CRES — REMOTE SENSING LABORATORY

AAFE RADSCAT DATA REDUCTION PROGRAMS USER'S GUIDE

CRES Technical Report 186-9

(NASA-CR-144992) AAFE RADSCAT DATA REDUCTION PROGRAMS USER'S GUIDE (Kansas Univ. Center for Research, Inc.) 146 p HC $6.00 CSCL 04B

N76-22643

Unclas G3/43 27389

John P. Claassen

Supported by:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Langley Research Center
Hampton, Virginia

CONTRACT NAS 1-10048

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.
2385 Irving Hill Rd.— Campus West Lawrence, Kansas 66044

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-144992		

4. Title and Subtitle	5. Report Date
AAFE RADSCAT Data Reduction Programs User's Guide	
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
John P. Claassen	Technical Report 186-9
	10. Work Unit No.
9. Performing Organization Name and Address University of Kansas Center for Research, Inc. Space Technology Center, 2291 Irving Hill Road - Campus West Lawrence, Kansas 66044	11. Contract or Grant No. NAS 1-10048
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Langley Research Center Hampton, Virginia 23365	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The theory, design and operation of the computer programs which automate the reduction of joint radiometer and scatterometer observations conducted by the AAFE RADSCAT instrument is presented. The programs reduce scatterometer measurements to the normalized scattering coefficient; whereas the radiometer measurements are converted into antenna temperatures. The programs are both investigator and user oriented. Supplementary parameters are provided to aid in the interpretation of the observations. A hierarchy of diagnostics is available to evalute the operation of the instrument, the conduct of the experiments and the quality of the records.

General descriptions of the programs and their data products are also presented. This document therefore serves as a user's guide to the programs and is therefore intended to serve both the experimenter and the program operator.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Data Reduction Radiometer Measurements Scatterometer Measurements Remote Sensing Earth Resources	Unclassified-Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
UNCLASSIFIED	UNCLASSIFIED	144	

*For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151

THE UNIVERSITY OF KANSAS SPACE TECHNOLOGY LABORATORIES
2291 Irving Hill Dr.— Campus West Lawrence, Kansas 66044
Telephone:

AAFE RADSCAT DATA REDUCTION PROGRAMS

USER'S GUIDE

CRES Technical Report 186-9

John P. Claassen

Supported by:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Langley Research Center
Hampton, Virginia
CONTRACT NAS 1-10048

FOREWORD

The value of microwave scatterometers and radiometers as remote sea wind sensors has been independently demonstrated by a number of investigators. However, near-simultaneous observations by a composite radiometer and scatterometer (RADSCAT) instrument have been proposed as a method of making improved wind estimates. The improvement is derived from the complementary as well as supplementary features of the sensors. To demonstrate this potential a joint effort among New York University, General Electric Space Division, the University of Kansas and NASA Langley Research Center was undertaken through the Advanced Application Flight Experiment program of NASA. This document is submitted in support of these efforts.

Specifically, this document was prepared by the Remote Sensing Laboratory of the University of Kansas Center For Research, Inc. under NASA Contract NAS 1-10048. The principal investigator under the contract is Dr. R. K. Moore and project engineer is Dr. A. K. Fung. Several individuals rendered valuable assistance in the development of the computer programs. The control and integration of the routines were partially achieved by Glen E. Elliott of the University of Kansas. John L. Mitchell of LTV operated the programs repeatedly on the CDC-6600 while they were under final scrutiny by the author.

ABSTRACT

The theory, design and operation of the computer programs which automate the reduction of joint radiometer and scatterometer observations conducted by the AAFE RADSCAT instrument is presented. The programs reduce scatterometer measurements to the normalized scattering coefficient; whereas the radiometer measurements are converted into antenna temperatures. The programs are both investigator and user oriented. Supplementary parameters are provided to aid in the interpretation of the observations. A hierarchy of diagnostics is available to evaluate the operation of the instrument, the conduct of the experiments and the quality of the records.

General description of the programs and their data products are also presented. This document therefore serves as a user's guide to the programs and is therefore intended to serve both the experimenter and the program operator.

TABLE OF CONTENTS

TABLE OF CONTENTS (CONTINUED)

TABLE OF CONTENTS (CONTINUED)

LIST OF FIGURES

LIST OF FIGURES (CONTINUED)

LIST OF TABLES

I. INTRODUCTION

The AAFE RADSCAT* Data Reduction Package, as prepared by the University of Kansas, is a family of computer programs** which together form a basis for automated reduction and presentation of scatterometer and radiometer measurements taken by the RADSCAT instrument. The primary member in the family validates the raw RADSCAT data records and uses instrument and aircraft parameters to yield emission and scattering characteristics of the scene represented by the data. The scatterometer data is reduced to the normalized scattering coefficient whereas the radiometer data is reduced to the antenna temperature. Other programs in the family are employed to maintain a file of instrument characteristics, list the output (reduced data), and prepare duplicate output tapes for use on other computing machines. Two engineering routines are also provided to compute certain antenna parameters essential to the reduction of the microwave data; these routines are not employed regularly. The programs were written in FORTRAN IV, developed on a HW 635 system and then modified and demonstrated on a CDC-6600.

The program outputs provide a comprehensive basis from which to interpret the results of RADSCAT experiments. The basic output parameters have been augmented with others to reconstruct the influence of the aircraft attitude on the experiment, to validate the measurements, and to reflect the status of the RADSCAT instrument. Full evaluation of the data is accomplished by examining contents of an activity report as well as the output records. A fuller data base from which to analyze RADSCAT data is only provided when these outputs are augmented with sea/ground truth and certain aircraft parameters available from other sources. The programs are therefore largely experimenter oriented.

Attempts have been made to generalize the reduction algorithms so as to easily adapt the programs for changes in the character or operation of the instrument. In doing so the user is expected to provide many more of the calibration and

---

*A composite radiometer and scatterometer sensor.

**The family of programs does not include a merge and edit routine which prepares the raw input data. This routine was developed by the Flight Instrumentation Division of NASA Langley Research Center.

data parameters than would otherwise by required. For this reason a historical file is maintained on the instrument parameters. It is also provided to trace the development, modifications and aging of the instrument.

No attempts have been made to develop efficient coding; rather extensive efforts have been made to develop programs which will, to some degree, document themselves. Self documentation is provided by a liberal insertion of comment cards. This approach had been instituted with the full realization that the programs are likely to undergo changes as experience or changes in the instrument dictate them. It is hopeful that this approach together with this document will ease the insertion of program modifications. However, the authors can assume no responsibility for changes made without their knowledge. It is strongly recommended that changes to the reduction algorithms be made only with the approval of an engineer who is thoroughly familiar with the operation of the instrument, the theory behind the algorithms and the needs of the investigators.

Some strides have been made to design production type programs. External directives give the users the ability to select the files to be processed or the service function to be executed. Unrecognized directives will halt the execution of the program. Data entries are also validated and when too many bad files are encountered the job is aborted. Activity reports and termination statements are provided so that the user can trace the progress of the job.

The details on the use, theory and design of these programs are presented in the remainder of this document. To place these programs in context with RADSCAT experiments the technical background is briefly described in Section II. The experimenter and programmer will find Section III helpful in understanding the general features and data products of the programs. He may wish to refer to Section IV and the Appendices for more specific information. The program operator will find Section IV essential to the operation of the programs. Again he may wish to consult the appendices for specific help. The cognizant engineer will want to become familiar with all aspects of this document to develop a thorough understanding of the program content. To a large extent this document assumes some knowledge of the operation of the RADSCAT instrument*. The programmer and cognizant

*A description of the instrument may be found in "Field Service Procedure Handbook," vol. 1, prepared by General Electric Space Division, Philadelphia, Pennsylvania under NASA Contract NAS 1-10161.

engineer will also want to consult the document which describes the merge and edit routine which prepares the raw data records. This document is in preparation at NASA Langley Research Center. Those wishing to learn the details on the theory and design of the various routines in the family of programs will find the appendices helpful.

II. TECHNICAL BACKGROUND

A. Introduction

This section is primarily written for the investigator or programmer not familiar with scatterometric or radiometric measurements. The motivation for joint measurements is presented and the elementary theory behind the measurements is discussed. The instrument and its relation to the C-130 aircraft is also briefly described. It is hoped that this section will place the AAFE RADSCAT Data Reduction Programs in context of experiments.

B. Motivation for Joint Measurements

In recent years a strong interest has developed in interpreting the microwave signatures of natural surfaces. In these efforts a number of potential earth resources and scientific applications have been demonstrated by measuring the emission or scattering properties of various scenes. The scatterometer is employed to measure the scattering characteristic whereas the radiometer is used to measure the emission properties. To date the investigations have been largely conducted with one instrument or the other. However, several investigators have proposed that joint scattering and emission measurements would form a stronger interpretational basis, because each sensor measures a different aspect of the bi-static scattering characteristic of the surface. From an entirely different viewpoint the sensors are complementary in nature. It is known that the radiometer is very sensitive to clouds and rain whereas the scatterometer is relatively insensitive to clouds and rain. Over the ocean this property has potential in correcting the scatterometer measurements for small attenuations caused by clouds.

C. Description of the RADSCAT Sensor

The AAFE RADSCAT instrument is a composite sensor consisting of a pencil-beam microwave scatterometer and radiometer. It was designed to conduct measurements from an aircraft and is capable of operating at either of two frequencies, 9.3 GHz or 13.9 GHz. In three of its four modes of operation it makes scatterometer and radiometer measurements on a near-simultaneous basis. In the fourth mode it makes radiometer measurements only. Observations can be conducted at any one of six adjustable angles covering the sector between nadir and 60 degrees.

The AAFE RADSCAT instrument was primarily designed to perform experiments over the ocean to verify the remote anemometric capability of the composite sensor. Although primarily designed for oceanographic work, the extension to observations of agrarian or urban scenes is straightforward. The instrument is capable of conducting experiments at one of four altitudes, viz, 2000, 5000, 10,000 and 20,000 feet.* The wide range of altitudes is provided to study atmospheric effects such as humidity, temperature, clouds and rains. To avoid difficulties introduced by operating from behind a radome, the RADSCAT instrument mounts on the tail gate of the NASA/MSC C-130 aircraft. The Litton Navigator LTN-51 aboard the C-130 aircraft is extremely useful to the interpretation and reduction of the RADSCAT observations. It provides aircraft attitude and location information from which the point of observation on the surface may be established. The attitude parameters are especially useful when measurements are performed from a roll maneuver, since these parameters enable one to compute the incident angle. The radar altimeter information provided by the aircraft is essential to the inversion of the scatterometer measurements.

D. Estimating the Scattering and Emission Properties from Measurements

The scatterometer is a radar which is capable of accurately measuring the backscatter properties of a rough surface. The backscatter ability of the surface is denoted by the normalized scattering coefficient σ^o which is given in terms of the scattering cross-section per unit area. In general it is dependent upon the point of observation, the incident angle, frequency, polarization, etc. It is well known that the radar return is predicted by (see Figure 1).

$$P_r = \frac{\lambda^2}{(4\pi)^3} P_t \iint_{4\pi} \frac{g^2(\theta,\phi)\ \sigma^o(\theta',\phi')}{R^4}\, dA \qquad (1)$$

where σ^o = normalized scattering coefficient

P_r = received power

P_t = transmitted power

λ = wavelength

g = antenna gain function

R = radar range to elemental area dA

*The radiometer by itself can operate at any altitude.

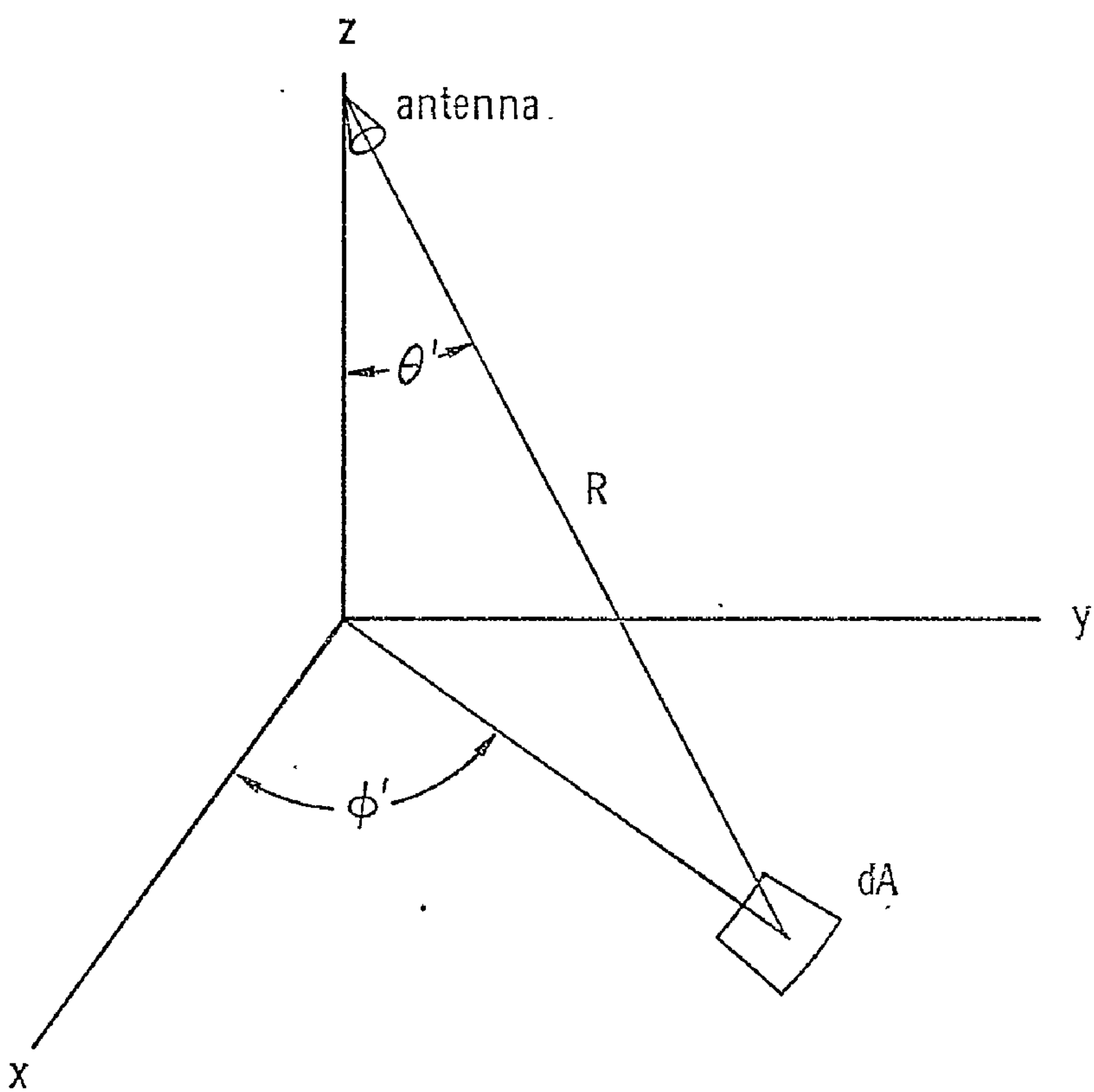


FIGURE 1. RADAR GEOMETRY

For narrow (or equivalently narrow) beam scatterometers the above integral expression may be inverted rather simply for σ^o. As a consequence the problem of accurately determining σ^o reduces to the problem of accurately measuring the transmitted and receiver powers, the pattern function, look angle, and radar range. The area of illumination may be shown to be related to the pattern function, look angle, and radar range. The scatterometer provides outputs which are measures of P_r and P_t. The power transfer function of scatterometer completes the association between the outputs to yield P_r and P_t*. A member of the family of programs actually employs the scatterometer transfer function and an algebraic approximation of equation (1) to estimate σ^o.

The radiometer, on the other hand, is a very sensitive receiver capable of measuring very low level emissions which emanate naturally from objects. In the microwave region the natural emission is governed by the Jeans-Rayleigh law which states that the power emitted is proportional to the physical temperature T of the body. In actuality, however, different objects at different temperatures emit different powers depending until their relative ability to absorb energy. Better absorbers emit more efficiently than poor ones. The relative ability to emit is described in terms of a parameter called emissivity ϵ. The emitted power per unit bandwidth in direction θ is given by**

$$P(\theta) = \frac{4\pi k \epsilon(\theta) T}{\lambda^2} \tag{2}$$

where k is Boltzmann's constant.

We may now suppose that the surface emits in accord with this relation. Then a radiometer whose antenna gain function is given by $g(\theta,\phi)$ will measure (ignoring atmospheric contributions) a total power per unit bandwidth given by***

$$P_a = \frac{1}{\Omega} \iint_{4\pi} g(\theta,\phi)\, k\, T_B(\theta)\, d\Omega \tag{3}$$

*Actually only the ratio P_r/P_t is needed.

**For one polarization.

***This relation is only representative of the nature of the problem. See Claassen, J. P. and A. K. Fung, "An Efficient Method for Inverting Antenna Temperatures for the Apparent Temperature Distribution" University of Kansas Center For Research, Inc., TR 186-8, January 1973, for a precise relationship.

where

$$\Omega = \iint_{4\pi} g(\theta, \phi)\, d\Omega \qquad (4)$$

and

$$T_B = \epsilon(\theta)\, T \qquad (5)$$

The antenna power may be related to antenna temperature by $P_a = kT_a$. For narrow beam efficient antennas the antenna temperature is an approximate measure of the brightness temperature T_B in the look direction. The radiometer therefore must accurately measure the antenna power. The RADSCAT data reduction programs use the calibration data and radiometer transfer function to estimate T_a for each polarization.

III. GENERAL DESCRIPTION

A. Introduction

This section is primarily written to familiarize the investigator or programmer with the family of RADSCAT data reduction programs. The functions of the particular programs and the relationships between the programs are presented. Input and output data products are described to various degrees.

B. Summary of Programs

The AAFE RADSCAT Data Reduction Package consists of a family of six programs:

1. Conversion Program (CONVERT)
2. Output Program (OUTPUT)
3. Program for the Storage and Retrieval of Instrument Characteristics (ICHAR)
4. Output Translation and Duplication Program (TDUPE)
5. Equivalent Beamwidth Program (WIDTH)
6. Antenna Gain and Efficiency Program (GAIN)

The actual reduction of data is accomplished in CONVERT. The input data for CONVERT is prepared by an edit and merge routine developed by NASA Langley Research Center. The merged tape consists of RADSCAT and aircraft data. To assist in the computations, program ICHAR prepares a historical file of instrument characteristics. CONVERT selects a particular set of characteristics from the file and applies them to the input data. The computed results are stored on an output tape. Program OUTPUT will list the contents of the output tape. To prepare duplicate output tapes for use on other computing machines program TDUPE may be employed.

WIDTH and GAIN are special routines which compute several antenna parameters. The parameters are employed as elements of the instrument characteristics. Both programs require antenna pattern information to compute the parameters. Their use will seldom be required unless the antenna pattern changes.

Of the six programs the initial four programs are production types and the last two are engineering application programs.

C. Conversion Program

The reduction of RADSCAT data is performed entirely in CONVERT. By means of external directives, physical files of a specified type are processed with a designated set of instrument characteristics withdrawn from a separate file source. External directives permit the user to initiate processing of input files by type or sequentially regardless of type. The files are organized by flight runs and are labeled in accord with a procedure instituted by the Flight Operations Division of NASA/MSC.

Directive options are given by MISFLT, FLTLIN, SPECIF, or ALL. When the first directive is appended with mission and flight numbers, all files from that mission and flight are processed in the order that they appear on the tape. Mission, flight and line numbers must appear with the second directive option. Files of the designated line type will thus be processed. Consecutive directives which exhaust the line types will order the output files by line type. This feature is helpful to the interpreter who often associates an experiment condition with a line type. The third directive is employed when a specific file designated by mission, flight, line and run is to be processed. This directive is appropriate for a quick or selective look at the experiment results. The last directive (ALL) will initiate processing of all files regardless of type. Combinations of directives may be employed; however, indiscriminant use of the processing types is cautioned against since duplicate processing of files may occur. For each directive the file search is effectively conducted from the beginning of the tape.

Each of the above directives must include a designator which selects a set of instrument characteristics from a historical file of characteristics or which permits them to be read from cards. It is likely that the most current characteristics will be chosen; however, any set on the file may be selected. Alternatively, characteristics may be entered from the card reader. However, operation under this latter option is restricted to one data file per directive. Entry through the card reader is mainly reserved for testing a set of instrument characteristics before storage on the historical file. Also as a part of the same directive the number of files to be processed of the indicated type and the line density of the activity report must be designated on the control card. The latter parameter can be used to limit the number of data records on which reports are given.

During processing CONVERT generates a conversion report. The report routinely issues statements on the calibration parameters and critical receiver temperatures as they are computed from the input file. It will also routinely state that a record had been reduced. The reports are appended with record numbers. Other special statements are generated whenever critical situations are encountered in the records. When these special situations occur, the record is normally processed regardless; however, that record is flagged by an interpretational code which becomes a part of the output record. In some cases, such as invalid data, the record is bypassed and an appropriate error message is generated. When an excessive number of bad records is encountered, the file is bypassed and an appropriate message printed. These messages are primarily intended to reflect the status of the processing, the condition of the instrument, the conduct of the experiment, and the quality of the data. A sample listing of the conversion report is illustrated in Figure 2.

As processing progresses output records are accumulated in blocks (arrays) and the blocks are written to the output tape. A description of the contents of the output tape appears in the following subsection.

D. Output Program

Program OUTPUT will list the processed records produced in the conversion program. External directives similar to those in CONVERT permit listing files of designated types or all files regardless of type. Again general directives are given by MISFLT, FLTLIN, SPECIF or ALL. The appropriate mission, flight, line and run designators must accompany the directives.

A sample listing produced by OUTPUT is shown in Figure 3. A double page of output consists of the file label, column headings, 80 records and a key. The set of instrument characteristics which produced the results is identified by set number and date in the last entry of the file label. Each measurement record occupies one line on the listing. A record number is appended to each line and should agree with the number assigned in the conversion report. The time of measurement in mean Greenwich time is presented in the second column. It is to be read in hours, minutes, seconds and 1/10th seconds. The measurement can therefore be associated with data from other sources. The mode from which the instrument conducted the measurement is reflected in column 3. The incident and cross-tracks angles appear in the next two columns. They were computed from aircraft attitude parameters and the RADSCAT view angle with respect to the aircraft platform. The cross-track angle is that angle

CONVERSION REPORT

MISSION-207 FLIGHT- 12 DATE- 7 25 1972 FLT LINE- 2 RUN- 3 FREQUENCY- 13.9 FEED- VG SW IPAR= 1 2 1 71

RECORD NO.*	MESSAGE *	PARAMETERS
	* CALIBRATION DATA *	CHAN 1= .270 CHAN 2= .410 CHAN 3= 1.190 CHAN 4= 1.285
	*	RCAL= 2.406 RBASE= .160 NCAL= 6 NBCAL= 6
1	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.8 KELVIN
1	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.1 SW= 311.3 AIR= 288.0 GDE= 312.7
1	* EXCESSIVE DEPOLARIZATION *	DEPOL= .021
1	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 246.6 SCATTERING COEF=7.720364E+00 CHANNEL= 2 DOP BAND= 7
2	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.8 KELVIN
2	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.1 SW= 311.3 AIR= 288.0 GDE= 312.7
2	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 248.6 SCATTERING COEF=6.311449E+00 CHANNEL= 2 DOP BAND= 7
3	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.8 KELVIN
3	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.1 SW= 311.3 AIR= 288.0 GDE= 312.7
3	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 248.4 SCATTERING COEF=5.924877E+00 CHANNEL= 2 DOP BAND= 7
4	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.8 KELVIN
4	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.1 SW= 311.3 AIR= 288.0 GDE= 312.7
4	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 249.8 SCATTERING COEF=7.028195E+00 CHANNEL= 2 DOP BAND= 7
5	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.8 KELVIN
5	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.1 SW= 311.3 AIR= 287.9 GDE= 312.7
5	* EXCESSIVE DEPOLARIZATION *	DEPOL= .021
5	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 249.4 SCATTERING COEF=6.206822E+00 CHANNEL= 2 DOP BAND= 7
6	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.8 KELVIN
6	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.1 SW= 311.3 AIR= 287.6 GDE= 312.7
6	* EXCESSIVE DEPOLARIZATION *	DEPOL= .022
6	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 249.1 SCATTERING COEF=5.944944E+00 CHANNEL= 2 DOP BAND= 7
7	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.8 KELVIN
7	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.2 SW= 311.3 AIR= 287.9 GDE= 312.7
7	* EXCESSIVE DEPOLARIZATION *	DEPOL= .020
7	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 248.2 SCATTERING COEF=7.507535E+00 CHANNEL= 2 DOP BAND= 7
8	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.8 KELVIN
8	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.2 SW= 311.3 AIR= 288.0 GDE= 312.7
8	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 249.0 SCATTERING COEF=7.141559E+00 CHANNEL= 2 DOP BAND= 7
9	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.8 KELVIN
9	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.1 SW= 311.3 AIR= 288.0 GDE= 312.7
9	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 249.2 SCATTERING COEF=8.071541E+00 CHANNEL= 2 DOP BAND= 7
10	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =337.6 KELVIN
10	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.1 SW= 311.3 AIR= 288.0 GDE= 312.7
10	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 248.6 SCATTERING COEF=7.185143E+00 CHANNEL= 2 DOP BAND= 7
11	* RECVR TEMP NOT WITHIN OPERATING RANGE *	LOCATION NO. 1 TEMPERATURE =338.0 KELVIN
11	* CRITICAL TEMPERATURES *	WARM= 311.8 HOT= 415.0 OMT= 312.1 SW= 311.3 AIR= 288.1 GDE= 312.9
11	** COMPLETED RECORD COMPUTATIONS **	ANTENNA TEMP= 243.5 SCATTERING COEF=5.522865E+00 CHANNEL= 2 DOP BAND= 7

