (Yes)? N
       Bearing from reference site.
       Enter in degrees clockwise from north,
       i.e. north = 0, east = 90, south = 180, west = 270.
       Answer can be in decimal degrees (X.Y) or in deg., min., sec.
       (X,Y,Z), and must be between 0.0 and 360.0 degrees
4) Bearing (   0.0 deg)?
```

Explanation:

If the main beam direction is toward the receiver, the bearing is set to 0.0 and the validation level is set to 0. This means that PATTRN will calculate one vertical cut transmitter antenna pattern that will be used for all of the circuits. If the main beam is not toward the receiver, the user is asked for the bearing and the validation level is 1. IONCAP will then use a different vertical cut antenna pattern for each circuit as calculated by PATTRN for the bearing between the transmitter and the circuit.

2.2.7 Antenna Name

Program Prints:

```
Antenna name (up to 8 characters)
5) Antenna name (BOTS111)?
```

Explanation:

The antenna name is the user's identification of the antenna data base. Associated with each antenna name will be a transmitter site latitude and longitude, language, validation level, antenna bearing, and antenna configuration. This information will be displayed for the user in VALSUM when the user is asked to choose an antenna from the available IONCAP data (VALSUM question 3).

2.2.8 Antenna Configurations

Program Prints:

```
    Antenna configurations:
           L = List current set of antenna configurations
           D = Delete an antenna configuration
           A = Add an antenna configuration
           C = Change an antenna configuraion
           N = No change
    10)  Antenna configuration (Add)?

    Up to 12 frequencies (between 6 and 26 MHz)
    (A carriage return exits this mode)
           Frequency band (6.1)?

    Number of horizontal bays (1 to 14)
           Number of bays (4)?

Curtain excitation
Mode
 1          +00
 2          0+0
 3          00+
 4          ++0
 5          +0+
 6          0++
 7          +-0
 8          +0-
 9          0+-
10          +++
11          ++-
12          +-+
13          +--
           Curtain excitation ( 1)?

Horizontal slew angle from the normal to the array
The 15 slew angles are:
    -30            30
    -25            25
    -21            21
    -17            17
    -13            13
    - 9             9
    - 5             5
      0
           Horizontal slew angle ( 0)?

    Frequency at which the antenna is designed between 8.00 and 20.00 MHz
           Antenna design frequency (8.75 MHz)?

    Height above ground of lowest stack (1.00 to 100.0 m)
    One-half wavelength stack height above ground at 8.75 MHz is 17.13 meters
           Lowest stack height above ground (17.13 m)?
    10) Antenna configuration (Add)?
```

Explanation:

An antenna configuration consists of the antenna's frequency, number of horizontal bays, curtain excitation mode, horizontal slew angle, antenna design frequency, and height above ground of lowest stack. The user may enter one antenna configuration for each frequency with a maximum of 12 frequencies. Each frequency may be used for only one antenna configuration. Definitions of the antenna parameters and details on the antenna pattern calculations are described in the report by Kuester (1987).

2.2.9 AM and PM Time Blocks

Program Prints:

```
AM time of day to begin analysis and predictions
(between 0100 and 2400 hours UT)
11) AM start time (0100)?

AM time of day to end analysis and predictions
(between 0100 and 2400 hours UT)
12) AM end time (1200)?

PM time of day to begin analysis and predictions
(between 0100 and 2400 hours UT)
13) PM start time (1300)?

PM time of day to end analysis and predictions
(between 0100 and 2400 hours UT)
14) PM end time (2400)?
```

Explanation:

The user must input two time blocks, one AM and one PM, of the daily broadcast schedule. In the example, IONCAP will be run for the hours 04, 05, 06, 07, 16, 17, 18, 19, and 20. The data associated with these hours will be used to find one best frequency using the appropriate algorithms. VALSUM is currently set so that only one broadcast schedule of daily time blocks can be implemented for the entire month and sunspot number sequence.

2.2.10 Broadcast Language

Program Prints:

```
Broadcast language
       The first four characters of the language are entered; for example,
       POLI = Polish
       ENGL = English
21) Language (POLI)?
```

Explanation:

The first four characters of a language file must be entered. SKYVOA will search the named language file for test points that are candidates for circuits to be processed by IONCAP. The following parameters will cull the list of test points in the language file.

2.2.11 Language Area

Program Prints:

```
Language area under which to categorize data base (four characters)
       ex. ENG2 = English to Africa
22) Language Area (TEST)?
```

Explanation:

When running VALSUM, the user will be asked for the language category and will be given a list of all the available data bases for that category. The user will then be able to select the data base to be used in VALSUM.

2.2.12 Sector

Program Prints:

```
Sector beginning bearing (0.0 to 360.0 deg)
23) Sector beginning bearing (30.0 deg)?

Sector ending bearing (0.0 to 360.0 deg)
24) Sector ending bearing (50.0 deg)?

Sector beginning range (0.0 to 10000.0 km)
25) Sector beginning range (1000.0 km)?

Sector ending range (1001.0 to 12000.0 km)
26) Sector ending range (5000.0 km)?
```

Explanation:

The user must select the area for which the test point IDs are to be included. SKYVOA then searches the language file and keeps any test point IDs which are in the sector.

2.2.13 Test Point List

Program Prints:

```
Modify TEST POINT ID list
       A = Add
       C = Change
       D = Delete
       L = List
       N = No change
27) TEST POINT ID list (A)?
```

Explanation:

The test point ID list contains all test points found in the language file that are also in the sector. The user can list the test points, delete test points from the list, or add test points to the list. For example, if the user wishes to add a test point, the user should type "A" to question 27. The program will respond with:

```
TEST POINT IDs to be used in calculations (5 numeral value; 1 to 400 Test
Point IDs)
       10XXX = TEST POINT ID for region 1
       20XXX = TEST POINT ID for region 2, etc.
       a carriage return terminates TEST POINT ID input
TEST POINT ID?

Type site lat (followed by carriage return) and site lon (return) for each
of the sites.  Enter the reference site location first.
       Limits are - 0N <= lat <=  90N
                    0S <= lat <=  90S
                    0W <= lon <= 180W
                    0E <= lon <= 180E
The default hemispheres are N and E.  The S and W locations can be
specified by adding an S to the latitude value or adding a W to the
longitude value.
Inputs of the form X,Y,Z imply degrees, minutes, and seconds.
Inputs of the form X.Y imply decimal degrees.
27)  TEST POINT ID 10009 site lat (  0.0000 deg N or  0 0 0 dms N)?
27)  TEST POINT ID 10009 site lon (  0.0000 deg E or  0 0 0 dms E)?
```

If an invalid ID is entered, the program will respond, for example, with:

```
TEST POINT ID 10999 is not in the sector.
```

Currently the program is limited to 400 test point circuits. The user must limit the number of test point IDs in the list to 400.

When all test point IDs are entered as desired, the user should respond with an "N" to question 27, indicating no change.

2.2.14 Summary and Process

Program Prints:

Do you want a summary of the input data (Y or N)?
Do you want to process this data (Y or N)?

Explanation:

The user may elect to print a summary of all the parameters and values chosen. In the summary, each parameter has a question number associated with it and the value selected. After reviewing the summary, the user can return to the menu, select EDIT, and modify parameters by entering the question numbers of the parameters to be changed.

After the summary question, the user is asked if the data set is to be processed. If "yes", the data are processed and the user is returned to the menu. The "processed data" are a set of input data that programs IONCAP and PATTRN use, as described in the next section. Figure 3 shows the input data set (card images) for program IONCAP and Figure 4 shows the input data set for program PATTRN.

```
PROCEDURE CIRCUITS
LABEL     MIDB10074
CIRCUIT    3083N     3527E     4800N     2300E
ANTENNA      2   18                                                   -1
ANTENNA      1   18
EXECUTE
LABEL     MIDB10014
CIRCUIT    3083N     3527E     4880N     2250E
ANTENNA      1   18
EXECUTE
LABEL     MIDB10346
CIRCUIT    3083N     3527E     4900N     2500E
ANTENNA      1   18
EXECUTE
END
COMMENT   DATE                COMPUTER  BOULDER CYBER 840
COMMENT   TRANSMIT LOCATION   MIDB LATITUDE    30.83000 LONGITUDE   35.27000
COMMENT   ANTENNA ID VALTEST   VALIDATION   1    BEARING   334.9 DEG NO CONF 5
COMMENT   ANTENNA FRQ   610 BAYS  4 EXCT 10 SLEW   0 DSGN FRQ   875 STKH  1713
COMMENT   ANTENNA FRQ   970 BAYS  4 EXCT  4 SLEW   0 DSGN FRQ   875 STKH  1713
COMMENT   ANTENNA FRQ  1370 BAYS  4 EXCT 10 SLEW   0 DSGN FRQ  1900 STKH   789
COMMENT   ANTENNA FRQ  1772 BAYS  4 EXCT  4 SLEW   0 DSGN FRQ  1900 STKH   789
COMMENT   ANTENNA FRQ  2588 BAYS  4 EXCT  1 SLEW   0 DSGN FRQ  1900 STKH   789
COMMENT   STARTHRAM  1  ENDHRAM  5  STARTHRPM 14  ENDHRPM 21
COMMENT   LANGUAGE             MIDB
COMMENT   SECTOR     BEARING   334.9 -   7.8 DEG DIST   1524.7 -  2427.0 KM
METHOD      15
ANTENNA      2   18
ANTENNA      1   18
EXECUTE
ANTENNA      1   18
EXECUTE
ANTENNA      1   18
EXECUTE
METHOD      20    1
SYSTEM    354.0 145.  .10  90.73.00 3.00  .10
FPROB      1.0  1.0  1.0  0.0
FREQUENCY  6.10 9.7013.7017.7225.88
SUNSPOT    10.0120.0
MONTH      1986    1    4    7   10
TIME         1    5    1    1
CIRCUITS
TIME        14   21    1    1
CIRCUITS
QUIT
EOR
```

[Note: Cards and input fields are explained further in IONCAP User's Manual (Teters et al., 1983).]

Figure 3. An example input data set for program IONCAP as created by program SKYVOA.

```
  30.83   35.27   5  28 1 334.9
 610    4   10    0  875 1713
 970    4    4    0  875 1713
1370    4   10    0 1900  789
1772    4    4    0 1900  789
2588    4    1    0 1900  789
  48.00   23.00
  48.80   22.50
  49.00   25.00
EOR
```

Figure 4. An example data set for program PATTRN as created by program SKYVOA.

2.3 Program IONCAP

2.3.1 Modifications to IONCAP

IONCAP writes its output to a file that in most cases is assumed to be the line printer file. That file must be saved (on magnetic tape or as a disk file) for processing by VALDPR. The preservation of the file can be controlled by the site job control structure.

A second consideration is the calculation by IONCAP of long circuit paths exceeding 10000 km. For these cases, IONCAP will produce different output tables than those expected by VALDPR. The user must avoid selecting paths longer than 10000 km.

2.3.2 IONCAP Input Data

A sample of the IONCAP input data as created by Program SKYVOA (see Section 2.2) is shown in Figure 3. The test point ID (e.g. MIDB10012) must be specified in a LABEL card in the IONCAP input; this is used by VALDPR to identify circuits.

The antenna card of the IONCAP input must be set to antenna type 18 for IONCAP to read an external antenna file (see Control Card 43 of VOA Standard 16775.02). The antenna pattern data must be on the antenna file (LU26). Appendix B provides one possible program, called PATTRN, that could be used to create the antenna file.

2.3.3 IONCAP Output Data

The output of IONCAP must be saved on a medium that can be read by VALDPR. The implementation given here uses magnetic tape with the format given in Section 2.4.2.

2.4 Program VALDPR

2.4.1 Function

The purpose of VALDPR is to create data bases from IONCAP's output that can be used by VALSUM. The data base names are six characters each, the last two are always "DB". The structure of this data base will vary with the computer on which VALSUM is implemented. In the original implementation, the data base consists of a large array IVALDT, which can store 4 IONCAP output variables for each combination of 12 frequencies, 4 months, and 2 sunspot numbers, up to 24 hours, and up to 400 circuits. The array is stored in virtual memory. For computers that do not permit initialized virtual memory, IVALDT could be a function call that accesses a random-access-disk data base.

2.4.2 Program VALDPR's Input Data Format

Program VALDPR reads a tape of output from program IONCAP. The tape is blocked in 128 80-character logical records to give 10240 characters for each physical record. The tape density is 1600 BPI on 9-track tapes and the ASCII character code is used to allow data portability. Each tape can contain a complete set of data (2 sunspot numbers, 4 months, up to 24 hours, up to 12 frequencies) for approximately 80 circuit paths. Each tape has three sections: the IONCAP control statements; the antenna gain pattern data; and the performance analysis tables. Each section is described below.

<u>IONCAP control statements</u> At the beginning of each tape is a list of the IONCAP control statements used to set up the IONCAP executions (see Figure 3 for an example). Program VALDPR reads the control statements and selects those statements that have the phrase COMMENT in the first seven columns. The COMMENT statements include information about the data base name, transmitter site name, language category, validation level, antenna bearing, broadcast sector, broadcast time blocks, number of transmit frequencies, data base creation date, and transmitter latitude and longitude. Associated with each operating frequency are the antenna characteristics: the number of bays, the excitation mode, the horizontal slew angle, the design frequency, and the lower antenna height. Program VALDPR stores these data on file VALIND, which provides an index to all of the IONCAP data bases. Figure 5 shows an example of file VALIND.

File Name	Site Name	Ant Name	Lang	Val	Ant Brg	Sector Brg		Sector Range	
VBNH	MIDB	SRMODE2	UKR1	1	352.0	334.9	7.8	1524.7	2427.0

TIME BLOCKS AM	TIME BLOCKS PM	NUM CONFS	DATE	TLAT	TLONG
1- 5	14-21	9	06 FEB 1987	30.8	35.3

FREQ	BAYS	EXCIT	SLEW	DSGN FRQ	STK HGT
720	4	4	0	875	1713
1185	4	1	0	875	1713
1535	4	4	0	1900	789
2165	4	4	0	1900	789
607	4	10	0	875	1713
970	4	4	0	875	1713
1370	4	10	0	1900	789
1772	4	4	0	1900	789
2588	4	1	0	1900	789

Figure 5. An example of the index file VALIND which contains the list of available data bases for VALSUM. This contains the information for data base VBNH.

The test point ID, month, sunspot number, hour, and frequency are used to index the data base. In addition, the azimuth from the transmitter to the receiver in degrees and the circuit ground path distance in kilometers are stored.

Antenna gain pattern data The antenna patterns generated by program PATTRN are written to tape when IONCAP Method 15 is executed. If the main beam direction is toward the receiver, the IONCAP tape will contain the receiver antenna pattern and one transmitter antenna pattern following the IONCAP input. If the main beam direction is not toward the receiver, the receiver antenna pattern will be followed by a transmitter pattern for each test point ID. An example of the Method 15 output is shown in Figure 6, and that output is described in Teters, et al., (1983).

```
1                    METHOD 15   IONCAP 85.04   PAGE   1

ITS- 1 ANTENNA PACKAGE                           ANTENNA PATTERN

FREQUENCY RANGE  ANTENNA TYPE     HEIGHT   LENGTH    ANGLE   AZIMUTH    EX(
1)     EX(2)     EX(3)    EX(4)  CONDUCT.  DIELECT.
  2.0 TO  30.0    CONS17               .000     .000      .000      .000      .0
00      .000      .000      .000      .010    10.000
        2     3     4     5     6     7     8     9    10    11    12    13
14    16    18    20    22    24    26    28    30
 G 28 -12.0 -12.0 -12.0 -12.0 -12.0 -12.0 -12.0  -7.8  -7.8  -7.8  -7.8  -8.0 -
8.0  -8.0  -5.5  -5.5  -5.5  -5.5  -5.5  -5.5  -5.5 28 G
 R 26  -1.1  -1.1  -1.1  -1.1  -1.1  -1.1  -1.1   -.2   -.2   -.2   -.2    .5
 .5    .5 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 26 R
 E 24   4.6   4.6   4.6   4.6   4.6   4.6   4.6   3.8   3.8   3.8   3.8   5.6
5.6   5.6  -9.8  -9.8  -9.8  -9.8  -9.8  -9.8  -9.8 24 E
 E 22   8.5   8.5   8.5   8.5   8.5   8.5   8.5   6.4   6.4   6.4   6.4   9.1
9.1   9.1 -10.8 -10.8 -10.8 -10.8 -10.8 -10.8 -10.8 22 E
 S 20  11.3  11.3  11.3  11.3  11.3  11.3  11.3   8.1   8.1   8.1   8.1  11.7 1
1.7  11.7 -15.5 -15.5 -15.5 -15.5 -15.5 -15.5 -15.5 20 S
 18  13.4  13.4  13.4  13.4  13.4  13.4  13.4   9.2   9.2   9.2   9.2  13.6  1
3.6  13.6    .2    .2    .2    .2    .2    .2    .2 18
 16  14.8  14.8  14.8  14.8  14.8  14.8  14.8   9.8   9.8   9.8   9.8  14.9  1
4.9  14.9   6.7   6.7   6.7   6.7   6.7   6.7   6.7 16
 14  15.7  15.7  15.7  15.7  15.7  15.7  15.7   9.9   9.9   9.9   9.9  15.8  1
5.8  15.8  10.7  10.7  10.7  10.7  10.7  10.7  10.7 14
 12  16.0  16.0  16.0  16.0  16.0  16.0  16.0   9.6   9.6   9.6   9.6  16.0  1
6.0  16.0  13.1  13.1  13.1  13.1  13.1  13.1  13.1 12
 10  15.8  15.8  15.8  15.8  15.8  15.8  15.8   8.9   8.9   8.9   8.9  15.8  1
5.8  15.8  14.4  14.4  14.4  14.4  14.4  14.4  14.4 10
  8  15.0  15.0  15.0  15.0  15.0  15.0  15.0   7.7   7.7   7.7   7.7  14.9  1
4.9  14.9  14.6  14.6  14.6  14.6  14.6  14.6  14.6  8
  6  13.3  13.3  13.3  13.3  13.3  13.3  13.3   5.7   5.7   5.7   5.7  13.2  1
3.2  13.2  13.7  13.7  13.7  13.7  13.7  13.7  13.7  6
  4  10.4  10.4  10.4  10.4  10.4  10.4  10.4   2.5   2.5   2.5   2.5  10.3  1
0.3  10.3  11.3  11.3  11.3  11.3  11.3  11.3  11.3  4
  2   4.7   4.7   4.7   4.7   4.7   4.7   4.7  -3.3  -3.3  -3.3  -3.3   4.6
        2     3     4     5     6     7     8     9    10    11    12    13
14    16    18    20    22    24    26    28    30
0                         FREQUENCY IN MEGAHERTZ

0                           ANTENNA EFFICIENCY

        .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0
 .0    .0    .0    .0    .0    .0    .0    .0    .0
        2     3     4     5     6     7     8     9    10    11    12    13
14    16    18    20    22    24    26    28    30
0                         FREQUENCY IN MEGAHERTZ
```

[Note: The Method 15 output table usually has lines that are 130 columns wide. Here they are broken into lines that do not exceed 80 columns wide.]

Figure 6. Receiver antenna pattern which is used for all receivers.

```
 ITS- 1 ANTENNA PACKAGE                                   ANTENNA PATTERN

FREQUENCY RANGE  ANTENNA TYPE     HEIGHT    LENGTH     ANGLE   AZIMUTH    EX(
1)      EX(2)      EX(3)      EX(4)  CONDUCT.  DIELECT.
 2.0 TO  30.0   SWWHIP              .000      .000      .000      .000      .0
00       .000       .000       .000      .010     10.000
         2     3     4     5     6     7     8     9    10    11    12    13
14    16    18    20    22    24    26    28    30
 G 28   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2
-.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2 28 G
 R 26   -.1   -.1   -.1   -.1   -.1   -.1   -.1   -.1   -.1   -.1   -.1   -.1
-.1   -.1   -.1   -.1   -.1   -.1   -.1   -.1   -.1 26 R
 E 24    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0
 .0    .0    .0    .0    .0    .0    .0    .0    .0 24 E
 E 22    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0
 .0    .0    .0    .0    .0    .0    .0    .0    .0 22 E
 S 20    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0
 .0    .0    .0    .0    .0    .0    .0    .0    .0 20 S
 18     .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0    .0
 .0    .0    .0    .0    .0    .0    .0    .0    .0 18
 16    -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2
-.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2   -.2 16
 14    -.5   -.5   -.5   -.5   -.5   -.5   -.5   -.5   -.5   -.5   -.5   -.5
-.5   -.5   -.5   -.5   -.5   -.5   -.5   -.5   -.5 14
 12    -.9   -.9   -.9   -.9   -.9   -.9   -.9   -.9   -.9   -.9   -.9   -.9
-.9   -.9   -.9   -.9   -.9   -.9   -.9   -.9   -.9 12
 10   -1.3  -1.3  -1.3  -1.3  -1.3  -1.3  -1.3  -1.3  -1.3  -1.3  -1.3  -1.3  -
1.3   -1.3  -1.3  -1.3  -1.3  -1.3  -1.3  -1.3  -1.3 10
  8   -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  -
1.8   -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  -1.8  8
  6   -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  -
3.2   -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  -3.2  6
  4   -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  -
5.4   -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  -5.4  4
  2 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0 -1
1.0 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0 -11.0  2
  0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0 -2
0.0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0 -20.0  0
         2     3     4     5     6     7     8     9    10    11    12    13
14    16    18    20    22    24    26    28    30
0                           FREQUENCY IN MEGAHERTZ

0                             ANTENNA EFFICIENCY

   -4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -
4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -4.8  -4.8
         2     3     4     5     6     7     8     9    10    11    12    13
14    16    18    20    22    24    26    28    30
0                           FREQUENCY IN MEGAHERTZ
```

Figure 7. Sample of transmitter antenna pattern which depends upon transmitter antenna characteristics and transmitter-receiver geometry.

<u>Performance analysis tables</u> The output from IONCAP Method 20 is the final output on the IONCAP tape. Program VALDPR saves the data pertaining to mode, take off angle, required power gain, and multipath probability for each circuit for each sunspot number and hour. An example of this is shown in Figure 8.

```
1                                  METHOD 20   IONCAP 85.04   PAGE   1
          JAN    1986               SSN =  10.
 MIDB10074                                    AZIMUTHS          N. MI.      KM
 30.83 N   35.26 E - 48.00 N   23.00 E    334.88  146.99    1173.9   2173.9
                                   MINIMUM ANGLE    .1  DEGREES
 ITS- 1 ANTENNA PACKAGE
 XMTR    2.0  TO   30.0  CONS17         H      .00 L      .00 A      .0  OFF AZ     .
 RCVR    2.0  TO   30.0  SWWHIP         H      .00 L      .00 A      .0  OFF AZ     .
 POWER = 354.000 KW  3 MHZ NOISE = -145.0 DBW   REQ. REL = .90  REQ. SNR = 73.
 MULTIPATH POWER TOLERANCE =  3.0 DB   MULTIPATH DELAY TOLERANCE =   .100 MS
  UT  MUF
 1.0 10.6   7.2 11.9 15.4 21.7   .0   .0   .0   .0   .0   .0   .0 FREQ
      1F2   1F2  1F2  1F2  1F2    -    -    -    -    -    -    - MODE
     11.9   7.9 12.9 12.9 12.9    -    -    -    -    -    -    - ANGLE
      7.8   7.6  7.8  7.8  7.8    -    -    -    -    -    -    - DELAY
     336.  252. 361. 361. 361.    -    -    -    -    -    -    - V HITE
      .50   .98  .23  .00  .00    -    -    -    -    -    -    - F DAYS
     128.  114. 139. 190. 271.    -    -    -    -    -    -    - LOSS
      61.   68.  46.  -3. -81.    -    -    -    -    -    -    - DBU
      -67   -58  -83 -134 -215    -    -    -    -    -    -    - S DBW
     -164  -159 -166 -169 -173    -    -    -    -    -    -    - N DBW
      97.  102.  83.  35. -42.    -    -    -    -    -    -    - SNR
       2.  -16.  16.  64. 126.    -    -    -    -    -    -    - RPWRG
      .89  1.00  .69  .03  .00    -    -    -    -    -    -    - REL
      .00   .00  .00  .00  .00    -    -    -    -    -    -    - MPROB
      .45   .78  .31  .03  .00    -    -    -    -    -    -    - S PRB
      25.   10.  25.  25.   9.    -    -    -    -    -    -    - SIG LW
      11.    5.  18.  25.   5.    -    -    -    -    -    -    - SIG UP
      61.   68.  46.   6.   2.    -    -    -    -    -    -    - VHFDBU
      25.   10.  25.   6.   6.    -    -    -    -    -    -    - VHF LW
      11.    5.  18.   8.   8.    -    -    -    -    -    -    - VHF UP
        F    F2    F    F    F    -    -    -    -    -    -    - VHFMOD
      26.   12.  26.  26.  11.    -    -    -    -    -    -    - SNR LW
      14.   10.  20.  27.  10.    -    -    -    -    -    -    - SNR UP
```

Figure 8. An example of program IONCAP Method 20 output, the HF circuit predictions.

2.4.3 Program VALDPR Source Code

Because program VALDPR must work with IONCAP output and also develop the data base for VALSUM, VALDPR is heavily burdened with input/output tasks. To aid the user to adapt or develop the program for the user's machine, an example source program for VALDPR is provided in Appendix C to show one possible implementation of VALDPR.

2.5 VALSUM Data Files

VALSUM requires four files: the user input file, which can be created by VALSUM or it can be a file previously created by the user via VALSUM; the language files, such as UKR1, which should contain no more than 400 test point IDs; the circuit file, such as VBNHIN, which contains all of the circuits that have been processed by IONCAP; and VALIND, the file that contains information about the available data bases.

The language file has three fields as shown in Figure 9. The first field in the first five columns contains the test point ID number. The next field in columns 6 through 20 contains the test point latitude and longitude. The last field in columns 21 through 25 contains the assigned population weight for the language at the test point.

10122	04450N03380E	50
10110	04520N02950E	75
10136	04628N03044E	50
10138	04650N03522E	63
10139	04690N03700E	75
10137	04700N03300E	50
10384	04775N03550E	63
10109	04800N02650E	63
10074	04800N02300E	50
10131	04825N02900E	75
10014	04880N02250E	63
10347	04900N03900E	63
10135	04900N03700E	75
10134	04900N03500E	63
10133	04900N03300E	75
10132	04900N03100E	75
10346	04900N02500E	63
10379	05000N03100E	75
10380	05000N02900E	75
10381	05000N02700E	75
10128	05050N03700E	75
10098	05080N02410E	75
10127	05100N03500E	75
10378	05100N03300E	75
10125	05160N02900E	50
10124	05180N02700E	25
10126	05190N03100E	63
10121	05250N03400E	75

Figure 9. An example of the language file used by program VALSUM.

The circuit file contains the list of circuit IDs that have been processed by the IONCAP program. Figure 10 shows an example of this file. The first line contains the data base creation date. The second line contains the transmitter latitude and longitude. The third line contains the number of circuits and number of frequencies. The a.m. start hour, a.m. end hour, p.m. start hour, and p.m. end hour for calculations are contained in line 4. Line 5 has the frequencies (in a 12 * 4 format) at which the predictions have been calculated. The remaining lines contain the circuit ID, bearing (deg E of N) from the transmitter and distance in kilometers from the transmitter for each circuit ID. When a user specifies a particular language, VALSUM compares the test point IDs in the language file with those in the circuit file. This is accomplished by concatenating the transmitter name with the test point ID from the language file. VALSUM then searches through the circuit file looking for a match with the circuits that have been processed by IONCAP. Those test points that fail to match are listed to the user's terminal.

```
Tue 17 Feb 1987
30.83   35.26
23    9
1  5 14 21
720118515352165 607 970137017722588
MIDB10110    344.2   1674.2
MIDB10381    344.5   2239.7
MIDB10124    346.2   2427.6
MIDB10131    346.5   2008.0
MIDB10136    347.8   1767.2
MIDB10380    348.0   2194 2
MIDB10125    349.2   2365.6
MIDB10132    351.1   2051.6
MIDB10379    351.8   2160.7
MIDB10126    352.8   2368.3
MIDB10137    354.5   1808.1
MIDB10133    355.3   2029.0
MIDB10122    355.6   1525.1
MIDB10378    355.9   2250.2
MIDB10121    357.9   2411.4
MIDB10134    359.5   2020.2
MIDB10127    359.5   2242.6
MIDB10138    359.9   1742.2
MIDB10384       .5   1881.2
MIDB10128      3.3   2191.6
MIDB10135      3.7   2025.4
MIDB10139      4.3   1792.8
MIDB10347      7.8   2044.5
```

Figure 10. An example of the data base INDEX file used by program VALSUM.

The fourth file, VALIND, has been discussed in Section 2.4.2 of this report.

3. VALSUM PROGRAM INSTRUCTIONS

3.1 General

VALSUM is an interactive program that allows the user to select various predetermined variables from an output data base produced by IONCAP and to display the IONCAP data in user-selected table formats. The input instructions for IONCAP follow the steps prescribed in paragraph 3.2 of VOA (1985) Standard 16775.01. The output from IONCAP is processed in the manner previously described in Chapter 2 of this User's Guide to form a data file, called IVALDT, for use with VALSUM.

The program will start after the user properly logs on to the system and types VALSUM. An example of the dialog between the user and the program is given in Appendix D. The following sections describe the various parts of the dialog with the text that the program prints at the beginning of each section followed by an explanation.

3.2 VALSUM Data Set Name

Program Prints:

```
    Data set name (must be 6 alpha characters).
        Entering an existing data set name will cause the program to use the
        data set for default values and to replace it with new data.
        Entering a new data name will cause the program to use program
        defined defaults and to create a new data set.
    Data set name?
```

Explanation:

As the user enters values defining the transmitter name, the broadcast language, the test point locations, etc., the program will create a VALSUM data set that saves the user's values for the VALSUM calculations and outputs. Before the user exits the program, the data set is saved on a disk file. At the beginning of the VALSUM program, the user will be asked for a six-character file name. If a previously defined file is named, the data set values in that file will become the default values for the questions during the dialog. If a new file is named, default values will be supplied by the program.

3.3 Menu

Program Prints:

```
Choose from the menu:
     V = Verbose dialog
     C = Concise dialog
     E = Edit data
     S = Summary of data
     P = Process last data set entered
     Q = Quit
Menu (Verbose)?
```

Explanation:

The user is presented with a menu that allows for either verbose or concise dialog screen presentations for entering the data. Other items on the menu are an "edit data" mode which allows the user to return to a user specified question number and change the appropriate values. The "summary of data" option provides a list of all the input questions and the user responses. The "process last data set entered" permits the user to execute the data stream that has been entered, reviewed, or edited. The "quit" option allows the user to exit from the VALSUM program.

To return to the menu from any question, the user should type two colons, i.e., "::".

3.4 Input Data Questions

The following parameters require values selected by the user in order to carry out the calculations and produce the output tables and plots.

3.4.1 Transmitter Site Name

Program Prints:

```
Transmitter site name
     TNGR = Tangier
     UDRN = Udorn
     PTRC = Puerto Rico
     CRSN = Sri Lanka
     BTSW = Botswana
1) Transmitter site name (TNGR)?
```

Explanation:

Transmitter site names are permitted to have four alphanumeric characters. These characters must have been entered on the label card image of IONCAP and must be in the transmitter name location in the IONCAP output file that has been processed by VALDPR and has been made available in the input data file for VALSUM.

Note that for this and all other questions, a default value is entered in parentheses for each question. By typing a carriage return to the question, the program understands that the default value is to be used as the response to the question.

3.4.2 Language Area

Program Prints:

```
Language area used for IONCAP execution
     The first four characters of the language are entered, for example,
              POLI = Polish
              ENGL = English
2) Language area (ENGL)?
```

Explanation:

The language area is the language classification of the IONCAP run. The program will then provide the user a list of all of the antennas that were run for that language/transmitter site name such as shown in Section 3.4.3, if there is at least one available data base. The statement printed by the program to the user that "There is no data for this transmitter site name/language" means that there is no VALDPR file for that transmitter site name/language or that the user entered the wrong transmitter name or language. The program will return to questions 1 and 2.

3.4.3 Transmitter Antenna Number

Program Prints:

Ant Name	Valid Level	Boresight Bearing (deg EofN)	Oper Freq (MHz)	Design Freq (MHz)	Lowest Elem Ht (m)	Bays	Excite Mode	Horiz Slew (deg)
1) SRMODE2	1	352.0	7.20	8.75	17.13	4	4	0
			11.85	8.75	17.13	4	1	0
			15.35	19.00	7.89	4	4	0
			21.65	19.00	7.89	4	4	0
			6.07	8.75	17.13	4	10	0
			9.70	8.75	17.13	4	4	0
			13.70	19.00	7.89	4	10	0
			17.72	19.00	7.89	4	4	0
			25.88	19.00	7.89	4	1	0

```
Transmitter antenna number is to be chosen from the list of available
IONCAP data
3) Transmitter antenna (1)?
```

Explanation:

The user must select the data base to be used in VALSUM from this list of available antennas.

3.4.4 Circuit Selection Option

Program Prints:

```
Circuit selection option
     L =   Language:  all circuits of a particular language are chosen
           from the TEST POINT data base
     S =   Sector:  the points contained within a specific range of
           azimuth and distance are chosen
4) Circuit selection option (Language)?
```

Explanation:

Under the circuit selection option the user may choose either a language or a sector. A language file containing test points of that language must be specified regardless of which option is chosen.

3.4.5 Language

Program Prints:

```
Broadcast language
Language area to be studied is specified.  This area may be a subset of
the language area specified in IONCAP for the chosen antenna.  The four
character code of language is entered.  Any points that do not have data
will be displayed.
5)  Language (UKR1)?
```

Explanation:

The first four characters of a language file must be entered. If the language option was chosen, VALSUM will search the language file to verify that all test points identified in the file have IONCAP data in the IVALDT data base. Those not having data will be listed to the user's terminal.

3.4.6 Sector

Program Prints:

```
Sector beginning bearing (0.0 to 360.0 deg E of N)
6) Sector beginning bearing (30.0 deg E of N)?
```

```
Sector ending bearing (0.0 to 360.0 deg E of N)
7) Sector ending bearing (50.0 deg E of N)?

Sector beginning range (0.0 to 10000.0 km)
8) Sector beginning range (1000.0 km)?

Sector ending range (1001.0 to 12000.0 km)
9) Sector ending range (5000.0 km)?
```

Explanation:

If "sector" was chosen as a circuit option, then the user must answer these questions. VALSUM then searches the language file for the test point IDs which are in the sector, and saves these IDs if they are in the IVALDT data base.

3.4.7 Test Point List

Program Prints:

```
Modify TEST POINT ID list
     A = Add
     C = Change
     D = Delete
     L = List
     N = No change
10) TEST POINT ID list (A)?
```

Explanation:

If "language" was selected previously, the test point ID list includes all test points found in the language file that are also in the IVALDT data base. If "sector" was chosen, the test point ID list consists of all test point IDs in the language file, which are also in the sector. The user may list the test points, delete test points from the list, or add test points to the list. For example, if the user wishes to add a test point, the user should type "A: to question 10. The program will respond with:

```
TEST POINT IDs to be used in calculations (5 numeral value; 1 to 400 Test
Point IDs)
     10XXX = TEST POINT ID for region 1
     20XXX = TEST POINT ID for region 2, etc.
     a carriage return terminates TEST POINT ID input
TEST POINT ID?
```

If an ID number that is not in the data base is entered, the program will respond, for example, with:

```
TEST POINT ID?  10999
Circuit number TNGR10999 is not in data file.
```

Currently the program is limited to 400 test point circuits. The user must limit the number of test point IDs in the list to 400.

When all test point IDs are entered as desired, the user should respond with an "N" to question 10, indicating no change. VALSUM will, in the execution stage, combine the transmit site code with the test point ID to form a nine-place circuit ID which is used to search the IVALDT file for the appropriate variables from the desired circuits.

3.4.8 Spill-Over Region

Program Prints:

```
Spill-over region calculations option
     Y = Yes, calculate power to specified spill-over region circuits
     N = No, no calculations
11) Spill-over calculations (Yes)?
```

Explanation:

The spill-over region is the region considered to be excluded from the best frequency selection process but is a region for which the user wishes to know something about the coverage. If this option is selected, the user must specify the desired circuits in the same manner as was used for the primary coverage area (described above in paragraph 3.4.4 on Circuit Selection Option).

3.4.9 Required Power Gain Cuts

Program Prints:

```
Culling process to be used to calculate the required power gain
     Most stringent is Cut 1; least stringent is Cut 5.  For further
     explanation, see paragraph 3.5 of VOA Std 16775.01
Cut          Brief Description
 1    =      Meets the highest required gains with 100% area coverage
 2    =      Meets 90% of the highest required gains with 100% area
             coverage
 3    =      Meets 90% of all circuit hours with Case 1 Best Frequency
 4    =      Meets 90% of all circuit hours with Case 2 Best Frequency
             (90% circuit reliability for 90% of the locations)
 5    =      Meets 90% of all listener hours with Case 2 Best Frequency and
             population weights
19) Cut number to be used to determine required power gain (5)?
```

Explanation:

The process, defined in VOA Standard 16775.01, paragraph 3.5 and 3.6, describes an interactive design approach consisting of five steps or cuts. The first cut is the most stringent and provides the required power gain needed to achieve the minimum required signal-to-noise ratio on 90 percent of the broadcast hours in a month at the most difficult test points in the specified region. The assumption for the frequency of operation during each broadcast time block is the one that minimizes the highest required power gain in the listening area (see paragraph 3.4.1 in VOA Standard 16775.01 for the Case 1 Best Frequency).

The second cut is the same as Cut 1 with the exception that the highest required gains are limited or clipped at the upper decile value of the distribution of highest required gains from each hour in the broadcast time blocks over the months and sunspot numbers. The Case 1 Best Frequency is used to obtain the required gain values in this cut.

The third cut is the same as the second cut except that the limiting upper decile value is obtained from the entire distribution of required gains from each broadcast hour in the time blocks over the months and sunspot numbers at each test point for the Case 1 Best Frequency.

The fourth cut is the same as the third cut but uses the Case 2 Best Frequency for finding the distribution of required power gain values. The Case 2 Best Frequency algorithm attempts to find the frequency that minimizes the required gain for the majority of the test points while not letting any test point in the listening area at a given time block have a required gain value 6 dB greater than the minimum highest required gain. The procedure for the Case 2 Best Frequency is given in paragraph 3.4.2 of the VOA (1985) Standard 16775.01.

The fifth cut, which is an optional cut, is the same as the fourth cut except that the required power gains for each test point are weighted by the population weighting factor associated with each test point in the test point data file.

3.4.10 Time Block Output

Program Prints:

```
Time blocks given in output
     S  =   Single table for all time blocks combined
     M  =   Multiple tables for each time block and all time blocks
            combined
20) Output time blocks (Multiple table)?
```

Explanation:

The user specifies if the output tables are to be shown for the combined time blocks (i.e., entries from individual time blocks are superimposed in one table showing the worst case or highest required power gain). If so desired, the user selects the "single table" option. If it is also desired to see the required gain values displayed as a function of elevation angle and frequency for each specified time block, then the user selects the "multiple tables" option.

3.4.11 Time Block List

Program Prints:

```
Modify time blocks list
     A = Add
     C = Change
     D = Delete
     L = List
     N = No change
21) Time blocks list (A)?
```

If "A" is selected, then the program prints:

```
Time block starting hour = 1-24 implies 0100 UT to 2400 UT
Time block ending hour   =  1-24 implies 0159 UT to 2459 UT
     a carriage return terminates time block input
Time block starting hour?
Time block ending hour?
```

Explanation:

The user must specify the time blocks (i.e., set of continuous hours) of the daily broadcast schedule. For example, an entry of 05 to 07 hours UT will select the data base associated with these hours from the IONCAP data base (IVALDT). The data associated with these hours will be used to find one Best Frequency using the appropriate algorithms. VALSUM is currently set so that only one broadcast schedule of daily time blocks can be implemented for the entire month and sunspot number sequence. The default time blocks are the two time blocks that were run for IONCAP.

To supply additional time blocks, the user selects "add" from the menu presented. This allows the user to enter a beginning and ending hour for a time block. If more than one time block is desired, the user continues to enter time blocks (to a maximum of 12 blocks) until all of the desired time blocks have been specified. Any additional time blocks must be a subset of the time blocks that were run in IONCAP. A carriage return terminates the time block input. At that point the user may list the blocks entered, edit the list by the "add" or "delete" menu items, and when satisfied with the time block entries, proceed to the next question by entering "no change".

3.4.12 Transmitter Frequency List

Program Prints:

```
Modify HF frequencies list
     A = Add
     D = Delete
     L = List
     N = No change
22) Transmitter frequency (N)?
```

Explanation:

IONCAP was initially set to run transmit frequencies that were specified in SKYVOA. For some design applications it may be desirable to eliminate one or more of the frequencies and their associated data from the VALSUM design process. If so, the user indicates "delete" in order to reject those frequencies. If no change is desired, an entry of "N" will allow the user to proceed to the next question. If the user is returning to this question and wishes to add a frequency previously deleted, he or she may do so. Note that no frequency may be added that was not used in the generation of the IONCAP data.

3.4.13 Number of Transmitter Frequencies

Program Prints:

```
Number of transmitter frequencies to be considered per time block
     1 =  Single frequency per time block (only the best frequency is
          printed)
     2 =  Two frequencies per time block (the best and second best
          frequencies are printed)
23) Number of frequencies per time block (1)?
```

Explanation:

In some design applications it may be desirable to consider use of more than one antenna/transmitter pair to satisfy a language requirement from a single transmit site. The user is allowed to select either the case of one frequency used per time block or the case of two simultaneous frequencies per time block. If the latter case is selected, the best frequency algorithm for the cut chosen is looped through twice per time block with the first Best Frequency eliminated after the first pass. The second pass produces the second Best Frequency. The user should be aware that for difficult coverage situations such as a language area falling in the one-hop/two-hop F_2-layer transition region, the two-frequency selection process may or may not attempt to find the best set of one-hop and two-hop frequencies. The use of test point elimination in VALSUM (described in Section 3.4.4 on Circuit Selection Option) is better suited for separating the listening area into two different coverage regions.

3.4.14 Supplemental Plots and Tables

Program Prints:

```
Calculate and output supplemental plots and tables
24) Output supplemental plots and tables (No)?
```

Explanation:

If the user wishes to see the required power gain values shown as a function of takeoff and azimuth angle for each of the best frequencies selected by VALSUM, then this question should be answered with a "yes" entry.

3.4.15 VALSUM Identification

Program Prints:

```
Identification to be printed on output tables and plots (up to 20
characters)
25) VALSUM identification (VALSUM test run)?
```

Explanation:

The user is allowed 20 characters to label the VALSUM printouts with a desired name.

3.4.16 Sunspot Numbers

Program Prints:

```
Modify sunspot number list
     A = Add
     D = Delete
     L = List
     N = No change
26) Sunspot Number (N)?
```

Explanation:

IONCAP was initially set to run two sunspot numbers. See 3.4.12 Transmitter Frequency List explanation for a discussion on modifying the list.

3.4.17 Months

Program Prints:

```
Modify month list
     A = Add
     D = Delete
     L = List
     N = No change
27) Month number (N)?
```

Explanation:

IONCAP was initially set to run 4 months. See 3.4.12 Transmitter Frequency List explanation for a discussion on modifying the list.

3.5 Summary and Process

Program Prints:

```
Do you want a summary of the input data (Y or N)?
Do you want to process this data (Y or N)?
```

Explanation:

The user may elect to print a summary of all the parameters and values chosen. In the summary, each parameter has a question number associated with it and the values selected. After reviewing the summary, the user can return to the menu, select EDIT, and modify parameters by entering the question numbers of the parameters to be changed.

After the summary question, the user is asked if the data set is to be processed. If "yes", the data is processed and the user is returned to the menu.

4. DESCRIPTION OF VALSUM OUTPUT

4.1 Circuit Data

The first output table lists all of the circuit IDs (i.e., the transmitter site designator and the test point IDs) the user specified either for an entire language area or by keyboard entry to form a region. The longitude and latitude as they appear in the IONCAP output for the receive site location are printed out for each test point along with the great circle route azimuth and distance as seen from the transmit location. Also, the population weighting factor appearing in the test point data file for the user-specified language is listed. (Note: This value is displayed for all cuts, but for cuts other than Cut 5, the value used in the calculations is 1.) Below the circuit listing is a table of transmitter site specific data that provides the following:

Total Azimuth Range	The total arc in the azimuth plane expressed in degrees that describes the receive area as seen from the transmitter site.
Geometric Center	The azimuth in degrees from true north centered on the mid point of the total azimuth range, above.
Azimuth Weighted by Distance	The mean azimuth in degrees from true north weighted by distance so that longer paths tend to influence the azimuth.
Azimuth Weighted by Population	The mean azimuth in degrees from true north weighted by population so that more densely populated test points tend to influence the azimuth.
Azimuth Weighted by Distance and Population	The mean azimuth in degrees from true north weighted by the distribution of population and path distance so that longer circuits in the more heavily populated areas influence the azimuth.
Mean Distance	The average of the distances in kilometers from the transmitter site to the receive test point locations.
Standard Deviation	The standard deviation of the path distances about the assumed mean distance.

An example output is shown in Figure 11 for Cut 5, which includes the population weighting factors from the test point data file.