FIGURE 2. SAMPLE OF THE CONVERSION REPORT

MICROWAVE DATA

MISSION-207 FLIGHT- 12 DATE- 7 25 1972 FLT LINE- 2 RUN- 3 FREQUENCY- 13.9 FEED- NG SW IPAR= 1 2 1 73

RECORD NO.	TIME	MODE	INCID ANGLE (DEG)	CROSS ANGLE (DEG)	XMIT POL	REC POL	S INT TIME (SEC)	SCAT	SCAT DB	RAD (DEG K)	DEPOL FACTOR	S/R CODE	ALTITUDE (FEET)	FLT DIRECT (DEG)
1	151746.7	F.A.	13.7	-11.4	H	H	.595	7.7204E+00	8.9	246.8	.021	110000	9966	2.5
2	151748.2	F.A.	13.8	-10.3	H	H	.595	6.3114E+00	8.0	248.6	0.000	100000	9959	2.7
3	151750.4	F.A.	13.8	-9.0	H	H	.595	5.9249E+00	7.7	248.4	0.000	100000	9957	2.6
4	151751.1	F.A.	13.7	-9.9	H	H	.595	7.0282E+00	8.5	249.8	0.000	100000	9956	2.3
5	151751.9	F.A.	13.6	-10.7	H	H	.595	6.2068E+00	7.9	249.4	.021	110000	9956	2.0
6	151752.6	F.A.	13.6	-10.8	H	H	.595	5.9449E+00	7.7	249.1	.022	110000	9959	1.9
7	151753.3	F.A.	13.5	-10.5	H	H	.595	7.5075E+00	8.8	248.2	.020	110000	9958	1.8
8	151754.1	F.A.	13.5	-9.7	H	H	.595	7.1420E+00	8.5	249.0	0.000	100000	9955	1.8
9	151754.8	F.A.	13.3	-6.9	H	H	.595	8.0719E+00	9.1	249.2	0.000	100000	9953	1.8
10	151756.3	F.A.	13.4	-8.6	H	H	.595	7.1891E+00	8.6	2 8.6	0.000	100000	9956	1.6
11	151757.8	F.A.	13.5	-9.5	V	V	.595	5.5229E+00	7.4	243.5	0.000	100000	9956	1.3
12	151758.5	F.A.	13.6	-9.3	V	V	.595	5.5486E+00	7.4	244.0	0.000	100000	9954	1.2
13	151759.2	F.A.	13.5	-8.9	V	V	.595	5.3070E+00	7.3	242.9	0.000	100000	9955	1.2
14	151800.0	F.A.	13.4	-8.2	V	V	.595	5.6805E+00	7.5	244.0	0.000	100000	9955	1.1
15	151800.7	F.A.	13.4	-7.4	V	V	.595	5.7777E+00	7.6	243.8	0.000	100000	9955	1.1
16	151802.2	F.A.	13.3	-7.2	V	V	.595	5.7537E+00	7.6	243.2	0.000	100000	9953	.9
17	151804.4	F.A.	13.3	-7.8	V	V	.595	7.1411E+00	8.5	244.4	0.000	100000	9951	.6
18	151805.1	F.A.	13.4	-7.6	V	V	.595	5.3751E+00	7.3	242.7	0.000	100000	9952	.6
19	151805.9	F.A.	13.3	-7.1	V	V	.595	7.2444E+00	8.6	243.7	0.000	100000	9956	.6
20	151806.6	F.A.	13.2	-6.7	V	V	.595	7.6415E+00	8.8	244.4	0.000	100000	9956	.6
21	151815.8	F.A.	43.0	-2.5	V	V	.595	3.2273E-01	-4.9	250.1	0.000	100000	9951	103.4
22	151816.7	F.A.	43.0	-2.8	V	V	.802	2.4405E-01	-6.1	250.1	0.000	100000	9953	359.8
23	151818.6	F.A.	42.9	-2.6	V	V	.802	2.4335E-01	-6.1	250.4	0.000	100000	9952	359.9
24	151820.5	F.A.	42.7	-2.2	V	V	.802	2.4358E-01	-6.1	249.7	0.000	100000	9952	359.8
25	151821.4	F.A.	42.8	-2.2	V	V	.802	2.4347E-01	-6.1	249.3	0.000	100000	9953	359.7
26	151822.4	F.A.	42.7	-2.4	V	V	.802	2.0886E-01	-6.8	250.5	0.000	100000	9952	359.7
27	151823.3	F.A.	42.8	-2.5	V	V	.802	2.0911E-01	-6.8	250.5	0.000	100000	9953	359.6
28	151824.3	F.A.	42.7	-2.5	V	V	.802	2.0958E-01	-6.8	250.1	0.000	100000	9957	359.6
29	151825.2	F.A.	42.8	-2.2	V	V	.802	2.0990E-01	-6.8	250.1	0.000	100000	9957	359.7
30	151827.1	F.A.	42.7	-1.9	V	V	.802	2.0854E-01	-6.8	249.3	0.000	100000	9961	359.6
31	151828.0	F.A.	42.8	-1.9	V	V	.802	2.0997E-01	-6.8	249.5	0.000	100000	9960	359.6
32	151829.9	F.A.	43.0	-1.9	V	V	.802	2.4451E-01	-6.1	249.3	0.000	100000	9963	359.4
33	151830.9	F.A.	43.0	-2.0	V	V	.802	2.4449E-01	-6.1	249.3	0.000	100000	9963	359.4
34	151831.8	F.A.	43.0	-1.9	V	V	.802	2.4402E-01	-6.1	250.1	0.000	100000	9961	359.4
35	151832.8	F.A.	42.9	-1.8	V	V	.802	2.4336E-01	-6.1	250.5	0.000	100000	9962	359.5
36	151833.7	F.A.	42.9	-1.6	V	V	.802	2.4434E-01	-6.1	249.7	0.000	100000	9958	359.5
37	151835.6	F.A.	42.8	-1.2	H	H	.802	8.2189E-04	-30.9	247.0	0.000	100000	9959	359.4
38	151836.5	F.A.	42.8	-1.2	H	H	.802	7.9444E-04	-31.0	246.0	0.000	100000	9957	359.3
39	151837.5	F.A.	42.8	-1.3	H	H	.802	7.4506E-04	-31.3	246.8	0.000	100000	9960	359.3
40	151838.4	F.A.	42.8	-1.3	H	H	.802	6.5443E-04	-31.8	246.2	0.000	100000	9960	359.3
41	151839.4	F.A.	42.8	-1.4	H	H	.802	7.7676E-04	-31.1	246.4	0.000	100000	9960	359.4
42	151840.3	F.A.	42.7	-1.3	H	H	.802	7.2953E-04	-31.4	245.0	0.000	100000	9960	359.5
43	151842.2	F.A.	42.9	-1.0	H	H	.802	8.5995E-04	-30.7	246.0	0.000	100000	9960	359.4
44	151844.1	F.A.	42.9	-1.1	H	H	.802	6.8271E-04	-31.7	247.0	0.000	100000	9960	359.4
45	151846.0	F.A.	42.9	-1.3	H	H	.802	6.8509E-04	-31.6	245.2	0.000	100000	9963	359.4
46	151847.9	F.A.	42.9	-1.1	H	H	.802	7.5142E-04	-31.2	247.2	0.000	100000	9966	359.6
47	151848.8	F.A.	43.0	-1.0	H	H	.802	7.4572E-04	-31.3	247.8	0.000	100000	9964	359.5
48	151857.8	F.A.	13.3	-2.2	H	H	.595	5.6747E+00	7.5	247.8	0.000	100000	9960	359.7
49	151858.5	F.A.	13.1	-2.3	H	H	.595	5.5253E+00	7.4	248.3	0.000	100000	9960	359.7
50	151859.3	F.A.	13.1	-2.1	H	H	.595	7.4056E+00	8.7	249.1	0.000	100000	9962	359.8
51	151900.7	F.A.	13.2	-1.5	H	H	.595	6.7056E+00	8.3	248.9	0.000	100000	9959	359.8
52	151901.5	F.A.	13.3	-1.1	H	H	.595	5.6748E+00	7.5	249.7	0.000	100000	9961	359.9
53	151902.2	F.A.	13.3	-1.0	H	H	.595	6.4946E+00	8.1	248.9	0.000	100000	9961	359.9
54	151903.7	F.A.	13.3	-1.4	H	H	.595	5.5778E+00	7.5	248.2	0.000	100000	9960	359.8
55	151904.4	F.A.	13.2	-1.4	H	H	.595	6.0592E+00	7.8	249.5	0.000	100000	9961	359.8
56	151905.2	F.A.	13.3	-1.5	H	H	.595	6.0606E+00	7.8	249.3	0.000	100000	9961	359.9
57	151906.6	F.A.	13.2	-1.5	H	H	.595	5.4558E+00	7.4	247.8	0.000	100000	9959	360.0
58	151907.4	F.A.	13.3	-1.4	H	H	.595	5.8421E+00	7.7	248.4	0.000	100000	9960	360.0
59	151908.1	F.A.	13.2	-1.3	H	H	.595	5.4304E+00	7.3	248.4	0.000	100000	9960	213.1
60	151909.6	F.A.	13.3	-1.1	H	H	.595	6.9278E+00	8.4	248.5	0.000	100000	9960	.1
61	151910.3	F.A.	13.2	-1.2	H	H	.595	5.7202E+00	7.6	249.3	0.000	100000	9959	.1
62	151911.8	F.A.	13.2	-1.2	H	H	.595	6.7520E+00	8.3	247.6	0.000	100000	9957	.1
63	151912.5	F.A.	13.2	-1.3	V	V	.595	5.9210E+00	7.7	244.2	0.000	100000	9958	.2
64	151913.3	F.A.	13.2	-1.2	V	V	.595	4.7140E+00	6.7	244.4	0.000	100000	9956	.3
65	151914.0	F.A.	13.3	-1.2	V	V	.595	4.8794E+00	6.9	244.6	0.000	100000	9954	.3
66	151915.5	F.A.	13.3	-1.1	V	V	.595	4.9261E+00	6.9	243.4	0.000	100000	9954	.5
67	151916.2	F.A.	13.2	-1.2	V	V	.595	5.9873E+00	7.8	244.4	0.000	100000	9954	.5
68	151917.7	F.A.	13.3	-1.3	V	V	.595	4.4515E+00	6.5	245.0	0.000	100000	9951	.4
69	151918.4	F.A.	13.2	-1.3	V	V	.595	5.3231E+00	7.3	243.4	0.000	100000	9950	.4
70	151919.2	F.A.	13.2	-1.3	V	V	.595	5.7957E+00	7.6	243.6	0.000	100000	9951	.5
71	151920.6	F.A.	13.3	-1.0	V	V	.595	5.5595E+00	7.5	244.2	0.000	100000	9950	.6
72	151921.4	F.A.	13.3	-.6	V	V	.595	5.2076E+00	7.2	244.0	0.000	100000	9951	.8
73	151922.1	F.A.	13.2	-.4	V	V	.595	5.0653E+00	7.0	245.0	0.000	100000	9952	.9
74	151923.6	F.A.	13.2	-.4	V	V	.595	4.7133E+00	6.7	244.0	0.000	100000	9955	.9
75	151924.3	F.A.	13.3	-.7	V	V	.595	4.9975E+00	7.0	243.6	0.000	100000	9954	.9
76	151925.1	F.A.	13.2	-1.1	V	V	.595	6.0345E+00	7.8	244.2	0.000	100000	9954	.9
77	151925.8	F.A.	13.2	-1.4	V	V	.595	5.9154E+00	7.7	243.6	0.000	100000	9953	.9
78	151926.5	F.A.	13.2	-1.4	V	V	.595	5.1604E+00	7.1	244.2	0.000	100000	9954	1.1
79	151927.3	F.A.	13.1	-1.5	V	V	.595	5.3982E+00	7.3	243.3	0.000	100000	9958	1.1
80	151928.0	F.A.	13.1	-1.4	V	V	.595	4.6699E+00	6.7	244.2	0.000	100000	9961	1.2

KEY TO CODES-

0 NO FLAG
1 POSSIBLY OUTSIDE RANGE GATE
10 OUTSIDE DYNAMIC RANGE
100 EXCESSIVE DOPPLER
1000 POLAR REVERSAL
10000 EXCESSIVE DEPOL
100000 REC TEMP ABNORMAL
1000000 WARM LOAD TEMP ESTIMATED
COMBINATIONS OF FLAGS CAN OCCUR.

FIGURE 3. SAMPLE LISTING OF OUTPUT RECORDS

between the "negative" flight direction and the azimuthal look direction. The positive cross-track angle is defined to the right of the aircraft when looking aft. It is induced by aircraft roll and drift angle. The incident and cross-track angles are precisely computed for any aircraft attitude. The angles enable the investigator to locate the illumination area. The scatterometer transmit polarization and the scatterometer and radiometer receive polarization are presented in the next two columns, respectively. The scatterometer integration time is indicated in the following column. The radiometer integration time is not listed since it is constant (128 milliseconds) regardless of mode or angle. The normalized scattering coefficient (m^2/m^2) is shown in decimal as well as dB units. The radiometer antenna temperature is shown in column 11.

A depolarization factor appears in column 12. The depolarization factor when expressed as a percentage indicates the percent power transmitted at the opposite surface polarization. Whenever DEPOL exceeds 50 per cent a polarization reversal flag is generated under the S/R validation code (column 13) and the complementary depolarization factor (1-DEPOL) is posted in column 12. If a reversal is indicated and the (complementary) depolarization factor is small, the polarization designators in columns 6 and 7 may indeed be regarded in the opposite state. Polarization reversals will frequently occur when observations are conducted from an aircraft roll condition. When DEPOL exceeds 2 per cent a flag in the S/R code is raised.

Other flag conditions can occur under the S/R code. The flags are used as a quick indicator of an abnormal condition. The interpretations of flags are given in a key at the bottom of every double page. The range gate flag is specifically set whenever the aircraft is not within $\pm$ 300 feet of the range gate setting. On some occasions this condition may be purposely generated to check the scatterometer zeros. A dynamic range flag is set whenever the scatterometer return is above or below the dynamic capability of the scatterometer receiver. If any of the receiver temperatures are not within the proper operating range a flag is set. Whenever the receiver temperature data is missing for long periods of time (data drop out), the receiver temperatures are estimated and a flag is generated. The other keys are self-explanatory. For additional assistance in diagnosing the flagged condition the conversion report may be consulted.

The latter two columns denote aircraft radar altitude and flight direction. Other aircraft parameters are available from the listings of the aircraft mission tape.

Not listed but present on the tape is the ambient air temperature at altitude. This latter parameter is available should the investigator wish to use it to estimate the apparent temperature distribution*.

E. Program ICHAR

Program ICHAR maintains a historical tape file of RADSCAT instrument characteristics. The historical file will (eventually) consist of many sets of instrument parameters which have been entered chronologically and labeled numerically. The historical file is maintained to chart the changes in the instrument parameters as may occur through updating, instrument modifications or simply aging. The Conversion Program withdraws an appropriate set of parameters from the historical file and applies the withdrawn parameters to reduce the measurements.

ICHAR performs several useful functions. It will initially establish the file. Once the file has been established it will permit the user to append new sets of characteristics. To assist in the maintenance of the file, ICHAR may be used to print and punch old sets of characteristics. The punched sets may be modified, for example, and submitted to ICHAR as a new set to be appended to file.

F. Program TDUPE

Program TDUPE provides a master duplicate tape of the output tape formed by the Conversion Program for use on other computing machines. The bit conversion problem between machines is avoided by scaling and integerizing all the numerical data before transferring the output data to the duplicate tape. This technique has been successfully applied to tapes prepared by a CDC-6600 and read on a Honeywell 635. The CDC 6600 is a 60 bit machine which performs "ones complement" arithmetic whereas the Honeywell is a 36 bit machine which performs "twos complement" arithmetic. The scale factors may be withdrawn from the listing of TDUPE shown in Figure D-2. Each output file consists of a header label (a flight run), the instrument characteristics applied to the data, and the output data records.

---

*See for example, Claassen, J. P. and A. K. Fung, "An Efficient Technique for Estimating the Apparent Temperature Distribution," University of Kansas Center for Research, Inc., Technical Report 186-8, January 1973.

Additional integerized tapes may be prepared from the master duplicate tape by using system utility programs.

G. Special Programs

In addition to the above production type programs two engineering routines WIDTH and GAIN are included in the reduction package. Program WIDTH computes the equivalent beamwidth when it is presented with the mainbeam antenna pattern. The equivalent beamwidth parameter is essential to the inversion of the scatterometer data. The program GAIN computes the beam solid angle and directivity. The initial factor is important to the inversion of the radiometer measurements whereas the second is required in the inversion of the scatterometer measurements.

Functional representations of the antenna pattern are required by both programs.

IV. OPERATING INSTRUCTIONS

A. Introduction

General operating requirements, preparation of data input files, construction of the job deck, descriptions of the data products, flags and aborts, etc., are treated in this section. It is primarily written for the user having little or no knowledge of the theory and design of the programs. The reader will find that Section II and Section III will be helpful in understanding the contents of this section. Supplementary as well as detailed information on these programs may be found in the Appendices. Sufficient information is intended to permit the user to operate the primary programs CONVERT, OUTPUT, ICHAR and TDUPE. The remaining programs WIDTH and GAIN are engineering routines and should be handled by an engineer (or equivalent) familiar with the RADSCAT antenna. However, it is also strongly advised that a competent engineer, thoroughly familiar with the RADSCAT instrument, be involved in the preparation of the data and in the operation of the primary programs.

System control cards are not specified. It is presumed that this section together with sufficient knowledge of the operating system will dictate the necessary system control cards.

B. Conversion Program (CONVERT)

1. General

This program reduces 50 word input records to 20 word output records. Various computational algorithms described in detail in the Appendices are applied to the input records to yield data products of interest to the investigator. The input records are members of a file having an indefinite number of records. The files are elements of a multi-file tape. Each file is labeled with the entries shown in Table 1. The files are footed with an "end of file" mark. It is anticipated that files (flight runs) from several missions can be stored on a single tape. Tapes of this type are designated Raw Data Tapes.

To initiate processing of any of the files or all of them, processing directives are employed. The directives permit seeking files of certain types and must appear as data cards in the job deck. Each processing directive must be accompanied with a file referral argument to select a set of instrument parameters from the Instrument

Characteristics Tape* on which sets of instrument parameters are indexed numerically (and presumably stored chronologically). A set of these parameters is employed by CONVERT to reduce the raw data records. Alternatively, the instrument parameters can be entered from the card reader should the file index be set at zero. Files processed under a particular directive are transferred to the Output Tape. Various processing directives may be applied successively. In each instance the file search is effectively initiated from the beginning of the tape. Processing continues until all directives have been exhausted.

Other types of directives perform service functions. The directive POST permits newly processed files to be appended to an old Output Tape. This directive, when used, must precede all processing directives. The PRINT and PUNCH commands will list and punch, respectively, a set or several consecutive sets of instruments parameters from the Instrument Characteristics Tape. When the PUNCH command is employed, a listing of the set is also generated.

The composition of a CONVERT job deck is illustrated in Figure 4.

2. Preparation of the Raw Data Tape

As mentioned above each file is identified by a header label. The first seven items in the label (Table 1) are integer variables. The integer variables must be right justified. Note that the date requires three words. The sixth and seventh items are appended to the label to assist in selecting the proper subset of instrument characteristics from the set index designated on the processing directive card. See Section IV D for a description of the instrument characteristic files. The frequency is specified in GHz. The feed type is declared in the seventh word and is an alphanumeric word.

Presently two feed types are employed by the RADSCAT instrument. One feed uses a mechanical waveguide switch and the other a ferrite circulator. The abbreviations WG SW and CIRC are recommended as designators. To establish a convention, they should be left justified. The label must be created with an unformatted WRITE statement.

Beneath the header label are a sequence of raw data records each a mixture of real and integer words. The records are blocked by groups of 10 records. The blocks

*This tape is physically separate from the Raw Data Tape.

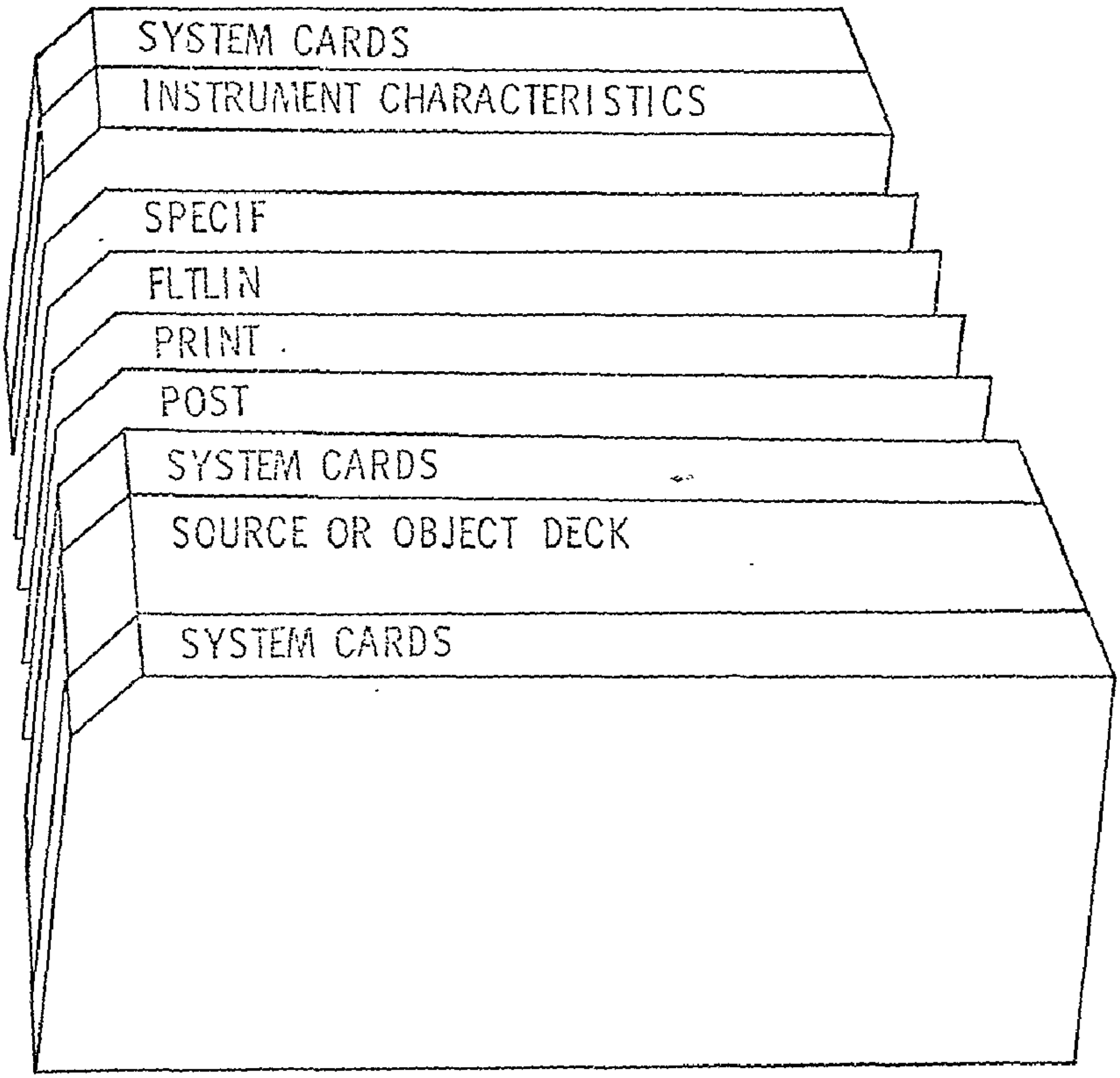


FIGURE 4. JOB DECK FOR PROGRAM CONVERT

Table 1 . File Label Entries Illustrated

Position	Name	Entry
1	Mission	207
2	Flight	11
3	Month	7
4	Day	21
5	Year	1972
6	Line	4
7	Run	9
8	Frequency	13.9
9	Feed	WG SW

must be created by unformatted WRITE statements. When insufficient records exist to fill the last block on a file, the remaining records must be filled with 99999.0 (floating point). These entries are used to flag the end of the records (file). In addition an EOF is appended to the foot of the file. As many files as practical may be created on the tape. The end of the files must be flagged with the label END OF REEL which fills the beginning of a nine word array and in effect is another file label.

The files are composed of three record types:

1. Normal Calibration
2. Baseline Calibration
3. Measurements

Calibration records may appear anywhere in the file and as frequently as desired. In a properly constructed file both types of calibrations should appear in the first records. However, should one (or both) type(s) of calibration be missing from the top of the file, the program will search for calibration records deeper in the file. Records bypassed in the search are transferred to a scratch unit for subsequent processing. Once found, the calibration information is extracted and applied to the bypassed records. Records following the calibration records are also treated with the same parameters until new calibration records are encountered. When the calibration information is totally missing or incomplete, the file is bypassed and an appropriate error message is printed.

The entries in the records are shown in Table II. Type and units of the variables are indicated. Parenthetical designators behind some of the entry names describe the computed results stored in those positions during processing. The integer

TABLE II RAW DATA VECTOR

NAME : DATA

POSITION	ENTRY	TYPE	UNITS
1	TIME	R	HR MIN SEC
2	FREQUENCY INDICATOR	I	—
3	MODE INDICATOR	I	—
4	DATA TYPE INDICATOR	I	—
5	ANGLE DESIGNATOR	I	—
6	TRANS POL INDICATOR	I	—
7	REC POL INDICATOR	I	—
8	SCAT 1 (OR CONVERTED SCAT)	R	VOLTS
9	SCAT 2	"	"
10	SCAT 3	"	"
11	SCAT 4	"	"
12	RAD POL INDICATOR	I	—
13	RAD (OR CONVERTED RAD)	R	VOLTS
14	SCAT ANGLE (OR INCID ANGLE)	R	"
15	RAD ANGLE (OR X-TRACK ANGLE)	R	"
16	RANGE GATE INDICATOR	I	—
17	T_1, 9.3 DICKE SWITCH	R	VOLTS
18	T_2, TRIPLEXER	R	"
19	T_3, T/R CIRCULATOR	R	"
20	T_4, 13.9 DICKE SWITCH	R	"
21	T_5, 9.3 TDA	R	"
22	T_6, 9.3 WARM LOAD	R	"
23	T_7, 9.3 LIMITER	R	"
24	T_8, BRACKET	R	"
25	T_9, COMMON T/R GUIDE	R	"
26	T_{10}, 11.7 DICKE SWITCH	R	"
27	T_{11}, 13.9 WARM LOAD	R	"
28	T_{12}, POLAR SWITCH OR CIRC.	R	"
29	T_{13}, SCAT CALIB SWITCH	R	"
30	T_{14}, 9.3 IMAGE REJECTION FILTER	R	"

TABLE II RAW DATA VECTOR (continued)

NAME : DATA

POSITION	ENTRY	TYPE	UNITS
31	T_{15}, BASEPLATE	R	VOLTS
32	T_{16}, OMT	R	"
33	T_{17}, HOT LOAD	R	VOLTS
34	V_{agc}	R	VOLTS
35	V_{supply}	R	"
36	FLAG (T_{11} MISSING)	I	—
37	EMPTY		
38	"		
39	"		
40	"		
41	RADAR ALTITUDE	R	FEET
42	GRD SPEED	R	FT/SEC
43	PITCH	R	DEGREES
44	DRIFT	R	DEGREES
45	ROLL	R	DEGREES
46	AMBIENT AIR TEMP	R	°C
47	FLIGHT DIRECTION	R	DEGREES
48	EMPTY		
49			
50			

indicator variables are defined in Table III. These indicators are employed to cue certain program actions or to index variables and must be right justified. Items 2 through 13 and 16 are extracted from the RADSCAT microwave data channel. Items 17 through 32 are retrieved from the RADSCAT ATM channel. Item 33 is presently entered by hand but may eventually appear on the ATM channel. Items 34 and 35 are presently ignored but may also eventually appear on the ATM channel. Item 36 is presently entered by hand and is used to flag missing temperature data (DATA(17) - DATA(32)). Items 41 to 45 and 47 are provided by the Litton Navigator LTN-51 aboard the C-130. Item 46 is provided by an external sensor aboard the aircraft.

The measurement records must be completed as shown in Table II. However, the calibration records may be abbreviated to include only the data type indicator (DATA(4)) and the calibration entries in the appropriate SCAT and/or in the RAD position. The normal calibration records must be ordered by SCAT channels (one through four) with the normal RAD calibration occurring in the fourth record (containing the SCAT channel 4 calibration). The routine which extracts calibration data will orient itself on the calibration record for SCAT channel 1 in an attempt to meet contingencies where calibration record drop-outs occur. Any number of normal calibration sequences (calibrations on channels one through four in each sequence) may occur successively. All will be averaged. Typically the instrument produces three sequences of normal calibrations. The baseline calibration record simply requires the appropriate RAD baseline calibration in the DATA(13) position (as well as the type indicator in DATA(4)) and may occur anywhere in the file except within a sequence of normal calibrations. It is essential that the baseline and normal calibrations be grouped by three or more consecutive records to assure that a statistically significant average will be established. The program will average consecutive baseline records, even if they are interspersed with sets of normal calibrations. The normal calibrations will also be averaged in this circumstance.

A routine prepared by Langley Research Center actually merges data from several sources to meet the above specifications.

3. Preparation and Use of the Control Cards

There are four types of processing directives and three service directives. The directives consist of a control word and some arguments. All control words are left justified; the arguments are right justified. Sample directives and their formats are defined in Table IV.

TABLE III. DEFINITION OF INTEGER INDICATOR VARIABLES

	0	1	2	3	4	5	6
Frequency Indicator		9.3 GHz	11.4*GHz	13.9 GHz			
Data Type	Measurement	Normal Calibration	Base Calibration				
Angle Designator		X	X	X	X	X	X
Polarization Indicator	Horizontal	Vertical					
Range Gate	2000 ft.	5000 ft.	10,000 ft.	20,000 ft.			
Mode Indicator**	R.O.	S.S.	F.A.	A.A.			

*NOT OPERATIONAL

**R.O. = RADIOMETER ONLY
S.S. = SHORT SCAT
F.A. = FIXED ANGLE
A.A. = ALTERNATING ANGLE

TABLE IV. CONTROL CARD OPTIONS ILLUSTRATED

CONTROL WORD	ARGUMENTS						
	MISSION	FLIGHT	LINE	RUN	LINE PRINT MODULUS	ICHAR SET NO.	FILES TO BE PROCESSED
A6, 4X	I5	I5	I5	I5	I5	I5	I5
SPECIF	207	12	2	3	1	2	1
	208						
FLTLIN	208	1	4	--	5	3	81
MISFLT	209	4	--	--	10	3	10
ALL	---	---	--	--	10	2	999
POST	---	---	--	--	--	--	---
PRINT	---	---	--	--	--	2	2
PUNCH	---	---	--	--	--	1	1

The control word for the processing directives designates the classification of the file search to be conducted. The search is conducted by comparing the arguments, mission, flight, line and run on the directive with the file labels on the Raw Data Tape. The depth of comparison in the label will depend on the type of processing directive. ALL will process all files on the tape, MISFLT will process all lines and runs under the designated mission and flight, FLTLIN will process all runs from the specified mission, flight, and line. SPECIF will process only a particular file labeled by the specified mission, flight, line and run. When consecutive processing directives are employed, the file search is effectively conducted from the beginning of the tape.

Arguments other than the mission, flight, line and run must also be provided. The sixth word LNPMOD controls the density of records on which reports will be made in the activity report. The use of this parameter will limit the amount of print out acquired during the reduction of records. It will not affect the density of output records.

The set of instrument characteristics (ICU) to be applied to the specified files is designated in the seventh word. The entry specifies the file position on a master tape of characteristics. If a zero is entered, the program will expect instrument parameters to be read from cards appended behind the program control card. When characteristics are read from the card reader, processing is limited to one physical file regardless of the directive. This control option is primarily reserved to test a set of instrument characteristics. See Section IV D for a description of the format required for the instrument characteristics. If 999 is specified, the last set of characteristics on the master (historical) tape will be used.

The last (eighth) entry NFP on the control card specifies the number of files to be processed under the directive. To process all files under the directive simply make NFP sufficiently large.

The special service command POST may be used to position a previously produced output tape to a point behind the last file on the tape. The use of this command will, therefore, permit new files to be appended to an old output tape. This command, when used, must appear before any processing directives. There are no arguments with this directive.

The two service directives PRINT and PUNCH are provided to list and list and punch a designated set of instrument characteristics, respectively. The desired set number is entered in columns 36 through 40 of the directive card. If n-1 sets behind the designated set are to be listed or punched also, NFP should be set to n; otherwise it is set equal to one. NFP appears in column 41 through 45 of the directive card.

4. Peripheral and Memory Requirement

The input and output unit requirements are illustrated in Figure 5. Under a processing directive units 01, 02, 04, 06, 08 are required; under the service directives PRINT or PUNCH units 04, 06, and 43 are required. Unit 08 is a scratch unit, either disc or tape, perferably disc. It is used to temporarily store bypassed records when it is necessary to search for calibration data.

Approximately 17000 decimal words of memory were required to load this program on a CDC-6600 to include buffer space and system routines.

5. Flags and Aborts

The activity report produced by the program conveys various messages to indicate the status of the processing. It is normal for the program to echo check the directives, to list the progress on the processing, and to indicate a normal termination. The Conversion Report will also make regular statements reflecting the progress of the processing and the quality of the measurements. A sample listing of the Conversion Report is shown in Figure 2 of Section III. In some instances however, improper use of the program or construction of the data will result in an error message or an abort.

When an abort condition is encountered, an error message is generated, an abort code is given, an end of file mark is appended to the output tape on unit 02, an END OF REEL statement is appended, and control is yielded to the system. Table V defines the various abort conditions. When invalid data records are encountered, an error message will identify the improper element and the entire record will be dumped in octal. The message will appear within the Conversion Report. When more than 10 bad records are encountered in a file, the remaining records in the file are bypassed and processing is directed to the next internal activity. When more than three bad files are encountered the processing is aborted.

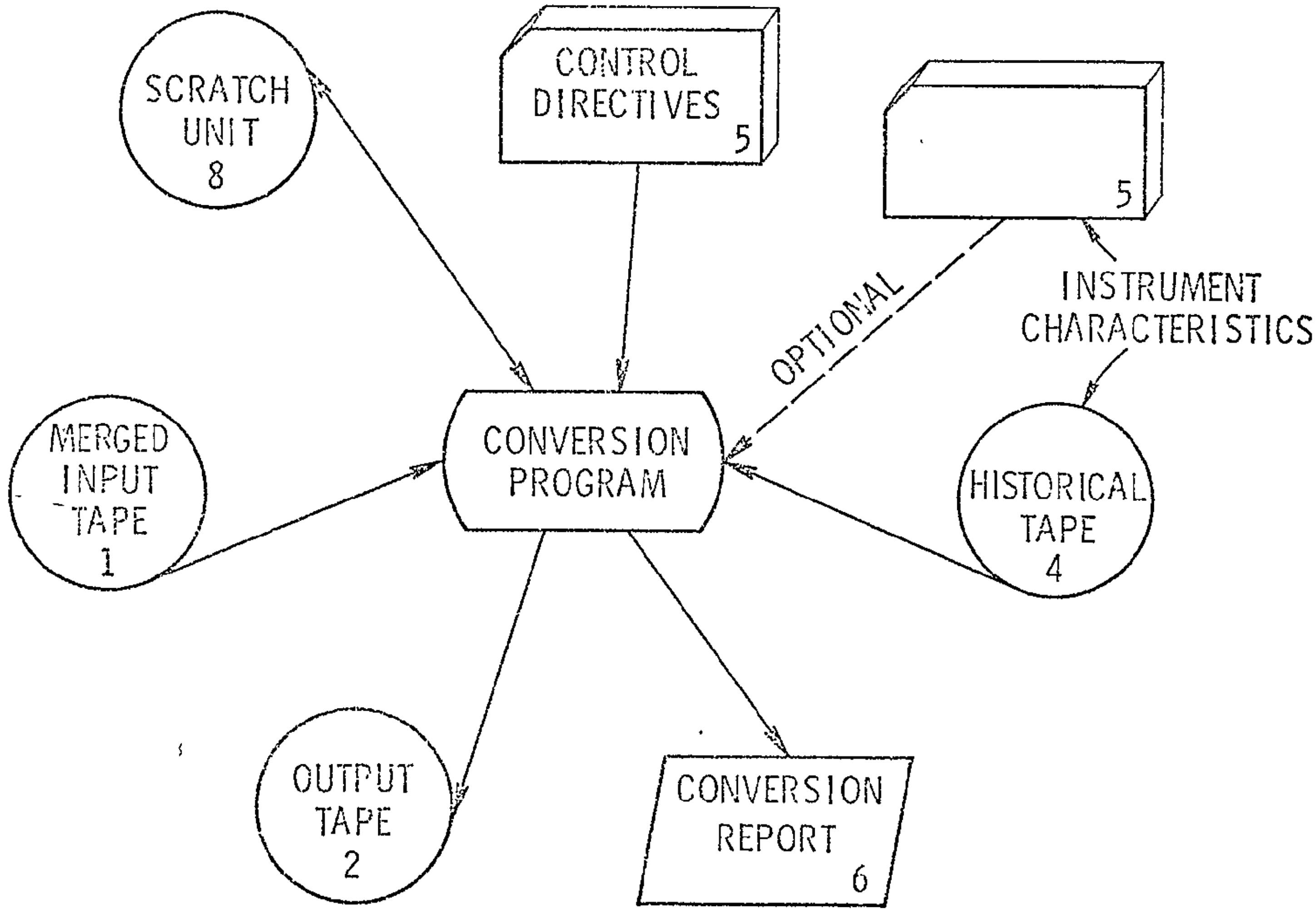


FIGURE 5. PERIPHERAL REQUIREMENTS FOR THE PROGRAM CONVERT

Table V. Definition of Abort Codes

UD	= Unrecognized Directive
IC	= Instrument Characteristic Missing
MF	= Missing Raw Data File
FF	= Unable to Find Instrument Characteristics with the Feed and Frequency Designated
ER	= Too Many File Errors
NFP	= NFP less than 1

6. Description of the Data Products

Normally two data products are generated by CONVERT, an activity report and an output tape. When the PUNCH option is employed a set or sets of punched instrument parameters can also occur. The character of the punched output is identical to that described in Section III D. The activity report reflects the program progress and largely contains items in the Conversion Report. See Figure 2 of Section III for an illustration of the Conversion Report. The Output Tape, however, will be described.

The output file structure is similar to that of the input raw data files. The file consists of the file label and blocks of output records. However, occurring between the label and output record blocks is the set of instrument parameters that had been used to process the input records. The entries in the file label are the same as that for the input files shown in Table I. The label and instrument parameters are created with separate unformatted WRITE statements. The entries in the instrument characteristics are described in Section III D. The output records occurring beneath the instrument parameters are blocked in 20 records groups. Each block is created with a binary WRITE statement. The elements in the output records consist of mixed integer and floating point entries. The entries are defined in Table VI. When insufficient records exist to fill the the last block in the file, the remaining entries are filled with integer zeros. An "end of file" mark is created after the last block in the file.

When all directives have been exhausted or the program aborted an "end of reel" statement is appended behind the last output file.

TABLE VI OUTPUT DATA VECTOR

POSITION	ENTRY	TYPE	UNITS
1.	TIME	R	HR, MIN, SEC
2.	MODE	I	—
3.	INCIDENCE ANGLE	R	DEGREES
4.	CROSS TRACK ANGLE	R	DEGREES
5.	TRANSMIT POLARIZATION	I	—
6.	RECEIVE POLARIZATION	I	—
7.	SCAT INTEGRATION TIME	R	SECONDS
8.	NORMALIZE SCATTERING COEF.	R	M^2/M^2
9.	NORMALIZE SCATTERING COEF.	R	dB
10.	ANTENNA TEMPERATURE	R	DEGREES KELVIN
11.	DEPOLARIZATION FACTOR	R	—
12.	DATA VALIDATION CODE	I	—
13.	ALTITUDE		FEET
14.	FLIGHT DIRECTION		DEGREES wrt TRUE NORTH
15.	AMBIENT AIR TEMP		DEGREES KELVIN
16-20.	EMPTY		

C. Output Program

1. General

The Output Program simply lists the output records prepared by program CONVERT. The output listings are organized by files which are selected by external directives similar to those described for the Conversion Program. A sample of a listing is illustrated in Figure 3 of Section III. The file heading together with instrument characteristic identifiers compose the page heading. Beneath the heading are listed 80 record images; each image consists of the first fourteen entries in an output record as defined in Table VI. The fifteenth entry is not printed but is available on tape should the investigator choose to use it to further reduce the radiometer data. Each line of print-out is appended with the record number. The record number corresponds to those used in the Conversion Report. The 80 records are intended to fill two consecutive pages of print-out where the page size is 8-1/2" by 15". Appended to the foot of each double page is a key defining the S/R validation code in column 13.

The instrument characteristics although present on the output tape are not listed. The set which has produced the data is identified in the heading on the right by the set number and date. No option is provided to print the characteristics.

The job deck for the program OUTPUT is organized identically to the conversion program (however, no instrument characteristics must appear in the data deck). File types are selected from the output by proper use of the directives:

1. ALL
2. MISFLT
3. FLTLIN
4. SPECIF

The meaning and use of these directives have been presented in Sections III B.1 and III B.2. The format of these control cards is identical to those described above. In the argument field of the control card the appropriate file identifiers must be present. No provision is made to limit the number of files to be processed, nor can the record density be controlled through a line print modulus. It is, of course, meaningless to specify the instrument characteristics ICU.

2. Program Messages

In addition to listing the output file, OUTPUT will keep the user posted on its internal activity as it searches for the specified files. The directive is initially echo checked, bypassed files are identified and the user is notified when the reel is encountered. Unrecognized directives will terminate the processing.

3. Peripheral and Memory Requirement

Peripheral requirements are illustrated in Figure 6. The program required approximately 7000 decimal words of memory on a CDC 6600 to include buffer space and system routines.

D. Program ICHAR

1. General

The primary purpose of program ICHAR is to maintain a file of instrument characteristics. The file is maintained not only to serve as a source of parameters to reduce raw data records but also to serve as a history of the changes undergone by the RADSCAT instrument. The file can consist of many sets of instrument parameters which are appended to the file numerically. The user may also identify the set by a date which is entered with the parameters.

Each set of instrument characteristics can consist of four subsets. The four subsets permit characterizing the instrument for a combination of two feeds and two frequencies. Each subset is composed of a set of radiometer parameters and a set of scatterometer parameters. The subsets are identified by feed and frequency. These identifiers are used by the Conversion Program to select the appropriate subset of characteristics.

This file maintenance program is operated by means of external control cards. Four directive (processing) options are offered:

1. INITIALIZE
2. APPEND
3. PRINT
4. PUNCH

They must be left justified and occupy columns 1 through 10 of the control card. INITIALIZE will create the historical file by storing the first set of characteristics on the tape. The set must appear behind the control card. APPEND will permit other sets to be added to the existing historical file. The characteristics set to be appended must appear behind the APPEND command card. When several sets are added at one operation, each set must be led by an APPEND card and intervening PRINT or PUNCH commands are not permitted. When sets are appended to the file, the old sets are transferred from the old tape to a new tape and the new sets are then

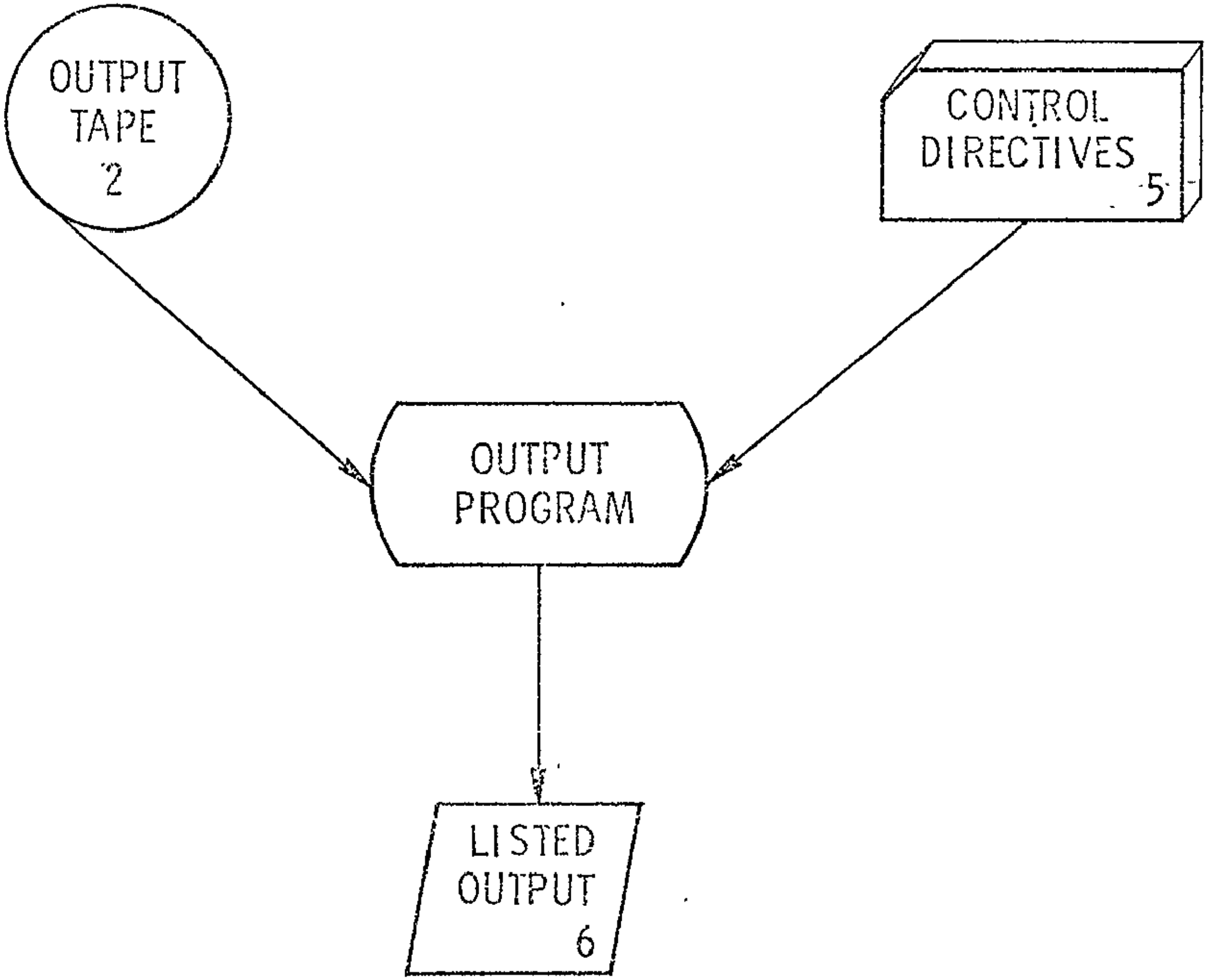


FIGURE 6. PERIPHERAL REQUIREMENTS FOR THE PROGRAM OUTPUT

added to the new tape. Before appending a new set of characteristics, it is advisable to have tested them in the Conversion Program by entering them from the card reader.

The PRINT and PUNCH commands perform the same service as they do in the Conversion Program. However, one command must be used for each set to be printed or punched. The set to be printed or punched must be specified on the control card in columns 11 through 15. The set number must be right justified. A listing accompanies the punched output. The PRINT or PUNCH may preceed or follow a set of APPEND commands but must not intermingle with them. These commands are provided so that new sets may be formed easily from old sets of characteristics.

A sample deck is illustrated in Figure 7.

2. Preparation of the Instrument Characteristics

The instrument parameters cards must be prepared in groups of four subsets. When insufficient data exists to complete one of the subsets, a complete subset may be repeated an appropriate number of times to fill the requirement. A sample subset is shown in Figure 8. The lines actually represent card images (except for the word SET). The entries in the upper half are radiometer parameters whereas the entries in the lower half are scatterometer parameters. This subset would be used for measurements taken at 13.9 GHz with a waveguide switch (WG SW) as the polarization switch. The radiometer and scatterometer parameters are identified with the set number of which it is a member and with a date of entry (origin). The first line of radiometer or scatterometer parameters must be punched with the format 4I5, 10X, E10.3, A10, E10.3 whereas the remaining lines must be punched according to the format 7E10.3. Excluding the set number and date, the radiometer parameters array consist of 75 words and the scatterometer parameter array consists of 85 words. Not all words are necessarily filled.

The entries in the radiometer and scatterometer arrays are defined in Tables VII and VIII, respectively. Many of the entries can be clarified by referring to a description of the reduction algorithms in the Appendices. These algorithms have been developed from the transfer and calibration characteristics of the RADSCAT instrument. As a consequence the entries may change (and perhaps the program algorithms) when the instrument is modified. Many of the entries must be based on special calibrations and measurements. Explanations are required for some of the

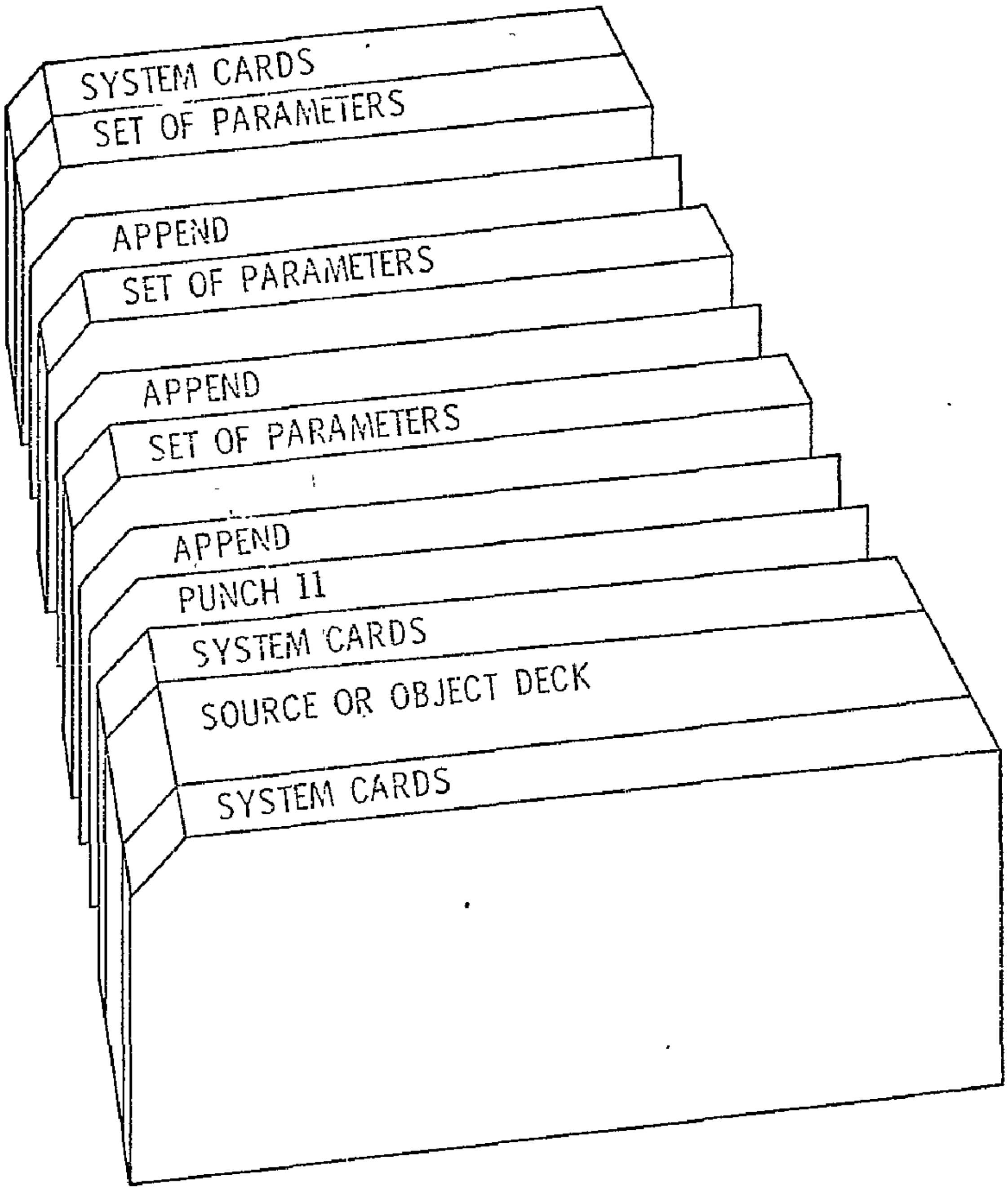


FIGURE 7. JOB DECK FOR PROGRAM ICHAR

SET 1 DATE	10 23 72	1.3900E+10	WG SW	0.		
0.	3.9900E+02	1.0540E+00	1.9950E+02	4.0000E-01	8.0000E-02	8.0000E-02
1.0200E-02	1.0150E-02	6.0000E-02	6.0000E-02	0.	0.	-0.
-0.	1.2800E-01	1.0000E-01	3.2500E+02	3.0500E+02	-8.7800E+00	7.9924E+01
-8.7800E+00	7.9871E+01	-8.7800E+00	7.9169E+01	-8.7800E+00	7.8879E+01	-8.7800E+00
7.9476E+01	-8.7800E+00	7.8730E+01	-8.7800E+00	7.9415E+01	-8.7800E+00	7.9704E+01
-8.7800E+00	7.9388E+01	-8.7800E+00	7.9889E+01	-8.7800E+00	8.0178E+01	-8.7800E+00
7.8826E+01	-8.7800E+00	7.9757E+01	-8.7800E+00	7.9309E+01	-8.7800E+00	7.9573E+01
-8.7800E+00	7.9379E+01	1.0000E+00	-2.7318E+02	-0.	-0.	-0.
-0.	-0.	-0.	-0.	-0.	-0.	-0.
-0.	-0.	-0.	-0.	-0.	-0.	-0.
-0.	-0.					

SET 1 DATE	10 23 72	1.3900E+10	WG SW	1.0000E-01		
5.8000E-01	5.9500E-01	6.3700E-01	7.0700E-01	8.0200E-01	9.2400E-01	3.0000E-01
3.7568E-01	3.7916E-01	0.	0.	4.9753E-01	5.0213E-01	2.0893E-07
1.0000E+00	1.0000E-02	3.1623E-04	3.1623E-06	-8.0000E+03	3.5000E+03	2.5000E+02
1.4125E+04	1.4125E+04	1.4000E+00	1.0500E+01	1.4000E-01	-6.3400E+00	5.0000E+01
2.2514E-01	2.2514E-02	0.	0.	4.6000E-02	7.1000E-02	9.6000E-02
1.3000E-01	1.6500E-01	2.1300E-01	2.6200E-01	2.8600E-01	3.0000E-01	6.4200E-01
5.1500E-01	6.0000E-01	7.6900E-01	8.8500E-01	1.0000E+00	1.1730E+00	1.3460E+00
1.3460E+00	1.3460E+00	1.2500E+00	1.1540E+00	1.0960E+00	1.0380E+00	1.0190E+00
1.0000E+00	1.0560E+00	1.1120E+00	1.2170E+00	1.3230E+00	1.3730E+00	1.4230E+00
1.4850E+00	1.5380E+00	1.4420E+00	1.3460E+00	1.1570E+00	9.6900E-01	7.6000E-01
5.5000E-01	4.2100E-01	2.9700E-01	2.3300E-01	1.7300E-01	1.4400E-01	1.1500E-01
1.0000E-01	8.5000E-02	-0.	-0.	-0.		

FIGURE 8. SAMPLE OF INSTRUMENT CHARACTERISTICS

TABLE VII RADIOMETER PARAMETERS

VARIABLE NAME : RPAR

POSITION	ENTRY		TYPE	UNITS
1	FREQUENCY		R	GHZ
2	FEED (WG SW OR CIRC)		A	UNITLESS
3	CROSS FEED CONSTANT	ξ_{HV}	R	"
4	" "	ξ_{VH}	R	"
5	HOT LOAD REFERENCE CONST	σ_1	R	°K
6	WARM LOAD REFERENCE CONST	σ_2	R	"
7	HOT LOAD BIAS FACTOR	σ_3	R	"
8	WARM LOAD BIAS FACTOR	σ_4	R	UNITLESS
9	SWITCH COMPENSATION FACTOR	ρ_{1H}	R	UNITLESS
10	" " "	ρ_{1V}	R	"
11	OMT " "	ρ_{2H}	R	"
12	" " "	ρ_{2V}	R	"
13	CUTLER FEED " "	ρ_{3H}	R	"
14	" " "	ρ_{3V}	R	"
15	INPUT GUIDE " "	ρ_{4H}	R	"
16		ρ_{4V}	R	"
17	RESIDUAL "	ρ_{5H}	R	
18	"	ρ_{5V}	R	
19	MEASUREMENT PERIOD	τ	R	SEC
20	CALIBRATION PERIOD	τ_C	R	SEC
21	UPPER TEMP LIMIT	UP	R	°K
22	LOWER TEMP LIMIT	LOW	R	°K
23	THERMISTOR CONVERSION FACTOR	α_1	R	°C/VOLT
24	" " "	β_1	R	°C
25	" " "	α_2	R	°C/VOLT
26	" " "	β_2	R	°C
27	" " "	α_3	R	°C/VOLT
28	" " "	β_3	R	°C
29	" " "	α_4	R	°C/VOLT
30	" " "	β_4	R	°C

TABLE VII RADIOMETER PARAMETERS (continued)

POSITION	ENTRY				TYPE	UNITS
31	THERMISTOR	CONVERSION	FACTOR	α_5	F6.2	°C/VOLT
32	"	"	"	β_5	"	°C
33	"	"	"	α_6	"	°C/VOLT
34	"	"	"	β_6	"	°C
35	"	"	"	α_7	"	°C/VOLT
36	"	"	"	β_7	"	°C
37	"	"	"	α_8	"	°C/VOLT
38	"	"	"	β_8	"	°C
39	"	"	"	α_9	"	°C/VOLT
40	"	"	"	β_9	"	°C
41	"	"	"	α_{10}	"	°C/VOLT
42	"	"	"	β_{10}	"	°C
43	"	"	"	α_{11}	"	°C/VOLT
44	"	"	"	β_{11}	"	°C
45	"	"	"	α_{12}	"	°C/VOLT
46	"	"	"	β_{12}	"	°C
47	"	"	"	α_{13}	"	°C/VOLT
48	"	"	"	β_{13}	"	°C
49	"	"	"	α_{14}	"	°C/VOLT
50	"	"	"	β_{14}	"	°C
51	"	"	"	α_{15}	"	°C/VOLT
52	"	"	"	β_{15}	"	°C
53	"	"	"	α_{16}	"	°C/VOLT
54	"	"	"	β_{16}	"	°C
55	"	"	"	α_{17}	"	°C/VOLT
56	"	"	"	β_{17}	"	°C
57	EMPTY					
.	.					
.	.					
.	.					
75	EMPTY					

TABLE VIII SCATTEROMETER PARAMETERS

POSITION	ENTRY		TYPE	UNITS
1	FREQ		R	HZ
2	FEED (WG SW OR CIRC)		A	—
3	CALIBRATION PERIOD	C	R	SEC
4	INT. TIME #1	1	R	"
5	INT. TIME #2	2	R	"
6	INT. TIME #3	3	R	"
7	INT. TIME #4	4	R	"
8	INT. TIME #5	5	R	"
9	INT. TIME #6	6	R	"
10	SHORT SCAT INT TIME	7	R	"
11	TRANSFER FUNCTION	G_{HT}	R	—
12	" "	G_{VT}	R	—
13	" "	G_{VHT}	R	—
14	" "	G_{HVT}	R	—
15	" "	G_{HR}	R	—
16	" "	G_{VR}	R	—
17	" "	G_C	R	—
18	CAL ATTN CHAN 1	A_1	R	—
19	" " " 2	A_2	R	—
20	" " " 3	A_3	R	—
21	" " " 4	A_4	R	—
22	FILTER LOWER LIMIT (NEG)		R	HZ
23	FILTER UPPER LIMIT (POS)		R	HZ
24	FILTER RESOLUTION		R	HZ
25	ANT GAIN H	Γ_H	R	—
26	ANT GAIN V	Γ_V	R	—
27	CHAN #1 SATURATION LEVEL		R	VOLTS
28	EMPTY			—
29	MINIMUM SQUARE LAW LEVEL FOR ALL CHANNELS		R	VOLTS
30	ANGLE CONVERSION FACTOR		R	DEG/VOLT

TABLE VIII SCATTEROMETER PARAMETERS (continued)

POSITION	ENTRY		TYPE	UNITS
31	ANGLE CORRECTION FACTOR*		R	DEG.
32	EQUIVALENT BEAMWIDTH H θeq		R	RADIANS
33	EQUIVALENT BEAMWIDTH V θeq		R	"
34	EMPTY			—
35	"			—
36	"			—
37	"			—
38	RELATIVE FILTER GAIN,	f= 3000	R	—
39	"	f= 2750	R	—
40	"	f= 2500	R	—
41	"	f= 2250	R	—
42	"	f= 2000	R	—
43	"	f= 1750	R	—
44	"	f= 1500	R	—
45	"	f= 1250	R	—
46	"	f= 1000	R	—
47	"	f= 750	R	—
48	"	f= 500	R	—
49	"	f= 250	R	—
50	"	f= 0	R	—
51	"	f= -250	R	—
52	"	f= -500	R	—
53	"	f= -750	R	—
54	"	f=-1000	R	—
55	"	f=-1250	R	—
56	"	f=-1500	R	—
57	"	f=-1750	R	—
58	"	f=-2000	R	—
59	"	f=-2250	R	—
60	"	f=-2500	R	—

*Includes relative pitch between aircraft frame and antenna gimbal.

TABLE VIII SCATTEROMETER PARAMETERS (continued)

POSITION	ENTRY		TYPE	UNITS
61	RELATIVE FILTER GAIN,	f=-2750	R	—
62	"	f=-3000	R	—
63	"	f=-3250	R	—
64	"	f=-3500	R	—
65	"	f=-3750	R	—
66	"	f=-4000	R	—
67	"	f=-4250	R	—
68	"	f=-4500	R	—
69	"	f=-4750	R	—
70	"	f=-5000	R	—
71	"	f=-5250	R	—
72	"	f=-5500	R	—
73	"	f=-5750	R	—
74	"	f=-6000	R	—
75	"	f=-6250	R	—
76	"	f=-6500	R	—
77	"	f=-6750	R	—
78	"	f=-7000	R	—
79	"	f=-7250	R	—
80	"	f=-7500	R	—
81	"	f=-7750	R	—
82	"	f=-8000	R	—
83	EMPTY			
84	"			
85	"			

entries. The thermistor conversion factors in RPAR(23) through RPAR(56) convert the thermistor voltages in DATA(17) through DATA(33) to temperature in centigrade (see Table II). A sampled representation of the scatterometer doppler filter frequency characteristic appears in SPAR(34) through SPAR(85). The sample frequency interval is designated in SPAR(24) and the frequency domain of the doppler filter is specified in SPAR(22) and SPAR(23).

3. Peripheral and Memory Requirements

The peripheral requirements are illustrated in Figure 9. To load the program required approximately 10,000 decimal words of memory on a CDC-6600 to include buffer space and system routines.

E. Program TDUPE

Program TDUPE scales and integerizes output files prepared by program CONVERT and transfers the result to another tape. The resulting tape serves as a master tape which may be duplicated and sent to the investigator for use on another type of computing machine.

The scaling and transferring of an output tape is implemented by an external control card appended to the source or object deck. The card simply states the files by mission and flight which are to be transferred. The control card requires a 2I10 format. The input files must be located on tape unit 01. The specified file types are transferred to a tape on unit 02. An activity report, stating the status on the transfers, is generated. The program requires approximately 8000 decimal words on a CDC-6600. The peripheral requirements are illustrated in Figure 10.

F. Special Programs

Several engineering application programs are provided to compute and/or verify certain antenna parameters employed in the instrument parameters. Program WIDTH computes the equivalent pencil beamwidth, an important parameter to the inversion of the scatterometer data (SPAR(32) and SPAR(33)). The program GAIN computes the antenna gains SPAR(25) and SPAR(26) and the beam solid angle.