```
NEW_VALSUM_TEST          7:20 AM   FRI., 13   MAR., 1987 T:MIDB L:UKR1 R:0 Cut:5
Case:2    Main Region    Antenna Name: SRMODE2
```

Circuit ID	Weight	Latitude	Longitude	Azimuth	Distance
10074	50	48.00	23.00	334.88	2173.90
10014	63	48.80	22.50	335.34	2268.60
10346	63	49.00	25.00	339.78	2196.70
10098	75	50.80	24.10	340.61	2404.30
10109	63	48.00	26.50	341.19	2048.60
10110	75	45.20	29.50	344.21	1674.20
10381	75	50.00	27.00	344.45	2239.70
10124	25	51.80	27.00	346.18	2427.60
10131	75	48.25	29.00	346.46	2008.00
10136	50	46.28	30.44	347.76	1767.20
10380	75	50.00	29.00	348.02	2194.20
10125	50	51.60	29.00	349.24	2365.60
10132	75	49.00	31.00	351.14	2051.60
10379	75	50.00	31.00	351.75	2160.70
10126	63	51.90	31.00	352.75	2368.30
10137	50	47.00	33.00	354.49	1808.10
10133	75	49.00	33.00	355.26	2029.00
10122	50	44.50	33.80	355.60	1525.10
10378	75	51.00	33.00	355.89	2250.20
10121	75	52.50	34.00	357.92	2411.40
10134	63	49.00	35.00	359.45	2020.20
10127	75	51.00	35.00	359.53	2242.60
10138	63	46.50	35.22	359.90	1742.20
10384	63	47.75	35.50	.55	1881.20
10128	75	50.50	37.00	3.28	2191.60
10135	75	49.00	37.00	3.65	2025.40
10139	75	46.90	37.00	4.28	1792.80
10347	63	49.00	39.00	7.80	2044.50

```
Total azimuth range:                 32.92 degrees
Geometric center:                   351.83 degrees from true north
Azimuth weighted by distance:       351.56 degrees from true north
Azimuth weighted by population:     352.21 degrees from true north
Azimuth weighted by dist/pop:       351.97 degrees from true north
Mean distance:                     2082.63 kilometers
Standard deviation:                 236.73 kilometers
```

Figure 11. An example circuit output table for program VALSUM Cut 5.

4.2 Required Power Gain Table

The first summary table of ionospheric propagation data shows the required power gain as obtained by the user-selected cut (described in Section 3.4). The required power gains in decibels are presented as a function of elevation angle and frequency. When entries are duplicated, only the largest value is printed. Negative values of required gain indicate that there is excess in relative to 90 percent reliability of achieving the minimum required signal-to-noise ratio used in IONCAP. If an actual transmit antenna pattern was used in IONCAP, then the required power gains are in terms of decibels relative to the pattern assumptions used in IONCAP.

The table presents the elevation angle requirements for all seasons and sunspot numbers for each test point in the specified listening area. An example output is shown in Figure 12. As may be seen, the azimuthal spread is suppressed in this output format. If the azimuthal dependence is desired, the user must request the tables for takeoff and azimuth angle.

NEW_VALSUM_TEST 7:20 AM FRI., 13 MAR., 1987 T:MIDB L:UKR1 R:0 Cut:5
Case:2 Main Region Antenna Name: SRMODE2
Highest/90% Required Power Gains for Best Frequency for All Time Blocks

	Frequency										
Angle	7.20 #	11.85 #	15.35 #	21.65 #	6.07 #	9.70 #	13.70 #	17.72 #	25.88 #	Angle # = Frequency selected	Maximum
17	-10	-19	-19			-15	-18			17	-10
16	-10	-17	-19			-18	-18			16	-10
15	-13	-19	-18			-18	-21			15	-13
14	-13	-21	-21			-20	-21			14	-13
13	-10	-14	-14			-17	-21			13	-10
12	-10	-13	-14			-12	-20			12	-10
11	-10	-11	-16			-15	-20			11	-10
10	-10	-11	-15			-12	-16			10	-10
9	-10	-10	-18			-14	-15			9	-10
8	-10	-10	-16			-14	-16			8	-10
7	-10	-10	-16			-16	-17			7	-10
6	-11	-15								6	-11
5	-13									5	-13
4	-10									4	-10
3	-10									3	-10
2	-10									2	-10
1										1	
0										0	
	*****	23777	14632	0	0	18290	14632	0	0		

Number of Times per Frequency
Total Number of Samples: 59144

SSN	Mean	SD	Upper Decile (calculated)	Upper Decile (actual)
10	-15.7	7.00	-6.7	101.0
120	-20.8	4.99	-14.4	101.0
Both	-18.2	6.59	-9.8	101.0

Figure 12. An example required power gain and elevation angle table for all time blocks for program VALSUM.

The table in Figure 12 presents data for all time blocks or hours, depending on the requested cut number. If "multiple" tables were requested by the user, subsequent tables are output for each time block specified in the VALSUM input. These time-block-dependent tables allow the user to determine the gain requirement for various broadcast schedules. For example, if high required gains are encountered in the time block from 1700 to 2000 hours, the required gain values may be reduced substantially if a frequency change is allowed at 1800 hours.

The user is provided with a quick summary of the highest required gains as a function of takeoff angle at the far right of the table in Figure 12. At the bottom of the table below the frequency columns the user is provided with the number of occurrences when the frequency was found to be the best. Also at the bottom of the table the total sample size is given. This number may be greater than the number of visible entries in the table since only the highest required gains are presented.

A separate table is given below the required gain table for all time blocks in cuts 2 through 5. This table, as may be seen in the example of Figure 12, presents the mean, standard deviation and upper decile for the low, high, and combined sunspot numbers. In cuts 2 through 5, the upper decile value for both sunspot numbers is the limiting value for the required gain values shown in the table above. In other words, no gain values will be shown that exceed the upper decile of the required gains. When a higher value is found, it is printed as the limiting value by VALSUM.

Another table is shown below the required power gain tables for individual time blocks (multiple table option). This table, as shown in Figure 13, gives the Best Frequency found for each month and sunspot number during that time block and for the assumptions corresponding to the user requested required gain cut.

NEW_VALSUM_TEST 7:20 AM FRI., 13 MAR., 1987 T:MIDB L:UKR1 R:0 Cut:5
Case:2 Main Region Antenna Name: SRMODE2
Highest/90% Required Power Gains for Best Frequency by Time Block
Time: 1 to 5 UT

Frequency

Angle	7.20 #	11.85 #	15.35 #	21.65 #	6.07 #	9.70 #	13.70 #	17.72 #	25.88 #	Angle	Maximum # = Frequency selected
29										29	
28										28	
27		-10								27	-10
26		-10								26	-10
25										25	
24		-10								24	-10
23	-10	-10								23	-10
22	-10									22	-10
21	-10	-16								21	-10
20	-10	-20				-10				20	-10 *
19	-10	-24				-11				19	-10
18	-10	-17				-16				18	-10
17	-10	-19				-15				17	-10
16	-18	-17				-18				16	-17
15	-13	-19				-18				15	-13
14	-17	-21				-20				14	-17
13	-13	-18				-17				13	-13
12	-15	-13				-12				12	-12
11	-10	-11				-15				11	-10
10	-10	-11				-12				10	-10
9	-10	-10				-14				9	-10
8	-10	-10				-14				8	-10
7	-12					-16				7	-12
6	-12									6	-12
5	-13									5	-13
4	-10									4	-10
3	-10									3	-10
2	-10									2	-10
1										1	
0										0	
	*****	9145	0	0	0	18290	0	0	0		

Number of Times per Frequency
Total Number of Samples: 7624

Best Frequencies
Month

1	2	3	4	Sunspot Number
7.20	7.20	7.20	7.20	10
7.20	9.70	11.85	9.70	120

Figure 13. An example required power gain and elevation angle table for an individual time block from program VALSUM.

If the user requested that a second Best Frequency be found for each time block, then separate tables, as described above, are printed out for the second Best Frequency.

4.3 Best Frequency For All Time Blocks

The Best Frequency table (see Figure 14) is printed automatically and shows the Best Frequency for each time block, month, and sunspot number. The Best Frequency is dependent on the required gain cut selected by the user. Cuts 1 through 3 will depict the Case 1 Best Frequency while Cuts 4 and 5 will show the Case 2 Best Frequency. If a second Best Frequency is requested, a separate table is produced showing the second Best Frequencies.

```
NEW_VALSUM_TEST          7:20 AM   FRI., 13   MAR., 1987 T:MIDB L:UKR1 R:0 Cut:5
Case:2     Main Region    Antenna Name: SRMODE2
                          Best Frequencies for All Time Blocks
```

	SSN = 10				SSN = 120			
Blk Hr	JAN.	APR.	JULY	OCT.	JAN.	APR.	JULY	OCT.
1	7.20	7.20	7.20	7.20	7.20	9.70	11.85	9.70
2	7.20	7.20	7.20	7.20	7.20	13.70	15.35	11.85

Figure 14. An example best frequency table from program VALSUM.

4.4 Circuits Having Highest Required Gain

The circuit IDs having the highest required gain for the Best Frequency algorithm selected by the user (i.e., Cuts 1-3 use Case 1, and Cuts 4 and 5 use Case 2) are shown in tabular form for the time blocks, months, and sunspot numbers. An example table is shown in Figure 15. This table is useful in the process of editing out troublesome points when coverage is being restricted to only a portion of the listening area. It also indicates when the gain requirement is being driven by a single point or a few points in the listening area. The circuits having the highest required gain may vary depending on the Best Frequency algorithm being used and if a second Best Frequency has been requested.

```
NEW_VALSUM_TEST          7:20 AM   FRI., 13  MAR., 1987 T:MIDB L:UKR1 R:0 Cut:5
Case:2    Main Region    Antenna Name: SRMODE2

              Circuits Having Highest Required Gain by Time Block
                            for Best Frequency

                                 SSN =  10
```

Hours	JAN.	APR.	JULY	OCT.
1 to 5	10122	10139	10347	10347
14 to 21	10074	10074 10014	10074 10014	10074 10014

SSN = 120

Hours	JAN.	APR.	JULY	OCT.
1 to 5	10122	10122	10347	10122
14 to 21	10074 10014	10122	10122	10074 10014

Figure 15. An example circuits with highest required gain table from program VALSUM.

4.5 Summary of Highest Required Gains

This table is related to the previous table (Section 4.4) and gives the actual required power gain for the circuits having the highest required gain. The table, as shown in Figure 16, is also presented in terms of months and sunspot numbers. However, the entries in decibels of required gain are given for each hour within all time blocks specified by the user.

```
|NEW_VALSUM_TEST         7:20 AM   FRI., 13  MAR., 1987 T:MIDB L:UKR1 R:0 Cut:5
Case:2    Main Region    Antenna Name: SRMODE2
                         Summary of Highest Required Gains
                                for Best Frequency
```

		SSN = 10				SSN = 120			
Blk	Hr	JAN.	APR.	JULY	OCT.	JAN.	APR.	JULY	OCT.
1	4	-15.00	-11.00	1.00	-12.00	-17.00	-14.00	-9.00	-15.00
1	5	-12.00	2.00	7.00	-3.00	-14.00	-12.00	22.00	-14.00
2	14	-4.00	7.00	12.00	-1.00	2.00	-15.00	-14.00	-8.00
2	15	-13.00	2.00	11.00	-12.00	-13.00	-17.00	-15.00	-10.00
2	16	-13.00	-11.00	-9.00	-13.00	-15.00	-19.00	-17.00	-11.00
2	17	-14.00	-13.00	-10.00	-14.00	-16.00	-20.00	-18.00	-12.00
2	18	-15.00	-14.00	-11.00	-15.00	-18.00	-22.00	-20.00	-13.00
2	19	-16.00	-15.00	-12.00	-16.00	-18.00	-17.00	-21.00	-14.00

Figure 16. An example highest required gain summary table from program VALSUM.

4.6 Distribution of Most Probable Modes

For each specified circuit, the total number of occurrences of a particular mode as the most probable is shown for each hour period, month, sunspot number, and Best Frequency. This table allows the user to determine if mixed modes are being predicted for the listening area. The table also shows the percent of the total hours each mode is required for the combined listening area. For each mode, the maximum range of takeoff angles is given along with the average value. An example of this table is shown in Figure 17.

```
NEW_VALSUM_TEST          7:20 AM   FRI., 13  MAR., 1987 T:MIDB L:UKR1 R:0 Cut:5
Case:2   Main Region     Antenna Name: SRMODE2
         Distribution of Most Probable Modes for Best Frequency
```

Number of Times Mode Required for Listed Circuits

Mode	Circuit 10128	10098	10127	10378	10125	10124	10126	10121
1F1	0	0	0	0	0	0	0	0
2F1	150	225	150	150	100	75	126	150
3F1	0	0	0	0	0	0	0	0
1F2	7050	6900	6975	6975	4600	2300	5796	6900
2F2	600	675	675	675	500	225	630	750
3F2	0	0	0	0	0	0	0	0
1E	0	0	0	0	0	0	0	0
2E	0	0	0	0	0	0	0	0
3E	0	0	0	0	0	0	0	0

Mode	%	Angle Max.	Avg.	Min.
1F1	.13	9.40	8.37	7.70
2F1	.97	21.70	19.86	18.50
3F1	0.00	0.00	0.00	0.00
1F2	92.47	22.00	10.09	4.50
2F2	4.56	39.00	20.35	15.80
3F2	0.00	0.00	0.00	0.00
1E	.33	3.90	2.80	2.20
2E	1.55	11.60	9.36	7.70
3E	0.00	0.00	0.00	0.00

Total samples: 190216

Figure 17. An example of most probable modes table from program VALSUM.

4.7 Occurrences of Frequency/Takeoff Angle

The number of occurrences that a particular frequency and takeoff angle are required over the full collection of hours, months, and sunspot numbers for all circuits specified by the user is shown in an example table given in Figure 18. This table is useful in describing the range of takeoff angles that must be accommodated by the antenna in order to excite the least-loss ionospheric modes. The percent of the time and locations for which a particular angle is required is shown to the far right of the table. Also, the percent of the time a particular frequency band is required is shown at the bottom of the table below each frequency band column. It should be noted that the frequency distribution is bivariant since only low and high sunspot numbers have been used in the analysis.

NEW_VALSUM_TEST 7:20 AM FRI., 13 MAR., 1987 T:MIDB L:UKR1 R:0 Cut:5
Case:2 Main Region Antenna Name: SRMODE2
Occurrences of Frequencies/Takeoff Angles for All Hours at Best Frequency

Frequency (MHz)

Angle	7.20	11.85	15.35	21.65	6.07	9.70	13.70	17.72	25.88		Angle	% Occurrences per Angle
	#	#	#	#	#	#	#	#	#	# = Frequency selected		
23		251	25								23	.5
22		1040		50							22	1.8
21		2054	100					50			21	3.7
20		2692	100	50			100	50			20	5.1
19		2142	50	100			50	50			19	4.0
18		1439	338	125			50	125			18	3.5
17		1227	614	363			275	201			17	4.5
16		1238	827	376			464	588			16	5.9
15		2378	726	613			713	539			15	8.4
14		4511	840	903			615	388			14	12.3
13		6363	1453	1190			1465	1492			13	20.2
12		7508	2145	1668			2396	1491			12	25.7
11		12019	3106	1792			3168	1804			11	37.0
10		14677	2706	2344			3497	2720			10	43.9
9		20409	4448	2366			2942	2254			9	54.8
8		17799	2918	1703			2067	1916			8	44.6
7		12825	1854	989			488	964			7	28.9
6		6448	751								6	12.2
5		1239									5	2.1
4		100									4	.2
3		351									3	.6
2		175									2	.3
1											1	
0											0	
		-20.6	40.2	24.7	0.0	0.0	30.9	24.7	0.0	0.0		

% Occurrences per Frequency
Total Number of Samples: 59144

Figure 18. An example frequency/takeoff angle table from program VALSUM.

4.8 Spill-Over Region

If the user has specified a spill-over region, then all of the tables previously described will be produced for the set of specified circuits in the spill-over region. The Best or Second Best Frequency used to find the required gain values in the spill-over region are the same as those found for the specified cut number used in analyzing the primary listening area. The user cannot specify a spill-over region if a primary target area has not been previously defined in VALSUM.

4.9 Supplemental Plots and Tables

If the user selected the Supplemental Plots and Tables option, azimuth angle vs. takeoff angle plot and the set of output tables will be generated, one table for each Best Frequency as shown in the Best Frequency table. The table presents the required power gain values as a function of azimuth and takeoff angle. The azimuths are shown relative to the geometric center of the listening area as seen from the transmitter site. These tables are very useful in determining if a particular antenna pattern is adequate in terms of beam position for the intended listening area. An example of this table is given in Figure 19.

NEW_VALSUM_TEST 7:20 AM FRI., 13 MAR., 1987 T:MIDB L:UKR1 R:0 Cut:5
Case:2 Main Region Antenna Name: SRMODE2
Antenna pattern of maximum required gain for best frequency at 7.20 MHz
Maximum cut-off gain set to -9.77 dB
Azimuth of 0 deg is equal to bearing of 351.8 deg

Takeoff Angle

Azim	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
-7		-10			-17	-14	-10	-15	-13	-14	-19	-17	-18	-13	-17	-10	-16	-10	-10	-10	-10		
-6																							
-5				-14	-12	-15	-13	-10	-10	-19	-16	-15	-15		-10	-10	-10	-10	-10	-10			
-4																							
-3	-10				-18	-16	-10	-16	-13	-10	-17	-17	-16	-17	-19		-10	-10	-10	-10	-10		
-2				-16	-13	-15	-14	-19	-12						-10	-10	-10	-10	-10	-10			
-1																							
0					-18	-17	-10	-10	-14	-19	-15	-15	-13					-10	-10	-10	-10		
1				-16	-13	-15	-14	-12	-17						-10	-10	-10	-10	-10	-10			
2	-10							-22	-19	-10	-17	-18	-15	-22	-19								
3						-20	-18	-10	-15	-19	-16	-16	-13						-10	-10			
4			-10		-17	-15	-16	-15	-13	-20	-23	-22	-19	-21	-18	-14	-10	-10	-10	-10	-10		
5																							
6				-15	-13	-14	-15	-12	-16						-10	-10	-10	-10	-10	-10			
7						-19	-18	-10	-15	-16	-15	-15	-13						-10	-10			

Figure 19. An example azimuth angle vs. takeoff angle plot from program VALSUM.

5. CONCLUSIONS

The Validation Summary (VALSUM) program provides a means for ascertaining and validating antenna performance parameters for HF broadcast coverage. The technique is a statistical reduction of point-to-point circuit analyses produced by the NTIA/ITS Ionospheric Communication Analysis and Prediction (IONCAP) program. The VALSUM user is provided with two alternative ways of devising a "best" frequency broadcast schedule so that design parameters for the transmit antenna can be developed from a collection of receive points over a given time block, month, and sunspot number sequence. A variety of output table options are available from VALSUM for determining the range and distribution of takeoff angles, azimuth angles, required power gain, and the frequencies needed to provide a desired level of coverage over a specified area.

6. REFERENCES

CCIR (1984), Antenna Diagrams, ITU, Geneva, Switzerland.

Kuester, Nancy A. (1987), Gain Evaluation for an Idealized Curtain Array Antenna, NTIA Report 87-215, May.

Teters, L.R., J.L. Lloyd, G.W. Haydon, and D.L. Lucas (1983), Estimating the Performance of Telecommunication Systems Using the Ionospheric Transmission Channel; Ionospheric Communications Analysis and Prediction Program User's Manual, NTIA Report 83-127, July, (NTIS Order No. PB84-111210; manual and computer program on magnetic tape, NTIS Order No. PB85-225084).

VOA (1985), Voice of America Engineering Standard 16775.01; High Frequency (Shortwave) Broadcast System Design; Chapter 1: Requirements Definition, Voice of America, Office of Engineering and Technical Operations, Washington, D.C., July 31.

VOA (1985), Voice of America Engineering Standard 16775.02; Estimating the Performance of Telecommunication Systems Using the Ionospheric Transmission Channel; Ionospheric Communications Analysis and Prediction Program User's Manual, NTIA Report 83-127, Voice of America, Office of Engineering and Technical Operations, Washington, D.C.

7. ACKNOWLEDGMENTS

The development of the IONCAP Summary Program was sponsored by the USIA/Voice of America. Mr. George Lane and Mr. David Loudin of the Voice of America and Mr. Mike Toia, formerly of the Voice of America, were instrumental in developing the requirements, the algorithms, and the input/output structures for the HF broadcast system design and validation programs.

APPENDIX A. EXAMPLE DIALOG BETWEEN PROGRAM SKYVOA AND A USER

```
                    Institute for Telecommunication Sciences

          Ionospheric Communications Analysis and Prediction Program
                             Voice of America
                               Version 1.0

                          Thu 19 Feb 1987  14:01:00


Choose from the menu:
    H = Help
    D = Program Description
    C = Concise Dialog
    V = Verbose Dialog
    E = Edit Data
    S = Summary of Data
    P = Process Last Data Set Entered
    Q = Quit


Menu (Verbose)?