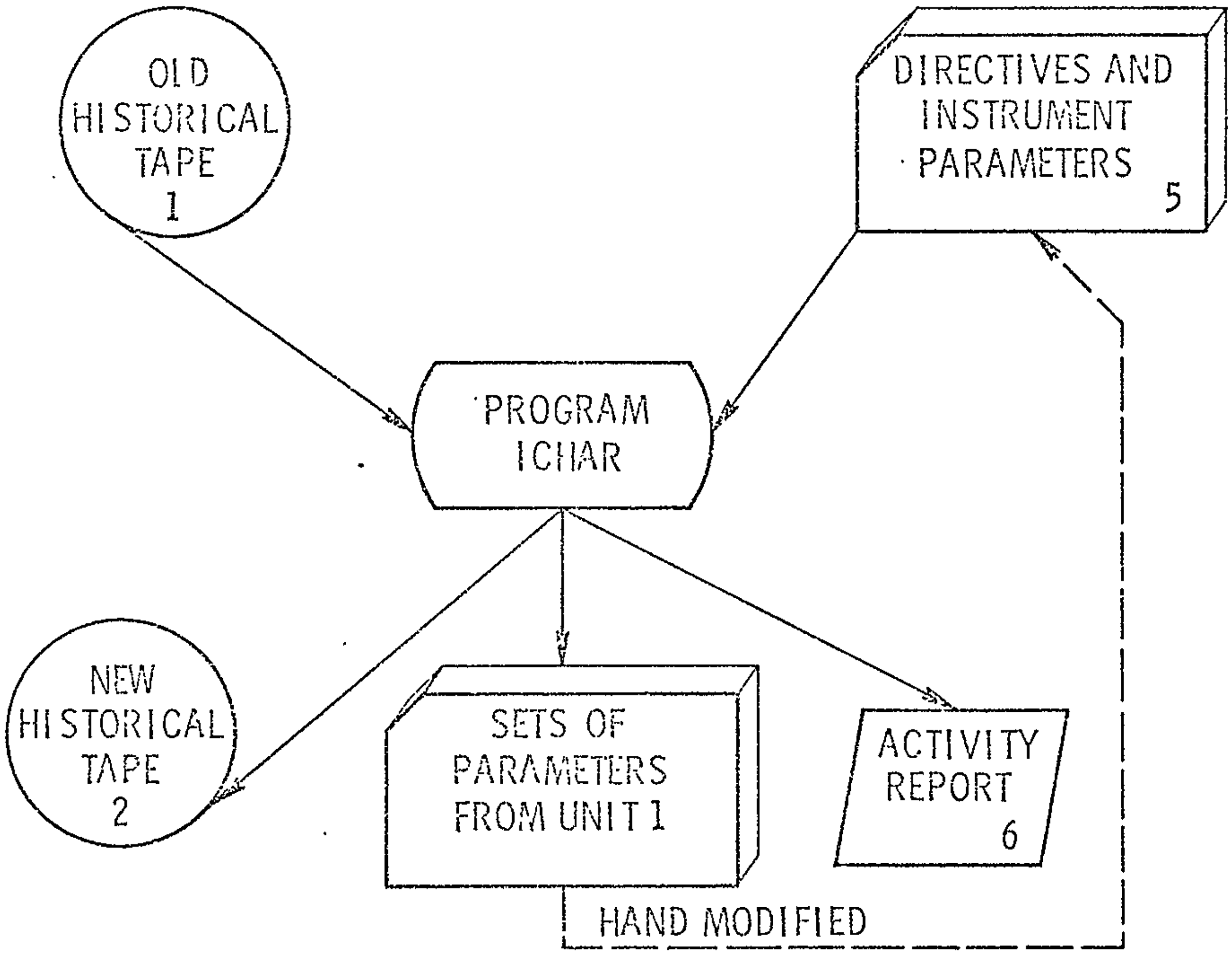


FIGURE 9. PERIPHERAL REQUIREMENTS FOR PROGRAM ICHAR

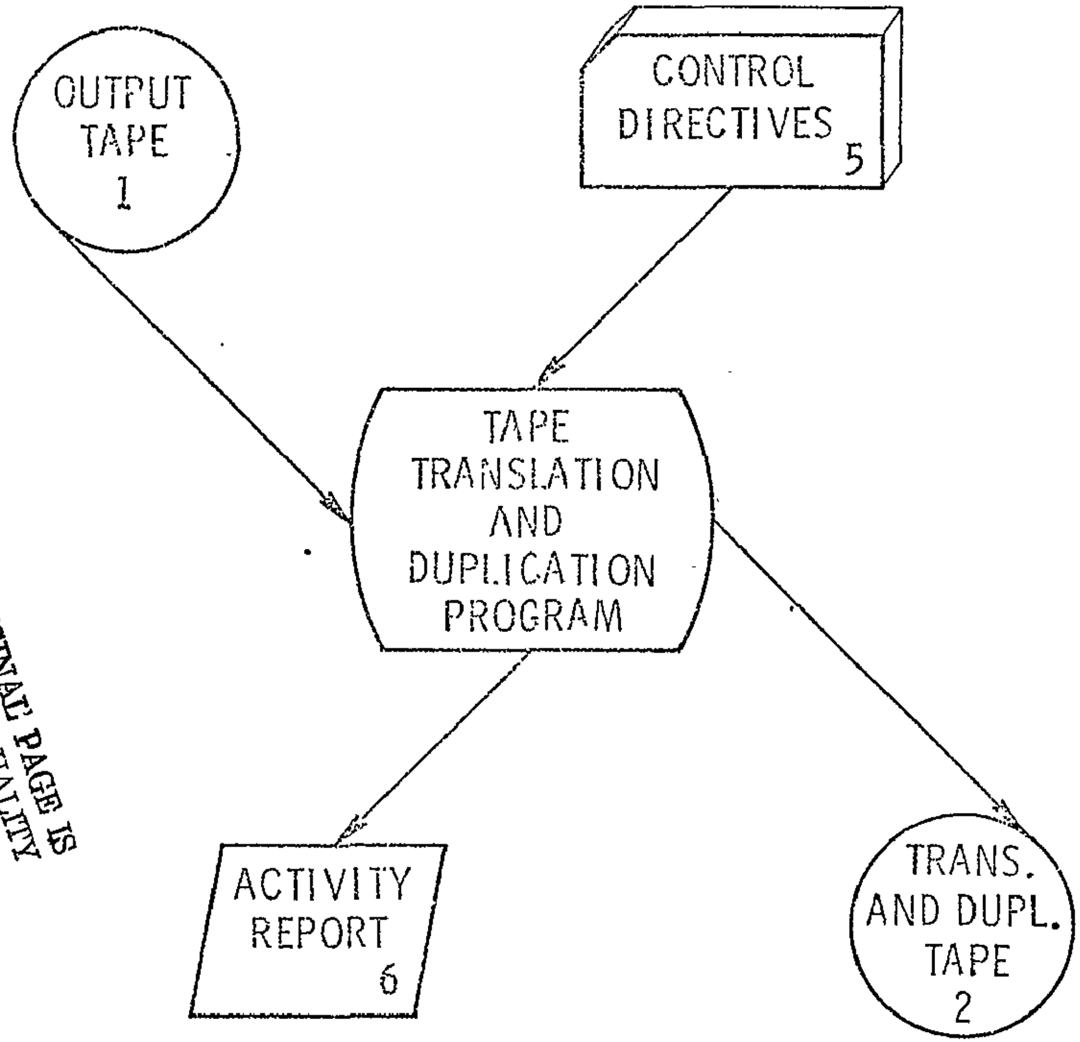


FIGURE 10. PERIPHERAL REQUIREMENTS FOR PROGRAM TDUPE

Program WIDTH is self sufficient when antenna pattern information is provided in the FUNCTION subroutine FUN. Presently functional representations of the main beam out to 6 degrees from boresight are provided. This routine may be replaced by a sampled version of the main beam. However, interpolation between sampled values must be provided since a double integration routine requires values dictated by the Gaussian Legendre quadrature technique.

Program GAIN requires both polarized and cross polarized power patterns. Functional representations of the polarized pattern is now given in FUN1 and the cross polarized pattern in FUN2. These may be replaced with sampled valued representation, if desired. Again, interpolation will be required.

For more specific information see Appendix E.

APPENDICES

APPENDIX A

CONVERSION PROGRAM

I. INTRODUCTION

The conversion program (CONVERT) applies many reduction algorithms to the raw RADSCAT data to provide the investigator with a set of comprehensive data products with which to interpret the scattering and emission characteristics of the scene represented by the data. CONVERT consists of the Control Section and the following subroutines:

1. IREAD	7. CALIB
2. READ1	8. RCONV
3. WRITE2	9. ANGLE
4. ABORT	10. DOPCHK
5. SEARCH	11. SCONV
6. CRUNCH	12. THERMO

The theory and design of these routines are treated in this appendix.

II. GENERAL OVERVIEW

The Control Section reads and interprets external directives. On the basis of the directive, the appropriate files are retrieved and presented to the processing subroutines. Subroutine SEARCH executes entry into the data records of the file in search of the appropriate calibration records. When found, the calibration information is extracted and averaged in subroutine CALIB. When sufficient calibration data cannot be found, the file is bypassed. The actual processing is performed by subroutine CRUNCH which applies various algorithms and calibration information to reduce the data. Many of the algorithms are embodied in subroutines, bearing descriptive names, which are called from CRUNCH.

Subroutine ANGLE computes the incident and cross-track angles using aircraft attitude and look angle information. Subroutine THERMO checks internal receiver temperatures. The antenna temperature is computed in RCONV using the transfer function of the RADSCAT radiometer, the temperatures developed in THERMO and RAD calibration parameters. When scatterometer data is present, the maximum on-scale measurement is selected from the four SCAT channels with an in-line routine in CRUNCH. Using aircraft ground speed, DOPCHK computes the doppler shift in the scatterometer signal. The doppler filter gain is chosen accordingly. The scatterometer transfer function is applied in subroutine SCONV to compute the scatterometer normalized input power. An in-line routine in CRUNCH applies antenna inversion parameters to the input power to compute the normalized scattering coefficient.

When additional calibration records are encountered, the old calibration information is replaced by the new calibration information by subroutine CALIB.

Subroutine IREAD and READ1 perform the file read functions, whereas subroutine WRITE2 performs the write functions.

Subroutine ABORT is employed by the CONTROL Section to abort the job when an abnormal condition is encountered.

III. CONTROL SECTION

A. Theory and Design

The control section was developed to give the user a large amount of flexibility in the choice of the instruments characteristics and the files to be processed. As many or as few files of a particular type can be processed. Some files can be treated with one set of instrument characteristics and others with another. These features permit the user to (1) test a set instrument parameters, (2) group the files by type, (3) process different types of files with different instrument parameters, (4) etc. The flexibility in the choice of parameters was incorporated with the full realization that the instrument would undergo modifications.

Control of the program is executed by the choice of appropriate directives entered as data cards. The types of controls have already been discussed in Section II and III of the text and will not be treated here again. The method by which control cards are read and files are retrieved is shown in the logic diagram of Figure A-1.

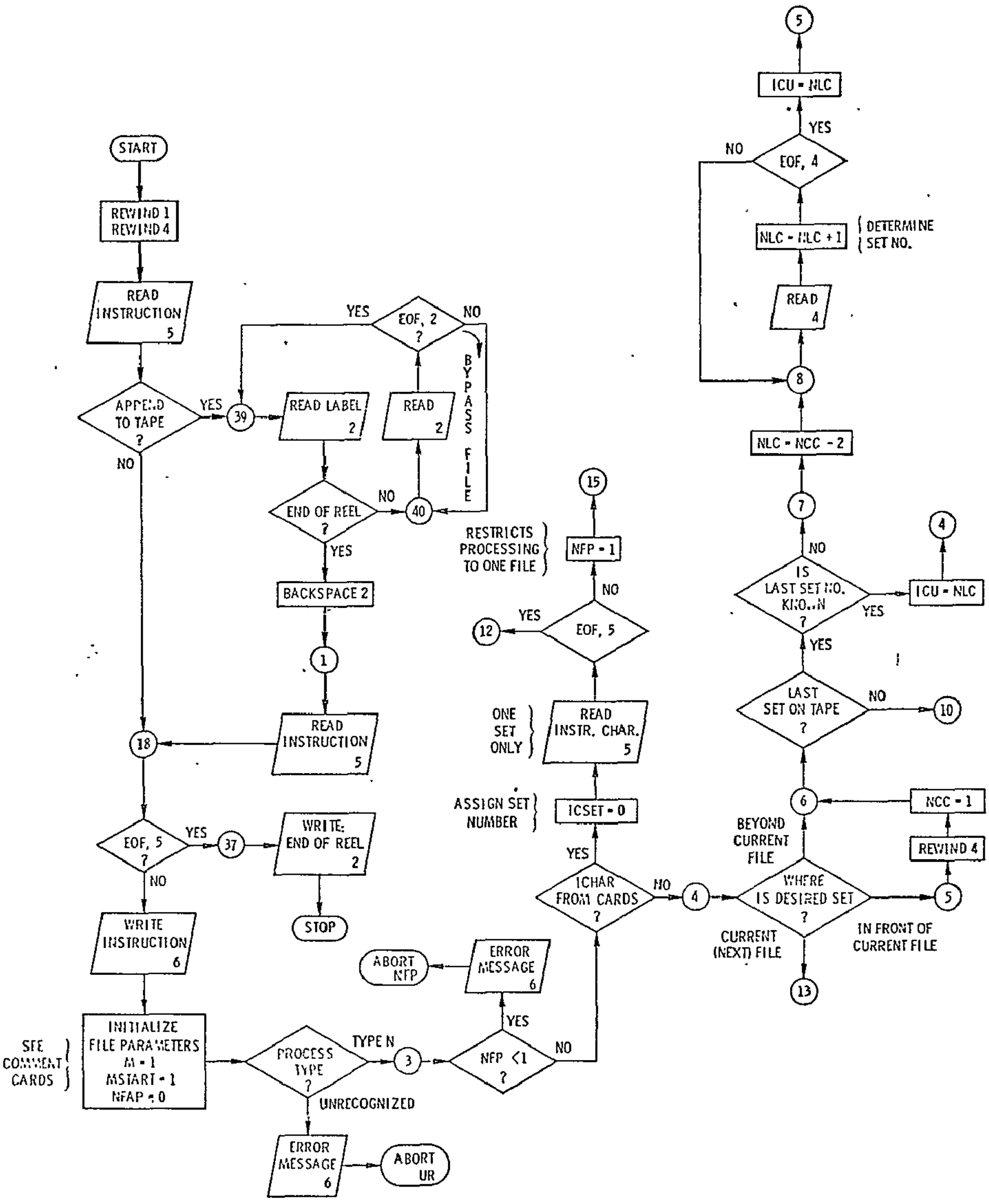


FIGURE A-1(a). DESCRIPTIVE LOGIC DIAGRAM FOR THE CONTROL SECTION

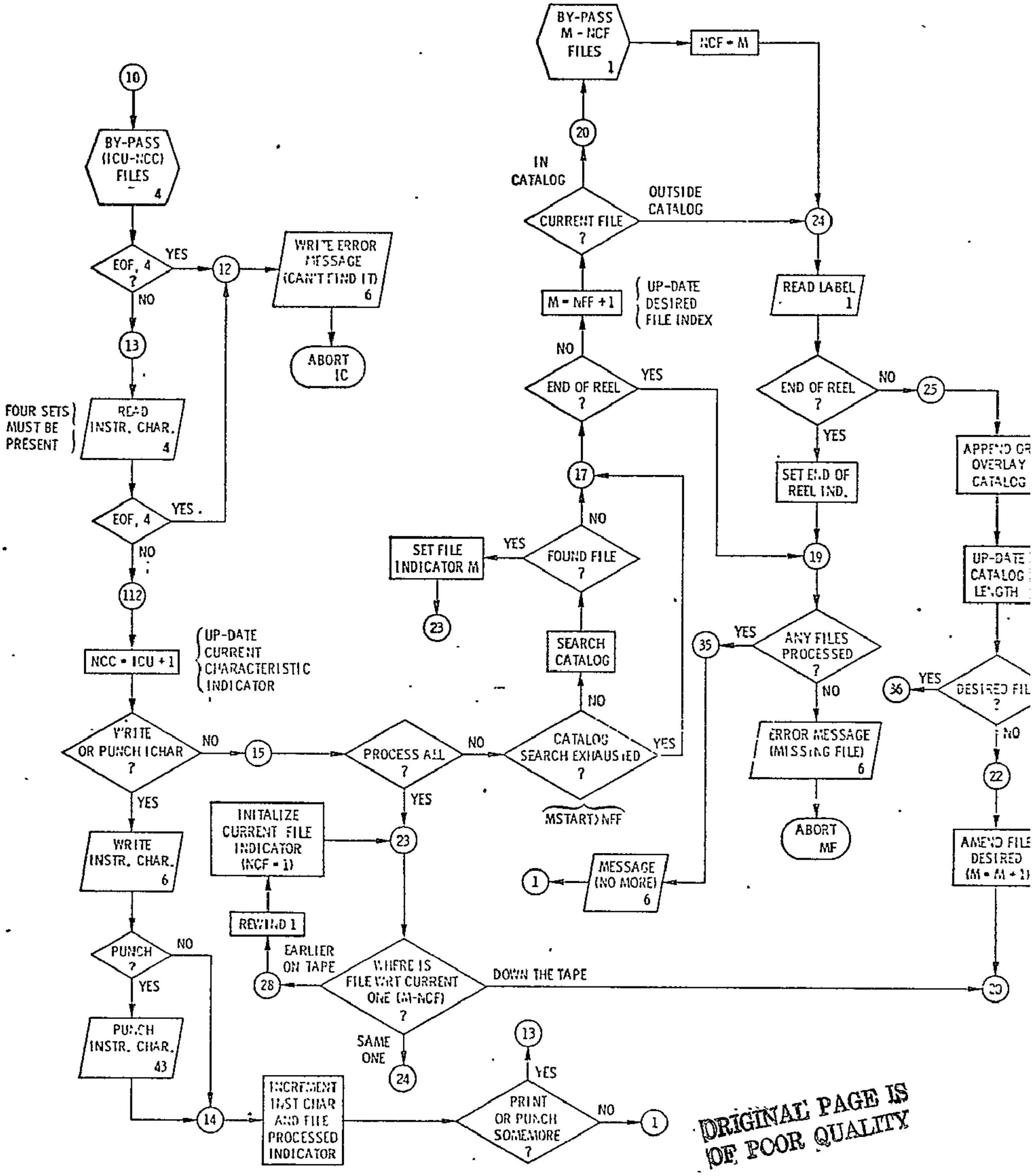


FIGURE A-1. (b) DESCRIPTIVE LOGIC DIAGRAM FOR FOR THE CONTROL SECTION

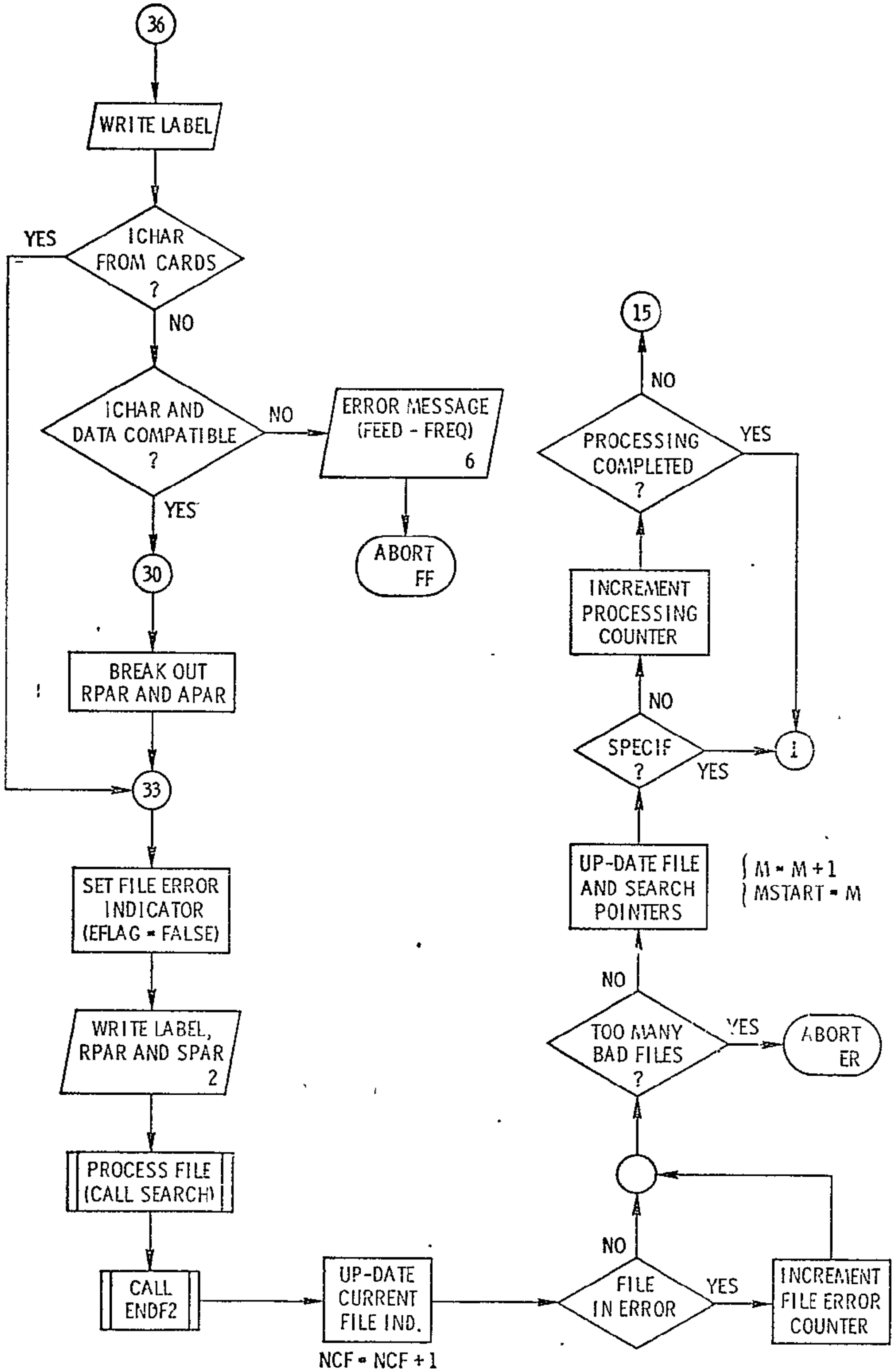


FIGURE A-1. (c) DESCRIPTIVE LOGIC DIAGRAM FOR THE CONTROL SECTION

The numbers on the connecting points coincide with the statement numbers of the text shown in Figure A-2.

An instruction is read from the card reader. If it is POST, the files on an old output tape are bypassed in statements between 39 and 1 of the diagram. Another instruction is then read. The instruction is echo checked, file parameters are initialized and in the subsequent blocks of Figure A-1, the processing type is identified. Thereafter the specified instrument characteristic is read from unit 04 or 05. If the last set of characteristics on a file code 04 is desired and its file number is not known (ICU=999), the last file number is determined in statements between 6 and 5. If, on the other hand, they are to be read from cards (ICU=0), the appropriate action is taken and control is advanced to statement 15. If a specific file is designated, an appropriate number of files is bypassed until tape unit 04 is positioned on the correct file (see steps between statements 10 and 13).

If the set is to be punched or printed, the appropriate action is taken in steps preceeding statement 14. If NFP>1, NFP-1 subsequent sets are also punched or printed. (See steps between 112 and 13). When completed, the next control card is read.

When the control card is a processing directive, the instrument characteristic tape will have been positioned to the correct file (set of parameters) or the set will have been entered from cards. Steps following 15 search for the appropriate data file. The search is governed by the type of directive. Bypassed files are cataloged in steps between 25 and 22. If the desired file is in the catalog and the current file is strictly in the catalog, the appropriate number of files are bypassed in steps between 22 and 24. If the file is beyond the catalog, the catalog is updated as the program advances the tape in search of the file.

Once found, the file label is listed and the appropriate subset of instrument parameters is withdrawn from tape if they weren't entered from the reader. SEARCH is then called to find the calibration data. SEARCH in turn calls CRUNCH to process the measurement records. Once completed control returns to the Control Section where additional files are sought or another directive is read.

B. Program Listing

The Control Section Listing is shown in Figures A-2(a) through A-2(e). The variables are defined in the listing.

```
      PROGRAM CONVERT(INPUT,OUTPUT,PUNCH,TAPE1,TAPE2,TAPE4,TAPE8,
     *   TAPE5=INPUT,TAPE6=OUTPUT,TAPE43=PUNCH)
CCCCS          CONVERSION CONTROL SECTION
C
C              THIS PROGRAM WAS PREPARED BY
C
C                   JOHN P. CLAASSEN
C                   GLEN E. ELLIOTT
C
C              UNIVERSITY OF KANSAS CENTER FOR RESEARCH
C
C
C                             PROGRAM DESCRIPTION.
C              FILE CODES USED.
C
C              01 -- INPUT RAW DATA.
C              02 -- OUTPUT PROCESSED DATA.
C              04 -- INPUT INSTRUMENT CHARACTERISTICS FILE.
C              05 -- SYSTEM INPUT - DIRECTIVE CARDS.
C              06 -- SYSTEM OUTPUT - PRINT LINES.
C              08 -- SCRATCH FILE - USED BY COMPUTATION SECTION.
C              43 -- SYSTEM OUTPUT - PUNCHED CARDS.
C
C              VARIABLE NAMES USED.
C
C     LBLTBL   ≠LABEL TABLE≠.  TABLE CONTAINING
C              LABEL INFORMATION (MISSION, FLIGHT, LINE, RUN) FOR ALL
C              DATA FILES BYPASSED SO FAR.
C
C     NWPST    ≠NEW POSITION.≠ ARRAY CONTAINING LABEL INFO (M,F,L,R)
C              OF NEXT FILE TO BE PROCESSED.
C
C     NCF      ≠NUMBER OF CURRENT FILE.≠  POSITION POINTER FOR
C              FILE CODE 01.
C
C     NFF      ≠NUMBER OF FINAL FILE.≠  INDEX OF LAST ENTRY IN
C              ≠LBLTBL≠ TABLE.
C
C     NSTART   STARTING POINT FOR SEARCH IN ≠LBLTBL≠ TABLE.
C
C     MI       INPUT FILE TO USE. RELATIVE POSITION ON FILE CODE 01.

C     EORI     ≠END OF REEL INDICATOR.≠  LOGICAL VARIABLE, SET TRUE
C              WHEN END OF REEL ON FILE CODE 01 IS DETECTED, INDICATING
C              ≠LBLTBL≠ TABLE IS COMPLETE.
C
C     IEOR     INTEGER ARRAY CONTAINING WORDS ≠END OF REEL≠.
C
C     IPTYPE   INTEGER ≠PROCESSING TYPE≠ ARRAY, CONTAINING RECOGNIZED
C              PROCESSING TYPES.
C
C     JPTYPE   INPUT DESIRED PROCESSING TYPE.
C
C     N        INDEX OF POSITION OF ≠JPTYPE≠ IN ≠IPTYPE≠.
C
C     ICU      ≠INSTRUMENT CHARACTERISTICS TO USE≠.  NUMBER OF THE I.C.
C              SET TO BE USED.  REFLECTS THE POSITION OF THE SET ON
C              FILE CODE 04.
C              0 INDICATES I.C. TO BE READ FROM CARDS (FC=05),
C              999 MEANS USE THE LAST SET ON FILE 04.
C
C     NCC      ≠NUMBER OF CURRENT CHARACTERISTICS≠.  REFLECTS POSITION C
C              FILE CODE 04.
C
C     NLC      ≠NUMBER OF LAST CHARACTERISTIC≠.  USED WHEN ICU=999.
C              NLC=0 INDICATES THIS VALUE IS YET UNKNOWN.
C
C     RPAR     ≠RADIOMETER PARAMETERS≠.
C     SPAR     ≠SCATTEROMETER PARAMETERS≠.
C
C     PAR      ≠PARAMETERS≠  ARRAY CONTAINING UP TP 4 SETS OF RADSCAT
C              INSTR PARAMETERS READ FROM FILE CODE 04 OR 05. RAD
C              PARAMETERS COMPOSE THE FIRST 75 ENTRIES OF PAR AND SCAT
C              PARAMETERS OCCUPY THE NEXT 85 ENTRIES IN EACH SUBSET.
C              THE APPROPRIATE SUBSET OF PARAMETERS IS SELECTED ON THE
C              BASIS OF FREQUENCY AND FEED.
C
C     NFP      ≠NUMBER OF FILES TO PROCESS≠ USING THE PROCESSING TYPE
C              GIVEN BY ≠JPTYPE≠ AND DESCRIPTION IN ≠NWPST≠.
C              NFP IS IGNORED IF JTYPE = SPECIF.  IF ICU = 0,
C              NFP IS SET TO 1.
C
C     NFAP     ≠NUMBER OF FILES ALREADY PROCESSED.≠  COMPARED TO ≠NFP≠
C              TO CHECK FOR TERMINATE CONDITION.
C
```


FIGURE A-2. (a) LISTING OF CONTROL SECTION

```
          C       LNPMOD  *LINE PRINTING MODULUS*. USED TO REDUCE OUTPUT              00000102
          C               FROM COMPUTATION SECTION.  0 IS TAKEN AS 1.                 00000103
          C                                                                           00000104
          C       NERR    *NUMBER OF ERRORS*. NUMBER OF TIMES COMPUTATION SECTION     00000105
          C               RETURNED ERROR CONDITION.  NERR = 4 CAUSES ABORT.           00000106
          C
          C       ERRFLG  BAC FILE ERROR FLAG.
          C                                                                           00000107
          C                                                                           00000108
          C                              DIRECTIVE CARD USAGE                         00000109
          C                                                                           00000110
          C       PROCESSING                   MEANING                                00000111
          C        TYPE                                                               00000112
          C       ALL     START FROM FIRST OF INPUT FILE AND PROCESS WITHOUT          00000113
          C                REGARD TO LABEL INFORMATION.                               00000114
          C                                                                           00000115
          C       MISFLT  CHECK LABEL ONLY FOR CORRECT MISSION AND FLIGHT.            00000116
          C                                                                           00000117
          C       FLTLIN  CHECK FOR CORRECT MISSION, FLIGHT, AND FLIGHT LINE.         00000118
          C                                                                           00000119
          C       SPECIF  CHECK FOR MISSION, FLIGHT, FLT. LINE, AND RUN.              00000120
          C                                                                           00000121
          C       PRINT   PRINT OUT VALUES OF INSTRUMENT CHARACTERISTICS.             00000122
          C                                                                           00000123
          C       PUNCH   PRINT AND PUNCH I.C.*S.                                     00000124
          C                                                                           00000125
          C       POST    MAY BE USED AS THE FIRST PROCESSING TYPE.  IF SO,           00000138
          C               THE PROGRAM WILL SPACE DOWN THE OUTPUT TAPE SEARCHING       00000139
          C               FOR THE END OF REEL.  POST IS ILLEGAL EXCEPT ON THE         00000140
          C               FIRST DIRECTIVE CARD.                                       00000141
          C
          C               DIRECTIVE CARD FORMAT.                                      00000126
          C                                                                           00000127
          C       JPTYPE      MISS  FLT LINE  RUN LNPM  ICU  NFP                      00000128
          C        A6    4X    I5    I5   I5   I5   I5   I5   I5                    00000129
          C                                                                           00000130
          C       ALL          *     *    *    *    1    3    6                       00000131
          C       MISFLT      207   12    *    *    5    3    2                       00000132
          C       FLTLIN      207   12    1    *   10    3    5                       00000133
          C       SPECIF      207   12    1    2    1    3    1                       00000134
          C       PUNCH        *     *    *    *    *    4    6
          C       PRINT        *     *    *    *    *    5    7
          C       POST         *     *    *    *    *    *    *

          C                                                                           00000135
          C               * INDICATES FIELD IS IGNORED BY PROGRAM.                    00000136
          C                                                                           00000137
          C                                                                           00000142
000003            DIMENSION NWPST(4), LBLTBL(4,300), IEOR(2), RPAR(75), SPAR(85),     00000018
                 *           PAR(160,4), IPTYPE(6), IDAT(3)                           00000019
000003            DIMENSION LBL(9)                                                    00000020
000003            EQUIVALENCE (LBL(8),FREQ),(LBL(9),FEED)                             00000021
000003            EQUIVALENCE (IDAT(1),ID1),(IDAT(2),ID2),(IDAT(3),ID3)               00000022
000003            LOGICAL  EORI, ERRFLG                                               00000023
000003            DATA IEOR   /10HEND OF REE,1HL/                                     00000024
000003            DATA IPTYPE/6HFLTLIN,6HMISFLT,6HSPECIF,3HALL,5HPRINT,5HPUNCH/       00000025
000003            DATA IPOST/4HPOST/                                                  00000026
000003            DATA EORI/.FALSE./, NCC,NCC/0,1/, NCF,NFF/1,0/, NERR/0/             00000027
000003            REWIND 1                                                            00000143
000005            REWIND 4                                                            00000144
000007            READ(5,500)  JPTYPE, NWPST, LNPMOD, ICU, NFP                        00000145
000025            IF(JPTYPE.NE.IPOST) GO TO 18                                        00000146
000027     39     READ(2) LB1,LB2                                                     00000147
000036            IF(LB1.NE.IEOR(1)) GO TO 40                                         00000148
000040            IF(LB2.NE.IEOR(2)) GO TO 40                                         00000149
000042            BACKSPACE 2                                                         00000150
000044            GO TO 1                                                             00000151
000045     40     READ(2)                                                             00000152
000050            IF(EOF,2) 39, 40                                                    00000153
          C                                                                           00000154
          C               READ DIRECTIVES                                             00000155
          C                                                                           00000156
000053      1     READ(5,500)  JPTYPE, NWPST, LNPMOD, ICU, NFP                        00000157
000071    500     FORMAT(A6,4X,7I5)                                                   00000158
000071     18     IF(EOF,5)  37,41                                                    00000159
000074     41     WRITE(6,600) JPTYPE, NWPST, LNPMOD, ICU, NFP                        00000160
000112    600     FORMAT(1H1,//10X,17H CONTROL CARD IS ,                              00000161
                 *            A6,4X,7I5)                                              00000162
          C                                                                           00000163
          C               WHAT MANNER TO PROCESS                                      00000164
          C                                                                           00000165
000112            M=1                                                                 00000166
000113            MSTART=1                                                            00000167
000114            NFAP=0                                                              00000168
000115            DO 2 N=1,6                                                          00000169
000117      2     IF(JPTYPE.EQ.IPTYPE(N)) GO TO 3                                     00000170
000123            WRITE(6,601)                                                        00000171
```


FIGURE A-2. (b) LISTING OF CONTROL SECTION

```
000126    601   FORMAT(10X,30HUNRECOGNIZED PROCESSING FORMAT)
000126          CALL ABORT(2HCF)
000130    3     IF(NFP.GE.1) GO TO 59
000133          WRITE(6,610) NFP
000140    610   FORMAT(//10X,*NFP LT 1*,5X,*NFP = *,I5//)
000140          CALL ABORT(3HNFP)
          C
          C              FIND INSTRUMENT CHAR.
          C
000142    59    IF(ICU.NE.0) GO TO 4
          C                                 READ FROM CARDS
000143          ICSET=0
000144          READ(5,4303) NUMBER,ID1,ID2,ID3,RPAR(1),RPAR(2),RPAR(3)
000166          IF(EOF,5) 12, 60
000171    60    WRITE(6,603) ICSET,ID1,ID2,ID3,RPAR(1),RPAR(2),RPAR(3)
000213          DO 61 J=4,75,7
000215          L=J+6
000217          IF(L.GT.75) L=75
000222          READ(5,4304)(RPAR(K),K=J,L)
000235          IF(EOF,5) 12, 61
000240    61    WRITE(6,604)(RPAR(K),K=J,L)
000255          READ(5,4303) NUMBER,ID1,ID2,ID3,SPAR(1),SPAR(2),SPAR(3)
000277          IF(EOF,5) 12, 62
000302    62    WRITE(6,603) ICSET,ID1,ID2,ID3,SPAR(1),SPAR(2),SPAR(3)
000324          DO 63 J=4,85,7
000326          L=J+6
000330          IF(L.GT.85) L=85
000333          READ(5,4304)(SPAR(K),K=J,L)
000346          IF(EOF,5) 12, 63
000351    63    WRITE(6,604)(SPAR(K),K=J,L)
000366          NFP=1
000367          GO TO  15
000370    4     IF(ICU-NCC) 5,13,6
000373    5     REWIND 4
000375          NCC=1
000376    6     IF(ICU.NE.999) GO TO 10
          C                                 FIND LAST SET ON FILE.
000400          IF(NLC.EQ.0) GO TO 7
000401    42    ICU=NLC
000403          GO TO 4
000403    7     NLC=NCC-2
000405    8     READ(4)
000410          NLC=NLC+1

000412          IF(EOF,4) 45, 8
000415    45    ICU=NLC
000417          GO TO 5
          C                                 POSITION TO THE SET
000417    10    NC1=NCC+1
000421          DO 11 I=NC1,ICU
000422          READ(4)
000425          IF(EOF,4) 12,11
000430    11    CONTINUE
000433          GO TO 13
000433    12    WRITE(6,602)
000437    602   FORMAT(10X,34HNO SUCH INSTRUMENT CHARACTERISTICS)
000437          CALL ABORT(2HIC)
          C                                 READ THEM
000441    13    READ(4) ICSET,ID1,ID2,ID3,PAR
000456          IF(EOF,4)  12,112
000461    112   NCC=ICU+1
000463          IF(N.LT.5) GO TO 15
          C
          C              WRITE (AND PUNCH) INST. CHAR.
          C
000466          DO 55 I=1,4
000467          WRITE(6,603)     ICU,ID1,ID2,ID3,PAR(1,I),PAR(2,I),PAR(3,I)
000515    603   FORMAT(//* SET*,I5,*  DATE*,3I5,E15.4,5X,A10,E15.4)
000515          DO 54 J=4,75,7
000517          L=J+6
000521          IF(L.GT.75) L=75
000524    54    WRITE(6,604)(PAR(K,I),K=J,L)
000543    604   FORMAT(7E15.4)
000543          WRITE(6,603)     ICU,ID1,ID2,ID3,PAR(76,I),PAR(77,I),PAR(78,I)
000572          DO 55 J=79,160,7
000574          L=J+6
000576          IF(L.GT.160) L=160
000601    55    WRITE(6,604)(PAR(K,I),K=J,L)
000622          IF(N.NE.6) GO TO 14
000624          DO 57 I=1,4
000626          LL=1
000627          WRITE(43,4303)     ICU,ID1,ID2,ID3,PAR(1,I),PAR(2,I),PAR(3,I),
               *         ICU,I,LL
000664    4303  FORMAT(4I5,10X,E10.3,A10,E10.3,12X,I2,I1,I5)
000664          DO 56 J=4,75,7
000666          LL=LL+1
000670          L=J+6
```


FIGURE A-2. (c) LISTING OF CONTROL SECTION

```
C00672    56    WRITE(43,4304)(PAR(K,I),K=J,L),ICU,I,LL
C00715    4304  FORMAT(/E10.3,2X,I2,I1,I5)
000715          LL=LL+1
CC0717          WRITE(43,4303)    ICU,IDAT,PAR(76,I),PAR(77,I),PAR(78,I),
              *        ICU,I,LL
000747          DO 57 J=79,16C,7
000751          LL=LL+1
C00753          L=J+6
C00755    57    WRITE(43,4304)(PAR(K,I),K=J,L),ICU,I,LL
C01002    14    ICU=ICU+1
C01004          NFAP=NFAP+1
C01005          IF(NFAP.LT.NFP) GO TO 13
C01007          GO TO 1
          C
          C             FIND THE DATA RECORD.
          C
001010    15    IF(N.EQ.4) GO TC 23
C01012          IF(MSTART.CT.NFF) GO TO 17
          C
          C             SEE IF WE*VE BEEN BY IT ALREADY.
          C
001016          DO 16 M=MSTART,NFF
C01017          IF(NWPST(1).NE.LBLTBL(1,M)) GO TO 16
C01022          IF(NWPST(2).NE.LBLTBL(2,M)) GO TO 16
001025          IF(N.EQ.2) GO TC 23
C01027          IF(NWPST(3).NE.LBLTBL(3,M)) GO TO 16
C01032          IF(N.EQ.1) GO TC 23
001033          IF(NWPST(4).NE.LBLTBL(4,M)) GO TO 16
C01036          GO TO 23
C01036    16    CONTINUE
001041    17    IF(EORI) GO TO 19
          C
          C             GUESS IT*S FURTHER DOWN THE TAPE.
          C
001043          M=NFF+1
C01045          IF(NCF.GT.NFF) GC TO 24
001050    20    M1=M-1
001052          DO 27 I=NCF,M1
001054    26    READ(1)
C01057          IF(EOF,1) 27, 26
001062    27    CONTINUE
001065          NCF=M
001066    24    READ(1) LBL

001073          IF(LBL(1).NE.IEOR(1)) GO TO 25
C01075          IF(LBL(2).NE.IEOR(2)) GC TO 25
C01077          EORI=.TRUE.
C01100    19    IF(NFAP.NE.0) GC TO 35
C01101          WRITE(6,606)
001105    606   FORMAT (10X,19HNC SUCH DATA RECORD)
C01105          CALL ABORT(2HFF)
001107    35    WRITE(6,607) NFAP
C01115    607   FORMAT(10X,I3,32H FILES FOUND.  CAN FIND NC MORE.)
001115          GO TO 1
C01116    25    LBLTBL(1,NCF)=LBL(1)
C01121          LBLTBL(2,NCF)=LBL(2)
001124          LBLTBL(3,NCF)=LBL(6)
001126          LBLTBL(4,NCF)=LBL(7)
C01130          NFF=MAX0(NFF,NCF)
001133          IF(N.EQ.4) GO TC 36
001135          IF(NWPST(1).NE.LBLTBL(1,M)) GO TO 22
001140          IF(NWPST(2).NE.LBLTBL(2,M)) GO TO 22
C01143          IF(N.EQ.2) GC TC 36
001144          IF(NWPST(3).NE.LBLTBL(3,M)) GO TO 22
C01147          IF(N.EQ.1) GC TC 36
C01150          IF(NWPST(4).NE.LBLTBL(4,M)) GO TO 22
C01153          GO TO 36
001153    22    M=M+1
001155          GC TO 20
001155    23    IF(M-NCF) 28, 24, 20
001160    28    REWIND 1
001162          NCF=1
001163          GO TO 23
          C                           FOUND IT. IT*S THIS ONE.
001164    36    WRITE(6,608) LBL, ICSET, ID1, ID2, ID3
C01202    608   FORMAT(1H1,51X,17HCCNVERSICN REPORT,
              * //10X,8HMISSICN-,I3,9H  FLIGHT-,I3,
              *        7H  DATE-,2I3,I5,2X,11H  FLT LINE-,1X, I5,6H  RUN-,I3,
              *        13H    FREQUENCY-,F5.1,7H  FEED-,2X,A6,
              *        6H IPAR=,I2,3I3////)
001202          IF(ICU.EQ.0) GO TO 33
          C                           GET CORRECT INST. CHAR.
C01203          DO 29 K=1,4
C01205          IF(ABS(FREC-PAR(5,K)/1.0 E9).GT.1.0) GO TO 29
001215          IF(IFEED.EQ.PAR(2,K)) GO TO 30
001221    29    CONTINUE
001223          WRITE(6,605) FREQ, FEED
```

FIGURE A-2. (d) LISTING OF CONTROL SECTION

```
001233     605  FORMAT (10X,17HNO SUCH FREQ-FEED,F10.5,1X,A10)
001233          CALL ABORT(2HFF)
001235      30  DO 31 I=1,75
001237      31  RPAR(I)=PAR(I,K)
001246          DO 32 I=1,65
001247      32  SPAR(I)=PAR(I+75,K)
           C                               CALL COMPUTATION SECTION.
001256      33  ERRFLG=.FALSE.
           C                               LABEL OUTPUT TAPE
001257          WRITE(2) LBL
001264          WRITE(2) ICSET, ID1, ID2, ID3, RPAR, SPAR
001303          CALL SEARCH(SPAR,RPAR,LNPMOD,ERRFLG)
001306          CALL ENDF2
001307          NCF=NCF+1
001311          IF(ERRFLG) NERR=NERR+1
001313          IF(NERR.GT.3) CALL ABORT(2HER)
001317          M=M+1
001321          MSTART=M
           C                               MORE FILES TO PROCESS
001322          IF(N.EQ.3) GO TO 1
001324          NFAP=NFAP+1
001325          IF(NFAP.GE.NFP) GO TO 1
001327          GO TO 15
           C
           C                  TERMINATIONS.
001330      37  WRITE(2) IECR, (I,I=3,9)
001342          WRITE(6,700)
001346     700  FORMAT(1H1,///11X,10(1H*),18HNORMAL TERMINATION,10(1H*))
001346          STOP
001350          END
```

FIGURE A-2. (e) LISTING OF CONTROL SECTION

C. Peripheral Requirements

UNIT 01	RADSCAT Raw Data Tape
UNIT 02	Output Tape
UNIT 04	Characteristics Tape
UNIT 05	Card Reader
UNIT 06	Printer
UNIT 08	Scratch Unit (Perferably Disc)
UNIT 43	Punch

IV. SUBROUTINES

A. Subroutine Search

1. Theory and Design

This routine serves as an entry and exit point for the selected file. One of its primary purposes is to find calibration data of both kinds. It anticipates that both NORMAL and BASELINE calibrations are in the opening records of the file. If this is the case the processing of the remainder of the file is given to CRUNCH. Otherwise, a search for the missing calibrations is conducted deeper into the file. The bypassed files are stored on a scratch unit (08) for subsequent processing. When both types of calibrations are found, the bypassed records on file code 08 are processed by CRUNCH and the remainder of the records on file code 01 are also.

If the return from CRUNCH is without an error flag, control is simply returned to the Control Section. If not, an error message is given and the remainder of the file bypassed before returning.

See Figure A-3 for a descriptive logic diagram of SEARCH.

2. Program Listing and Variables

The listing for SEARCH is shown in Figure A-4(a) and A-4(b). The definitions of the variables are given in Table A-1. The entries in DATA were defined in Section III B.

Table A-1 Definition of Variable Used in SEARCH

DATA	=	Vector Containing Raw RADSCAT DATA
IDATA	=	Integer Equivalent of DATA
SCAL	=	SCAT Calibration Vector (Channels 1-4)

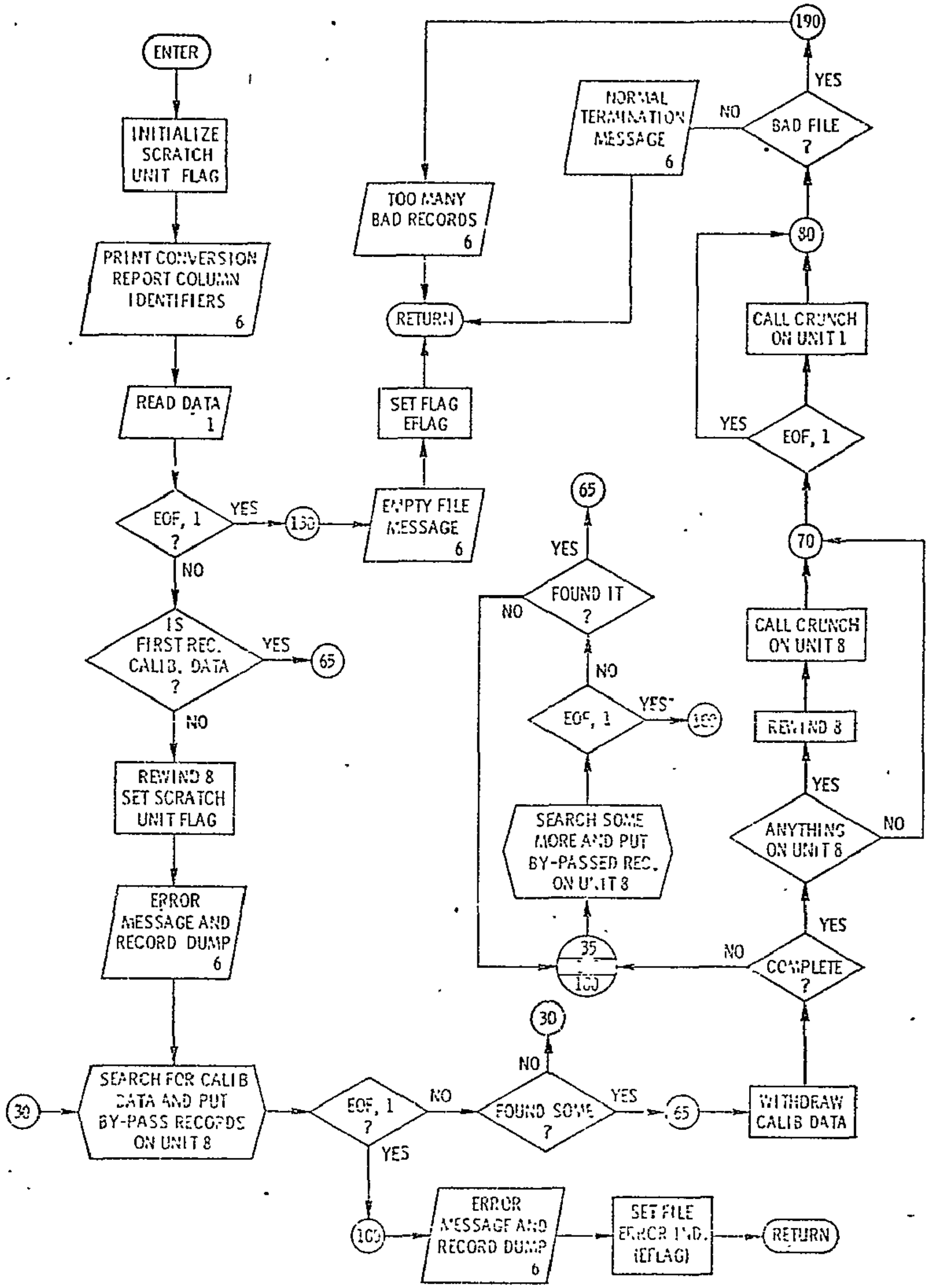


FIGURE A-3. DESCRIPTIVE LOGIC DIAGRAM FOR SEARCH

```
                SUBROUTINE SEARCH(SPAR,RPAR,LNPMOD,EFLAG)
        CSEARCH                        CALIB SEARCH ROUTINE
        C                SUBROUTINE SEARCH(SPAR,RPAR,LNPMOD,EFLAG)
        C
        C                THIS PROGRAM WAS PREPARED BY
        C
        C                     JOHN P. CLAASSEN
        C                     GLEN E. ELLIOTT
        C
        C                UNIVERSITY OF KANSAS CENTER FOR RESEARCH
        C
        C
        C                THIS SUBROUTINE EXECUTES ENTRY INTO THE CONTENTS OF THE
        C                FILE AND SEEKS CALIBRATION INFORMATION, BOTH NORMAL AND
        C                BASELINE. THE PROGRAM ANTICIPATES THAT BOTH TYPES OF
        C                CALIBRATIONS ARE PRESENT IN THE INITIAL RECORDS.  HOWEVER
        C                WHEN THIS NOT THE CASE, THE FILE IS SEARCHED UNTIL BOTH
        C                TYPES ARE FOUND.  BY-PASSED RECORDS ARE STORED ON UNIT 8
        C                SUBSEQUENT PROCESSING.
        C
C00007          DIMENSION DATA(50), IDATA(50),
              *    SPAR(65), RPAR(75), SCAL(4), RCAL(2)
C00007          COMMON /INPUT/ EOF1, DATA
C00007          EQUIVALENCE (DATA(1),IDATA(1))
C00007          LOGICAL  EFLAG, EOF1, EOF8
C00007          LOGICAL NORMAL, BASE
        C
        C             READ DATA RECORD FROM TAPE UNIT 1
        C
C00007          IPASS=0
000010          WRITE(6,60)
C00013    60    FORMAT(12H RECORD NO.*,13X,7HMESSAGE,19X,1H*,20X,
              *        10HPARAMETERS///)
000013          CALL READ1
C00014          IF(EOF1) GO TO 130
        C
        C             CHECK IF RECORD IS CALIB DATA
        C
C00020    10    IF (IDATA(4) .EQ. 0 .OR. IDATA(4).EQ.2) GO TO 50
        C
        C                PREPARE DISC FOR BY-PASSED RECORDS
        C

C00027          REWIND 8
C00031          IPASS=1
        C
        C             ERROR MESSAGE
        C
C00032          WRITE(6,20) IDATA(4)
000040    20    FORMAT(5X,1H*,5X,27HFILE STRUCTURE INCOMPATIBLE,7X,1H*,5X,
              *          33HFIRST RECORD NOT CALIBRATION DATA,5X, 6HMODE= ,I3/)
000040          WRITE(6,120) DATA
        C
        C             SEARCH FOR CALIBRATION RECORD
        C
C00046    30    WRITE(8) DATA
C00053    35    CALL READ1
C00054          IF(EOF1) GO TO 100
        C
        C             CHECK IF CALIB DATA
        C
000060    40    IF (IDATA(4) .NE. 0 .AND. IDATA(4).NE.2) GO TO 30
000067    50    ICOUNT=0
C00070          IL=0
        C
        C             WITHDRAW CALIBRATION INFORMATION
        C
C00071          NORMAL = .FALSE.
C00072          BASE = .FALSE.
C00073    65    CALL CALIB(SCAL,RCAL,RPAR,NORMAL,BASE)
C00077          IF (.NOT. NORMAL .AND. .NOT. BASE) GO TO 35
C00104          IF (.NOT. NORMAL) GO TO 160
000106          IF (.NOT. BASE) GO TO 180
C00107          IF(IPASS.EQ.0) GO TO 70
000110          ENDFILE 8
C00112          REWIND 8
000114          CALL CRUNCH(8,DATA,IDATA,IL,ICOUNT,SCAL,RCAL,
              *          SPAR,RPAR,LNPMOD,EFLAG,EOF8)
000131    70    IF(EOF1) GO TO 80
C00135          CALL CRUNCH(1,DATA,IDATA,IL,ICOUNT,SCAL,RCAL,
              *          SPAR,RPAR,LNPMOD,EFLAG,EOF1)
000152          IF(EFLAG) GO TO 190
        C
        C             NORMAL TERMINATION MESSAGE
        C
C00156    80    WRITE (6,90) IL, ICOUNT
```


FIGURE A-4. (a) SOURCE LISTING FOR SUBROUTINE SEARCH

```
000166    90     FORMAT (I6,5X,1H*,2X,36HCOMPLETED COMPUTATIONS FOR THIS FILE,
                *        1X,1H*,5X,10HTHERE WERE,I5,
                *        24H WARNING FLAGS GENERATED)
000166           RETURN
          C
          C          ERROR MESSAGE
          C
000167   100     WRITE (6,110)
000173   110     FORMAT (11X,1H*,5X,27HFILE STRUCTURE INCOMPATIBLE,7X,1H*,5X,
                *        25HUNABLE TO FIND CALIB DATA)
000173           WRITE(6,120) DATA
000201   120     FORMAT(/10(5O22/)/)
          C
          C          ABORT FILE AND SEEK NEXT INSTRUCTION
          C
000201           EFLAG=.TRUE.
000204           RETURN
000205   130     WRITE(6,140)
000211   140     FORMAT(11H EMPTY FILE)
000211           LFLAG=.TRUE.
000214           RETURN
          C
          C              SEARCH FOR MISSING CALIB DATA
          C
000215   150     WRITE(8) DATA
000222   160     CALL READ1
000223           IF(EOF1) GO TO 100
000227   165     IF (IDATA(4) .NE. 0) GO TO 150
000230           GO TO 65
000231   170     WRITE (8) DATA
000236   180     CALL READ1
000237           IF(EOF1) GO TO 100
000243   185     IF (IDATA(4) .NE. 2) GO TO 170
000245           GO TO 65
000246   190     WRITE(6,195)
000252   195     FORMAT(///11X,1H*,5X,28HEXCESSIVE NO. OF BAD RECORDS,6X,1H*)
000252           IF(EOF1) RETURN
000257   200     READ(1)
000262           IF(EOF,1) 250,200
000270   250     RETURN
000271           END
```

ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE A-4. (b) SOURCE LISTING FOR SUBROUTINE SEARCH

(Continued)
Table A-1 Definition of Variable Used in SEARCH entries in DATA

RCAL	=	RAD Calibrations (NORMAL, BASE)
EOF1	=	End of File for Unit 1
EOF8	=	End of File for Unit 8
EFLAG	=	File Error Flag
IPASS	=	Flag for Data on Unit 8
NORMAL	=	Flag for Normal RAD Cals
BASE	=	Flag for Baseline RAD Cals
ICOUNT	=	RADSCAT Performance Error Counter
IL	=	Record Counter

B. Subroutine CRUNCH

1. Theory and Design

The reduction algorithms are applied in or from subroutine CRUNCH. The routine was designed to process RADSCAT data whether it consisted of radiometer measurements (RAD only mode) or both scatterometer and radiometer measurements (alternating angles, fixed angle, or short SCAT modes). In processing the records on an independent basis, there is a tacit assumption that coupling between polarization is negligible. It was also necessary to equip the routine with the ability to read records from file code 01 (raw data tape) or file code 08 (scratch unit). A description of the internal operation of CRUNCH is shown in the logic diagram of Figure A-5.

In the opening steps of CRUNCH the data record is extracted from the file and verified by an in-line routine. The first sixteen entires in the data record are checked for type and magnitude. If an invalid entry is present, an error message identifying the bad entry is given and a bad record counter is incremented (NGR). If more than 10 bad records are encountered, an error flag (EFLAG) is set and control is returned to SEARCH. If not, the next record in read (statement 100) and the process repeated. If the record is valid, it is further examined to determine if it is a calibration record. If it is, CALIB is called (statement 220) to extract and average the calibrations. The next record is then read and the process

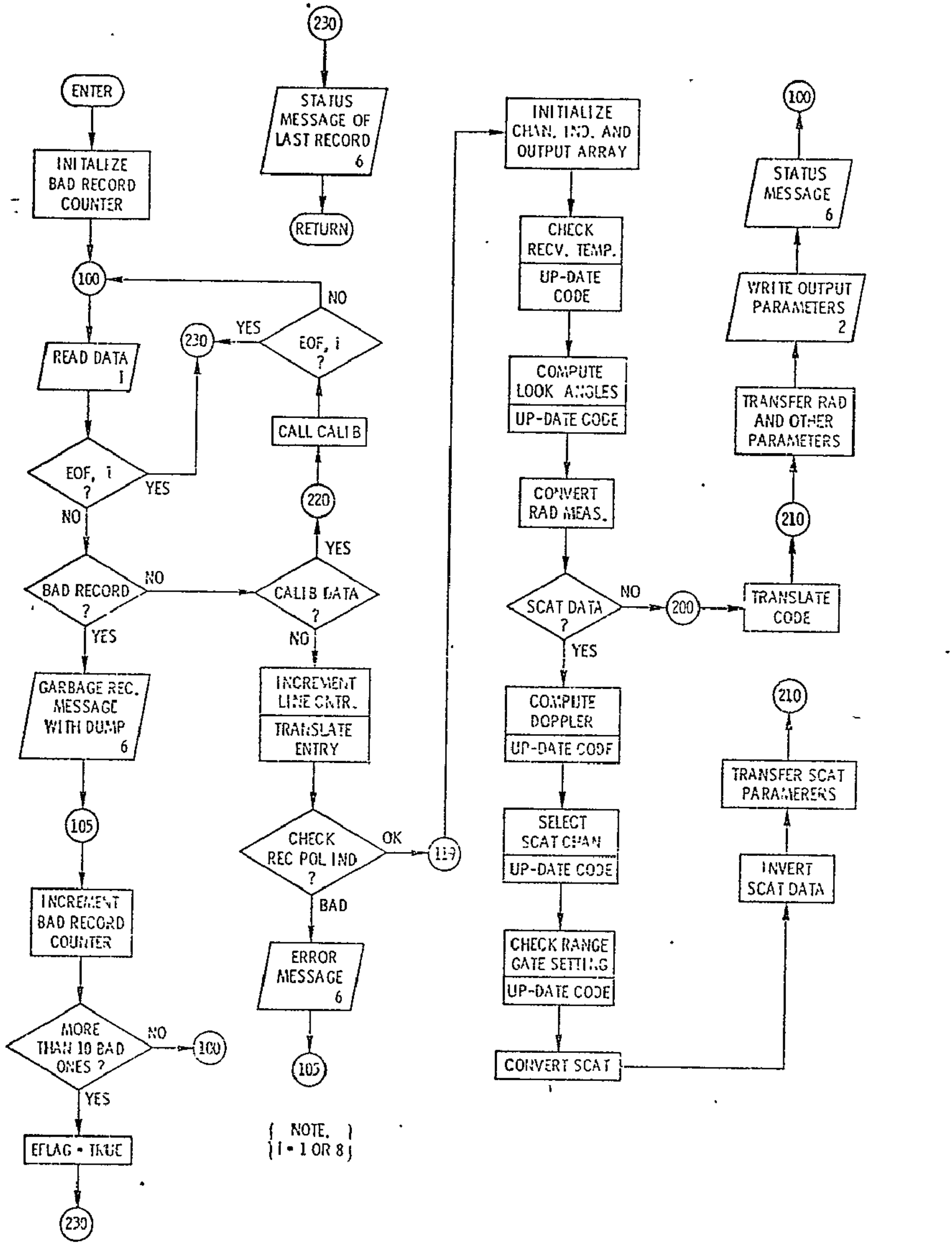


FIGURE A-5. DESCRIPTIVE LOGIC DIAGRAM FOR SUBROUTINE CRUNCH repeated. If it is not calibration data, the mode entry is translated and the SCAT and RAD receiver polarization are checked for agreement when the instrument is not in the "RAD only" mode. If they disagree the record is declared invalid and treated as a bad record.* If the measurement occurred in the "RAD only" mode or the polarization agree, the record is then processed in the steps following and including statement number 119.