TRANSMITTER CHARACTERISTICS

Transmitter site name
    TNGR = Tangier
    UDRN = Udorn
    PTRC = Puerto Rico
    CRSN = Sri Lanka
    BOTS = Botswana
1) Transmitter site name (TNGR)? MIDB

Type site lat (followed by carriage return) and site lon (return)
for each of the sites.  Enter the reference site location first.
    Limits are-  0N <= lat <=   90N
                 0S <= lat <=   90S
                 0W <= lon <=  180W
                 0E <= lon <=  180E
The default hemispheres are N and E.  The S and W locations can be specified
by adding an S to the latitude value or adding an W to the longitude value.
Inputs of the form X,Y,Z imply degrees, minutes and seconds.  Inputs of the
form X.Y imply decimal degrees.
    1) Transmitter site lat ( 43.0000 deg N or  43  0  0 dms N)? 30.83
    1) Transmitter site lon (  26.0000 deg E or   26  0  0 dms E)? 35.27

Power delivered to transmitting antenna (between    .10 and 1000.0 kW)
    2) Power delivered to transmitting antenna (  1.00 kW)? 354

Main beam direction toward receiver
```

```
    Y=Yes
    N=No
    3) Main beam direction toward receiver (Yes)? N

Bearing from reference site.
Enter in degrees clockwise from north,
i.e. north = 0, east = 90, south = 180, west = 270.
Answer can be in decimal degrees(X.Y) or in deg. min. and sec. (X,Y,Z), and
must be between   0.0 and 360.0 degrees
    4) Bearing (   0.0 deg)? 334.9

Antenna name (up to 8 characters)
    5) Antenna name (BOTS111 )? VALTEST

SYSTEM CHARACTERISTICS

Antenna configurations:
    L = List current set of antenna configurations
    D = Delete an antenna configuration
    A = Add an antenna configuration
    C = Change an antenna configuration
    N = No change
    10) Antenna configuration (Add)?

Up to 12 frequencies (between 6 and 26 MHz)
(A carriage return exits this mode)
    Frequency (  6.1)? 6.1
Number of horizontal bays ( 1 to 14)
    Number of bays ( 4)?

Curtain excitation
Mode
     1   +00
     2   0+0
     3   00+
     4   ++0
     5   +0+
     6   0++
     7   +-0
     8   +0-
     9   0+-
    10   +++
    11   ++-
    12   +-+
    13   +--
         Curtain excitation ( 1)? 10

Horizontal slew angle from the normal to the array
The 15 slew angles are:
    -30         30
    -25         25
    -21         21
    -17         17
    -13         13
```

```
    -9          9
    -5          5
     0
    Horizontal slew angle (  0)?

Frequency at which the antenna is designed between  8.00 and 20.00 MHz
    Antenna design frequency ( 8.75 MHz)?

Height above ground of lowest stack ( 1.00 to 100.0 m)
One-half wavelength stack height above ground at  8.75 MHz is 17.13 meters
    Lowest stack height above ground (17.13 m)?
    Frequency (  6.1)? 9.7
    Number of bays ( 4)?
    Curtain excitation (10)? 4
    Horizontal slew angle (  0)?
    Antenna design frequency ( 8.75 MHz)?
    Lowest stack height above ground (17.13 m)?
    Frequency (  9.7)? 13.7
    Number of bays ( 4)?
    Curtain excitation ( 4)? 10
    Horizontal slew angle (  0)?
    Antenna design frequency (19.00 MHz)?
    Lowest stack height above ground ( 7.89 m)?
    Frequency ( 13.7)? 17.72
    Number of bays ( 4)?
    Curtain excitation (10)? 4
    Horizontal slew angle (  0)?
    Antenna design frequency (19.00 MHz)?
    Lowest stack height above ground ( 7.89 m)?
    Frequency ( 17.7)? 25.88
    Number of bays ( 4)?
    Curtain excitation ( 4)? 1
    Horizontal slew angle (  0)?
    Antenna design frequency (19.00 MHz)?
    Lowest stack height above ground ( 7.89 m)?
    Frequency ( 25.9)?
10) Antenna configuration (Add)? L
    Antenna configuration(s)
```

	Oper Freq (MHz)	Design Freq (MHz)	Lowest Elem Ht (m)	Bays	Excite Mode	Horiz Slew (deg)
1)	6.1	8.75	17.13	4	10	0
2)	9.7	8.75	17.13	4	4	0
3)	13.7	19.00	7.89	4	10	0
4)	17.7	19.00	7.89	4	4	0
5)	25.9	19.00	7.89	4	1	0

```
10) Antenna configuration (Add)?  N

AM time of day to begin analysis and predictions
(between 0100 and 2400 hours UT)
11) AM start time (0100)? 0100

AM time of day to end analysis and predictions
```

```
(between 0100 and 2400 hours UT)
12) AM end time (1200)? 0500

PM time of day to begin analysis and predictions
(between 0100 and 2400 hours UT)
13) PM start time (1300)? 1400

PM time of day to end analysis and predictions
(between 0100 and 2400 hours UT)
14) PM end time (2400)? 2100

DATA BASE PARAMETERS

Broadcast language
The first four characters of the language are entered; for example,
    POLI = Polish
    ENGL = English
21) Language (POLI)? UKR1

Language area under which to catagorize data base (4 characters)
    ex. ENG2 = English to Africa
22) Language area (TEST)? MIDB

Sector beginning bearing (  0.0 to 360.0 deg)
23) Sector beginning bearing (334.9 deg)? 334.9

Sector ending bearing (  0.0 to 360.0 deg)
24) Sector ending bearing (359.9 deg)? 7.8

Sector beginning range (  0.0 to 10000 km)
25) Sector beginning range (1525. km)? 1524.7

Sector ending range (1526. to 12000 km)
26) Sector ending range (2427. km)? 2427.0
TEST POINT ID 10124 is not in the new sector limits

Modify TEST POINT ID list
    A = Add
    C = Change
    D = Delete
    L = List
    N = No change
27) TEST POINT ID list (A)? N

Do you want a summary of the input data  (Y or N)? Y

TRANSMITTER CHARACTERISTICS

 1) Transmitter site name:                              MIDB
                                                         30.8 N    35.3 E
 2) Power delivered to transmitting antenna:            354.0 kW
 3) Main beam direction toward receiver antenna:         No
 4) Bearing:                                            334.9
 5) Antenna name:                                       VALTEST
```

SYSTEM CHARACTERISTICS

10) Antenna configurations:

	Oper Freq (MHz)	Design Freq (MHz)	Lowest Elem Ht (m)	Bays	Excite Mode	Horiz Slew (deg)
1)	6.10	8.75	17.13	4	10	0
2)	9.70	8.75	17.13	4	4	0
3)	13.70	19.00	7.89	4	10	0
4)	17.72	19.00	7.89	4	4	0
5)	25.88	19.00	7.89	4	1	0

11) AM start time: 100
12) AM end time: 500
13) PM start time: 1400
14) PM end time: 2100

DATA BASE PARAMETERS

21) Broadcast language: UKR1
22) Language area: MIDB
23) Sector beginning azimuth: 334.9 deg
24) Sector ending azimuth: 7.8 deg
25) Sector beginning range: 1525. km
26) Sector ending range: 2427. km
27) TEST POINT IDS:

Site	Latitude (deg)		Longitude (deg)		Great circle Distance (km)	Bearing from Ref site (deg E of N)
10122	44.5000 N	44,30, 0 N	33.8000 E	33,48, 0 E	1524.7	355.6
10110	45.2000 N	45,12, 0 N	29.5000 E	29,30, 0 E	1674.0	344.2
10136	46.2800 N	46,16,48 N	30.4400 E	30,26,24 E	1766.8	347.7
10138	46.5000 N	46,30, 0 N	35.2200 E	35,13,12 E	1741.6	359.9
10137	47.0000 N	47, 0, 0 N	33.0000 E	33, 0, 0 E	1807.7	354.5
10109	48.0000 N	48, 0, 0 N	26.5000 E	26,30, 0 E	2048.3	341.2
10131	48.2500 N	48,15, 0 N	29.0000 E	29, 0, 0 E	2007.6	346.4
10014	48.8000 N	48,48, 0 N	22.5000 E	22,30, 0 E	2268.3	335.3
10134	49.0000 N	49, 0, 0 N	35.0000 E	35, 0, 0 E	2019.6	359.4
10133	49.0000 N	49, 0, 0 N	33.0000 E	33, 0, 0 E	2028.5	355.2
10132	49.0000 N	49, 0, 0 N	31.0000 E	31, 0, 0 E	2051.2	351.1
10346	49.0000 N	49, 0, 0 N	25.0000 E	25, 0, 0 E	2196.3	339.8
10379	50.0000 N	50, 0, 0 N	31.0000 E	31, 0, 0 E	2160.1	351.7
10380	50.0000 N	50, 0, 0 N	29.0000 E	29, 0, 0 E	2193.7	348.0
10381	50.0000 N	50, 0, 0 N	27.0000 E	27, 0, 0 E	2239.2	344.4
10098	50.8000 N	50,48, 0 N	24.1000 E	24, 6, 0 E	2403.9	340.6
10127	51.0000 N	51, 0, 0 N	35.0000 E	35, 0, 0 E	2241.9	359.5
10378	51.0000 N	51, 0, 0 N	33.0000 E	33, 0, 0 E	2249.6	355.9
10125	51.6000 N	51,36, 0 N	29.0000 E	29, 0, 0 E	2365.1	349.2
10126	51.9000 N	51,54, 0 N	31.0000 E	31, 0, 0 E	2367.7	352.7
10121	52.5000 N	52,30, 0 N	34.0000 E	34, 0, 0 E	2410.7	357.9
10139	46.9000 N	46,54, 0 N	37.0000 E	37, 0, 0 E	1792.2	4.3
10384	47.7500 N	47,45, 0 N	35.5000 E	35,30, 0 E	1880.6	.5
10347	49.0000 N	49, 0, 0 N	39.0000 E	39, 0, 0 E	2043.7	7.8
10135	49.0000 N	49, 0, 0 N	37.0000 E	37, 0, 0 E	2024.7	3.6

```
10128  50.5000 N  50,30, 0 N   37.0000 E  37, 0, 0 E    2190.9           3.3

Do you want to process this data  (Y or N)? Y


Choose from the menu:
    H = Help
    D = Program Description
    C = Concise Dialog
    V = Verbose Dialog
    E = Edit Data
    S = Summary of Data
    P = Process Last Data Set Entered
    Q = Quit


Menu (Edit)?

Question number? 10

Antenna configurations:
    L = List current set of antenna configurations
    D = Delete an antenna configuration
    A = Add an antenna configuration
    C = Change an antenna configuration
    N = No change
10) Antenna configuration (Add)? D
    Antenna configuration? 5
10) Antenna configuration (Add)? D
    Antenna configuration? 4
10) Antenna configuration (Add)? D
    Antenna configuration? 3
10) Antenna configuration (Add)? D
    Antenna configuration? 2
10) Antenna configuration (Add)? C
    Antenna configuration? 1

Up to 12 frequencies (between 6 and 26 MHz)
(A carriage return exits this mode)
    Frequency (  6.1)? 7.2
Number of horizontal bays ( 1 to 14)
    Number of bays ( 4)?

Curtain excitation
Mode
 1   +00
 2   0+0
 3   00+
 4   ++0
 5   +0+
 6   0++
 7   +-0
 8   +0-
 9   0+-
```

```
10   +++
11   ++-
12   +-+
13   +--
    Curtain excitation (10)? 4

Horizontal slew angle from the normal to the array
The 15 slew angles are:
    -30          30
    -25          25
    -21          21
    -17          17
    -13          13
     -9           9
     -5           5
      0
    Horizontal slew angle (  0)?

Frequency at which the antenna is designed between  8.00 and 20.00 MHz
    Antenna design frequency ( 8.75 MHz)?

Height above ground of lowest stack ( 1.00 to 100.0 m)
One-half wavelength stack height above ground at  8.75 MHz is 17.13 meters
    Lowest stack height above ground (17.13 m)?
10) Antenna configuration (Add)?
    Frequency (  7.2)? 11.85
    Number of bays ( 4)?
    Curtain excitation ( 4)? 1
    Horizontal slew angle (  0)?
    Antenna design frequency ( 8.75 MHz)?
    Lowest stack height above ground (17.13 m)?
    Frequency ( 11.9)? 15.35
    Number of bays ( 4)?
    Curtain excitation ( 1)? 4
    Horizontal slew angle (  0)?
    Antenna design frequency (19.00 MHz)?
    Lowest stack height above ground ( 7.89 m)?
    Frequency ( 15.4)? 21.65
    Number of bays ( 4)?
    Curtain excitation ( 4)?
    Horizontal slew angle (  0)?
    Antenna design frequency (19.00 MHz)?
    Lowest stack height above ground ( 7.89 m)?
    Frequency ( 21.7)?
10) Antenna configuration (Add)? L
    Antenna configuration(s)
```

	Oper Freq (MHz)	Design Freq (MHz)	Lowest Elem Ht (m)	Bays	Excite Mode	Horiz Slew (deg)
1)	7.2	8.75	17.13	4	4	0
2)	11.9	8.75	17.13	4	1	0
3)	15.4	19.00	7.89	4	4	0
4)	21.7	19.00	7.89	4	4	0

10) Antenna configuration (Add)? N

Question number?

Menu (Concise)? S

TRANSMITTER CHARACTERISTICS

1) Transmitter site name:	MIDB 30.8 N 35.3 E
2) Power delivered to transmitting antenna:	354.0 kW
3) Main beam direction toward receiver antenna:	N
4) Bearing:	334.9
5) Antenna name:	VALTEST

SYSTEM CHARACTERISTICS

10) Antenna configurations:

	Oper Freq (MHz)	Design Freq (MHz)	Lowest Elem Ht (m)	Bays	Excite Mode	Horiz Slew (deg)
1)	7.20	8.75	17.13	4	4	0
2)	11.85	8.75	17.13	4	1	0
3)	15.35	19.00	7.89	4	4	0
4)	21.65	19.00	7.89	4	4	0

11) AM start time:	100
12) AM end time:	500
13) PM start time:	1400
14) PM end time:	2100

DATA BASE PARAMETERS

21) Broadcast language:	UKR1
22) Language area:	MIDB
23) Sector beginning azimuth:	334.9 deg
24) Sector ending azimuth:	7.8 deg
25) Sector beginning range:	1525. km
26) Sector ending range:	2427. km

27) TEST POINT IDS:

Site	Latitude (deg)		Longitude (deg)		Great circle Distance (km)	Bearing from Ref site (deg E of N)
10384	47.7500 N	47,45, 0 N	35.5000 E	35,30, 0 E	1880.6	.5
10128	50.5000 N	50,30, 0 N	37.0000 E	37, 0, 0 E	2190.9	3.3
10135	49.0000 N	49, 0, 0 N	37.0000 E	37, 0, 0 E	2024.7	3.6
10139	46.9000 N	46,54, 0 N	37.0000 E	37, 0, 0 E	1792.2	4.3
10347	49.0000 N	49, 0, 0 N	39.0000 E	39, 0, 0 E	2043.7	7.8
10014	48.8000 N	48,48, 0 N	22.5000 E	22,30, 0 E	2268.3	335.3
10346	49.0000 N	49, 0, 0 N	25.0000 E	25, 0, 0 E	2196.3	339.8
10098	50.8000 N	50,48, 0 N	24.1000 E	24, 6, 0 E	2403.9	340.6
10109	48.0000 N	48, 0, 0 N	26.5000 E	26,30, 0 E	2048.3	341.2
10110	45.2000 N	45,12, 0 N	29.5000 E	29,30, 0 E	1674.0	344.2
10381	50.0000 N	50, 0, 0 N	27.0000 E	27, 0, 0 E	2239.2	344.4
10131	48.2500 N	48,15, 0 N	29.0000 E	29, 0, 0 E	2007.6	346.4
10136	46.2800 N	46,16,48 N	30.4400 E	30,26,24 E	1766.8	347.7
10380	50.0000 N	50, 0, 0 N	29.0000 E	29, 0, 0 E	2193.7	348.0
10125	51.6000 N	51,36, 0 N	29.0000 E	29, 0, 0 E	2365.1	349.2
10132	49.0000 N	49, 0, 0 N	31.0000 E	31, 0, 0 E	2051.2	351.1
10379	50.0000 N	50, 0, 0 N	31.0000 E	31, 0, 0 E	2160.1	351.7
10126	51.9000 N	51,54, 0 N	31.0000 E	31, 0, 0 E	2367.7	352.7
10137	47.0000 N	47, 0, 0 N	33.0000 E	33, 0, 0 E	1807.7	354.5
10133	49.0000 N	49, 0, 0 N	33.0000 E	33, 0, 0 E	2028.5	355.2
10122	44.5000 N	44,30, 0 N	33.8000 E	33,48, 0 E	1524.7	355.6
10378	51.0000 N	51, 0, 0 N	33.0000 E	33, 0, 0 E	2249.6	355.9
10121	52.5000 N	52,30, 0 N	34.0000 E	34, 0, 0 E	2410.7	357.9
10134	49.0000 N	49, 0, 0 N	35.0000 E	35, 0, 0 E	2019.6	359.4
10127	51.0000 N	51, 0, 0 N	35.0000 E	35, 0, 0 E	2241.9	359.5
10138	46.5000 N	46,30, 0 N	35.2200 E	35,13,12 E	1741.6	359.9

Do you want to process this data (Y or N)? Y

Menu (Edit)? Q

APPENDIX B. SOURCE CODE FOR AN EXAMPLE PROGRAM TO CREATE AN IONCAP ANTENNA FILE

```
      PROGRAM PATTRN
C
C
C*********************************************************************
C
C                       MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  PATTRN                  FILE NAME:
C  LOAD MODULE:  PATTRN                  HID:
C  CPU TYPE:  CYBER 840                  LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*********************************************************************
C
C                              PURPOSE
C
C  This program creates antenna patterns to be used in IONCAP.
C  The input to PATTERN is TAPE55 and the binary output is stored on
C  TAPE26 which is the name of the antenna file for IONCAP.
C
C*********************************************************************
C
C                       TECHNICAL INFORMATION
C
C  See purpose.
C
C*********************************************************************
C
C                          SUBPROGRAM USAGE
C
C  DAZEL - computes distance and bearing between xmtr and receiver
C  GAIN - computes power gain dBi
C  MAKDAT - write binary antenna pattern to TAPE26
C
C*********************************************************************
C
C                           CHANGE HISTORY
C
C*********************************************************************
C
C                          MNEMONIC LISTING
C
C  PARAMETERS:
C    None
C
C
C  COMMON BLOCKS:
C ANTDAT
C  NOSTAK - NUMBER OF VERTICAL STACKS
```

```
C  STKSPM - SPACING BETWEEN STACKS
C  NUMBAY - NUMBER OF BAYS
C  BAYSPM - SPACING BETWEEN BAYS
C  DIPLNM - LENGTH OF DIPOLE RADIATOR
C  RRSPM - SPACING BETWEEN RADIATORS AND REFLECTING SCREEN
C  STKHTM - HEIGHT OF LOWEST STACK ABOVE GROUND
C  STKRAT - CURRENT RATIO FOR STACKS
C  BAYPHS - RELATIVE PHASE FOR THE BAYS
C  BAYRAT - CURRENT RATIO FOR THE BAYS
C  OFMHZ - ANTENNA OPERATING FREQUENCY
C  DFMHZ - ANTENNA DESIGN FREQUENCY
C  VSLEW - VERTICAL SLEW ANGLE
C  HSLEW - HORIZONTAL SLEW ANGLE
C AZEL
C   ZTLAT - latitude (decimal degrees north of equator) of point T
C   ZTLON - longitude (decimal degrees east of prime (Greenwich)
C          meridian) of point T
C   ZTHT  - height (meters above mean sea level) of point T
C   ZRLAT - latitude (decimal degrees north of equator) of point R
C   ZRLON - longitude (decimal degrees east of prime meridian of point R
C   ZRHT  - height (meters above mean sea level) of point R
C   ZTAZ  - azumuth (decimal degrees clockwise from north) at T of R
C   ZRAZ  - azimuth (decimal degrees clockwise from north) at R of T
C   ZTELV - elevation angle (decimal degrees above horizontal at T
C          of straight line between T and R
C   ZRELV - elevation angle (decimal degrees above horizontal at R)
C          of straight line between T and R
C   ZTAKOF - take-off angle (decimal degrees above horizontal at T)
C          of refracted ray between T and R (assumed 4/3 earth radius)
C   ZRAKOF - take-off angle (decimal degrees above horizontal at R)
C          of refracted ray between T and R (assumed 4/3 earth radius)
C   ZD    - straight line distance (kilometers) between T and R
C   ZDGC  - great circle distance (kilometers) between T and R
C
C HELP
C   GAINAR - antenna pattern
C
C
C**********************************************************************
C**********************************************************************
C  EXECUTABLE CODE
C**********************************************************************
C
C
    DIMENSION SPACE(2), XLEN(2), RLAT(50),
     +  RLONG(50), GAINS(91)
    INTEGER IBAND(12,6), IPHASE(14,8), ISRATIO(6,13)
    CHARACTER SKYDAT*6
    COMMON/ANTDAT/NOSTAK,STKSPM,NUMBAY,BAYSPM,DIPLNM,RRSPM,
     +  STKHTM,STKRAT,BAYPHS,BAYRAT,OFMHZ,DFMHZ,VSLEW,HSLEW
    COMMON/AZEL/ ZTLAT,ZTLON,ZTHT,ZRLAT,ZRLON,ZRHT,ZTAZ,ZRAZ,
     * ZTELV,ZRELV,ZD,ZDGC,ZTAKOF,ZRAKOF
    COMMON/HELP/GAINAR(30,91)
C
```