Initially the record counter is incremented, the SCAT channel indicator initialized and the output array cleared. CRUNCH then calls THERMO to compute and check receiver temperatures. An interpretation code is updated after which subroutine ANGLE is called. ANGLE computes the incident and cross-track angles and again the interpretational code is updated. Subroutine RCONV (radiometer convert) is called to compute the antenna temperature. If no scatterometer data is present, control advances to statement 200 where the computed results are transferred to the output array. Otherwise when SCAT data is present, DOPCHK is called to compute the doppler shift and set the doppler filter band index. The interpretational code is then updated.

In the following statements (up to number 170), the maximum on-scale SCAT measurement is selected from the four output channels (DATA(8) through DATA(11)). To select the appropriate channel output, the channels are tested for an output in excess of a value that would cause the output to saturate on the next higher sensitivity channel. Note that channel 1 (DATA(8)) is least sensitive whereas channel 4 (DATA (11)) is most sensitive. The selected measurement is stored in DATA(8) after scaling it with some channel dependent parameters. If the SCAT return fell below or above the channels, an interpretational flag is set in statements between 170 and 175.

Following the channel selection, the range gate setting is compared with the aircraft altitude. If the altitude is not within 300 feet of the range gate setting, an interpretation flag is set.

Subroutine SCONV (SCAT convert) is then called to compute the normalized input power. Once computed, the scatterometer measurement is inverted for antenna pattern effects with an in-line statement (number 190). The antenna inversion is based on the notion of an equivalent pencil beam replacing the actual main beam

*The implication is that the instrument didn't operate correctly.

antenna pattern. Computer simulations have shown that the method is extremely accurate (errors $\leqslant$ 0.1 dB) for narrow beam antennas. Briefly the equivalent beam-width notion is based on the fact that the normalized input power

$$I(\theta_o) = \iint \frac{P^2(\theta,\phi)\ \sigma^\circ(\theta')\ dA}{R^4}$$

where

= normalized scattering coefficient

= radar range to elemental area dA

= normalized antenna pattern

= elemental area

can be approximated by an algebraic expression through the following considerations. For narrow beam antenna

$$I(\theta_o) = \iint \frac{P^2(\theta,\phi)\ \sigma^\circ(\theta')\ \sin\theta\ R^2\quad d\theta\ d\phi}{R^4\ (\cos\theta')}$$

where $\cos\theta'$ is a projection factor, may be approximated by

$$I(\theta_o) \cong \frac{\sigma^\circ(\theta_o)}{R^2(\cos\theta_o)} \iint P^2(\theta,\phi)\ \sin\theta\ d\theta\ d\phi$$

The integral may be computed numerically from a description of the antenna pattern and equated to the effect produced by a pencil beam of width θeq, i.e.,

$$\iint P^2(\theta,\phi)\sin\theta\ d\theta\, d\phi = \int_0^{2\pi}\int_0^{\theta_{eq}/2} \sin\theta\ d\theta\ d\phi$$

$$\cong \frac{\pi\ \theta_{eq}^2}{4}$$

Therefore the normalized scattering coefficient is given by

$$\sigma^{\circ}(\theta_{\circ}) = \frac{4\ I\ (\theta_{\circ})\ R^{2}\ \cos\theta_{\circ}}{\pi\ \theta^{2}_{eq}}\ .$$

But note that $R = h/\cos\theta_{\circ}$ so that

$$\sigma^{\circ}(\theta_{\circ}) = \frac{4\ I\ (\theta_{\circ})\ h^{2}}{\pi\ \theta^{2}_{eq}\ \cos\theta_{\circ}}$$

where h is the aircraft altitude. The entry θ_{eq} is computed by the engineering routine WIDTH described in Appendix E as well as Section IV F of the text.

The scatterometer and radiometer products are then stored in the output array and the output array is transferred to the output tape (unit 02).

The next data record is then read (statement 100) and the sequence of events repeated. If an end of file is encountered on the input tape, control is returned to SEARCH.

2. Program Listing and Variables

The source listing for CRUNCH is shown in Figures A-6(a) through A-6(d). The variablesemployed in CRUNCH are defined in Table A-2. Entires in SPAR

Table A-2 Definition of Variables Used in CRUNCH

DATA	=	Input Raw Data Vector
IDATA	=	Integer Equivalent of DATA
SPAR	=	SCAT Parameters
RPAR	=	RAD Parameter
SCAL	=	SCAT Calibration Vector (Channel 1 to 4)
RCAL	=	RAD Calibration Vector (Normal and Base)
VERI	=	Data Validation Parameters
IVERI	=	Integer Equivalent of VERI
EOFIFC	=	End of File Indicator for FC 01
EFLAG	=	File Verification Indicator
IL	=	Record Count

```
                SUBROUTINE CRUNCH(IFC,DATA,IDATA,IL,ICOUNT,SCAL,RCAL,
              *         SPAR,RPAR,LNFMOD,EFLAG,EOFIFC)
       CCRUNCH                        CRUNCH
       C
       C               THIS PROGRAM WAS PREPARED BY
       C
       C                   JOHN P. CLAASSEN
       C                   GLEN E. ELLIOTT
       C
       C               UNIVERSITY OF KANSAS CENTER FOR RESEARCH
       C
       C
       C          THE PRIMARY FUNCTION OF THIS ROUTINE IS TO CONVERT SCATTER-
       C          OMETER AND RADIOMETER OUTPUT VOLTAGES INTO THEIR RESPECTIVE
       C          NORMALIZED INPUT POWERS. THE CONVERTED SCAT DATA IS
       C          SUBSEQUENTLY INVERTED FOR THE SCATTERING COEFFICIENT.
       C          PRIMARY COMPUTATIONS ARE PERFORMED IN SUBROUTINE CRUNCH
       C          WHICH CALLS A NUMBER OF IMPORTANT ROUTINES.  FROM CRUNCH
       C          INTERNAL CALIBRATION INFORMATION IS WITHDRAWN FROM THE DATA
       C          RECORDS AND APPLIED TO SUBSEQUENT MEASUREMENTS. THE VIEW
       C          ANGLE IS CORRECTED FOR AIRCRAFT ORIENTATION. A WARNING FLAG
       C          IS GIVEN WHEN THE ROLL ANGLE CAUSES EXCESSIVE DEPOLARIZATION.
       C          A FLAG IS ALSO GENERATED WHEN THE DOPPLER SHIFT EXCEEDS
       C          THE DOPPLER FILTER BANDWIDTH. THERMISTER DATA IS
       C          CHECKED FOR REASONABLE TEMPERATURES AND TEMPERATURES OF
       C          CRITICAL COMPONENTS ARE WITHDRAWN FOR SUBSEQUENT USE IN
       C          CONVERTING RADIOMETER MEASUREMENTS.
       C
000017          DIMENSION DATA(50), IDATA(50), SPAR(85), RPAR(75), SCAL(4),
              *       RCAL(2), ARRAY(20), IARRAY(20), ALT(4)
000017          DIMENSION ICHG(3)
000017          DIMENSION VERI(15,2), IVERI(15,2)
000017          DIMENSION VMAX(15), VMIN(15), IVMAX(15), IVMIN(15)
000017          EQUIVALENCE (VMAX(1),IVMAX(1)),(VMIN(1),IVMIN(1))
000017          EQUIVALENCE (VERI(1,1),VMIN(1)),(VERI(1,2),VMAX(1))
000017          EQUIVALENCE (VERI(1,1),IVERI(1,1))
000017          EQUIVALENCE (ARRAY(1),IARRAY(1))
000017          LOGICAL EOFIFC
000017          DATA (IVMIN(I),I=1,6)/1,0,0,1,0,0/
000017          DATA (IVMAX(I),I=1,6)/3,4,2,6,1,1/
000017          DATA (VMIN(I),I=7,10),(VMAX(I),I=7,10)/4*0.0, 4*10.5/
000017          DATA IVERI(11,1),IVERI(11,2)/0,1/

000017          DATA (VMIN(I),I=12,14),(VMAX(I),I=12,14)/3*0.0, 10.5, 10.5, 10.5/
000017          DATA IVERI(15,1), IVERI(15,2)/0,3/
000017          DATA ICHG/1,0,2/
000017          DATA ALT/2000.0, 5000.0, 10000.0, 20000.0/
       C
       C          READ IN NEXT RECORD
       C
000017          NGR=0
000020    100   IF(IREAD(IFC).NE.0) GO TO 230
000026    101   I=12
       C
       C                     VERIFY CONTENTS OF RECORD
       C
000027          IF(IDATA(I).LT.IVERI(I-1,1).OR.
              *            IDATA(I).GT.IVERI(I-1,2)) GO TO 115
000041          I=16
000042          IF(IDATA(I).LT.IVERI(I-1,1).OR.
              *            IDATA(I).GT.IVERI(I-1,2)) GO TO 115
000054          DO 111 I=2,7
000055    111   IF(IDATA(I).LT.IVERI(I-1,1).OR.
              *            IDATA(I).GT.IVERI(I-1,2)) GO TO 115
000071          DO 112 I=8,11
000072    112   IF(DATA(I).LT.VERI(I-1,1).OR.
              *            DATA(I).GT.VERI(I-1,2)) GO TO 115
000106          DO 113 I=13,15
000107    113   IF(DATA(I).LT.VERI(I-1,1).OR.
              *            DATA(I).GT.VERI(I-1,2)) GO TO 115
000123          GO TO 118
000123    115   WRITE(6,116) I
000131    116   FORMAT(//11X,1H*,11H BAD RECORD,28X,1H*,5X,9HCOMPONENT,I3,
              * 12H IS GARBAGE.//)
000131          WRITE(6,117) DATA
000143    117   FORMAT(/10(5O22/)/)
000143    105   NGR = NGR + 1
000145          IF(NGR.LE.10) GO TO 100
000152          EFLAG = .TRUE.
000153          GO TO 230
       C
       C          BRANCH IF CALIB DATA
       C
000154    118   IF (IDATA(4) .EQ. 0 .OR. IDATA(4).EQ.2) GO TO 220
       C
       C               TRANSLATE CERTAIN ENTRIES IN FILE
```

FIGURE A-6. (a) SOURCE LISTING FOR CRUNCH

```
        C
000163           IDATA(3)=IDATA(3)-1
        *        IJK=IDATA(4)
        *        IDATA(4)=ICHC(IJK+1)
        *        IDATA(6)=1-IDATA(6)
        C
        C                CHECK FOR AGREEMENT IN RECEIVE POL
        C
000165           IF(IDATA(7).EQ.IDATA(12).OR.IDATA(3).EQ.0) GO TO 119
000175           WRITE(6,1000) IDATA(7), IDATA(12)
000212    1000   FORMAT(//11X,1H*,31H RECEIVE POLARIZATIONS DISAGREE,8X,1H*,5X,
               * 9HSCAT REC=,I3,5X,8HRAD REC=,I3//)
000212           GO TO 105
        C
        C                INITIALIZE CHANNEL INDICATOR FLAG AND CLEAR OUTPUT ARRAY.
        C
000216     119   ICHAN=0
000217           DO 120 I=1,20
000221     120   IARRAY(I)=0
000224           IL = IL + 1
        C
        C            CHECK TEMPERATURES OF CRITICAL ELEMENTS
        C
000226           CALL THERMC (DATA,IDATA,RPAR,IL,LNPMOD,IFLAG,IWARM)
        C
        C            FORM ERROR CODE MESSAGE FOR SCAT AND RAD
        C
000234           IARRAY(12) = IWARM
000236           ICOUNT = ICOUNT + IWARM
000242           IARRAY(12) = IARRAY(12)*10 + IFLAG
000245           ICOUNT=ICOUNT + IFLAG
        C
        C                COMPUTE LOOK ANGLES.
        C
000246           CALL ANGLE (DATA,SPAR,IL,LNPMOD,IFLAG)
        C
        C            AMEND ERROR CODE
        C
000252           ICOUNT=ICOUNT+IFLAG/10
000262           IARRAY(12) = IARRAY(12)*100+IFLAG
        C
        C            CONVERT RADIOMETER DATA
        C

000264           CALL RCONV (DATA,IDATA,RPAR,RCAL,IL)
        C
        C            BRANCH IF RAD ONLY DATA
        C
000270           IF (IDATA(3) .EQ. 0) GO TO 200
        C
        C            CHECK DOPPLER
        C
000275           CALL DOPCHK (DATA,SPAR,IL,IBAND,LNPMOD,IFLAG)
        C
        C            AMEND INDICATOR
        C
000301           IARRAY(12) = IARRAY(12) *10+IFLAG
000304           ICOUNT=ICCUNT+IFLAG
        C
        C                     SELECT SCAT CHANNEL OUTPUT
        C
        C
        C            THIS SECTION SELECTS THE MAXIMUM ON-SCALE SCAT CHANNEL OUTPUT
        C            AND TRANSFORMS IT TO A NORMALIZED RECEIVER INPUT POWER.
        C            WHERE READING FALLS OUTSIDE OF DYNAMIC RANGE OF THE FOUR
        C            CHANNELS, THE DATA IS YET SELECTED, HOWEVER, A FLAG
        C            ACCOMPANIES THE VALUE. THE TRANSFORMED VALUE IS FOUND
        C            IN DATA(8).
        C
        C                 INITIALIZE FLAG
        C
000310           IFLAG =0
        C
        C                 DETERMINE WHAT CHANNEL MEASUREMENT FELL IN
        C
000311           IF (DATA (8) .GE. SPAR(29)) GO TO 140
000314           IF (DATA (9) .GE. SPAR(29)) GO TO 150
000317           IF (DATA (10).GE. SPAR(29)) GO TO 160
000322           IF (DATA (11).GE. SPAR(29)) GO TO 130
        C
        C            DATA FELL BENEATH RANGE
        C
000326           DATA(11)=DATA(11)+1.0 E-10
000330           IFLAG =1
        C
        C            APPLY SCALE FACTORS
        C
```

FIGURE A-6. (b) SOURCE LISTING FOR CRUNCH

```
C00331    130   DATA (8)= DATA (11)*SPAR(21)/SCAL(4)
C00336          ICHAN=4
C00337          GO TO 170
              C
              C        DATA FELL ABOVE RANGE
              C
000337    140   IF (DATA(8) .GE. SPAR(27)) IFLAG=1
              C
              C        APPLY SCALE FACTORS
              C
000344          DATA (8)= DATA (8)*SPAR(18)/SCAL(1)
000350          ICHAN=1
000351          GO TO 170
000351    150   DATA (8)= DATA(9)*SPAR(19)/SCAL(2)
000356          ICHAN=2
000357          GO TO 170
C00357    160   DATA (8)= DATA(10)*SPAR(20)/SCAL(3)
C00364          ICHAN=3
              C
              C              END SELECTION OF SCAT CHANNEL.
              C
              C        AMEND ERROR INDICATOR
              C
000365    170   IARRAY(12)=IARRAY(12)*10+IFLAG
000370          ICOUNT=ICOUNT+IFLAG
000371          IF(IFLAG.NE.1) GO TO 175
C00373          IF(MOD(IL,LNFMCD).EQ.0) WRITE(6,1700) IL
000410   1700   FORMAT(I6,5X,2H* 22HEXCEEDED DYNAMIC RANGE,16X,1H*)
              C
              C              CHECK RANGE GATE SETTING
              C
000410    175.  IFLAG = 0
C00411          IALT = IDATA(16)+1
000414          CHECK=ALT(IALT)
C00416          IF(DATA(41) .LT. CHECK+300. .AND. DATA(41) .GT. CHECK-300.)
             *          GO TO 177
C00431          IFLAG = 1
000432          WRITE(6,1760) CHECK,DATA(41)
C00445   1760   FORMAT(I6,5X,2H* ,32HNOT WITHIN 300 FT. OF RANGE GATE,6X,1H*,
             *  5X, 11HRANGE GATE=,F7.0,5X,9HALTITUDE=,F7.0)
C00445    177   IARRAY(12) = IARRAY(12)*10 + IFLAG
C00450          ICOUNT=ICOUNT+IFLAG
              C

              C              COMPLETE CONVERSION OF SCAT DATA
              C
000454    180   CALL SCONV (DATA,IDATA,SPAR,IBAND)
              C
              C              SCAT INVERSION.
              C
              C
              C              COMPARE SCAT POLARIZATIONS.
              C
C00457          IXMT=IDATA(6)
C00464          IREC=IDATA(7)
000466          IF(IXMT.EQ.IREC) GO TO 190
              C
              C              ERROR MESSAGE
              C
000467          IF(MOD(IL,LNFMCD).EQ.0) WRITE(6,1800) IL, IXMT, IREC
000511   1800   FORMAT(I6,5X,2H* ,37HUNABLE TO HANDLE CROSS POLARIZED DATA,
             *          1X,2H* ,4X,26HWILL ASSUME POLARIZED DATA, 5X,
             *          10HXMIT POL.=,I2,5X,10HREC. POL.=,I2)
              C
              C              INVERT SCAT DATA
              C
000511    190   DATA(8) = 4.0*DATA(8)*DATA(41)*DATA(41)/(COS(0.017453293*
             *            DATA(14))*3.14159265*SPAR(IXMT+32)*SPAR(IREC+32)*
             *            9.290304)
              C
              C              TRANSFER SCAT INT PERIOD
              C
000533          IND=IDATA(5)
C00535          ARRAY(7)=SPAR(IND+3)
000537          IF(IDATA(3).EQ.1) ARRAY(7)=SPAR(10)
              C
              C        TRANSFER SCAT DATA TO OUTPUT ARRAY
              C
000544          IARRAY(5) =IDATA (6)
C00546          IARRAY(6) =IDATA (7)
C00547          ARRAY(8) = DATA (8)
000551          ARRAY(9) = 10.0*ALOG10(DATA(8))
C00562          GO TO 210
              C
              C        TRANSFER RAD DATA TO OUTPUT ARRAY
              C
C00562    200   IARRAY(12)=IARRAY(12)*1000
```


FIGURE A-6. (c) SOURCE LISTING FOR CRUNCH

```
000565   210   ARRAY (10)= DATA (13)
         C
         C       TRANSFER OTHER SELECTED DATA
         C
000567         ARRAY(1)=DATA(1)
000570         IARRAY(2)=IDATA(3)
000572         ARRAY(3)=DATA(14)
000573         ARRAY(4)= DATA(15)
000575         ARRAY(11)=DATA(40)
000576         ARRAY (13)= DATA(41)
000600         ARRAY (14)= DATA (47)
000601         ARRAY (15)= DATA (46)
         C
         C               WRITE TO OUTPUT TAPE
         C
000603         CALL WRITE2(ARRAY)
         C
         C               INTERMEDIATE PRINT-OUT
         C
000604         IF(MOD(IL,LNPRCD).EQ.0) WRITE(6,2000) IL, ARRAY(10), ARRAY(8),
              *        ICHAN,IBAND
000643   2000  FORMAT(I6,5X,2H**,4X,29HCOMPLETED RECORD COMPUTATIONS,4X,2H**,
              *        4X,14H ANTENNA TEMP=,F6.1,2X,16HSCATTERING COEF=,E12.6,
              *        2X,8HCHANNEL=,I2,2X,9HDOP BAND=,I3)
000643         GO TO 100
000644   220   CALL CALIB(SCAL,RCAL,RPAR,NORMAL,BASE)
000650         IF(.NOT.ECFIFC) GO TO 100
000656   230   WRITE(6,2000) IL, ARRAY(10), ARRAY(8), ICHAN, IBAND
000702         RETURN
000703         END
```

FIGURE A-6. (d) SOURCE LISTING FOR CRUNCH

Table A-2 Definition of Variables Used in CRUNCH (Continued)

ICOUNT	=	Number of Interpretational Flags
LNPMOD	=	Line Print Modulus
IFC	=	Unit Code Number (1 or 8)
NGR	=	Number of Bad Records
ALT	=	Range Gate Settings
CHECK	=	One of Gate Settings
IXMT	=	Transmit Polarization Indicator
IREC	=	Receive Polarization Indicator
IND	=	SCAT Integration Selector
ARRAY	=	Output Array
IARRAY	=	Integer Equivalent

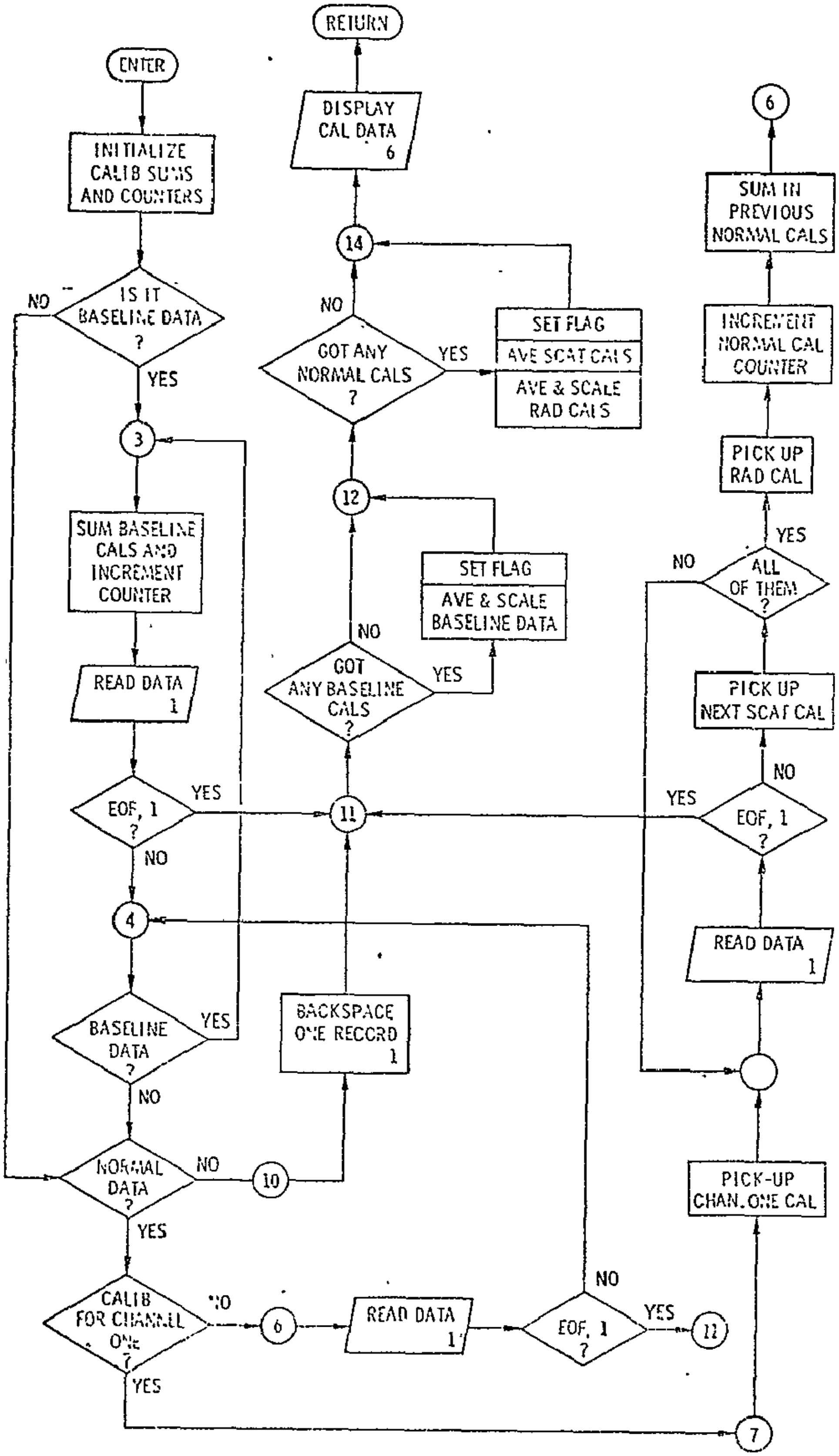
and RRAR are defined in Section III D; whereas, the entries in ARRAY are defined in Section III C.

C. Subroutine CALIB

1. Theory and Design

Subroutine CALIB extracts, averages and displays calibration parameters from calibration records. CALIB was designed to anticipate calibration records in a format generated by the RADSCAT instrument. Although additional flexibility is provided so that baseline records, which are entered by hand, may be inserted in almost any fashion. The baseline records may occur anywhere in the file except within a group of four normal calibration records. In regards to sequences of normal calibrations, the subroutine will orient itself on a set of four by interrogating whether the first normal calibration record causes SCAT outputs to saturate in channels 2, 3 and 4. This feature was embedded in the routine when it became apparent that the RADSCAT instrument would occasionally drop calibrations on the first few SCAT channels. A descriptive logic diagram for subroutine CALIB is shown in Figure A-7.

The baseline (BCAL) and normal calibration counters are initialized upon entry into the routine. Also the SCAT channel (CAL1(I), I = 1,4) and RAD accumulators (CAL1(5), RCL) are cleared. If the type of calibration record (normal or baseline, see Section III B) is normal (IDATA(4) = 0), processing is directed to statements including and following 4. Otherwise, if it is a baseline record, it is accumulated in RCL, the baseline counter is incremented, and the next record is



ORIGINAL PAGE IS OF POOR QUALITY

FIGURE A-7. DESCRIPTIVE LOGIC DIAGRAM FOR SUBROUTINE CALIB read. In statement 4 and the one following, the type of record is again determined. If additional baselines records are present (IDATA(4) = 2), the above processing is repeated; if it is normal calibration data, the record is examined to determine whether the calibration indeed occurred on channel 1 (see DO loop terminating in statement 5). If not, the next record is read and the type is again determined. Once the program is aligned on the calibration for channel one, it extracts it in statement 7 and pulls the SCAT calibrations from the subsequent records for channels 2, 3 and 4 and the normal RAD calibration from the last SCAT calibration record (Channel 4). The normal cal counter is incremented and the calibration accumulator update.

The next record is read and the type determined again. If it is a calibration record the above steps are repeated. If not (measurement record), the unblocking routine READ1 is called to backspace a record (statement 10). The steps 11 through 60 determine what kinds of calibrations were present and how many. The accumulated calibrations are averaged and scaled accordingly. The types of calibrations present are reflected by setting the logical variables NORMAL and BASE to TRUE. The calibration parameters are then printed and control is returned to the calling routine, either SEARCH or CRUNCH.

2. Program Listing and Variables

The source listing for CALIB is shown in Figure A-8. The definitions of the variables are listed in Table A-3.

Table A-3 Variables Used in CALIB

DATA	=	Input Raw Data Vector
IDATA	=	Integer Equivalent of DATA
SCAL	=	SCAT Calibration Vector
RCAL	=	RAD Calibration Vector
RPAR	=	RAD Parameters
NORMAL	=	Normal Cal Indicator
BASE	=	Baseline Cal Indicator
BCAL	=	Baseline Cal Record Counter
CAL	=	Normal Cal Record Counter
RCAL	=	Baseline Cal Accumulator
CAL1	=	Normal Cal Accumulator

```
                 SUBROUTINE CALIB(SCAL,RCAL,RPAR,NORMAL,BASE)
          CCALIB                       CALIBRATION SUBROUTINE
          C
          C                 THIS PROGRAM WAS PREPARED BY
          C
          C                      JOHN P. CLAASSEN
          C                      GLEN E. ELLIOTT
          C
          C                 UNIVERSITY OF KANSAS CENTER FOR RESEARCH
          C
          C            THIS SUBROUTINE EXTRACTS CALIBRATION INFORMATION FROM
          C            THE RADSCAT DATA FILE AND DISPLAYS IT.
          C
000010           COMMON /INPUT/ EOF1, DATA
000010           LOGICAL NORMAL, BASE, EOF1
000010           DIMENSION DATA(50), IDATA(1), SCAL(1), RCAL(1), RPAR(1)
000010           DIMENSION CAL1(5), CAL2(5)
000010           EQUIVALENCE (DATA(1),IDATA(1))
          C
          C                 INITIALIZE CAL COUNTERS
          C
000010           BCAL=0.0
000011           RCL=0.0
000012           CAL=0.0
          C
          C                 CLEAR CAL ACCUMULATORS
          C
000013           DO 1 I=1,5
000014     1     CAL1(I)=0.0
          C
          C                 BRANCH WHEN NOT BASELINE
          C
000017     2     IF(IDATA(4).NE.2) GO TO 4
          C
          C                 EXTRACT BASELINE
          C
000021     3     RCL=RCL+DATA(13)
          C
          C                 UPDATE BCAL COUNT
          C
000023           BCAL=BCAL+1.0
          C
          C                 READ NEXT RECORD
          C
000025           CALL READ1
000026           IF(EOF1) GO TO 11
000033     4     IF(IDATA(4).EQ.2) GO TO 3
          C
          C                 EXIT WHEN NO MORE CAL DATA
          C
000035           IF(IDATA(4).NE.0) GO TO 10
          C
          C                 CHECK IF RECORD CONTAINS CAL ON
          C                 FIRST SCAT CHANNEL
          C
000036           DO 5 I=2,4
000037     5     IF(DATA(I+7).LT.9.0) GO TO 6
000044           GO TO 7
          C
          C                 FIND FIRST SCAT CAL
          C
000044     6     CALL READ1
000045           IF(EOF1) 11,4
          C
          C                 EXTRACT SCAT 1
          C
000053     7     CAL2(1)=DATA(8)
          C
          C                 EXTRACT REMAINDER OF CALS
          C
000055           DO 8 I=2,4
000056           CALL READ1
000057           IF(EOF1) GO TO 11
000064    15     IF(IDATA(4).NE.0) GO TO 4
000065     8     CAL2(I)=DATA(I+7)
000071           CAL2(5)=DATA(13)
000073           CAL=CAL+1.0
          C
          C                 ACCUMULATE REPETITIVE CALS
          C
000075           DO 9 I=1,5
000076     9     CAL1(I)=CAL1(I)+CAL2(I)
000102           GO TO 6
          C
          C                 REPOSITION DATA RECORD
```

ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE A-8. (a) FORTRAN LISTING FOR CALIB

```
              C
  000102    10     CALL BCKSP1
              C
              C              CHECK FOR PRESENCE OF BASELINE CAL
              C
  000103    11     IF(BCAL.LT.1.0) GO TO 12
  000111           BASE = .TRUE.
              C
              C              AVERAGE AND TRANSLATE BASELINE
              C
  000111           RCAL(1)=(RCL/BCAL)*RPAR(19)/RPAR(20)
              C
              C              CHECK FOR PRESENCE OF NORMAL CAL
              C
  000115    12     IF(CAL.LT.1.0) GO TO 14
  000120           NORMAL = .TRUE.
  000120           DO 13 I=1,4
              C
              C              AVERAGE SCAT CALS
              C
  000122    13     SCAL(I)=CAL1(I)/CAL
              C
              C              AVERAGE AND TRANSLATE RAD CAL
              C
  000126           RCAL(2)=(CAL1(5)/CAL)*RPAR(19)/RPAR(20)
              C
              C              DISPLAY CAL DATA
              C
  000132    14     WRITE (6,100) (I,SCAL(I),I=1,4),RCAL(2), RCAL(1), CAL, BCAL
  000167   100     FORMAT (11X,1H*10X,16HCALIBRATION DATA,13X,1H*,3X,
                *          4(2X,4HCHAN,I2,1H=,F7.3),
                *          /51X,1H*,5X,5HRCAL=F7.3,2X,6HRBASE=,F7.3,
                *          7H  NCAL=,F6.0,9H   NBCAL=,F6.0)
  000167    60     RETURN
  000170           END
```

ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE A-8. (b) (continued)

D. Subroutine ANGLE

1. Introduction

This routine computes the incident angle and cross track angle both of which are required to define the beam position on the surface with respect to the aircraft flight vector. When roll or pitch becomes excessive, as the case may be, so as to make the polarization difficult to interpret a flag is generated. Irrespective of the flagged condition the data is reduced in the standard way. The theory by which these angles are computed and the method by which the angles are applied to determine the polarization break-up at the surface are described below.