```
      REAL STKRAT(14), BAYPHS(14), BAYRAT(14)
      DATA SPACE/17.13,7.89/,XLEN/8.57,3.94/,NOSTAK/6/
      DATA ISRATIO/1,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1,1,1,1,1,1,0,0,
      +  1,1,0,0,1,1,0,0,1,1,1,1,1,1,-1,-1,0,0,1,1,0,0,-1,-1,
      +  0,0,1,1,-1,-1,1,1,1,1,1,1,1,1,1,1,-1,-1,1,1,-1,-1,1,1,
      +  1,1,-1,-1,-1,-1/
      DATA IPHASE/0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,31,31,0,77,77,
      +  109,109,0,155,155,186,186,0,47,56,103,0,139,185,195,242,
      +  0,278,324,334,381,0,47,81,128,0,200,246,281,327,0,399,446,
      +  480,527,0,47,105,152,0,260,306,365,411,0,519,566,624,671,
      +  0,47,129,176,0,318,365,447,494,0,636,683,765,812,0,90,152,
      +  242,0,375,465,527,617,0,750,840,903,993,0,90,180,270,0,444,
      +  534,624,714,0,888,978,1068,1158/
C
C
C
      DATA BAYRAT/1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1.,1./
      DATA ZRHT/0./,ZTHT/0.0/
      DATA SKYDAT/'TAPE55'/
C
C
      OPEN (1,FILE='OUTPUT')
      JFIRST = 0
      JFLAG = 0
C   OPEN INPUT FILE AND READ DATA
C
      OPEN (55,FILE=SKYDAT,IOSTAT=IOS,ERR=2000)
C
5       READ (55,'(F7.2,X,F8.2,X,I3,X,I3,X,I1,X,F5.2)',END=3000,ERR=3001,
     +    IOSTAT=IOS) ZTLAT,ZTLON,NBAND,NPOINT,IVALID,BEAR
        DO 10 I=1,NBAND
        READ (55,'(6I5)')(IBAND(I,J),J=1,6)
   10 CONTINUE
        DO 20 I=1,NPOINT
          READ (55,'(F7.2,X,F8.2)') RLAT(I), RLONG(I)
   20 CONTINUE
C
C
C   CALCULATE ANTENNA PATTERNS FOR EACH RECEIVER SITE
         IF (IVALID .EQ. 0) THEN
           LOOP = 1
         ELSE
           LOOP = NPOINT
         ENDIF
C
        DO 1000 KOUNT = 1, LOOP
C
C   DETERMINE AZIMUTH
       ZRLAT = RLAT(KOUNT)
       ZRLON = RLONG(KOUNT)
       CALL DAZEL(0)
C   CALCULATE GAINS FOR EACH FREQUENCY
           IBEG = 1
           DO 40 I=1,NBAND
```

```
C  SET DEFAULT VALUES FOR GAIN
         IFLAG = 0
         OFMHZ = IBAND(I,1) / 100.0
         DFMHZ = IBAND(I,5) / 100.0
         STKHTM = IBAND(I,6) / 100.0
         BAYSPM = 0.5*(299.8/DFMHZ)
         DIPLNM = 0.5*(299.8/DFMHZ)
         RRSPM = 0.25*(299.8/DFMHZ)
         KSLEW = (ABS(IBAND(I,4))/4)+1
         NUMBAY = IBAND(I,2)
         STKSPM = 0.5*(299.8/DFMHZ)
         HSLEW = IBAND(I,4)
C
C  INITIALIZE BAY PHASE
      DO 500 K=1,14
        IF (IBAND(I,4) .GT. 0) THEN
          BAYPHS(K) = (IPHASE(K,KSLEW))
          BAYPHS(K) = -BAYPHS(K)
        ELSE
          BAYPHS(K) = IPHASE(K,KSLEW)
        ENDIF
500   CONTINUE
C
C  INITIALIZE STACK RATIO
       DO 510 J = 1,6
         STKRAT(J) = ISRATIO(J,IBAND(I,3))
510    CONTINUE
C
C  CALCULATE GAINS FOR EACH FREQUENCY BAND
        DO 70 L = 0,90
          THETAD = L
         IF (IVALID .EQ. 0) THEN
           AZIM = 0
         ELSE
           AZIM = ZTAZ - BEAR
        ENDIF
          CALL GAIN (XGAIN,XFACTR,IFLAG,THETAD,AZIM)
          IFLAG = 1
          GAINS (L+1) = XGAIN
   70     CONTINUE
          DO 80 M = IBEG,(IBAND(I,1)/100)+1
            DO 90 N = 1,91
              GAINAR(M,N) = GAINS(N)
   90       CONTINUE
   80     CONTINUE
         IF ((IBAND(I,1)/100) .EQ. 6) THEN
           IF ((IBAND(2,1)/100) .EQ. 7) THEN
             IBEG = 7
           ELSE
             IBEG = 8
           ENDIF
         ELSE
           IBEG = (IBAND(I,1)/100) + 2
         ENDIF
```

```
   40   CONTINUE
      IF (IBEG .LT. 30) THEN
        DO 100 M = IBEG,30
          DO 110 N = 1,91
            GAINAR(M,N) = GAINS(N)
  110     CONTINUE
  100   CONTINUE
      ENDIF
      IF (JFIRST .EQ. 0) THEN
        OPEN (88,FILE='TAPE26',FORM='UNFORMATTED',ERR=2000,IOSTAT=IOS,
     +  ACCESS='SEQUENTIAL')
        REWIND(88)
      ENDIF
        CALL MAKDAT (JFIRST)
        JFIRST = 1
 1000 CONTINUE
      GOTO 5
C
3001  WRITE (1,'("ERROR #",I5," READING FROM SKYDAT")') IOS
3000  CLOSE(55)
      CLOSE(88)
C
C
      STOP
2000  WRITE (1,'("ERROR #",I5," OPENING TAPE55")') IOS
      STOP
      END
       SUBROUTINE DAZEL(MODE)
C*********************************************************************
C
C                      MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  PATTRN                  FILE NAME:
C  LOAD MODULE:  DAZEL                   HID:
C  CPU TYPE:  CYBER 840                  LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*********************************************************************
C
C                             PURPOSE
C
C     Two modes--   0   input lat and lon of end point
C                       return distance and azimuth to end pt with elevations
C                   1   input bearing (azimuth) of end point
C                       return lat and lon of end point with elevations
C
C*********************************************************************
C
C                      TECHNICAL INFORMATION
C
C  See purpose.
C
C*********************************************************************
```

```
C
C                          SUBPROGRAM USAGE
C
C  None.
C
C*********************************************************************
C
C                          CHANGE HISTORY
C
C*********************************************************************
C
C                          MNEMONIC LISTING
C
C  PARAMETERS:
C     Two modes--   0   input lat and lon of end point
C                       return distance and azimuth to end pt with elevations
C                   1   input bearing (azimuth) of end point
C                       return lat and lon of end point with elevations
C
C  COMMON BLOCKS:
C    AZEL
C
C   ZTLAT - latitude (decimal degrees north of equator) of point T
C   ZTLON - longitude (decimal degrees east of prime (Greenwich)
C          meridian) of point T
C   ZTHT  - height (meters above mean sea level) of point T
C   ZRLAT - latitude (decimal degrees north of equator) of point R
C   ZRLON - longitude (decimal degrees east of prime meridian of point R
C   ZRHT  - height (meters above mean sea level) of point R
C   ZTAZ  - azumuth (decimal degrees clockwise from north) at T of R
C   ZRAZ  - azimuth (decimal degrees clockwise from north) at R of T
C   ZTELV - elevation angle (decimal degrees above horizontal at T
C          of straight line between T and R
C   ZRELV - elevation angle (decimal degrees above horizontal at R)
C          of straight line between T and R
C   ZTAKOF - take-off angle (decimal degrees above horizontal at T)
C          of refracted ray between T and R (assumed 4/3 earth radius)
C   ZRAKOF - take-off angle (decimal degrees above horizontal at R)
C          of refracted ray between T and R (assumed 4/3 earth radius)
C   ZD    - straight line distance (kilometers) between T and R
C   ZDGC  - great circle distance (kilometers) between T and R
      IMPLICIT DOUBLE PRECISION(A-L,N-Y)
C
C
C   WRITTEN BY KEN SPIES 5/79
C   REFRACTION AND ST. LINE ELEVATIONS BY EJH
C
      COMMON/AZEL/ ZTLAT,ZTLON,ZTHT,ZRLAT,ZRLON,ZRHT,ZTAZ,ZRAZ,
     * ZTELV,ZRELV,ZD,ZDGC,ZTAKOF,ZRAKOF
      DATA PI/3.14159265358979323846264300/,RERTH/6368.0D0/
      DATA DTOR/0.01745329252D0/,RTOD/57.29577951D0/
      IF(MODE .EQ. 1) GO TO 200
       TLATS=ZTLAT
       TLONS=ZTLON
```

```
      THTS=ZTHT*1.0E-3
      RLATS=ZRLAT
      RLONS=ZRLON
      RHTS=ZRHT*1.0E-3
      IF(TLATS-(-90.0))2,2,4
2     STOP
4     IF(TLATS-90.0)6,2,2
6     IF(RLATS-(-90.0))2,2,8
8     IF(RLATS-90.0)10,2,2
10    DELAT=RLATS-TLATS
      ADLAT=DABS(DELAT)
      DELON=RLONS-TLONS
      IF(DELON-(-180.0))12,16,16
12    DELON=DELON+360.0
      IF(DELON-(-180.0))12,20,20
16    IF(DELON-180.0)20,20,18
18    DELON=DELON-360.0
      IF(DELON-180.0)20,20,18
20    ADLON=DABS(DELON)
      DELHT=RHTS-THTS
      IF(ADLON-1.0E-5)22,22,55
22    IF(ADLAT-1.0E-5)24,24,40
C
C   POINTS T AND R HAVE THE SAME COORDINATES
C
24    ZTAZ=0.0
      ZRAZ=0.0
      IF(DELHT)25,30,35
25    ZTELV=-90.0
      ZRELV=90.0
      ZD=-DELHT
     ZDGC=0.0
      RETURN
30    ZTELV=0.0
      ZRELV=0.0
      ZD=0.0
     ZDGC=0.0
      RETURN
35    ZTELV=90.0
      ZRELV=-90.0
      ZD=DELHT
     ZDGC=0.0
      RETURN
C
C   POINTS T AND R HAVE SAME LONGITUDE, DISTINCT LATITUDES
C
40    IF(DELAT-0.0)42,42,45
42    ZTAZ=180.0
      ZRAZ=0.0
      GO TO 50
45    ZTAZ=0.0
      ZRAZ=180.0
50    GC=ADLAT*DTOR
      SGC=DSIN(0.5*GC)
```

```
       D=DSQRT(DELHT*DELHT+4.0*(RERTH+THTS)*(RERTH+RHTS)*SGC*SGC)
      ZD=D
       GO TO 140
C
C   POINTS T AND R HAVE DISTINCT LONGITUDES
C
55     IF(DELON-0.0)56,56,60
56     WLAT=RLATS*DTOR
       ELAT=TLATS*DTOR
       GO TO 65
60     WLAT=TLATS*DTOR
       ELAT=RLATS*DTOR
C
C   CALCULATE AZIMUTHS AT POINTS W AND E
C
65     SDLAT=DSIN(0.5*ADLAT*DTOR)
       SDLON=DSIN(0.5*ADLON*DTOR)
       SADLN=DSIN(ADLON*DTOR)
       CWLAT=DCOS(WLAT)
       CELAT=DCOS(ELAT)
       P=2.0*(SDLAT*SDLAT+SDLON*SDLON*CWLAT*CELAT)
       SGC=DSQRT(P*(2.0-P))
       SDLAT=DSIN(ELAT-WLAT)
       CWAZ=(2.0*CELAT*DSIN(WLAT)*SDLON*SDLON+SDLAT)/SGC
      SWAZ=SADLN*CELAT/SGC
      WAZ=DATAN2(SWAZ,CWAZ)*RTOD
       CEAZ=(2.0*CWLAT*DSIN(ELAT)*SDLON*SDLON-SDLAT)/SGC
      SEAZ=SADLN*CWLAT/SGC
      EAZ=DATAN2(SEAZ,CEAZ)*RTOD
110    EAZ=360.0-EAZ
       IF(DELON-0.0)111,111,115
111    ZTAZ=EAZ
       ZRAZ=WAZ
       GO TO 120
115    ZTAZ=WAZ
       ZRAZ=EAZ
C
C
C   COMPUTE THE STRAIGHT LINE DISTANCE AND GREAT CIRCLE ANGLE BETWEEN T AND R
C
120    D=DSQRT(DELHT*DELHT+2.0*(RERTH+THTS)*(RERTH+RHTS)*P)
      ZD=D
       CGC=1.0-P
      GC=DATAN2(SGC,CGC)
C
C   COMPUTE GREAT CIRCLE DISTANCE AND ELEVATION ANGLES
C
140   ZDGC=GC*RERTH
142    IF(DELHT .GE. 0) GOTO 145
       AHT=THTS
       BHT=RHTS
       GO TO 150
145    AHT=RHTS
       BHT=THTS
```

```
150     SAELV=0.5*(D*D+DABS(DELHT)*(RERTH+AHT+RERTH+BHT))/(D*(RERTH+AHT))
       ARG=DMAX1(0.0D0,(1.0D0-SAELV*SAELV))
        AELV=DATAN2(SAELV,DSQRT(ARG))
        BELV=(AELV-GC)*RTOD
        AELV=-AELV*RTOD
C    COMPUTE TAKE-OFF ANGLES ASSUMING 4/3 EARTH RADIUS
        R4THD=RERTH*4.0/3.0
        GC=0.75*GC
        SGC=DSIN(0.5*GC)
        P=2.0*SGC*SGC
        AALT=R4THD+AHT
        BALT=R4THD+BHT
        DA=DSQRT(DELHT*DELHT+2.0*AALT*BALT*P)
        SAELV=0.5*(DA*DA+DABS(DELHT)*(AALT+BALT))/(DA*AALT)
       ARG=DMAX1(0.0D0,(1.0D0-SAELV*SAELV))
        ATAKOF=DATAN2(SAELV,DSQRT(ARG))
        BTAKOF=(ATAKOF-GC)*RTOD
        ATAKOF=-ATAKOF*RTOD
        IF(DELHT-0.0)151,155,155
151     ZTELV=AELV
        ZRELV=BELV
        ZTAKOF=ATAKOF
        ZRAKOF=BTAKOF
        RETURN
155     ZTELV=BELV
        ZRELV=AELV
        ZTAKOF=BTAKOF
        ZRAKOF=ATAKOF
        RETURN
C
C          COMPUTE END POINT GIVEN DISTANCE AND BEARING
C
200        TLATR=ZTLAT*DTOR
           TLONR=ZTLON*DTOR
           TAZR=ZTAZ*DTOR
           GC=ZDGC/RERTH
           COLAT=PI/2.0 - TLATR
           COSCO=DCOS(COLAT)
           SINCO=DSIN(COLAT)
           COSGC=DCOS(GC)
           SINGC=DSIN(GC)
           COSB=COSCO*COSGC + SINCO*SINGC*DCOS(TAZR)
           ARG=DMAX1(0.0D0,(1.0D0-COSB*COSB))
           B=DATAN2(DSQRT(ARG),COSB)
           ARC=(COSGC-COSCO*COSB)/(SINCO*DSIN(B))
           ARG=DMAX1(0.0D0,(1.0D0-ARC*ARC))
           RDLON=DATAN2(DSQRT(ARG),ARC)
           ZRLAT=(PI/2.0 - DABS(B))*RTOD
           DRLAT=ZRLAT
           ZRLAT=DSIGN(DRLAT,COSB)
           ZRLON=ZTLON+(DABS(RDLON)*RTOD)
           IF(ZTAZ .GT. 180) ZRLON=ZTLON-(DABS(RDLON)*RTOD)
           THTS=ZTHT*1.0E-3
           RHTS=ZRHT*1.0E-3
```

```
      DELHT=RHTS-THTS
      SGC=DSIN(0.5*GC)
      D=DSQRT(DELHT*DELHT+4.0*(RERTH+THTS)*(RERTH+RHTS)*SGC*SGC)
      GO TO 142
      END
      SUBROUTINE MAKDAT(JFIRST)
C
C***********************************************************************
C
C                    MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  PATTRN                  FILE NAME:
C  LOAD MODULE:  MAKDAT                  HID:
C  CPU TYPE:  CYBER 840                  LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C***********************************************************************
C
C                           PURPOSE
C
C  This subroutine writes a binary antenna pattern to TAPE26.
C
C***********************************************************************
C
C                    TECHNICAL INFORMATION
C
C  See purpose.
C
C***********************************************************************
C
C                       SUBPROGRAM USAGE
C
C  None
C
C***********************************************************************
C
C                        CHANGE HISTORY
C
C***********************************************************************
C
C                       MNEMONIC LISTING
C
C  PARAMETERS:
C    JFIRST - 0 = first pattern
C             1 = pattern has already been written
C
C
C  COMMON BLOCKS:
C    HELP
C     GAINAR - array of gain values to be written to TAPE26
C
C
C***********************************************************************
C***********************************************************************
```

```
C  EXECUTABLE CODE
C*********************************************************************
C
C
C
C
      DIMENSION XFQS(3),XFQE(3),YNH(3),YNL(3),YND(3),
     +      YETA(3),TEY(3,4),COND(3),DIEL(3),ARRAY(30,91),AEFF(30),
     +  TOAZ(3),AREFF(30)
      COMMON/HELP/GAINAR(30,91)
      CHARACTER LABANT(2,3)*6
C
C
      DATA NA/1/,JTSANT/1/,XFQS/1.,1.,1./,XFQE/30.,30.,30./,
     +YNH/0.,0.,0./,YNL/0.,0.,0./,YND/0.,0.,0./,YETA/0.,0.,0./,
     +TEY/12*0./,COND/.01,.01,.01/,DIEL/10.,10.,10./,
     +TOAZ/0.,0.,0./,
     +AEFF/30*0./,AREFF/30*-4.8/
      DATA LABANT/6*'      '/
      DATA (ARRAY(1,I),I=1,91)/
     +     -20.0,-14.0,-11.0,-7.6,-5.4,-4.0,-3.2,-2.5,-1.8,-1.6,
     +     -1.3,-1.1,-.9,-.6,-.5,-.4,-.2,-.1,0,0,0,0,0,0,0,0,
     +     -.1,-.2,-.2,-.2,-.3,-.3,-.4,-.5,-.5,-.6,-.7,-.8,-.8,
     +     -.9,-1.,-1.1,-1.2,-1.4,-1.5,-1.6,-1.8,-1.9,-2.,-2.1,
     +     -2.3,-2.4,-2.6,-2.7,-2.9,-3.1,-3.2,-3.4,-3.6,-3.7,
     +     -3.9,-4.2,-4.4,-4.7,-5.0,-5.4,-5.7,-6.0,-6.4,-6.7,
     +     -7.1,-7.5,-7.9,-8.4,-8.8,-9.3,-9.8,-10.4,-10.9,-11.4,-12.,
     +     -12.6,-13.2,-13.9,-14.6,-15.4,-16.2,-17.2,-18.2,
     +     -19.6,-21.9/
C
C
C
      IF (JFIRST .EQ. 0) THEN
C
C  RECEIVER ANTENNA PATTERNS
C
      DO 20 I = 2,30
        DO 30 J = 1,91
          ARRAY(I,J) = ARRAY(1,J)
 30     CONTINUE
 20   CONTINUE
      LABANT(1,1) ='SWWHIP'
C
      WRITE (88) NA,JTSANT,(XFQS(ITY),XFQE(ITY),LABANT(1,ITY),
     + LABANT(2,ITY),YNH(ITY),YNL(ITY),YND(ITY),TOAZ(ITY),
     + (TEY(ITY,K),K=1,4),COND(ITY),DIEL(ITY),ITY=1,3),
     + ((ARRAY(I,J),I=1,30),J=1,91),(AREFF(I),I=1,30)
C
       ENDIF
C
C  TRANSMITTER ANTENNA PATTERNS
C
      LABANT(1,1) = 'CONS17'
C
```

```
      WRITE (88) NA,JTSANT,(XFQS(ITY),XFQE(ITY),LABANT(1,ITY),
     + LABANT(2,ITY),YNH(ITY),YNL(ITY),YND(ITY),TOAZ(ITY),
     + (TEY(ITY,K),K=1,4),COND(ITY),DIEL(ITY),ITY=1,3),
     + ((GAINAR(I,J),I=1,30),J=1,91),(AEFF(I),I=1,30)
      RETURN
      END
      SUBROUTINE GAIN(XGAIN,FACTOR,IFLAG,THETAD,PHID)
C
C*********************************************************************
C
C                     MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  PATTRN                   FILE NAME:
C  LOAD MODULE:  GAIN                     HID:
C  CPU TYPE:  CYBER840                    LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*********************************************************************
C
C                            PURPOSE
C
C  This subroutine calculates the gain of a curtain antenna
C  (documented in NTIA report)
C
C*********************************************************************
C
C                      TECHNICAL INFORMATION
C
C  See purpose.
C
C*********************************************************************
C
C                         SUBPROGRAM USAGE
C
C  DBLTRAP - calculates a double integral in theta and phi using a
C            simple trapizoidal rule approximation.
C
C*********************************************************************
C
C                          CHANGE HISTORY
C
C*********************************************************************
C
C                         MNEMONIC LISTING
C
C  PARAMETERS:
C       XGAIN   -  POWER GAIN IN dBi CALCULATED WITHIN THE SUBROUTINE
C       FACTOR  -  INTEGRATION FACTOR WHICH CAN BE CALCULATED IN GAIN OR
C                    PASSED BY USER
C       IFLAG   -  FLAG TO DETERMINE IF FACTOR IS TO BE PASSED INTO GAIN
C                    OR CALCULATED WITHIN GAIN
C                    IFLAG = 0, FACTOR IS TO BE CALCULATED
C                    IFLAG = 1, FACTOR IS PASSED IN
C       THETAD  -  ANGLE ABOVE THE HORIZON IN DEGREES
```