2. Theory and Design

a. Incident and Cross-Track Angles - Suppose the aircraft is vectored along the positive x axis of an unprimed coordinate system where the z axis corresponds to the local vertical. The orientation of the aircraft is represented in a primed coordinate system where x' axis is located along the fore-aft axis of the aircraft. They are related to the unprimed system by a drift angle ϕ_d about the z axis, a pitch angle θ_p with respect to the x, y plane and a roll angle ϕ_r about the x' axis (Figure A-9).*

The antenna points at an angle θ_a with respect to the $-z'$ axis. θ_a is assumed to be corrected for the relative angle between aircraft and antenna platforms. The true incident angle θ on the ground is desired. The incident angle may be derived from relation

$$\cos\theta = -\bar{\iota}_r \cdot \bar{\iota}_z \tag{1}$$

where τ_r is a unit vector in the boresight direction. Also of interest is the cross track angle Φ_c measured in the x,y plane with respect to the -x axis as shown in Figure A-10. Clearly we have

$$\tan\phi_c = \frac{\bar{\iota}_r \cdot \bar{\iota}_y}{-\bar{\iota}_r \cdot \bar{\iota}_x} \tag{2}$$

*It should be noted that these definitions of roll, pitch and drift coincide with those provided by the Litton Navigator LTN-51.

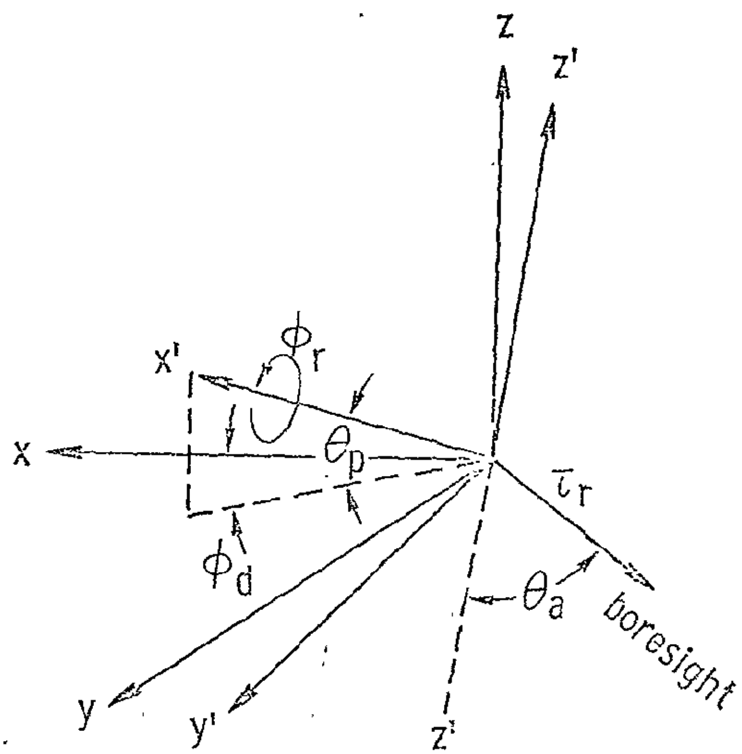


Figure A-9. Aircraft Flight Geometry

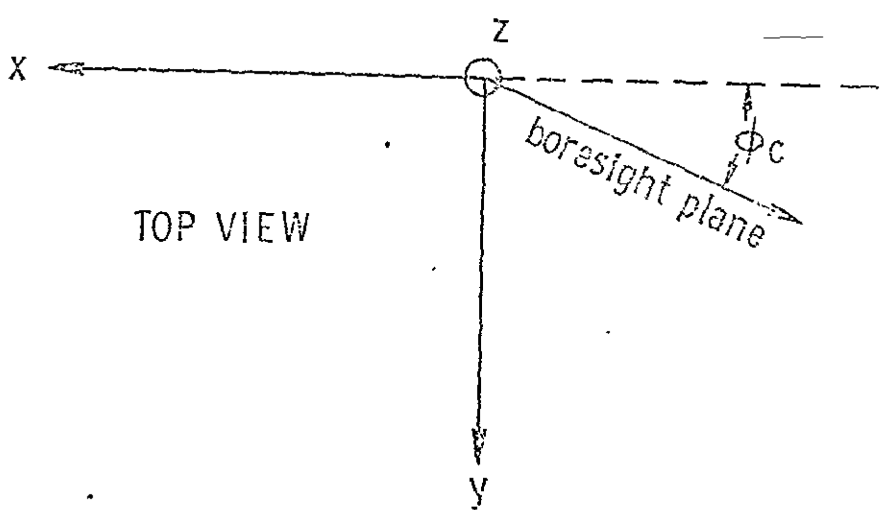


Figure A-10. Geometry Defining the Cross-Track Angle.

Now in the primed coordinate system the boresight axis is described by

$$\bar{\iota}_r = - \sin \theta_a \, \bar{\iota}_{x'} - \cos \theta_a \, \bar{\iota}_{z'} \tag{3}$$

The primed and unprimed coordinate systems are related by the following successive transformation

$$\begin{pmatrix} \bar{i}_{x'} \\ \bar{i}_{y'} \\ \bar{i}_{z'} \end{pmatrix} = ABC \begin{pmatrix} \bar{i}_{x} \\ \bar{i}_{y} \\ \bar{i}_{z} \end{pmatrix} \tag{4}$$

where

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi_r & \sin\phi_r \\ 0 & -\sin\phi_r & \cos\phi_r \end{pmatrix}$$

$$B = \begin{pmatrix} \cos\theta_p & 0 & \sin\theta_p \\ 0 & 1 & 0 \\ -\sin\theta_p & 0 & \cos\theta_p \end{pmatrix}$$

$$C = \begin{pmatrix} \cos\phi_d & \sin\phi_d & 0 \\ -\sin\phi_d & \cos\phi_d & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Substitution of (3) and (4) into (1) yields

$$\cos\theta = \sin\theta_p \sin\theta_a + \cos\theta_p \cos\theta_a \cdot \cos\phi_r \tag{5}$$

which reduces to the expected result when $\phi_r = 0$, namely, $\cos(\theta_p + \theta_a)$.

When the drift angle is considered zero, a similar approach will yield

$$\tan\phi_c\,(\phi_d = 0) = \frac{\sin\phi_r \cos\theta_a}{\sin\theta_a \cos\theta_p - \sin\theta_p \cos\theta_o \cos\phi_r} \tag{6}$$

When $\phi_d = 0$, it is clear from Figure A-11 that the correction for ϕ_d is simply given by

$$\phi_c(\phi_d = 0) = -[\phi_d - \phi_c(\phi_d = 0)] \tag{7}$$

or

$$\phi_c(\phi_d = 0) = -\phi_d + \tan^{-1}\left[\frac{\sin\phi_r \cos\theta_a}{\sin\theta_a\cos\theta_p - \sin\theta_p\cos\theta_a\cos\phi_r}\right] \tag{8}$$

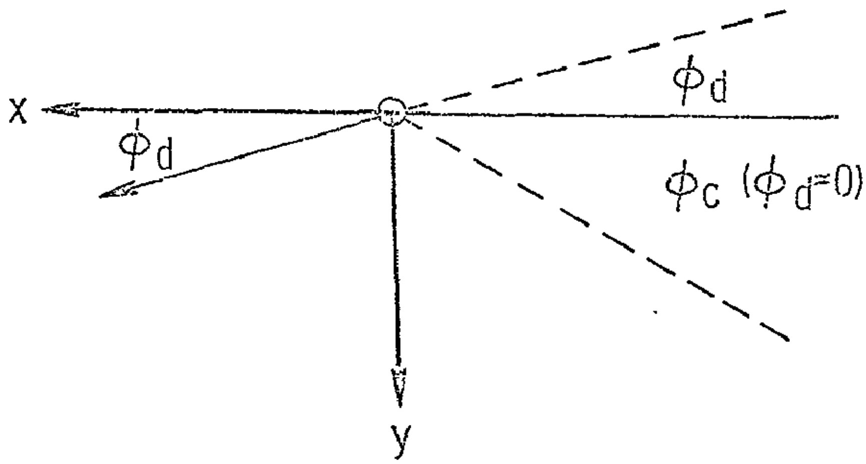


Figure A-11. Cross-Track Angle Correction for Drift.

Equations 5 and 8 thus define the beam position on the sea with respect to the aircraft trajectory.

b. Polarization Decomposition - When the aircraft pitches and rolls, the incident polarization can decompose into vertically and horizontally polarized components. When the unwanted component becomes excessive, the measurement becomes difficult to interpret; and, consequently, the reduced data should be flagged.* To determine the size of the undesired component, the degree of "de-polarization" is computed from considerations similar to those above.

*Correction for this situation is possible when polarized measurements are considered jointly.

Without loss of generality we may assume that the aircraft is vectored along the -x axis with zero drift. It is pitched at an angle of θ_p with respect to the -x axis and a positive roll of ϕ_r is induced about the -x' axis. The x,y plane forms the local horizontal and the z axis is pointed at nadir. See Figure A-12.

The antenna is boresighted in the x', z' plane with an angle θ_a from the z' axis. The vertical polarization emitted by the antenna is described by

$$\bar{\iota}_{\theta'} = \cos\theta_a \, \bar{\iota}_{x'} - \sin\theta_a \, \bar{\iota}_{z'} \tag{9}$$

along the boresight axis. The horizontal surface polarization at the boresight point is described as

$$\bar{\iota}_{\phi} = -\sin\phi_c \, \bar{\iota}_x + \cos\phi_c \, \bar{\iota}_y \tag{10}$$

where ϕ_c corresponds to the cross-track angle for the zero drift condition. Now the percent power de-polarization is approximately given by $100 \, |\bar{\iota}_{\theta'} \cdot \bar{\iota}_{\phi}|^2$ for points illuminated by the main beam.

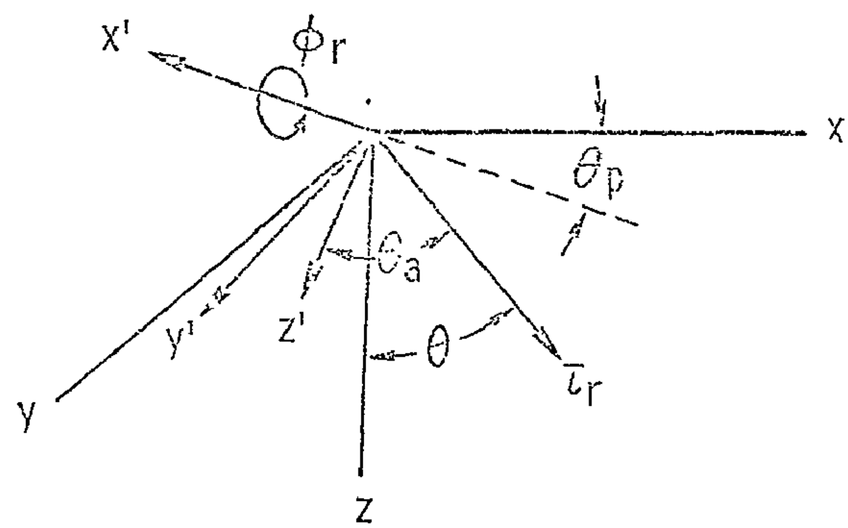


Figure A-12. "Depolarization" Geometry.

Now the transformation between the coordinates is given by

$$\begin{pmatrix} \bar{i}_{x'} \\ \bar{i}_{y'} \\ \bar{i}_{z'} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi_r & -\sin\phi_r \\ 0 & \sin\phi_r & \cos\phi_r \end{pmatrix} \cdot \begin{pmatrix} \cos\theta_p & 0 & \sin\theta_p \\ 0 & 1 & 0 \\ -\sin\theta_p & 0 & \cos\theta_p \end{pmatrix} \begin{pmatrix} \bar{i}_x \\ \bar{i}_y \\ \bar{i}_z \end{pmatrix} \tag{11}$$

or

$$\begin{pmatrix} \bar{i}_{x'} \\ \bar{i}_{y'} \\ \bar{i}_{z'} \end{pmatrix} = \begin{pmatrix} \cos\theta_p & 0 & \sin\theta_p \\ \sin\phi_r \sin\theta_p & \cos\phi_r & -\sin\phi_r \cos\theta_p \\ -\cos\phi_r \sin\theta_p & \sin\phi_r & \cos\phi_r \cos\theta_p \end{pmatrix} \begin{pmatrix} \bar{i}_x \\ \bar{i}_y \\ \bar{i}_z \end{pmatrix} \tag{12}$$

With the help of (12) we may thus write

$$\begin{aligned} |\bar{i}_{\theta'} \cdot \bar{i}_\phi| = |&\sin\phi_c(\cos\theta_a \cos\theta_p + \sin\theta_a \cos\phi_r \sin\theta_p) \\ &+ \cos\phi_r \sin\theta_a \sin\phi_r| \end{aligned} \tag{13}$$

Now a suitable criterion for excessive polarization may be formed by requiring that

$$|\bar{i}_{\theta'} \cdot \bar{i}_\phi|^2 < \epsilon \tag{14}$$

where $0 < \epsilon << 1$. However, when the aircraft roll becomes excessive so that $|\bar{i}_{\theta'} \cdot \bar{i}_\phi|^2 > 0.5$ the interpretation of the incident polarization should be reversed from that defined by the instrument. Under this circumstance the polarization criterion should be written as

$$1 - |\bar{i}_{\theta'} \cdot \bar{i}_\phi|^2 < \epsilon \tag{15}$$

where $0 < \epsilon << 1$. This latter criterion should be useful when RADSCAT observations are conducted from an aircraft in a roll (blank) attitude.

From Section IV.B.1 we noted that for zero drift

$$\tan \phi_c = \frac{\sin \phi_r \ \cos \theta_a}{\sin \theta_a \cos \theta_p + \sin \theta_p \cos \theta_a \cos \phi_r} \tag{16}$$

The drift angle does not influence the polarization decomposition as a little thought will substantiate. This is fortunate since drift angle is not well defined in a roll maneuver. As a consequence the criterion is now formed on the basis of measured and computed parameters.

Relationships (13), (14), (15) and (16) establish the depolarization parameters and the polarization reversal criterion.

3. Program Listing and Variables

The FORTRAN listing for ANGLE is shown in Figure A-13. The comment statements are sufficient to establish the logic of the program. The variables defined in Table A-4 should be helpful.

Table A-4 Definition of Variables in Subroutine ANGLE

DATA	=	Input Raw Data Vector
SPAR	=	SCAT Parameters
IL	=	Record Count
LNPMOD	=	Line Print Modulus
IFLAG	=	Excessive Depolarization and Reversal Flag
DEGRAD	=	Degress to Radians Conversion Factor
ANTA	=	Antenna Angle
PITCH	=	Aircraft Pitch
DRIFT	=	Aircraft Drift
ROLL	=	Aircraft Roll
THETA	=	Incident Angle
ARG1,ARG2	=	Intermediate Storage
PHI	=	Cross-track Angle
DEPOL	=	Depolarization Factor

```
                 SUBROUTINE ANGLE (DATA,SPAR,IL,LAPPOU,IFLAG)
            CANGLE                    COMPUTE INCIDENCE ANGLE
            C          SUBROUTINE ANGLE (DATA,SPAR,IL,LAPPOU,IFLAG)
            C
            C          THIS PROGRAM WAS PREPARED BY
            C
            C              JOHN P. CLAASSEN
            C              GLEN E. ELLIOTT
            C
            C          UNIVERSITY OF KANSAS CENTER FOR RESEARCH
            C
            C      THIS ROUTINE COMPUTES THE INCIDENT ANGLE THETA FROM AIRCRAFT
            C      AND ANTENNA PARAMETERS. CORRECTION IS MADE FOR THE RELATIVE
            C      ANGLE BETWEEN THE AIRCRAFT AND ANTENNA PLATFORM.
            C      THE BORESIGHT CROSS-TRACK ANGLE PHI IS ALSO COMPUTED.
            C      WHEN THE ROLL OR PITCH ANGLE BECOMES EXCESSIVE, AS THE CASE
            C      MAY BE, SO AS TO MAKE THE RETURN OR EMISSION POLARIZATION
            C      DIFFICULT TO INTERPRET, A FLAG IS GENERATED.
            C
C00010            DIMENSION DATA(1), SPAR(1)
            C
            C      DEGREE-RADIAN CONVERSION FACTOR.
            C
C00010            DATA DEGRAD/0.0174532925/
            C
            C      INITIALIZE FLAG
            C
000010            IFLAG=0
            C
            C      AVERAGE SCAT AND RAD ANGLES
            C
C00010            DATA(14)=(DATA(14)+DATA(15))/2.0
            C
            C      CONVERT TO DEGREES
            C
C00013            DATA(14)=DATA(14)*SPAR(30)
            C
            C      COMPENSATE FOR  ANTENNA PITCH RELATIVE TO AIRCRAFT
            C
C00015            DATA(14)=DATA(14)+SPAR(31)
            C
            C      CONVERT TO RADIANS
            C
C00017            ANTA= DATA(14)*DEGRAD
C00021            PITCH=DATA(43)*DEGRAD
C00022            DRIFT=DATA(44)*DEGRAD
C00024            ROLL =DATA(45)*DEGRAD
            C
            C      COMPUTE INCIDENT ANGLE
            C
000025            THETA = SIN(ANTA)*SIN(PITCH)+COS(PITCH)*COS(ANTA)*COS(ROLL)
C00043            THETA = ACOS(THETA)
            C
            C      CONVERT TO DEGREES.
            C
C00046            DATA(14)=THETA/DEGRAD
            C
            C      COMPUTE CROSS TRACK ANGLE
            C
C00052            ARG1=SIN(ROLL)*COS(ANTA)
C00057            ARG2=SIN(ANTA)*COS(PITCH)-SIN(PITCH)*COS(ANTA)*COS(ROLL)
C00075            IF (ABS(ARG2).LT. 1.E-07) GO TO 30
C00104            IF(ABS(ARG1).LT.1.0E-07) GO TO 10
C00107            PHI=ATAN2(ARG1,ARG2)
000112            GO TO 40
000115    10      PHI=0.
000116            IF(ARG2.LT.0.0) PHI=3.14159265
C00121            GO TO 40
000122    30      PHI=SIGN(1.57079633,ARG1)
C00125    40      DATA(15)=PHI/DEGRAD-DATA(44)
            C
            C      COMPUTE THE DEPOLARIZATION
            C
C00130            DEPOL= SIN(PHI)*
                 *      (COS(ANTA)*COS(PITCH)+SIN(ANTA)*COS(ROLL)*SIN(PITCH))+
                 *      COS(PHI)*SIN(ANTA)*SIN(ROLL)
            C
            C      CHECK FOR POLARIZATION REVERSALS.
            C
000162            DEPOL = DEPOL*DEPOL
000163            IF(DEPOL.GT.0.500) GO TO 50
000172            GO TO 60
C00172    50      DEPOL = 1.0 - DEPOL
            C
            C          FLAG REVERSAL
```


FIGURE A-15. (a) FORTRAN LISTING FOR SUBROUTINE ANGLE

```
        C
000174         IF(MOD(IL,LNPMCD).EQ.0) WRITE(6,500) IL
000210    500  FORMAT(I6,5X,2H* 21HPOLARIZATION REVERSAL,17X,1H*)
000210         IFLAG=1
        C
        C             FLAG EXCESSIVE DEPOL.
        C
000211    60   IF(DEPOL.GT.0.02) GO TO 70
000215         RETURN
000215    70   IFLAG=IFLAG+10
000216         IF(MOD(IL,LNPMCD).EQ.0) WRITE(6,700) IL, DEPOL
000235    700  FORMAT(I6,5X,2H* 24HEXCESSIVE DEPOLARIZATION,14X,1H*,
             *        5X,6HDEPOL=,F7.3)
000235         DATA(40)=DEPOL
000237         RETURN
000240         END
```


FIGURE A-13. (b) FORTRAN LISTING FOR SUBROUTINE ANGLE

E. Subroutine THERMO

Temperatures within the RADSCAT microwave assembly are monitored at various points. Certain of these temperatures are essential to the reduction of the radiometer data. All of the temperatures should be checked to determine whether the instrument is operating within its temperature limits.

The temperatures are monitored by means of thermistors. Above 283° the thermistor output voltages are linearly proportional to temperature. This routine converts the recorded thermistor voltages, DATA(17) through DATA(33), to degrees Kelvin by using linear conversion constants, RPAR(23) through RPAR(56). Once converted the computed temperatures are examined to see that they appear in the operating temperature range specified by RPAR(21) and RPAR (22). Those that fall outside this range are flagged with a message in the Conversion Report. A flag is also generated in the S/R validation code appearing in the output files. Routinely the temperatures critical to the conversion of radiometer data are reported by this routine.

The above steps are reflected in the FORTRAN listing for THERMO shown in Figure A-14. The variables employed in the routine are defined in Table A-5.

Table A-5 Definition of Variables Used in THERMO

DATA	=	Raw Data Vector
IDATA	=	Integer Equivalent of DATA
RPAR	=	Radiometer Parameters
IL	=	Record Number
LNPMOD	=	Line Print Modulus
IFLAG	=	Abnormal Receiver Temperature Flag
IWARM	=	Warm Load Temperature Estimate Flag

F. Subroutine RCONV

1. Introduction

This routine converts the radiometer output voltage into an antenna (input) temperature. Radiometer baseline calibration RCAL(1) and normal calibrations RCAL(2) are employed to properly scale the output voltage DATA(13) to a receiver input temperature. Thermal noise contributions from front-end elements are subtracted

```
              SUBROUTINE THERMO (DATA,IDATA,RPAR, IL,LNPMOD,IFLAG,IWARM)
        C
        CTHERMO                          CHECKS TEMPERATURE
        C
        C                 THIS PROGRAM WAS PREPARED BY
        C
        C                        JOHN P. CLAASSEN
        C                        GLEN E. ELLIOTT
        C
        C                 UNIVERSITY OF KANSAS CENTER FOR RESEARCH
        C
        C            THIS ROUTINE EXAMINES ATM VOLTAGES AND CONVERTS THEM TO
        C            TEMPERATURES. THE TEMPERATURE VALUES ARE CHECKED TO SEE
        C            IF THEY OCCUR IN THE PERMISSIBLE OPERATING RANGE.  WHEN
        C            THEY FALL OUTSIDE THE RANGE A FLAG IS GENERATED AND AN ERROR
        C            MESSAGE DISCLOSING THE BAD VALUE AND MONITORING LOCATION
        C            IS PRINTED. THE VOLTAGE TO TEMPERATURE PARAMETERS ARE
        C            ENTERED FROM RPAR. THE STARRED STATEMENT IS TO BE
        C            USED WHEN THE HOT LOAD TEMP IS AVAILABLE.
        C
C00012        DIMENSION DATA(1), RPAR(1), IDATA(1)
        C
        C                 INITIALIZE FLAGS
        C
C00012        IFLAG=0
C00012        IWARM = 0
        C
        C            CONVERT TO TEMPERATURES
        C
C00013        J = 22
C00014        DO 20 I= 1,17
C00016        K=I+J
C00020        DATA(I+16)=RPAR(K)*DATA(I+16)+RPAR(K+1)+273.18
C00025        T=DATA(I+16)
C00027        J = J+1
C00031        IF(I.EQ.17) GO TO 20
C00033        IF (T.LT.RPAR(21) .AND. T.GT. RPAR(22)) GO TO 20
        C
        C            FLAG AND PRINT MESSAGE WHEN NOT IN LIMITS.
        C
C00043   10   IFLAG=1
C00044        IF(MOD(IL,LNPMOD).EQ.0) WRITE (6,100) IL,I,T

C00066  100   FORMAT(I6,5X,41H* RECVR TEMP NOT WITHIN OPERATING RANGE *,
             *        5X,12HLOCATION NO. I3,5X,
             *        13HTEMPERATURE =,F5.1,7H KELVIN)
C00066   20   CONTINUE
        C
        C                 CHANGE AIR TEMP TO KELVIN
        C
C00070        DATA(46) = DATA(46) + 273.18
        C
        C                 CHECKS FOR ESTIMATED WARM LOAD TEMPERATURE
        C
C00072        IF (IDATA(36) .EQ. 0)  GO TO 30
C00074        IWARM = 1
C00074        IF (MOD(IL,LNPMOD) .EQ. 0)  WRITE(6,150)  IL
C00113   150  FORMAT (I6,5X,2H* ,31HWARM LOAD TEMPERATURE ESTIMATED,7X,,1H*)
        C
        C                 LIST CRITICAL TEMPERATURES
        C
C00113    30  IF(MOD(IL,LNPMOD).EQ.0) WRITE (6,200) IL, DATA(27), DATA(33),
             *        DATA(32), DATA(28), DATA(46), DATA(25)
C00167  200   FORMAT (I6,5X,2H* 21HCRITICAL TEMPERATURES,17X,
             *        1H*,5X,5HWARM=,F6.1,2X,4HHOT=,F6.1,2X,4HOMT=,F6.1,
             *        2X,3HSW=,F6.1,2X,4HAIR=,F6.1,2X,4HGDE=,F6.1)
C00167        RETURN
C00170        END
```

FIGURE A-14. FORTRAN LISTING FOR SUBROUTINE THERMO out to refer the receiver input temperature to the antenna terminals with the help of conversion constants RPAR and temperatures of input elements, DATA (17) to DATA (33).

2. Theory and Design

The radiometer conversion model was developed in TM 186-3.* In view of the involvement in the derivation, only an outline of the derivation will be presented here.

During baseline calibration an output proportional to a zero receiver intpu temperature is generated

$$V_b = -\frac{\tau_c}{RC}\left[V_r + \frac{G}{4(T_{1r}-T_{2r})}(T_{1r}+T_{2r}-T_{1r}-T_{2r})\right] \quad (1)$$

where
τ_c = calibration integration period
RC = output integrator time constant
G = receiver gain established by AGC circuit
V_r = output reference bias
T_{1r} = hot load temperature referred to receiver input
T_{2r} = warm load temperature referred to receiver input

During a normal calibration sequence the output integration produces a voltage given by

$$V_n = -\frac{\tau_c}{RC}\left[V_r - \frac{G}{4(T_{1r}-T_{2r})}(2T_{1r}-2T_{2r})\right] \quad (2)$$

This result yields the unknown gain factor

$$G = \frac{2RC}{\tau_c}(V_n - V_b) \quad (3)$$

where (1) has been employed. A measurement cycle yields an output voltage given by

$$V_m = -\frac{\tau}{RC}\left[V_r + \frac{G(2T_{ar}-T_{1r}-T_{2r})}{4(T_{1r}-T_{2r})}\right] \quad (4)$$

---

*Claassen, J. P., "The RADSCAT Radiometer Transfer Function and Its Application to the Reduction of RADSCAT Data," University of Kansas Center for Research, Inc. Lawrence, Kansas, December 1971.