```
C       PHID    -  ANTENNA BORESIGHT AZIMUTHAL ANGLE IN DEGREES
C
C
C  COMMON BLOCKS:
C    ANTDAT
C  NOSTAK - NUMBER OF VERTICAL STACKS
C  STKSPM - SPACING BETWEEN STACKS
C  NUMBAY - NUMBER OF BAYS
C  BAYSPM - SPACING BETWEEN BAYS
C  DIPLNM - LENGTH OF DIPOLE RADIATOR
C  RRSPM - SPACING BETWEEN RADIATORS AND REFLECTING SCREEN
C  STKHTM - HEIGHT OF LOWEST STACK ABOVE GROUND
C  STKRAT - CURRENT RATIO FOR STACKS
C  BAYPHS - RELATIVE PHASE FOR THE BAYS
C  BAYRAT - CURRENT RATIO FOR THE BAYS
C  OFMHZ - ANTENNA OPERATING FREQUENCY
C  DFMHZ - ANTENNA DESIGN FREQUENCY
C  VSLEW - VERTICAL SLEW ANGLE
C  HSLEW - HORIZONTAL SLEW ANGLE
C  FWAVE
C       EIL     -  0.5 * LENGTH OF DIPOLE IN WAVELENGTHS
C       XR      -  REFLECTOR TO DIPOLE SPACING IN WAVELENGTHS
C       C(8)    -  ARRAY OF RELATIVE STACK CURRENT MAGNITUDE RATIOS
C       R(14)   -  ARRAY OF RELATIVE BAY CURRENT MAGNITUDE RATIOS
C       PS(14)  -  ARRAY OF BAY CURRENT PHASE RATIOS IN RADIANS INCLUDING
C                     FREQUENCY ADJUSTMENT FOR OPERATING FREQUENCY NOT
C                     EQUAL TO DESIGN FREQUENCY
C       A(8)    -  ARRAY OF STACK CURRENT PHASE RATIOS IN RADIANS,
C                    CURRENTLY DEFAULTED TO ZERO
C       Y(14)   -  ARRAY CONTAINING THE SPACINGS BETWEEN THE BAY DIPOLES
C                     IN WAVELENGTHS (RELATIVE TO BAY 1)
C       Z(8)    -  ARRAY CONTAINING THE SPACINGS BETWEEN THE STACK DIPOLES
C                     IN WAVELENGTHS (RELATIVE TO STACK 1)
C CONST
C       PI      -  MATHEMATICAL SYMBOL PI = 3.14159
C       VOFL    -  VELOCITY OF LIGHT  FOR FREQUENCY IN MEGAHERTZ = 3.0E-2
C       PI2     -  2 * PI
C       PI02    -  PI / 2
C       D2R     -  CONVERSION FACTOR FOR DEGREES TO RADIANS CONVERSION
C                   = PI / 180.
C       R2D     -  CONVERSION FACTOR FOR RADIANS TO DEGREES CONVERSION
C                   = 180 / PI
C
C
C*********************************************************************
C*********************************************************************
C  EXECUTABLE CODE
C*********************************************************************
C
C
C
C*******************************************************************
C SUBROUTINE TO CALULATE THE GAIN OF A CURRENT ANTENNA ..............
C*********************************************************************
```

```
C
         COMMON/CONST/ PI,VOFL,PI2,PIO2,D2R,R2D
         DIMENSION BAYRAT(14),STKRAT(14),BAYPHS(14)
         COMMON/ANTDAT/NOSTAK,STKSPM,NUMBAY,BAYSPM,DIPLNM,RRSPM,
         1  STKHTM,STKRAT,BAYPHS,BAYRAT,OFMHZ,DFMHZ,VSLEW,HSLEW
         COMMON/FWAVE/EIL,XR,C,R,PS,A,Y,Z,WAVE,BETA
         DIMENSION C(8),R(14),PS(14),A(8),Y(14),Z(8)
C CONSTANTS
    WAVE = VOFL /OFMHZ
    BETA = PI2 / WAVE
    EL = DIPLNM * BETA
    Y(1) = 0.0
    H = STKHTM * BETA
    XR = RRSPM * BETA
    DO 30 I = 1,NOSTAK
         Z(I) = STKSPM * BETA *FLOAT(I-1) + H
         A(I) = 0.0
         C(I) = STKRAT(I)
 30    CONTINUE
         DO 40 I = 1,NUMBAY
             PS(I) = D2R * BAYPHS(I) * OFMHZ / DFMHZ
             R(I) = BAYRAT(I)
             IF (I.EQ.1) GO TO 40
             Y(I) = BETA * FLOAT(I-1) * BAYSPM
  40  CONTINUE
    EIL = EL / 2.0
    IF (IFLAG .GT. 0) GO TO 200
         TSTEPD = 1.0
         PSTEPD = 1.0
         TZD = 0.0
         PZD = -90.0
         TFD = 90.0
         PFD = 90.0
         CALL DBLTRAP(DINTGL,TSTEPD,PSTEPD,TZD,PZD,TFD,PFD)
         FACTOR = 4.0 * PI / DINTGL
 200  CONTINUE
      THETA = THETAD * D2R
      PHI = PHID * D2R
      XGAIN = F2 (THETA,PHI)
      IF (XGAIN . EQ .0.0) XGAIN = 1.E-10
      XGAIN = 10 * ALOG10(ABS(XGAIN * FACTOR))
 901  FORMAT(T1,E9.3,T11,E9.3,T22,E9.3)
 1000 FORMAT(E10.4)
  999 FORMAT(6F7.2)
      RETURN
      END
C***********************************************************************
      BLOCK DATA CONSTNT
C***********************************************************************
      COMMON/CONST/ PI,VOFL,PI2,PIO2,D2R,R2D
      DATA PI,PI2,PIO2 / 3.14159265359, 6.283185308, 1.570796327/
      DATA D2R,R2D,VOFL / .017453293, 7.2957795, 2.997925E2/
      END
C********************************************************************
```

```
      SUBROUTINE DBLTRAP(DINTGL,TSTEPD,PSTEPD,TZD,PZD,TFD,PFD)
C
C*******************************************************************
C
C                     MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  PATTRN                   FILE NAME:
C  LOAD MODULE:  DBLTRAP                  HID:
C  CPU TYPE:  CYBER840                    LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*******************************************************************
C
C                            PURPOSE
C
C  This subroutine calculates a double integral in theta and phi
C  using a simple trapezoidal rule approximation.
C
C*******************************************************************
C
C                     TECHNICAL INFORMATION
C
C  See purpose.
C
C*******************************************************************
C
C                        SUBPROGRAM USAGE
C
C  F2 - calculates the integrand.
C
C*******************************************************************
C
C                         CHANGE HISTORY
C
C*******************************************************************
C
C                        MNEMONIC LISTING
C
C  PARAMETERS:  n 94 7
C     DINTGL  -  INTEGRATION RESULT CALCULATED IN DBLTRAP
C     TSTEPD  -  THETA STEP SIZE IN DEGREES
C     PSTEPD  -  PHI STEP SIZE IN DEGREES
C     TZD     -  THETA LOWER LIMIT OF INTEGRATION IN DEGREES
C     PZD     -  PHI LOWER LIMIT OF INTEGRATION IN DEGREES
C     TFD     -  THETA UPPER LIMIT OF INTEGRATION IN DEGREES
C     PFD     -  PHI UPPER LIMIT OF INTEGRATION IN DEGREES
C
C
C COMMON BLOCKS:
C    CONST
C       PI      -  MATHEMATICAL SYMBOL PI = 3.14159
C       VOFL    -  VELOCITY OF LIGHT  FOR FREQUENCY IN MEGAHERTZ = 3.0E-2
C       PI2     -  2 * PI
C       PIO2    -  PI / 2
```

```
C       D2R      -  CONVERSION FACTOR FOR DEGREES TO RADIANS CONVERSION
C                     = PI / 180.
C       R2D      -  CONVERSION FACTOR FOR RADIANS TO DEGREES CONVERSION
C                     = 180 / PI
C
C
C*********************************************************************
C*********************************************************************
C  EXECUTABLE CODE
C*********************************************************************
C
C
C
      EXTERNAL F2
      COMMON/CONST/ PI,VOFL,PI2,PIO2,D2R,R2D
      THETAZ = TZD * D2R
      PHIZ = PZD * D2R
      PSTEP = PSTEPD * D2R
      TSTEP = TSTEPD * D2R
      THETAF = TFD * D2R
      PHIF = PFD *D2R
      TINT = 0.0
      T = THETAZ
      NTSTEPS = INT(ABS((THETAF - THETAZ) / TSTEP))
      NPSTEPS = INT(ABS((PHIF - PHIZ) / PSTEP))
      DO 80 IP1 = 1,NTSTEPS+1
         I = IP1 - 1
         P = PHIZ + PSTEP
      PINT = 0.0
      DO 90 J = 1,NPSTEPS-1
      PINT = PINT + F2(T,P)
         P = P + PSTEP
  90    CONTINUE
          PINT = PINT * PSTEP
          PINT = PINT + (PSTEP/2.) * (F2(T,P) + F2(T,PHIZ))
          IF (I.EQ.0 .OR. I.EQ.NTSTEPS) GO TO 91
          TINT = TINT + TSTEP * PINT *COS(T)
          GO TO 92
  91    TINT = TINT + TSTEP * PINT * COS(T) / 2.0
  92    CONTINUE
        T = T + TSTEP
  80  CONTINUE
      DINTGL =  TINT
      RETURN
      END
C***********************************************************************
      FUNCTION F2(THETA,PHI)

C*********************************************************************
C
C                       MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  PATTRN                     FILE NAME:
C  LOAD MODULE:  F2                         HID:
```

```
C  CPU TYPE:  CYBER840                    LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*********************************************************************
C
C                              PURPOSE
C
C  This function calculates the integrant of equation(10), NTIA Report
C
C*********************************************************************
C
C                        TECHNICAL INFORMATION
C
C  See purpose.
C
C*********************************************************************
C
C                          SUBPROGRAM USAGE
C
C  None
C
C*********************************************************************
C
C                           CHANGE HISTORY
C
C*********************************************************************
C
C                          MNEMONIC LISTING
C
C  PARAMETERS:
C    THETA - takeoff angle in radians
C    PHI - azimuthal angle in radians
C
C
C  COMMON BLOCKS:
C     CONST
C       PI      -  MATHEMATICAL SYMBOL PI = 3.14159
C       VOFL    -  VELOCITY OF LIGHT  FOR FREQUENCY IN MEGAHERTZ = 3.0E-2
C       PI2     -  2 * PI
C       PIO2    -  PI / 2
C       D2R     -  CONVERSION FACTOR FOR DEGREES TO RADIANS CONVERSION
C                   = PI / 180.
C       R2D     -  CONVERSION FACTOR FOR RADIANS TO DEGREES CONVERSION
C                   = 180 / PI
C     FWAVE
C       EIL     -  0.5 * LENGTH OF DIPOLE IN WAVELENGTHS
C       XR      -  REFLECTOR TO DIPOLE SPACING IN WAVELENGTHS
C       C(8)    -  ARRAY OF RELATIVE STACK CURRENT MAGNITUDE RATIOS
C       R(14)   -  ARRAY OF RELATIVE BAY CURRENT MAGNITUDE RATIOS
C       PS(14)  -  ARRAY OF BAY CURRENT PHASE RATIOS IN RADIANS INCLUDING
C                      FREQUENCY ADJUSTMENT FOR OPERATING FREQUENCY NOT
C                      EQUAL TO DESIGN FREQUENCY
C       A(8)    -  ARRAY OF STACK CURRENT PHASE RATIOS IN RADIANS,
C                    CURRENTLY DEFAULTED TO ZERO
```

```
C       Y(14)    -  ARRAY CONTAINING THE SPACINGS BETWEEN THE BAY DIPOLES
C                      IN WAVELENGTHS (RELATIVE TO BAY 1)
C       Z(8)     -  ARRAY CONTAINING THE SPACINGS BETWEEN THE STACK DIPOLES
C                      IN WAVELENGTHS (RELATIVE TO STACK 1)
C ANTDAT
C  NOSTAK - NUMBER OF VERTICAL STACKS
C  STKSPM - SPACING BETWEEN STACKS
C  NUMBAY - NUMBER OF BAYS
C  BAYSPM - SPACING BETWEEN BAYS
C  DIPLNM - LENGTH OF DIPOLE RADIATOR
C  RRSPM - SPACING BETWEEN RADIATORS AND REFLECTING SCREEN
C  STKHTM - HEIGHT OF LOWEST STACK ABOVE GROUND
C  STKRAT - CURRENT RATIO FOR STACKS
C  BAYPHS - RELATIVE PHASE FOR THE BAYS
C  BAYRAT - CURRENT RATIO FOR THE BAYS
C  OFMHZ - ANTENNA OPERATING FREQUENCY
C  DFMHZ - ANTENNA DESIGN FREQUENCY
C  VSLEW - VERTICAL SLEW ANGLE
C  HSLEW - HORIZONTAL SLEW ANGLE
C
C
C*********************************************************************
C*********************************************************************
C  EXECUTABLE CODE
C*********************************************************************
C
C
C
C************************************************************************
      COMPLEX FY, FZTHETA, FZPHI
      COMMON/CONST/ PI,VOFL,PI2,PIO2,D2R,R2D
      COMMON/ANTDAT/NOSTAK,STKSPM,NUMBAY,BAYSPM,DIPLNM,RRSPM,
     1  STKHTM,STKRAT,BAYPHS,BAYRAT,OFMHZ,DFMHZ,VSLEW,HSLEW
      DIMENSION C(8),R(14),PS(14),A(8),Y(14),Z(8)
      COMMON/FWAVE/EIL,XR,C,R,PS,A,Y,Z,WAVE,BETA
      DIMENSION BAYRAT(14),STKRAT(14),BAYPHS(14)
           PHID = PHI * R2D
           THETAD = THETA * R2D
           CPHI = COS(PHI)
           SPHI = SIN(PHI)
           CTHETA = COS(THETA)
           STHETA = SIN(THETA)
           SEIL = SIN(EIL)
           CEIL = COS(EIL)
           CPSI = CTHETA * SPHI
           SPSI = SQRT(1. - CPSI * CPSI)
           IF( ABS( SPSI ) .LE. 1.0E-6 ) SPSI = 1.0E-6
           EF = (COS(EIL * CPSI) - CEIL) / (SPSI * SPSI)
           FY = CMPLX(0.,0.)
           DO 50 I  = 1,NUMBAY
           ARG = Y(I) * CTHETA * SPHI + PS(I)
  50       FY = FY + R(I) * CMPLX(COS(ARG),SIN(ARG))
           FX =  SIN( XR * CTHETA * CPHI)
           FZPHI = CMPLX(0.,0.)
```

```
      DO 60 I =1,NOSTAK
          FZPHI = FZPHI +  C(I) * SIN(Z(I)* STHETA) *
   1                    CMPLX(COS(A(I)),SIN(A(I)))
 60    CONTINUE
      FZTHETA = FZPHI * STHETA * SPHI
      FZPHI = FZPHI * CPHI
      F2 = EF * EF  * FX * FX  *  FY * CONJG(FY)
   1       * (FZTHETA * CONJG(FZTHETA)  +  FZPHI * CONJG(FZPHI))
1111 FORMAT(T1,F7.2,T10,F7.2,T20,E10.4,'+I',E10.4,T45,E9.3,T56,E9.3,
   1  T65,E9.3,'+I',E9.3,T87,E9.3,'+I',E9.3,T109,E9.3,'+I',E9.3)
1110 FORMAT(T5,'THETA',T15,'PHI',T30,'F',T50,'EF',T61,'FX',T80,'FY',
   1  T99,'FZTHETA',T114,'FZPHI')
     RETURN
     END
```

APPENDIX C. SOURCE CODE FOR ONE POSSIBLE IMPLEMENTATION OF PROGRAM VALDPR

```
$EMA /ION/
$FILES (0,2)
      PROGRAM VALDPR(),  Validation data processor  <870325.1033>

C**********************************************************************
C
C                    MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                FILE NAME:
C  LOAD MODULE: VALDPR                 HID:
C  CPU TYPE:  HP1000                   LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C**********************************************************************
C
C                           PURPOSE
C
C  This program creates a data base from IONCAP output that can be used by
C  VALSUM.
C
C     NOTE: The IONCAP output processed by this program has all blank lines
C           removed.
C**********************************************************************
C
C                    TECHNICAL INFORMATION
C
C  See purpose.
C
C**********************************************************************
C
C                       SUBPROGRAM USAGE
C
C  LGBUF - HP buffer command
C  GETLN - reads block of data from tape
C  RD_COMMENT - reads comments from tape
C  TODAY - HP routine for date
C  RD_INDX - reads data base index of tapes that have been processed
C  OPNVM - HP routine to open VMA file
C  LURQ - HP routine to lock tape drive
C  VAL_HOUR - determines data base location of hour
C  WR_DAT - writes data to file
C  CLSVM - HP routine to close VMA file
C
C**********************************************************************
C
C                        CHANGE HISTORY
C
C**********************************************************************
C
C                       MNEMONIC LISTING
C
```

```
C  PARAMETERS:
C     None
C
C  COMMON BLOCKS:
$INCLUDE INDX_VALUES
$Include /voa/valsum/calc/ionema
$INCLUDE NUM
$INCLUDE VALUES
$INCLUDE INDX_NUM
C
C**********************************************************************
C**********************************************************************
C  EXECUTABLE CODE
C**********************************************************************
C
      LOGICAL EXIST, VALID
      REAL FRQS(12)
      INTEGER*2 TBUF(5120), NAME(3), IARL(2), IDXNAM(6), IHRS(24),
      +          IFRQS(12),INDX_FRQS(12)
      INTEGER IDATE(14)
      INTEGER*4 CIRCARR(400), CIRCID, CIROLD

      CHARACTER LINE*79,CMODE*2,MONCH*3,ILAT*1,TSITE*4,
      +  ILON*1,JLAT*1,JLON*1, DBNAME*4,STRING*80

      DATA CIROLD/10000/
      DATA TSITE/'TNGR'/
      DATA IARL/7,8/
      DATA IDXNAM/'    IN:BM:VM'/
      DATA NAME/'    DB'/
      DATA ISC/'JG'/,ICR/'VM'/

C     Use HP system routine to set large buffer for tape

      CALL LGBUF(TBUF,5120)

50    CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
      +  TSITE,*150,*100)
      GOTO 50

100   CALL RD_COMMENT(LINE,IHRS)    ! Read the comment lines in IONCAP output
      GOTO 200

150   WRITE(1,'(//" The tape is an IONCAP tape but not for VALSUM")')

      STOP

C  Use HP routine to get today's date

200   CALL TODAY(IDATE)

C  Ask for name of data base to be created
      WRITE (1,'("Data base name (4 characters)? _")')
      READ (1,'(A4)') DBNAME
```

```
C
C  Read index file and determine if this tape is part of an existing data base
C
      CALL RD_INDX (DBNAME,IFRQ_TOTAL,INDX_FRQS)
C
C  Rewrite index file
         OPEN (22,FILE='VALND2::VM',ERR=1000,IOSTAT=IOS,USE='NONEXCLUSIVE')
         OPEN (23,FILE='VALTMP::VM',ERR=1001,IOSTAT=IOS,USE='NONEXCLUSIVE')
15    READ (23,'(A80)',END=250,ERR=1002,IOSTAT=IOS) STRING
      WRITE (22,'(A80)') STRING
      GOTO 15
C  Use HP routine to OPEN VMA data base file

250   CLOSE (22)
      CLOSE (23,STATUS='DELETE')
      IOPTN = 2

      READ (DBNAME(1:2),'(A2)') NAME(1)
      READ (DBNAME(3:4),'(A2)') NAME(2)
      CALL OPNVM(NAME,IERR,IOPTN,ISC,ICR)

      IF (IERR .NE. 0) THEN

         WRITE (1,'("Error #",I5," OPENing data base ",3A2)') IERR, NAME

      STOP

      ENDIF

C  OPEN circuit index file

      IDXNAM(1) = NAME(1)

      IDXNAM(2) = NAME(2)

C  Use HP routine to LOCK tape drive to this program

         CALL LURQ(1,IARL,1)

C  OPEN FILE WHICH CONTAINS CIRCUIT INDEX.  THE FIRST LINE OF THIS
C  CONTAINS THE CREATION DATE, THE SECOND LINE CONTAINS THE XMTR LAT
C  AND LON.  THE 3RD LINE CONTAINS THE NUMBER OF CIRCUITS. THE FOLLOWING
C  LINES CONTAIN THE CIRCUIT IDS - EX. TNGR10001.
C
         OPEN (99,FILE=IDXNAM,IOSTAT=IOS,ERR=5000,USE='NONEXCLUSIVE')
C
C  READ INDEX FILE
C
      NCIR = 0
      READ (99,'(A2)',END=7,ERR=1098) IDUMMY
      READ (99,'(A2)',END=7,ERR=1098) IDUMMY
      READ (99,'(I5)',END=7,ERR=1098) NCIR
      READ (99,'(A2)',END=7,ERR=1098) IDUMMY
      READ (99,'(A2)',END=7,ERR=1098) IDUMMY
```

```
7     IF (NCIR .NE. 0) THEN
        DO 5 I=1,NCIR
          READ (99,'(4X,I5)',END=1098,ERR=1098) CIRCARR(I)
      5 CONTINUE
      REWIND (99)
      ENDIF
C
C  LINKNO IS INITIALIZED TO NUMBER OF CIRCUITS CURRENTLY IN THE FILE
C
      LINKNO = NCIR
      LINKLOC = LINKNO
      GOTO 350

C     NOW POSITIONED AT FIRST LINE OF A PAGE OF IONCAP OUTPUT
C     Locate start of IONCAP output tables

325   CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
      +   TSITE,*350,*100)
      GOTO 325

C     Now positioned at first data line of IONCAP output

350   CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
      +  TSITE,*900)

C     There are two output tables of results per page of IONCAP output.
C     ITP is used to make sure both tables are used.

      ITP = 0

      READ(LINE(13:15),'(A3)') MONCH
      IF(MONCH.EQ.'JAN') MONIN=1
      IF(MONCH.EQ.'APR') MONIN=2
      IF(MONCH.EQ.'JUL') MONIN=3
      IF(MONCH.EQ.'OCT') MONIN=4

      READ(LINE(40:42),'(I3)') ISUN

      IF(ISUN.EQ.10) ISIN=1
      IF(ISUN.EQ.120) ISIN=2

      CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
      +  TSITE,*900)

      READ(LINE(3:11),'(A4,I5)') TSITE,CIRCID

C     The circuit ID should be used to create an index of test points
C     stored in the data base.

C     If CIRCID has changed, increment INDEX

      IF(CIRCID.NE.CIROLD) THEN

         LINKNO = LINKNO + 1
```

```
      LINKLOC = LINKNO
      CIRCARR(LINKLOC) = CIRCID
      CIROLD=CIRCID

C  Check for duplicate circuit ID
      DO I=1,LINKNO-1
       IF (CIRCARR(I) .EQ. CIRCID) THEN
         LINKNO = LINKNO - 1
         LINKLOC = I
       ENDIF
      END DO

   ENDIF

   IF(LINE(45:52).NE.'AZIMUTHS') GOTO 900
   CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
   +  TSITE,*900)

   READ(LINE(1:79),9000) XD1,ILAT,XD2,ILON,XLAT,JLAT,XLON,JLON,AZ,DST

9000  FORMAT(2X,F5.2,X,A1,2X,F6.2,X,A1,3X,F5.2,X,A1,2X,F6.2,X,A1,4X,
   &        F6.2,20X,F7.1)

160   CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
   +  TSITE,*900)

   IF(LINE(67:71).NE.'FREQ ') GOTO 160

   READ( LINE(1:3),'(I3)') NOW_TIME

   ITP = ITP+1

   READ(LINE(2:55),'(I2,3X,F4.1,12F5.1)') ITIM,FMUF,(FRQS(I),I=1,12)
C
C  Save frequencies as integer values
   DO I = 1,12
     IFRQS(I) = NINT(FRQS(I)*100)
   END DO

C     Find the correct index for hour location in the data base IVALDT
   CALL VAL_HOUR( ITIM, ITIM_INDX,IHRS,NUM_HRS)
   WRITE(1,*) ITIM,MONIN,ISIN,LINKLOC,CIRCID,XLAT,XLON

   IF( MONIN .EQ. 1 ) THEN   ! Save xmtr lat & lon, rcvr lat & lon, az dist

      IF (ILAT .EQ. 'S') XD1 =-XD1
      IF (ILON .EQ. 'W') XD2 =-XD2
      TLOC(1) = XD1
      TLOC(2) = XD2
      IF (JLAT .EQ. 'S') XLAT =-XLAT
      IF (JLON .EQ. 'W') XLON =-XLON
      RLOC(1,LINKLOC) = XLAT
      RLOC(2,LINKLOC) = XLON
      BEAR(LINKLOC) = AZ
```

```
          DIST(LINKLOC) = DST

       END IF

170    CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
     +  TSITE,*900)

       DO I=1,NUM_CONF  ! get mode
C   Get frequency location
          IFQ = IFREQ_LOC (IFRQS(I),IFRQ_TOTAL,INDX_FRQS)

          J=(I-1)*5+12

          READ(LINE(J:J+3),'(I2,A2)') IHN,CMODE

          IF(CMODE.EQ.'F1') MODE=1
          IF(CMODE.EQ.'F2') MODE=2
          IF(CMODE.EQ.' E') MODE=3

          IVALUE = IHN*10 + MODE
       CALL WR_DAT(1,IFQ,ITIM_INDX,MONIN,ISIN,LINKLOC,IVALUE,IFRQ_TOTAL)

       END DO            !  mode

180    CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
     +  TSITE,*900)

       DO I=1,NUM_CONF  ! get take-off angle
C   Get frequency location
          IFQ = IFREQ_LOC (IFRQS(I),IFRQ_TOTAL,INDX_FRQS)

          J=(I-1)*5+12

          READ(LINE(J:J+3),'(F4.1)') FANG

          IVALUE = FANG*10.0
       CALL WR_DAT(2,IFQ,ITIM_INDX,MONIN,ISIN,LINKLOC,IVALUE,IFRQ_TOTAL)

       END DO            ! take-off angle

210    CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
     +  TSITE,*900)

       IF(LINE(67:71).NE.'RPWRG') GOTO 210

       DO I=1,NUM_CONF  ! get required power gain

C   Get frequency location
          IFQ = IFREQ_LOC (IFRQS(I),IFRQ_TOTAL,INDX_FRQS)
          J=(I-1)*5+12

          RPWRG = 1000.0

          READ(LINE(J:J+3),'(F4.0)') RPWRG
```

```
         IVALUE = NINT(RPWRG)
      CALL WR_DAT(3,IFQ,ITIM_INDX,MONIN,ISIN,LINKLOC,IVALUE,IFRQ_TOTAL)

      END DO           ! required power gain

      CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
      +  TSITE,*900)

230   CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
     +  TSITE,*900)

      DO I=1,NUM_CONF     ! get multipath probability
C  Get frequency location
         IFQ = IFREQ_LOC (IFRQS(I),IFRQ_TOTAL,INDX_FRQS)

         J=(I-1)*5+12

         READ(LINE(J:J+3),'(F4.2)') XMPROB

         IVALUE = XMPROB*100.0
      CALL WR_DAT(4,IFQ,ITIM_INDX,MONIN,ISIN,LINKLOC,IVALUE,IFRQ_TOTAL)

      END DO      ! multipath probability

      IF (NOW_TIME .EQ. IENDAM) THEN
         KTIME = IENDAM - ISTAM + 1
         IF (MOD(KTIME,2) .NE. 0) GOTO 500
      ELSE IF (NOW_TIME .EQ. IENDPM) THEN
         KTIME = IENDPM - ISTPM + 1
         IF (MOD(KTIME,2) .NE. 0) GOTO 500
      ENDIF

      IF(ITP.EQ.1) GOTO 160

500   CALL GETLN(LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,IFRQ_TOTAL,
     + TSITE,*350,*100)
      GOTO 500

900   CONTINUE

      WRITE(1,*) 'SYNTAX ERROR IN TAPE'
      GOTO 999
1098  WRITE (1,'("READ ERROR #",I5)') IOS
      GOTO 999
5000  WRITE (1,'("Error #",I5," OPENing index file")') IOS

999   CONTINUE
      CALL CLSVM

      STOP

1000  WRITE (1,'("ERROR #",I5,"OPENING VALIND")') IOS
      STOP
```

```
1001  WRITE (1,'("ERROR #",I5,"OPENING VALTMP")') IOS
      STOP
1002  WRITE (1,'("ERROR #",I5,"READING FROM VALTMP")') IOS
      STOP
      END

$EMA /ION/
      SUBROUTINE GETLN(LC,CIRCARR,LINKNO,XD1,XD2,IDATE,INDX_FRQS,
      +  IFRQ_TOTAL,TSITE,*,*)

C*******************************************************************
C
C                    MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                  FILE NAME:
C  LOAD MODULE:  GETLN                   HID:
C  CPU TYPE:  HP1000                     LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*******************************************************************
C
C                           PURPOSE
C
C  This routine reads an 80 character from tape
C
C*******************************************************************
C
C                    TECHNICAL INFORMATION
C
C  See purpose.
C
C*******************************************************************
C
C                       SUBPROGRAM USAGE
C
C*******************************************************************
C
C                        CHANGE HISTORY
C
C*******************************************************************
C
C                       MNEMONIC LISTING
C
C  PARAMETERS:
C    LC - character string read from tape
C    CIRCARR - array of circuit ids
C    LINKNO - circuit number
C    XD1 - circuit latitutde
C    XD2 - circuit longitude
C    IDATE - date
C    INDX_FRQS - frequencies in data base
C    IFRQ_TOTAL - total number of frequencies
C    TSITE - xmtr site name
C
```

```
C  COMMON BLOCKS:
$Include /voa/valsum/calc/ionema
$INCLUDE NUM
$INCLUDE VALUES
      COMMON /TSTUF/ TBLOC,IREC,ITB,ITLU,IEOF,IBF
C
C*****************************************************************
C*****************************************************************
C  EXECUTABLE CODE
C*****************************************************************
C

      CHARACTER*80 TBLOC(128), IFMT*80, TSITE*4
      INTEGER*4 CIRCARR(400)
      CHARACTER*79 LC
      INTEGER IDATE(14),INDX_FRQS(12)

      IREC=IREC+1

      IF(IREC.GT.IBF) THEN

         READ(ITLU,END=987,ERR=50) TBLOC

50      LEN = ITLOG()    ! HP function to tell how many characters from READ

      IBF = LEN/80

         IREC=1

      ENDIF

100   LC=TBLOC(IREC)(2:80)

C     The 'METHOD 20' line indicates the start of an output page

      IF(LC(33:41).EQ.'METHOD 20') RETURN 1

C     The 'COMMENT' line indicates the start of the COMMENTs from IONCAP

      IF( LC(1:7) .EQ. 'COMMENT' ) RETURN 2

      RETURN

987   CONTINUE

C     Write circuit index file

      WRITE (99,'(7A2,1A1)')(IDATE(I),I=1,8)
      WRITE (99,'(F6.2,X,F7.2)') XD1, XD2
      WRITE (99,'(2I5)') LINKNO, IFRQ_TOTAL
      WRITE (99,'(4I3)') ISTAM, IENDAM, ISTPM, IENDPM
      WRITE (IFMT,101) IFRQ_TOTAL
101   FORMAT ('(',I1,'I4)')
      WRITE (99,IFMT) (INDX_FRQS(I),I=1,IFRQ_TOTAL)
```

```
      DO I=1,LINKNO
         WRITE (99,'(A4,I5,F6.1,2X,F7.1)')TSITE(1:4),CIRCARR(I),BEAR(I),
      +        DIST(I)
      END DO
      CALL CLSVM                ! HP routine to close VMA file
      CLOSE (99)

      STOP 'EOF'
1000  WRITE (LUT,'("ERROR #",I5," OPENING VALIND")') IOS
      STOP
1001  WRITE (LUT,'("ERROR #",I5," OPENING VALTMP")') IOS
      STOP
1002  WRITE (LUT,'("ERROR #",I5," READING FROM VALTMP")') IOS

      END
      BLOCK DATA

      COMMON /TSTUF/ TBLOC,IREC,ITB,ITLU,IEOF,IBF
$INCLUDE NUM

      CHARACTER*80 TBLOC(128)

      DATA IREC/0/,ITB/0/,ITLU/7/,IEOF/0/,IBF/0/

      DATA NUM_PARAM /4/, NUM_MON /4/, NUM_SSN /2/

      END

$EMA /ION/
      SUBROUTINE WR_DAT( IPARAM,IFREQ,IHR,IMON,ISSN,ICKT,IVALUE,
    +  IFRQ_TOTAL)

C**********************************************************************
C
C                    MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                    FILE NAME:
C  LOAD MODULE:  WR_DAT                    HID:
C  CPU TYPE:  HP1000                       LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C**********************************************************************
C
C                           PURPOSE
C
C     Locate the correct location in the IVALDT file to write the parameter's
C      value.  By using a single dimensioned file the program can make use of
C      a variable number of frequencies
C     (IFRQ_TOTAL) and hours (NUM_HRS).
C
C**********************************************************************
C
C                    TECHNICAL INFORMATION
```

```
C
C   See purpose.
C
C**********************************************************************
C
C                             SUBPROGRAM USAGE
C
C   None
C
C**********************************************************************
C
C                              CHANGE HISTORY
C
C**********************************************************************
C
C                             MNEMONIC LISTING
C
C   PARAMETERS:
C          IPARAM - no. of the parameter whose value is to be written to IVALDT
C          IFREQ  - frequency number
C          IHR    - hour number
C          IMON   - month number
C          ISSN   - sunspot number
C          ICKT   - circuit number
C          IVALUE - value of parameter to be written
C
C   COMMON BLOCKS:
$Include /voa/valsum/calc/ionema
$INCLUDE NUM
C
C**********************************************************************
C**********************************************************************
C   EXECUTABLE CODE
C**********************************************************************
C
      INTEGER*4 IREC,DCKT
      NUM_FQ = 12
      DCKT = ICKT
      IREC = IPARAM+NUM_PARAM*( (IFREQ - 1)+NUM_FQ*( (IHR - 1) +
     &       NUM_HRS*( (IMON - 1) + NUM_MON*( (ISSN - 1) + NUM_SSN*(
     &       (DCKT - 1) )))))

      IVALDT(IREC) = IVALUE

      RETURN

      END
$EMA /ION/
      SUBROUTINE RD_INDX (DBNAME,IFRQ_TOTAL,INDX_FRQS)
C
      LOGICAL EXIST, VALID
      CHARACTER DBNAME*4, DBASE*4, IDUMMY*1
      INTEGER*2 HRS(24), INDX_FRQS(12)
C
```

```
C***********************************************************************
C
C                       MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALSUM                   FILE NAME:
C  LOAD MODULE:  RD_INDX                  HID:
C  CPU TYPE:  HP1000                      LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C***********************************************************************
C
C                              PURPOSE
C
C  This subroutine reads the index file which contains a description of all of
C  the VALSUM data bases and determines if the new data is part of another
C  data base.
C
C***********************************************************************
C
C                        TECHNICAL INFORMATION
C
C  See purpose.
C
C***********************************************************************
C
C                           SUBPROGRAM USAGE
C
C***********************************************************************
C
C                            CHANGE HISTORY
C
C***********************************************************************
C
C                           MNEMONIC LISTING
C
C  PARAMETERS:
C    DBNAME - name of data base for new data
C    IFRQ_TOTAL - total number of configurations
C    INDX_FRQS - frequencies in data base
C
C  COMMON BLOCKS:
C
$INCLUDE LU
$INCLUDE INDX_NUM
$INCLUDE INDX_VALUES
$INCLUDE NUM
$INCLUDE VALUES
C
C***********************************************************************
C***********************************************************************
C  EXECUTABLE CODE
C***********************************************************************
C
      EXIST = .FALSE.
```

```
      OPEN (22,FILE='VALND2::VM',ERR=1000,IOSTAT=IOS,USE='NONEXCLUSIVE')
      OPEN (23,FILE='VALTMP::VM',ERR=1001,IOSTAT=IOS,USE='NONEXCLUSIVE')
C
5     DO 10 I=1,3
         READ (22,'(A1)',END=100) IDUMMY
10    CONTINUE
C
      READ (22,'(A4,2X,A4,2X,A8,2X,A4,3X,I1,2X,F5.1,1X,F5.1,1X,F5.1,1X,
     +      F7.1,1X,F7.1)') DBASE,XXSITE,ANTNAME,NLANG,NVALID,ZBEAR,
     +      BAZM,EAZM,BDST,EDST
      DO 20 I=1,5
         READ (22,'(A1)') IDUMMY
20    CONTINUE
      READ (22,'(I2,X,I2,1X,I2,X,I2,3X,I2,3X,A12,F7.1,X,F8.1)')
     +      IBEGHRA,IENDHRA,IBEGHRP,IENDHRP,N_CONF,IND_DATE,TLAT,TLONG
C
      CALL HOURS(IBEGHRA,IENDHRA,IBEGHRP,IENDHRP,HRS,N_HRS)
C
      DO 25 I = 1,3
         READ (22,'(A1)') IDUMMY
25    CONTINUE
      DO 30 I = 1,N_CONF
         READ(22,'(I5,5X,I2,5X,I2,3X,I3,4X,I5,5X,I5)',END=22,ERR=21)
     +     (NEWCONF(I,J),J=1,6)
30    CONTINUE
      IF (DBASE .EQ. DBNAME) THEN
         EXIST = .TRUE.
         CALL MATCH (VALID)
         IF (.NOT. VALID) THEN
           WRITE (1,'("Data base parameters and tape parameters do not",
     +     " match - PROGRAM ABORTED")')
           STOP
         ENDIF
C  Determine if duplicate frequencies
         DO L = 1,NUM_CONF
         DO J = 1,N_CONF
         IF (ICONFIG(L,1) .EQ. NEWCONF(J,1)) THEN
         WRITE (1,'("DUPLICATE FREQUENCY - PROGRAM ABORTED")')
         STOP
         ENDIF
         END DO
         END DO
C  Determine if too many frequencies
      IFRQ_TOTAL = NUM_CONF + N_CONF
      IF (IFRQ_TOTAL .GT. 12) THEN
         WRITE (1,'("Too many frequencies - PROGRAM ABORTED")')
         STOP
      ELSE
C  Set frequency defaults
         J = 0
         DO K = N_CONF+1,IFRQ_TOTAL
         J = J + 1
         NEWCONF(K,1) = ICONFIG(J,1)
         END DO
```

```
         DO K = 1,IFRQ_TOTAL
         INDX_FRQS(K) = NEWCONF(K,1)
         END DO
C  Write new data to indx file
         CALL WRITE_NEW_INDX(DBNAME,NEWCONF,IFRQ_TOTAL)
      ENDIF   !  adding to data base
      ELSE
C  Write old data
         CALL WRITE_OLD_INDX (DBASE)
      ENDIF
      GOTO 5
100   IF (.NOT. EXIST) THEN
C  Write new data to index file
         CALL WRITE_NEW_INDX(DBNAME,ICONFIG,NUM_CONF)
         IFRQ_TOTAL = NUM_CONF
         DO K = 1,IFRQ_TOTAL
         INDX_FRQS(K) = ICONFIG(K,1)
      END DO
      ENDIF
      CLOSE(22)
      CLOSE(23)
      RETURN
1000  WRITE (1,'("ERROR #",I5," OPENING VALIND")') IOS
      STOP
1001  WRITE (1,'("ERROR #",I5," OPENING VALTMP")') IOS
      STOP
21    WRITE (1,'("ERROR #",I5," READING FROM VALIND")') IOS
      STOP
22    WRITE (1,'("EOF READING FROM VALIND")')
      STOP
      END
      SUBROUTINE WRITE_NEW_INDX (DBNAME,NEWCONF,IFRQ_TOTAL)
C
      CHARACTER IDUMMY*1, DBNAME*4
      INTEGER NEWCONF(12,6)
C**********************************************************************
C
C                     MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                  FILE NAME:
C  LOAD MODULE:  WRITE_NEW_INDX          HID:
C  CPU TYPE:  HP1000                     LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C**********************************************************************
C
C                            PURPOSE
C
C  This subroutine writes the index file for the VALSUM program with
C  information for a new file.
C
C**********************************************************************
C
C                     TECHNICAL INFORMATION
```

```
C
C  See purpose.
C
C***********************************************************************
C
C                          SUBPROGRAM USAGE
C
C  None.
C
C***********************************************************************
C
C                           CHANGE HISTORY
C
C***********************************************************************
C
C                          MNEMONIC LISTING
C
C  PARAMETERS:
C     DBNAME - data base name
C     NEWCONF - configurations
C     IFRQ_TOTAL - total number of configurations
C
C  COMMON BLOCKS:
$INCLUDE NUM
$INCLUDE VALUES
C
C***********************************************************************
C***********************************************************************
C  EXECUTABLE CODE
C***********************************************************************
C
20    WRITE (23,'("File",2X,"Site",4X,"Ant",5X,"Lang",2X,"Val",2X,
     +        "Ant",11X,"Sector")')
      WRITE (23,'("Name",2X,"Name",4X,"Name",15X,"Brg",8X,"Brg",7X,
     +        "Range")')
      WRITE (23,'("----",2X,"----",2X,"--------",2X,"----",2X,"---",
     +       1X,"-----",1X,"-----------",1X,"---------------")')
      WRITE (23,'(A4,2X,A4,2X,A8,2X,A4,3X,I1,2X,F5.1,1X,F5.1,1X,F5.1,1X,
     +       F7.1,1X,F7.1,/)') DBNAME,SITE,ANTENA_ID,LANG,IVALID,XBEAR,
     +       BAZIM,EAZIM,BDIST,EDIST
      WRITE (23,'("TIME BLOCKS",3X,"NUM",5X,"DATE",7X,"TLAT",5X,
     +   "TLONG")')
      WRITE (23,'(1X,"AM",5X,"PM",3X,"CONFS")')
      WRITE (23,'("-----",1X,"-----",2X,"-----",1X,"-----------",
     +       X,"-------",X,"--------"/)')
      WRITE (23,'(I2,"-",I2,1X,I2,"-",I2,3X,I2,3X,A12,F7.1,X,F8.1/)')
     +       ISTAM,IENDAM,ISTPM,IENDPM,IFRQ_TOTAL,ION_DATE,XLAT,XLONG
        WRITE(23,'(" FREQ",3X,"BAYS",2X,"EXCIT",2X,"SLEW",2X,"DSGN FRQ",
     +  2X,"STK HGT",/,"-----",3X,"----",2X,"-----",2X,"----",2X,
     +  "--------",2X,"-------")')
      DO I = 1,IFRQ_TOTAL-NUM_CONF
        WRITE(23,'(I5,5X,I2,5X,I2,3X,I3,4X,I5,5X,I5)')
     +    (NEWCONF(I,J),J=1,6)
      END DO
```

```
      DO I = 1,NUM_CONF
        WRITE(23,'(I5,5X,I2,5X,I2,3X,I3,4X,I5,5X,I5)')
     +    (ICONFIG(I,J),J=1,6)
      END DO
      RETURN
C
      END
      SUBROUTINE WRITE_OLD_INDX(DBASE)
C
      CHARACTER IDUMMY*1, DBASE*4
C**********************************************************************
C
C                      MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                    FILE NAME:
C  LOAD MODULE:  WRITE_OLD_INDX               HID:
C  CPU TYPE:  HP1000                       LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C**********************************************************************
C
C                             PURPOSE
C
C  This subroutine writes the data which was previously in the index file.
C
C**********************************************************************
C
C                      TECHNICAL INFORMATION
C
C  See purpose.
C
C**********************************************************************
C
C                          SUBPROGRAM USAGE
C
C  None.
C
C**********************************************************************
C
C                           CHANGE HISTORY
C
C**********************************************************************
C
C                          MNEMONIC LISTING
C
C  PARAMETERS:
C    None.
C
C  COMMON BLOCKS:
$INCLUDE INDX_NUM
$INCLUDE INDX_VALUES
C
C**********************************************************************
C**********************************************************************
```

```
C  EXECUTABLE CODE
C*********************************************************************
C
20     WRITE (23,'("File",2X,"Site",4X,"Ant",5X,"Lang",2X,"Val",2X,
     +         "Ant",11X,"Sector")')
       WRITE (23,'("Name",2X,"Name",4X,"Name",15X,"Brg",8X,"Brg",7X,
     +         "Range")')
       WRITE (23,'("----",2X,"----",2X,"--------",2X,"----",2X,"---",
     +        1X,"-----",1X,"-----------",1X,"---------------")')
       WRITE (23,'(A4,2X,A4,2X,A8,2X,A4,3X,I1,2X,F5.1,1X,F5.1,1X,F5.1,1X,
     +        F7.1,1X,F7.1,/)') DBASE,XXSITE,ANTNAME,NLANG,NVALID,ZBEAR,
     +        BAZM,EAZM,BDST,EDST
       WRITE (23,'("TIME BLOCKS",3X,"NUM",5X,"DATE",7X,"TLAT",5X,
     +    "TLONG")')
       WRITE (23,'(1X,"AM",5X,"PM",3X,"CONFS")')
       WRITE (23,'("-----",1X,"-----",2X,"-----",1X,"-----------",
     +        X,"-------",X,"--------"/)')
       WRITE (23,'(I2,"-",I2,1X,I2,"-",I2,3X,I2,3X,A12,F7.1,X,F8.1/)')
     +        IBEGHRA,IENDHRA,IBEGHRP,IENDHRP,N_CONF,IND_DATE,TLAT,TLONG
         WRITE(23,'(" FREQ",3X,"BAYS",2X,"EXCIT",2X,"SLEW",2X,"DSGN FRQ",
     +   2X,"STK HGT",/,"-----",3X,"----",2X,"-----",2X,"----",2X,
     +   "--------",2X,"-------")')
       DO I = 1,N_CONF
         WRITE(23,'(I5,5X,I2,5X,I2,3X,I3,4X,I5,5X,I5)')
     +      (NEWCONF(I,J),J=1,6)
       END DO
       RETURN
C
       END
       FUNCTION IFREQ_LOC (IFRQ,IFRQ_TOTAL,INDX_FRQS)
C
       INTEGER IFRQS(12),INDX_FRQS(12)
C*********************************************************************
C
C                        MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                   FILE NAME:
C  LOAD MODULE:                           HID:
C  CPU TYPE:  HP1000                      LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*********************************************************************
C
C                              PURPOSE
C
C  This routine determines where the frequencies run with IONCAP are
C  stored in the data base.
C
C*********************************************************************
C
C                         TECHNICAL INFORMATION
C
C  See purpose.
C
```

```
C*********************************************************************
C
C                          SUBPROGRAM USAGE
C
C*********************************************************************
C
C                           CHANGE HISTORY
C
C*********************************************************************
C
C                          MNEMONIC LISTING
C
C  PARAMETERS:
C    IFRQ - frequency
C    IFRQ_TOTAL - total number of frequencies in data base
C    INDX_FRQS - frequencies in data base
C
C  COMMON BLOCKS:
C
C*********************************************************************
C*********************************************************************
C  EXECUTABLE CODE
C*********************************************************************
C
      IFREQ_LOC = 0
C Determine where frequencies are in data base
      DO 10 I = 1, IFRQ_TOTAL
        JFRQ = NINT(INDX_FRQS(I) / 10.0)
          IF (IFRQ .EQ. INDX_FRQS(I) .OR.  IFRQ/10 .EQ. JFRQ) THEN
            IFREQ_LOC = I
            RETURN
          ENDIF
 10     CONTINUE
      WRITE (1,'("ERROR IN FINDING FREQUENCY LOCATION")')
      STOP
      END
      SUBROUTINE RD_COMMENT(LINE,IHRS)
C
      CHARACTER CIRCARR(200)*9, LINE*79
      INTEGER*2 IHRS(24)
C
C*********************************************************************
C
C                     MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                   FILE NAME:
C  LOAD MODULE:  RD_COMMENT               HID:
C  CPU TYPE:  HP1000                      LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*********************************************************************
C
C                             PURPOSE
C
```

```
C  This subroutine reads the comments from the IONCAP tape.
C
C*********************************************************************
C
C                          TECHNICAL INFORMATION
C
C  See purpose.
C
C*********************************************************************
C
C                             SUBPROGRAM USAGE
C
C  GETLN - reads a line of data from the IONCAP tape - line 59.
C
C*********************************************************************
C
C                              CHANGE HISTORY
C
C*********************************************************************
C
C                             MNEMONIC LISTING
C
C  PARAMETERS:
C    None
C
C  COMMON BLOCKS:
C
$INCLUDE NUM
$INCLUDE VALUES
C
C*********************************************************************
C*********************************************************************
C  EXECUTABLE CODE
C*********************************************************************
C
      KOUNT = 0
C
C  Decode 1st comment line
      GOTO 10
5     CALL GETLN (LINE,CIRCARR,LINKNO,XD1,XD2,IDATE,*100,*10)
10    IF (LINE(11:14) .EQ. 'DATE') THEN
C  Read date
         ION_DATE(1:12) = LINE(21:32)
C  Read xmtr site name and coordinates
      ELSE IF (LINE(11:18) .EQ. 'TRANSMIT') THEN
         SITE(1:4) = LINE(31:34)
         READ (LINE(47:55),'(F9.5)') XLAT
         READ (LINE(67:76),'(F10.5)') XLONG
C  Read antenna name, validation type, bearing, and number of configurations
      ELSE IF (LINE(11:20) .EQ. 'ANTENNA ID') THEN
         ANTENA_ID(1:8) = LINE(22:29)
         READ (LINE(46:46),'(I1)') IVALID
         READ (LINE(61:65),'(F5.1)') XBEAR
         READ (LINE(79:79),'(I1)') NUM_CONF
```

```
C  Read configuration - frequency, number of bays, excitation, slew angle
      ELSE IF (LINE(11:17) .EQ. 'ANTENNA') THEN
         KOUNT = KOUNT + 1
         READ (LINE(23:27),'(I5)') ICONFIG(KOUNT,1)
         READ (LINE(34:35),'(I2)') ICONFIG(KOUNT,2)
         READ (LINE(42:43),'(I2)') ICONFIG(KOUNT,3)
         READ (LINE(50:52),'(I3)') ICONFIG(KOUNT,4)
         READ (LINE(63:67),'(I5)') ICONFIG(KOUNT,5)
         READ (LINE(74:78),'(I5)') ICONFIG(KOUNT,6)
C  Read AM and PM starting and ending hours
      ELSE IF (LINE(11:19) .EQ. 'STARTHRAM') THEN
         READ (LINE(21:22),'(I2)') ISTAM
         READ (LINE(33:34),'(I2)') IENDAM
         READ (LINE(47:48),'(I2)') ISTPM
         READ (LINE(59:60),'(I2)') IENDPM

         CALL HOURS(ISTAM,IENDAM,ISTPM,IENDPM,IHRS,NUM_HRS)

C  Read language name
      ELSE IF (LINE(11:18) .EQ. 'LANGUAGE') THEN
         LANG = LINE(31:34)
C  Read sector beginning and ending distance and azimuth
      ELSE IF (LINE(11:16) .EQ. 'SECTOR') THEN
         READ (LINE(32:36),'(F5.1)') BAZIM
         READ (LINE (40:44),'(F5.1)') EAZIM
         READ (LINE(56:62),'(F7.1)') BDIST
         READ (LINE(66:72),'(F7.1)') EDIST
      ENDIF
C
      GOTO 5
100   RETURN
      END
      SUBROUTINE MATCH (VALID)
C
      LOGICAL VALID
C
C*********************************************************************
C
C                    MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                  FILE NAME:
C  LOAD MODULE:  MATCH                   HID:
C  CPU TYPE:  HP1000                     LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*********************************************************************
C
C                           PURPOSE
C
C  This subroutine determines if the new tape and an existing data base match.
C
C*********************************************************************
C
C                    TECHNICAL INFORMATION
```

```
C
C  See purpose.
C
C*********************************************************************
C
C                          SUBPROGRAM USAGE
C
C*********************************************************************
C
C                           CHANGE HISTORY
C
C*********************************************************************
C
C                          MNEMONIC LISTING
C
C  PARAMETERS:
C    VALID - set to true if tape and data base match
C
C  COMMON BLOCKS:
C
$INCLUDE NUM
$INCLUDE VALUES
$INCLUDE INDX_NUM
$INCLUDE INDX_VALUES
C
C*********************************************************************
C*********************************************************************
C  EXECUTABLE CODE
C*********************************************************************
C
      VALID = .FALSE.
C
      IF (SITE .EQ. XXSITE .AND. NINT(XLAT) .EQ. NINT(TLAT) .AND.
     +  ISTAM .EQ. IBEGHRA .AND. IENDAM .EQ. IENDHRA .AND. ISTPM .EQ.
     +  IBEGHRP .AND. IENDPM .EQ. IENDHRP .AND. NINT(XLONG) .EQ.
     +  NINT(TLONG) .AND. ANTENA_ID .EQ. ANTNAME .AND. IVALID .EQ.
     +  NVALID .AND. NINT(XBEAR) .EQ. NINT(ZBEAR) .AND. LANG .EQ. NLANG
     +  .AND. BAZIM .EQ. BAZM .AND. EAZIM .EQ. EAZM .AND. BDIST .EQ.
     +  BDST .AND. EDIST .EQ. EDST .AND. NUM_HRS  .EQ. N_HRS) THEN
C
C  Data base name is VALID
        VALID = .TRUE.
      ENDIF
C
      RETURN
      END
      SUBROUTINE VAL_HOUR (IHOUR,NEW_LOC,IHRS,NHRS)
C
      INTEGER IHRS(24)
C*********************************************************************
C
C                      MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                FILE NAME:
```

```
C  LOAD MODULE:                          HID:
C  CPU TYPE:  HP1000                     LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C**********************************************************************
C
C                             PURPOSE
C
C  This routine determines where the hours run with IONCAP are stored in the
C  data base.
C
C**********************************************************************
C
C                        TECHNICAL INFORMATION
C
C  See purpose.
C
C**********************************************************************
C
C                          SUBPROGRAM USAGE
C
C**********************************************************************
C
C                           CHANGE HISTORY
C
C**********************************************************************
C
C                          MNEMONIC LISTING
C
C  PARAMETERS:
C    ISTRT - starting hour of hour loop
C    IEND - ending hour of hour loop
C    LOOPBEG - location to begin hour loop in data base
C    LOOPEND - location to end hour loop in data base
C
C  COMMON BLOCKS:
C
C**********************************************************************
C**********************************************************************
C  EXECUTABLE CODE
C**********************************************************************
C
C  Determine number of hours to be calculated
      DO 10 I = 1,NHRS
        IF (IHOUR .EQ. IHRS(I)) THEN
          NEW_LOC = I
        ENDIF
10    CONTINUE
      RETURN
      END
      SUBROUTINE HOURS (ISTAM, IENDAM, ISTPM, IENDPM, IHRS, NUMHRS)
C
      INTEGER IHRS(24)
C
```

```
C*********************************************************************
C
C                     MANAGEMENT INFORMATION
C
C  SYSTEM NAME:  VALDPR                 FILE NAME:
C  LOAD MODULE:  HOURS                  HID:
C  CPU TYPE:  HP1000                    LANGUAGE:  FORTRAN 77
C  COMPILER VERSION:  FORTRAN 77
C
C*********************************************************************
C
C                            PURPOSE
C
C  This subroutine determines which hours were calculated.
C
C*********************************************************************
C
C                     TECHNICAL INFORMATION
C
C  See purpose.
C
C*********************************************************************
C
C                        SUBPROGRAM USAGE
C
C*********************************************************************
C
C                         CHANGE HISTORY
C
C*********************************************************************
C
C                        MNEMONIC LISTING
C
C  PARAMETERS:
C    ISTAM - am starting hour
C    IENDAM - am ending hour
C    ISTPM - pm starting hour
C    IENDPM - pm ending hour
C    IHRS - hours calculated
C    NUMHRS - number of hours calculated
C
C  COMMON BLOCKS:
C
C*********************************************************************
C*********************************************************************
C  EXECUTABLE CODE
C*********************************************************************
C
C  Determine which hours were calculated and save in IHRS
        ITEMPAM = ISTAM
        ITEMPPM = ISTPM
        NUMHRAM = IENDAM - ISTAM + 1
        NUMHRPM = IENDPM - ISTPM + 1
        NUMHRS = NUMHRAM + NUMHRPM
```

```
      IHRS(1) = ISTAM
      DO 15 I = 2,NUMHRAM
      ITEMPAM = ITEMPAM + 1
      IHRS(I) = ITEMPAM
15    CONTINUE
      IHRS(NUMHRAM+1) = ISTPM
      DO 20 I = NUMHRAM+2,NUMHRPM+NUMHRAM
        ITEMPPM = ITEMPPM + 1
        IHRS(I) = ITEMPPM
20    CONTINUE
      RETURN
      END
```

APPENDIX D. EXAMPLE DIALOG BETWEEN PROGRAM VALSUM AND A USER

```
                    Institute for Telecommunication Sciences

           Ionospheric Communications Analysis and Prediction Program
                                     VALSUM
                                Voice of America
                                  Version 1.0

                           Wed 04 Mar 1987   13:13:49


Data set name (must be 6 alpha characters)
    Entering an existing data set name will cause the program to use the data set
    for default values and to replace it with new data.
    Entering a new data name will cause the program to use program defined defaults
and to create  a new data set
Data set name ? TESTS2


Choose from the menu:
    H = Help
    D = Program Description
    C = Concise Dialog
    V = Verbose Dialog
    E = Edit Data
    S = Summary of Data
    P = Process Last Data Set Entered
    Q = Quit


Menu (Verbose)?

DATA BASE PARAMETERS

Transmitter site name
    TNGR = Tangier
    UDRN = Udorn
    PTRC = Puerto Rico
    CRSN = Sri Lanka
    BTSW = Botswana
 1) Transmitter site name (MIDB)?

Language area used for IONCAP execution
The first four characters of the language are entered; for example,
    POLI = Polish
    ENGL = English
 2) Language area (UKR1)?
```

Ant Name	Valid Level	Boresight Bearing (deg EofN)	Oper Freq (MHz)	Design Freq (MHz)	Lowest Elem Ht (m)	Bays	Excite Mode	Horiz Slew (deg)
1) SRMODE2	1	352.0	7.20	8.75	17.13	4	4	0
			11.85	8.75	17.13	4	1	0
			15.35	19.00	7.89	4	4	0
			21.65	19.00	7.89	4	4	0
			6.07	8.75	17.13	4	10	0
			9.70	8.75	17.13	4	4	0
			13.70	19.00	7.89	4	10	0
			17.72	19.00	7.89	4	4	0
			25.88	19.00	7.89	4	1	0

Transmitter antenna number is to be chosen from the list of available IONCAP data
3) Transmitter antenna (1)?

Circuit selection option
L = Broadcast language: all circuits of a particular language are chosen from the TEST POINT data base
S = Sector: The points contained within a specified range of azimuth and distance are chosen
4) Circuit selection option (Language)?

Broadcast language
Language area to be studied is specified. This area may be a subset of the language area specified in IONCAP for the chosen antenna. The four character code of language is entered. Any points that do not have data will be displayed.
5) Language (UKR1)?

Modify TEST POINT ID list
A = Add
C = Change
D = Delete
L = List
N = No change
10) TEST POINT ID list (A)? L

TEST POINT ID(s)	Bearing (deg E of N)	Range (km)
1) 10122	355.8	1524.2
2) 10110	344.4	1672.2
3) 10136	347.9	1765.4
4) 10138	.1	1741.6
5) 10139	4.4	1792.7
6) 10137	354.6	1807.0
7) 10384	.7	1880.7
8) 10109	341.3	2046.1
9) 10074	335.0	2170.8
10) 10131	346.6	2006.0
11) 10014	335.4	2265.5
12) 10347	7.9	2044.6
13) 10135	3.8	2025.1
14) 10134	359.6	2019.5
15) 10133	355.4	2027.9

```
    16) 10132           351.3       2050.1
    17) 10346           339.9       2194.0
    18) 10379           351.9       2159.2
    19) 10380           348.1       2192.4
    20) 10381           344.6       2237.5
    21) 10128             3.4       2191.3
    22) 10098           340.7       2401.7
    23) 10127           359.6       2241.8
    24) 10378           356.0       2249.1
    25) 10125           349.3       2363.8
    26) 10124           346.3       2425.4
    27) 10126           352.9       2366.8
    28) 10121           358.0       2410.4
10) TEST POINT ID list (A)? N

Spill-over region calculations option
  Y = Yes, calculate power to specified spill-over region circuits
  N = No, no calculations
11) Spill-over calculations (No)?

Output Parameters

Culling process to be used to calculate the required power gain  Most stringent is
Cut 1; least stringent is Cut 5.  For further explanation, see paragraph 3.5 of VOA
Std 16775.01
Cut         Brief description
 1 = Meets the highest required gains with 100% area coverage
 2 = Meets 90% of the highest required gains with 100% area coverage
 3 = Meets 90% of all circuit hours with Case 1 Best Frequency
 4 = Meets 90% of all circuit hours with Case 2 Best Frequency
     (90% circuit reliability for 90% of the locations)
 5 = Meets 90% of all listener hours with Case 2 Best Frequency and
    population weights
 A = All cuts are calculated
19) Cut number to be used to determine required power gain (4)? 1

Time blocks given in output
  S = Single table for all time blocks combined
  M = Multiple tables for each time block and all time blocks combined
20) Output time blocks (Single table)? M

Modify time blocks list
    A = Add
    C = Change
    D = Delete
    L = List
    N = No change
21) Time blocks list (A)? L
    Time blocks(s)
     1)  1  5
     2) 14 21
21) Time blocks list (A)? N

Modify HF frequencies list
```

```
    A = Add
    D = Delete
    L = List
    N = No change
22) Transmitter frequency (N)? L
    Frequencies
     1)  7.20
     2) 11.85
     3) 15.35
     4) 21.65
     5)  6.07
     6)  9.70
     7) 13.70
     8) 17.72
     9) 25.88
22) Transmitter frequency (N)? D
    Frequency? 5
22) Transmitter frequency (N)?

Number of transmitter frequencies to be considered per time block
    1 = Single frequency per time block (only the best frequency is printed)
    2 = Two frequencies per time block (the best and second best frequencies are
        printed)
23) Number of frequencies per time block (1)? 2

Calculate and output supplemental plots and tables
24) Output supplemental plots and tables (No)? Y

Identification to be printed on output tables and plots
  (up to 20 characters)
25) VALSUM identification (NEW_VALSUM_TEST)?

Modify sunspot number list
    A = Add
    D = Delete
    L = List
    N = No change
26) Sunspot number (N)? L
    Sunspot number(s)
     1) 10.00
     2) 120.0
26) Sunspot number (N)? N

Modify month list
    A = Add
    D = Delete
    L = List
    N = No change
27) Month number (N)? L
    Month(s)
     1) January
     2) April
     3) July
     4) October
```

27) Month number (N)? N

Do you want a summary of the input data (Y or N)? Y

VALSUM SUMMARY

DATA BASE PARAMETERS

1) Transmitter site name:	MIDB
2) Language area:	UKR1
3) Transmitter antenna:	1
4) Circuit selection option:	Language
5) Language:	UKR1

10) TEST POINT ID list:

TEST POINT ID(s)	Bearing (deg E of N)	Range (km)
1) 10122	355.8	1524.2
2) 10110	344.4	1672.2
3) 10136	347.9	1765.4
4) 10138	.1	1741.6
5) 10139	4.4	1792.7
6) 10137	354.6	1807.0
7) 10384	.7	1880.7
8) 10109	341.3	2046.1
9) 10074	335.0	2170.8
10) 10131	346.6	2006.0
11) 10014	335.4	2265.5
12) 10347	7.9	2044.6
13) 10135	3.8	2025.1
14) 10134	359.6	2019.5
15) 10133	355.4	2027.9
16) 10132	351.3	2050.1
17) 10346	339.9	2194.0
18) 10379	351.9	2159.2
19) 10380	348.1	2192.4
20) 10381	344.6	2237.5
21) 10128	3.4	2191.3
22) 10098	340.7	2401.7
23) 10127	359.6	2241.8
24) 10378	356.0	2249.1
25) 10125	349.3	2363.8
26) 10124	346.3	2425.4
27) 10126	352.9	2366.8
28) 10121	358.0	2410.4

11) Spill-over calculations: No

Output Parameters

19) Cut number to be used to determine required power gain: 1

20) Output hours: Multiple table

21) Time blocks(s):

1) 1 5

2) 14 21

22) Transmitter frequency:

Frequency Band(s):

```
    1)  7.20
    2) 11.85
    3) 15.35
    4) 21.65
    5)  6.07     deleted
    6)  9.70
    7) 13.70
    8) 17.72
    9) 25.88
23) Number of frequencies per time block:                  2
24) Output supplemental plots and tables:                 Yes
25) VALSUM identification:                             NEW_VALSUM_TEST
26) Sunspot number(s):
    1) 10.00
    2) 120.0
27) Month number:
    Month(s):
    1) January
    2) April
    3) July
    4) October

Do you want to process this data  (Y or N)? N
```

FORM **NTIA-29**
(4-80)

U.S. DEPARTMENT OF COMMERCE
NAT'L. TELECOMMUNICATIONS AND INFORMATION ADMINISTRATION

BIBLIOGRAPHIC DATA SHEET

	1. PUBLICATION NO.	2. Gov't Accession No.	3. Recipient's Accession No.
	NTIA Report 87-220		

4. TITLE AND SUBTITLE User's Guide for the HF Broadcast Antenna Design and Validation Summary Programs	5. Publication Date July 1987 6. Performing Organization Code NTIA/ITS.S4
7. AUTHOR(S) Eldon J. Haakinson, Susan L. Rothschild, John R. Godwin 8. PERFORMING ORGANIZATION NAME AND ADDRESS National Telecommunications and Information Admin. Institute for Telecommunication Sciences 325 Broadway Boulder, CO 80303	9. Project/Task/Work Unit No. 9104473 10. Contract/Grant No.
11. Sponsoring Organization Name and Address Voice of America 330 Independence Ave, S.W. Washington, D.C. 20547	12. Type of Report and Period Covered 13.

14. SUPPLEMENTARY NOTES

15. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here.)*

A series of programs have been developed by the Institute for Telecommunication Sciences (ITS) that assist the Voice of America (VOA) to plan and operate high frequency (HF) broadcast stations. The VOA has specific bands available for use and has particular reception areas to cover. Because of the long distance between the broadcast transmitter and the reception areas, the primary mode of communication is via HF sky wave. Since the ionosphere, which supports HF sky wave, varies with time of day, season of the year, frequency, and sunspot number, the HF broadcast antenna design and validation summary programs must calculate the best frequency and the required antenna power gain and pattern for the various conditions that apply to a particular broadcast scenario. This document will guide the user through the use of the programs, provide samples of the input to each program, and give examples of each program's output.

16. Key Words *(Alphabetical order, separated by semicolons)*

curtain array antennas; HF broadcast; ionospheric propagation

17. AVAILABILITY STATEMENT	18. Security Class. *(This report)*	20. Number of pages
☒ UNLIMITED.	unclassified	109
☐ FOR OFFICIAL DISTRIBUTION.	19. Security Class. *(This page)* unclassified	21. Price:

☆U.S. Government Printing Office: 1980—678-495/529