C, 2

where T_{ar} is the temperature at the antenna port of the Dicke switch. This result may be written as

$$T_{ar} = \frac{T_{1r} + T_{2r}}{2} - \rho_m \left(T_{1r} - T_{2r} \right) \qquad (5)$$

where

$$\rho_m = \frac{V_m - V_b \, \tau/\tau_c}{(V_n - V_b)\, \tau/\tau_c} \qquad (6)$$

and where (1) and (3) have been employed. From pages 8-11 of TM 186-3 it is noted that certain leakages at the Dicke switch require that certain terms in (5) be modified with reflection factors R_a and R_c so that the modeling equation actually becomes

$$T_{ar}(1 - R_L) = \frac{T_{1r} + T_{2r}}{2}(1 - R_a) - \rho_m \left(T_{1r} - T_{2r} \right) \qquad (7)$$

Now the temperature at the antenna port of the Dicke switch can be described by

$$\begin{aligned} (1 - R_a)\, T_{ar} = {} & \rho_1 T_a + \rho_2 T_g + \rho_3 T_s + \rho_4 T_F \\ & + \rho_5 T_o + \rho_6 T_c + \rho_7 + \rho_8 T_L \end{aligned} \qquad (8)$$

where
T_a = desired antenna temperature
T_L = leakage from opposite port
T_g = internal guide temperature
T_S = temperature of polarization switch
T_F = temperature of feed between switch and OMT
T_o = temperature of OMT
T_c = temperature of cutler feed
ρ_i = referral constants $i = 1, 2, \ldots, 8$

It should be noted that the factor ρ_7 simply accounts for small but unknown contributions originating from front-end elements not well defined physically. Now let

$$\frac{T_{1r} + T_{2r}}{2}(1 - R_a) = \nu_1 T_1 + \nu_2 T_2 \qquad (9)$$

and

$$\rho_m (T_{1r} - T_{2r}) = \rho_m (v_3 T_1 - v_4 T_2) \tag{10}$$

Then (7) may be written as

$$\begin{aligned} \beta_1 T_a + \beta_8 T_L = v_1 T_1 + v_2 T_2 - \rho_m (v_3 T_1 - v_4 T_2) \\ - \beta_2 T_g - \beta_3 T_s - \beta_4 T_F - \beta_5 T_o \\ - \beta_6 T_c - \beta_7 \end{aligned} \tag{11}$$

or after division by β_1 as

$$\begin{aligned} T_a + \xi_c T_L = \sigma_1 T_1 + \sigma_2 T_2 - \rho_m (\sigma_3 T_1 - \sigma_4 T_2) \\ - \eta_1 T_g + \eta_2 T_s - \eta_3 T_F - \eta_4 T_o \\ - \eta_5 T_c - \eta_6 \end{aligned} \tag{12}$$

Now since T_F is not monitored it will be assumed that $T_F = a_o T_s + a_1 T_o$ where $a_o + a_s = 1$. In this case the conversion model simplifies to the form

$$\begin{aligned} T_a + \xi_c T_L = \sigma_1 T_1 + \sigma_2 T_2 - \rho_m (\sigma_3 T_1 - \sigma_4 T_2) - \sigma_5 T_g \\ - \sigma_6 T_s - \sigma_7 T_o - \sigma_8 T_c - \sigma_9 \end{aligned} \tag{13}$$

The leakage term T_L is actually the antenna temperature at the opposite polarization. This term can be estimated from measurements at that polarization. There is no quarantee that it is always available; however, over the sea it can be estimated from T_a with reasonable accuracy when the sky is clear. Hollinger* has shown that the polarized temperatures are related by

$$T_{av} = \frac{1-c}{1+c} T_{ah}$$

where

$C = 0.00012 \ \theta^2$

θ = observation angle in degrees

*Hollinger, J. P., "Passive Microwave Measurements of the Sea Surface," Journal of Geophysical Research, vol. 75, no. 27, pp. 5209-5213, September 1970.

The latter estimating technique is employed by the subroutine since temperatures at the opposite polarization are not always available. Fortunately ξ_c is small (estimated at .004) so that under some circumstances the leakage term can be ignored (by setting ξ_c =0).

The above model is incorporated in RCONV. The listing for the subroutine is shown in Figure A-15. The variables are defined in Table A-6.

Table A-6 Definition of Variables in Subroutine RCONV

I	=	Receive Polarization Index
J	=	Frequency Indicator
RPAR(5)	= σ_1	
RPAR(6)	= σ_2	
RPAR(7)	= σ_3	
RPAR(8)	= σ_4	
RPAR(I+9)	= σ_6	
RPAR(I+11)	= σ_7	
RPAR(I+13)	= σ_8	
RPAR(I+15)	= σ_5	
RPAR(17)	= σ_9	
EST	= ξ_c T_L	
DATA(28)	= T_S	
DATA(32)	= T_0	
DATA(46)	= T_c	
DATA(18)	= T_9	

G. Subroutine DOPCHK

1. Introduction

This subroutine computes the doppler frequency shift induced by the relative motion between aircraft and sea. The doppler shift is compared with the doppler filter characteristic to determine whether the shift is within the bandpass. When it is, a frequency index IBAND, which is employed to locate the filter gain, is computed. When the doppler shift lies outside the defined doppler filter characteristic, the band index is set at the appropriate extreme and a flag is set.

```
            SUBROUTINE RCONV (DATA,IDATA,RPAR,RCAL,IL)
        CRCONV                          CONVERT RADIOMETER
        C              SUBROUTINE RCONV (DATA,IDATA,RPAR,RCAL,IL)
        C
        C              THIS PROGRAM WAS PREPARED BY
        C
        C                  JOHN P. CLAASSEN
        C                  GLEN E. ELLIOTT
        C
        C              UNIVERSITY OF KANSAS CENTER FOR RESEARCH
        C
        C
        C          THIS PROGRAM CONVERTS RADIOMETER OUTPUT VOLTAGES TO ANTENNA
        C          TEMPERATURE.
        C
C00010      DIMENSION DATA(1), IDATA(1), RPAR(1), RCAL(1)
        C
        C
        C              FORM POLARIZATION AND FREQUENCY INDEX.
        C
C00010      I=IDATA(12)
C00011      J=IDATA(2)
        C
        C              EXTRACT LOAD TEMPERATURES
        C
C00012      T1 = DATA(33)
C00014      T2=DATA(22)
C00015      IF(J.EQ.3) T2=DATA(27)
        C
        C              FORM MEASUREMENT PARAMETER.
        C
000021      DATA(13)=-(DATA(13)-RCAL(1))/(RCAL(2)-RCAL(1))
        C
        C              APPLY TEMPERATURE FACTORS.
        C
C00026      DATA(13)=DATA(13)*(RPAR(5)*T1-RPAR(6)*T2)+RPAR(7)*T1+RPAR(8)*T2-
           *        RPAR(I+9)*DATA(28)-RPAR(I+11)*DATA(32)-RPAR(I+13)*
           *        DATA(46) - RPAR(15)*DATA(18)
        C
        C                  APPROXIMATE MISSING DATA
        C
        C

        C          APPROXIMATING PARAMETER
        C
C00062      C=0.00012*DATA(14)*DATA(14)
        C
        C          FOR HORIZONTAL POLARIZATION
        C
C00065      EST=DATA(13)*(1.0+C)/(1.0-C)
        C
        C          FOR VERTICAL POLARIZATION
        C
000071      IF (I .EQ.1) EST = DATA (13)*(1.0-C)/(1.0+C)
        C          FORM ANTENNA TEMPERATURES
        C
C00100      DATA(13) = DATA(13) - EST*RPAR(I+3)
C00104      RETURN
C00105      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE A-15. FORTRAN LISTING FOR SUBROUTINE RCONV

2. Theory and Design

For non-relativistic speeds the two way doppler frequency shift is known to be given by

$$f_d = 2 \langle \bar{v} \cdot \bar{i}_r \rangle \frac{f}{c} \tag{1}$$

where $\bar{i}_r$ = boresight unit vector
$\bar{v}$ = aircraft velocity
f = operating frequency
c = speed of light

Denote ϕ_c as the cross-track angle and θ as the incident angle (see subroutine ANGLE). Then the boresight vector is given by

$$\bar{i}_r = - \sin\theta \cos\phi_c \, \bar{i}_x + \sin\theta \sin\phi_c \, \bar{i}_y - \cos\theta \, \bar{i}_z \tag{2}$$

Also

$$\bar{v} = v \, \bar{i}_x \tag{3}$$

So

$$\bar{v} \cdot \bar{i}_r = - v \sin\theta \cos\phi_c \tag{4}$$

and

$$f_d = -2 \frac{f v}{c} \sin\theta \cos\phi_c \tag{5}$$

Expression (5) is the desired result.

Now the computer routine determines whether the doppler shift falls in the doppler bandwidth as defined by the upper and lower limits, SPAR(23) and SPAR(22), respectively. If f_d is between SPAR(22) and SPAR(23), f_d is divided by the doppler filter gain sample interval SPAR(24) and the result rounded off to determine the band index (IBAND) from the center frequency. When the shift is outside the bandpass, the band index is set at the appropriate extreme and a flag is set. An error message is also appended to the Conversion Report. A listing of DOPCHK is shown in Figure A-16. The variables employed by the routine are defined in Table A-7.

```
            SUBROUTINE DOPCHK (DATA,SPAR,IL,IBAND,LNPMOD,IFLAG)
      CDOPCHK                              CHECK DOPPLER SHIFT
      C              SUBROUTINE DOPCHK (DATA,SPAR,IL,IBAND,LNPMOD,IFLAG)
      C
      C              THIS PROGRAM WAS PREPARED BY
      C
      C                 JOHN P. CLAASSEN
      C                 GLEN E. ELLIOTT
      C
      C              UNIVERSITY OF KANSAS CENTER FOR RESEARCH
      C
      C
      C           THIS SUBROUTINE COMPUTES THE DOPPLER SHIFT INDUCED BY THE
      C           RELATIVE MOTION BETWEEN AIRCRAFT AND SEA. THE DOPPLER SHIFT
      C           IS COMPARED  WITH THE DOPPLER FILTER CHARACTERISTIC TO
      C           DETERMINE WHETHER SHIFT IS WITHIN THE BANDPASS AND WHEN
      C           IT IS TO SELECT THE DOPPLER FILTER GAIN  RELATIVE TO THE
      C           CALIBRATION SIGNAL.
      C
C00011      DIMENSION DATA(1), SPAR(1)
      C
C00011      DATA DEGRAD/0.0174532925/
      C
      C           INITIALIZE FLAG
      C
C00011      IFLAG=0
      C
      C           CONVERT AIRCRAFT SPEED TO METERS/SEC
      C
C00011      VEL=DATA (42)/3.048
      C
      C           CONVERT TO RADIANS
      C
000013      THETA = DATA (14)*DEGRAD
C00015      PHI=DATA(15)*DEGRAD
      C
      C              COMPUTE THE DOPPLER SHIFT
      C
C00017       DOP=-2.0*SPAR(1)*VEL/299793000.0*SIN(THETA)*COS(PHI)
      C
      C           IF DOPPLER SHIFT EXCESSIVE FORM FLAG
      C

C00036      IF(DOP.LT.SPAR(22)) GO TO 10
C00041      IF(DOP.GT.SPAR(23)) GO TO 20
      C
      C           FORM DOPPLER BAND INDEX
      C
000044      IBAND=-DOP/SPAR(24)+0.5
C00047      IF (DOP.GT.0.0)  IBAND= -DOP/SPAR(24)-0.5
C00053      RETURN
      C              FORM FLAG AND ASSIGN BAND
C00054  10  IBAND = -SPAR(22)/SPAR(24)
C00057      GO TO 30
C00060  20  IBAND = -SPAR(23)/SPAR(24)
C00063  30  IFLAG=1
C00064      IF(MOD(IL,LNPMOD).EQ.0) WRITE (6,40) IL,DATA(42),DATA(14),DOP
000115  40  FORMAT (I6,5X,1H*,1X,17HEXCESSIVE DOPPLER,21X,1H*,5X,
           *        13HAIRCRAFT SPD=,F5.1,5X,11HVIEW ANGLE=,F5.1,5X,
           *        14HDOPPLER SHIFT=,F6.0)
C00115      RETURN
000116      END
```

FIGURE A-16. FORTRAN LISTING OF SUBROUTINE DOPCHK

Table A-7 Definition of Variables

IFLAG	=	Flag for Excessive Doppler
VEL	=	Aircraft Ground Velocity
THETA	=	Incident Angle
PHI	=	Cross-track Angle
DOP	=	Doppler Shift
IBAND	=	Doppler Band Index
SPAR(22)	=	Lower Doppler Filter Limit
SPAR(23)	=	Upper Doppler Filter Limit
SPAR(24)	=	Doppler Filter Sample Interval

H. Subroutine SCONV

1. Introduction

SCONV converts scatterometer output measurements into a normalized input power. The scatterometer transfer function is employed to perform the conversion. The transfer function is developed below. Certain significant simplifications arise when SCAT normal calibrations are employed.

2. Theory and Design

The block diagram for the RADSCAT scatterometer is shown in Figure A-17. The front-end elements have been decomposed into a calibration element G_c, transmission elements G_{VT}, G_{HVT}, G_{HT}, and G_{VHT} and reception elements G_{VR}, G_{HVR}, G_{HR} and G_{VHR} for ease in analysis and reduction of data. The transfer functions account for mis-match and loss. Calibration attenuators A_1 through A_4 are employed to calibrate the output channels 1 through 4, respectively.

During calibration transmitter power is routed sequentially through A_1, A_2, A_3 and A_4 and the respective channel output is noted. The resulting output voltage for the i^{th} channel is thus given by

$$V_i^{(c)} = d^2 P_x G_c G_R A_i \alpha_i I_i \tau_c \qquad (1)$$

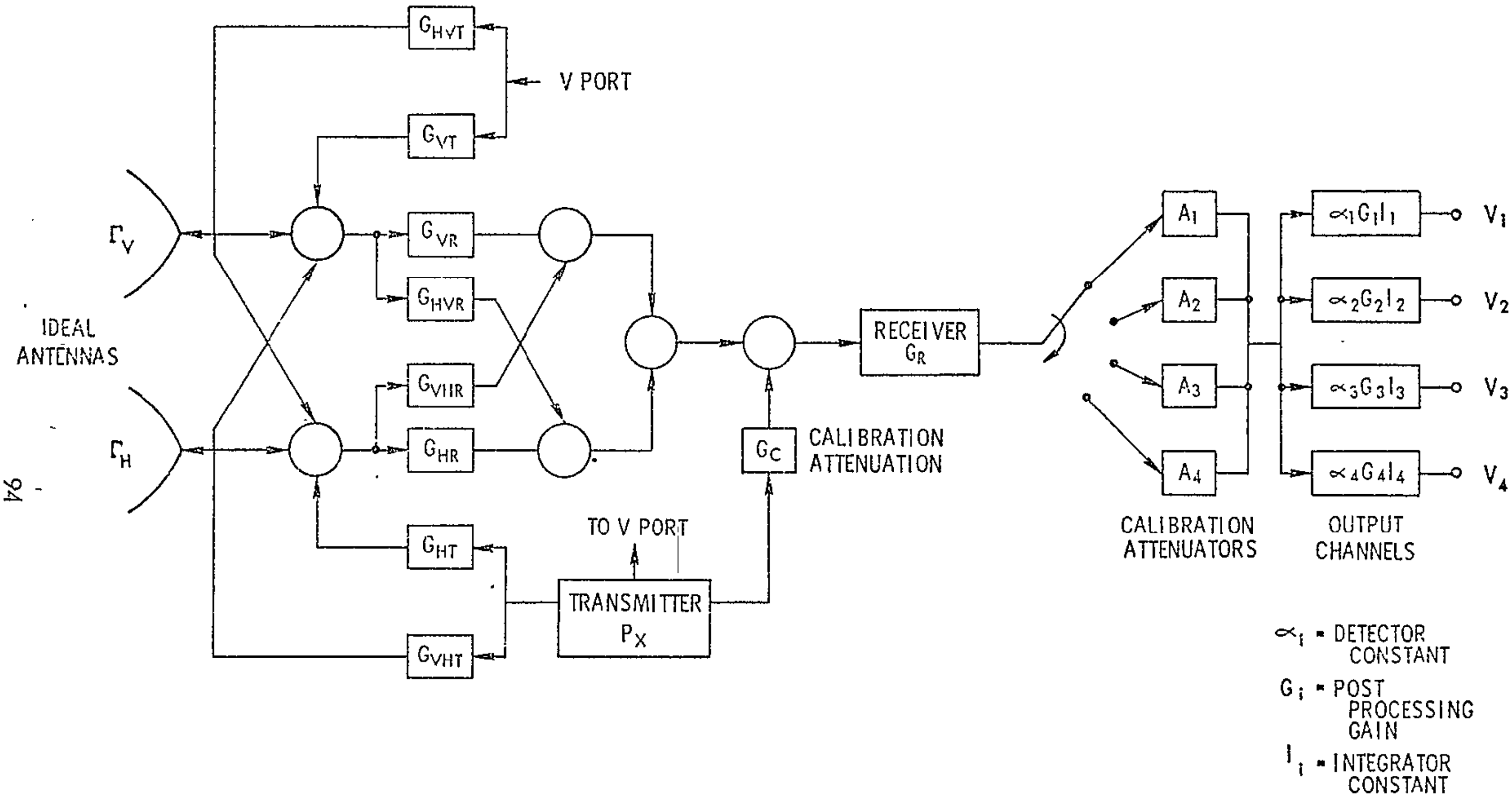


FIGURE A-17. BLOCK DIAGRAM FOR SCATTEROMETER where d = the effective duty factor

P_x = transmitter peak power

τ_c = calibration integration period

The factor d^2 occurs in the above expression since only the center frequency of the power spectrum is sampled by the receiver. The calibrations thus serve as a measure of the transmitted power

$$P_x = \frac{V_i^{(c)}}{d^2\, G_c\, G_R\, A_i\, \alpha_i\, G_i\, I_i\, \tau_c} \tag{2}$$

During measurements at VV polarization and incident angle θ_o the power arriving at the receiver input is given by

$$P_{VV}(\theta_o) = \frac{\lambda^2}{(4\pi)^3}\, d^2 P_x \,[\, G_{VV}\, \Gamma_V\, I_{VV}(\theta_o)\, \Gamma_V\, G_{VT} + G_{VHR}\, \Gamma_H\, I_{HV}(\theta_o)\, \Gamma_V\, G_{VT} + G_{HVR}\, \Gamma_H\, I_{HH}(\theta_o)\, \Gamma_H\, G_{HVT} + G_{VR}\, \Gamma_V\, I_{VH}(\theta_o)\, \Gamma_H\, G_{HVT}\,] \tag{3}$$

where

$$I_{mn} = \iint_{4\pi} \frac{P_m(\chi,\psi)\, P_n(\chi,\psi)\, \sigma_{mn}(\theta)}{R^4}\, dA \tag{4}$$

σ_{mn} = normalized scattering coefficient for transmission at polarization n and reception at polarization m

R = range to scattering element dA

P_m = normalized antenna pattern

T_m = antenna directivity at polarization m

λ = operating wavelength

Similarly for observations at HH polarization the power at the receiver input is given by

$$P_{HH}(\theta_o) = \frac{\lambda^2}{(4\pi)^3}\, d^2 P_x \,[\, G_{HR}\, \Gamma_H\, I_{HH}(\theta_o)\, \Gamma_H\, G_{HT} + G_{HVR}\, \Gamma_V + I_{VH}(\theta_o)\, \Gamma_H\, G_{HT} + G_{VHR}\, \Gamma_V\, I_{VV}(\theta_o)\, \Gamma_V\, G_{VHT} + G_{HR}\, \Gamma_H\, I_{HV}(\theta_o)\, \Gamma_V\, G_{VHT}\,] \tag{5}$$

The latter three terms in expressions (3) and (5) may be neglected since products such as $G_{HVR} I_{VH}$; $G_{VHR} G_{VHT}$, $I_{HV} G_{VHT}$, etc., are extremely small. Therefore (3) and (5) simplify to:

$$P_{VV}(\theta_o) = \frac{\lambda^2 d^2 G_x}{(4\pi)^3} G_{VR} \Gamma_V I_{VV}(\theta_o) \Gamma_V G_{VT} \tag{6}$$

and

$$P_{HH}(\theta_o) = \frac{\lambda^2 d^2 G_x}{(4\pi)^3} G_{HR} \Gamma_H I_{HH}(\theta_o) \Gamma_H G_{HT} \tag{7}$$

The corresponding on-scale output voltage for these measurements occur on say, channels j and k, respectively. The outputs are given by

$$V_{jm} = P_{VV}(\theta_o) \; G_R \, \alpha_j \, G_j \, I_j \, \tau_m \tag{8}$$

and

$$V_{km} = P_{HH}(\theta_o) \; G_R \, \alpha_k \, G_k \, I_k \, \tau_m \tag{9}$$

where τ_m = measurement integration time

$m = 1, 2, \ldots, 7$

A subscript is employed on τ to identify the different integration periods associated with six angle indicators (1 through 6) for the ALTERNATING and FIXED ANGLE MODES and the integration period τ_7 unique to the SHORT SCAT mode.

Now equations (8) and (9) may be employed to express the receiver input power. Thus we may write

$$\frac{V_{jm}}{G_R \, \alpha_j \, G_j \, I_j \, \tau_m} = \frac{\lambda^2 d^2 P_x}{(4\pi)^3} G_{VR} G_{HT} \Gamma_V^2 I_{VV}(\theta_o) \tag{10}$$

$$\frac{V_{km}}{G_R \, \alpha_j \, G_j \, I_j \, \tau_m} = \frac{\lambda^2 d^2 P_x}{(4\pi)^3} G_{HR} G_{HT} \Gamma_H^2 I_{HH}(\theta_o) \tag{11}$$

Now P_x may be replaced by expression (2) for the corresponding output channel to yield the desired results

$$I_{VV}(\theta_o) = \frac{V_{jm}}{V_j^{(c)}} \frac{\tau_c}{\tau_m} \frac{G_c A_j}{G_{VR} G_{VT} \Gamma_V^2} \frac{(4\pi)^3}{\lambda^2} \tag{12}$$

and

$$I_{HH}(\theta_o) = \frac{V_{km}}{V_k^{(c)}} \frac{\tau_c}{\tau_m} \frac{G_c A_k}{G_{HR} G_{HT} \Gamma_H^2} \frac{(4\pi)^3}{\lambda^2} \tag{13}$$

It is noted that the receiver gain terms have dropped out.

Subroutine SCONV implements the algebraic expressions for I_{VV} and I_{HH}. Actually the SCAT channel selection routine in CRUNCH applies the factors A_m and $V_m^{(c)}$ ($m = j$ or k) since they are channel dependent. Missing from these algebraic expressions is the doppler filter gain factor which is a function of the doppler frequency shift. The gain is normalized with respect to the gain at zero doppler since the calibrations are made there. An equally spaced sample version of this filter characteristic must appear in the lower part of SPAR. The filter gain at zero doppler must appear at SPAR(50). Down doppler filter gains must be stored in entries between SPAR(51) and SPAR(85) whereas entries for the up-doppler gain must appear between SPAR(34) and SPAR(50). Presently the gain function is sampled at 250 Hz intervals. To compensate the measurement the relative doppler gain must divide the measurement, DATA(8).

The FORTRAN coding for SCONV is shown in Figure A-18. The variables are defined in Table A-8.

Table A-8 Definition of Variables

DATA(8)	=	Selected Channel Output, V_{jm}
INT	=	Integration Period Index
IDATA(3)	=	RADSCAT Mode
SPAR(3)	=	Calibration Period
SPAR(INT+3)	=	Measurement Period
SPAR(IBAND+50)	=	Relative Doppler Filter Gain
SPAR(17)	=	Calibration Attenuation, G_c

```
                 SUBROUTINE SCONV (DATA,IDATA,SPAR,IBAND)
       CSCONV                         SCAT CONVERSION
       C                 SUBROUTINE SCONV (DATA,IDATA,SPAR,IBAND)
       C
       C                 THIS PROGRAM WAS PREPARED BY
       C
       C                        JOHN P. CLAASSEN
       C                        GLEN E. ELLIOTT
       C
       C                 UNIVERSITY OF KANSAS CENTER FOR RESEARCH
       C
       C
       C            THIS SUBROUTINE CONVERTS A SCATTEROMETER NORMALIZED OUTPUT
       C            VOLTAGE TO THE SCATTERING INTEGRAL.  THE RESULT IS LEFT
       C            IN DATA(8).
       C
C00007           DIMENSION DATA(1), IDATA(1), SPAR(1)
C00007           DATA PI/3.1415927/
       C
       C
       C
       C            APPLY CONVERSION PARAMETERS
       C
C00007           DATA(8)= DATA(8)*(4.*PI)**3/(299793000./SPAR(1))**2
       C
       C            SET INTEGRATION TIME INDEX
       C
C00014           INT = IDATA (5)
C00015           IF (IDATA (3) .EQ. 1) INT =7
       C
       C            COMPENSATE FOR INTEGRATION TIMES
       C
C00020           DATA(8)= DATA(8)*SPAR(3)/SPAR(INT+3)
       C
       C            SET POLARIZATION INDICES
       C
C00024           IXMIT =IDATA (6)
C00025           IREC = IDATA (7)
       C
       C            COMPENSATE FOR DOPPLER FILTER CHARACTERISTIC
       C
C00027           DATA(8)= DATA(8)/SPAR(IBAND+50)

       C
       C                  APPLY GAIN FACTORS
       C
C00031           DATA(8)=DATA(8)*SPAR(17)/(SPAR(IXMIT+11)*SPAR(IXMIT+25)*
                *      SPAR(IREC+15)*SPAR(IREC+25))
C00041           RETURN
C00042           END
```


FIGURE A-18. SOURCE LISTING FOR SUBROUTINE SCONV

Table A-8 Definition of Variables (Continued)

IXMIT	=	Transmit Polarization Index
IREC	=	Receive Polarization Index
SPAR(IXMIT+11)	=	Transmit Transfer Function
SPAR(IREC+15)	=	Receive Transfer Function
SPAR(IXMIT+25)	=	Transmit Antenna Gain
SPAR(IREC+25)	=	Receive Antenna Gain

V. SPECIAL ROUTINES

A. Tape Read Routines

FUNCTION subroutine IREAD and subroutine READ1 are employed to read the Input Raw Data records. IREAD permits CRUNCH to read records from either file code 01 or 08. IREAD calls READ1 to unblock records from file code 01. The records on the scratch unit (08) are not blocked and so can be read from IREAD. A second entry point in READ1 permits subroutine CALIB to backspace one physical record (DATA). Read functions on unit 1 may be performed by simply calling READ1. The source listings for IREAD and READ1 are shown in Figures A-19 and A-20, respectively.

B. Tape Write Routine

Subroutine WRITE2 performs WRITE functions on file code 02. Output records (IARRAY) are accumulated in IBLOCK until it is full at which time IBLOCK is written to file code 02. A second entry point in WRITE2 performs the END OF FILE function. However, the remainder of IBLOCK is first filled with zeros before putting an EOF mark on file code 02. See the listing for WRITE2 in Figure A-21 for further insight.

C. Subroutine ABORT

To purposely stop the execution of the program and abort the job, an abort function was provided. The circumstances underwhich an abort is executed are described in Section III B. If the system has an abort routine available that dumps more registers and memory than this fatal execution will, it is advisable to replace the one provided. A listing of subroutine ABORT is shown in Figure A-22.

```
              FUNCTION IREAD(IFC)
        C
        C              THIS PROGRAM WAS PREPARED BY
        C                    GLEN E. ELLIOTT
        C              UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.
        C
        C              THIS ROUTINE PERMITS SUBROUTINE CRUNCH TO READ FROM FILE
        C              CODE 1 OR 8.  RECORDS ON FILE UNIT 1 ARE BLOCKED IN
        C              GROUPS OF 10 PHYSICAL RECORDS.  EACH RECORD CONSISTING
        C              OF 50 WORDS.  SUBROUTINE READ1 UNBLOCKS THE RECORDS.
        C.             ON THE OTHERHAND THOSE RECORDS ON FILE UNIT 8 ARE NOT
        C              BLOCKED AND AS A CONSEQUENCE A SIMPLE BINARY READ
        C              IS EMPLOYED.
        C
000003        LOGICAL EOF1
000003        COMMON /INPUT/ EOF1,DATA(50)
000003        IREAD=0
000003        IF(IFC.EQ.8) GO TO 1
000006        CALL READ1
000007        IF(EOF1) IREAD=1
000013        RETURN
000015  1     READ(8) DATA
000022        IF(EOF,8) 2,3
000026  2     IREAD=1
000027  3     RETURN
000031        END
```

ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE A-19. LISTING FOR IREAD

```
              SUBROUTINE READ1
         C
         C             THIS PROGRAM WAS PREPARED BY
         C                  GLEN E. ELLIOTT
         C             UNIVERSITY OF KANSAS CENTER FOR RESEARCH,INC.
         C
         C             THIS SUBROUTINE READS BLOCKS OF 10 PHYSICAL RECORDS AND
         C             SEPARATES THEM. RECORDS FILLED WITH 99999. ARE BY-PASSED
         C             UNTIL AN EOF IS ENCOUNTERED.  A SECONDARY ENTRY POINT
         C             IS PROVIDED TO PERMIT BACKSPACING INDIVIDUAL RECORDS
         C             WITHIN A BLOCK.
         C
000002        COMMON/INPUT/IEOF,DATA(50)
000002        LOGICAL IEOF
000002        DIMENSION BLOCK(50,10)
000002        DATA NDX,NUT/0,1/
000002        IEOF=.FALSE.
000003        IF(NDX.NE.0) GO TO 10
000004        NDX=1
000005        READ(NUT) BLOCK
000012        IF(EOF,NUT) 100,10
000015   10   IF(BLOCK(1,NDX) .EQ. 99999.) GO TO 90
000022        DO 20 I = 1,50
000023   20   DATA(I) = BLOCK(I,NDX)
000033        NDX=NDX+1
000034        IF(NDX.GT.10) NDX=0
000037        RETURN
000040   90   READ(NUT)
000043        IF(EOF,NUT) 100, 90
000046  100   IEOF=.TRUE.
000047        NDX=0
000050        RETURN
000051        ENTRY BCKSP1
000056        IF(NDX.LT.1) NDX=11
000062        NDX=NDX-1
000063        RETURN
000064        END
```


FIGURE A-20. LISTING FOR READ1

```
                 SUBROUTINE WRITE2 (IARRAY)
          C
          C              THIS PROGRAM WAS PREPARED BY
          C                   GLEN E. ELLIOTT
          C              UNIVERSITY OF KANSAS CENTER FOR RESEARCH,INC.
          C
          C              THIS ROUTINE BLOCKS 20 WORD RECORDS INTO 20 RECORD
          C              GROUPS (BLOCKS) AND WRITES THE BLOCK OUT TO FILE UNIT
          C              02.  WHEN INSUFFICIENT RECORDS EXIST TO FILL A BLOCK
          C              THE REMAINING PORTION IS FILLED WITH ZEROES AS PROVIDED
          C              AT THE SECONDARY ENTRY POINT.  AN EOF IS APPENDED
          C              TO THE FILE THEREAFTER.
          C
000003           DIMENSION IBLOCK(20,20), IARRAY(1)
000003           DATA INDEX /1/
000003           DO 20 I = 1,20
000005     20    IBLOCK(I,INDEX) = IARRAY(I)
000015           INDEX = INDEX + 1
000016           IF (INDEX .LE. 20)  RETURN
000021           WRITE (2) IBLOCK
000026           INDEX = 1
000027           RETURN
000030           ENTRY ENDF2
000036           IF (INDEX .LE. 1)   GO TO 30
000042           DO 25 I = INDEX,20
000043           DO 25 J= 1,20
000044     25    IBLOCK(J,I) = 0
000054           WRITE (2)  IBLOCK
000061           INDEX=1
000062    30     ENDFILE 2
000064           RETURN
000065           END
```


FIGURE A-21. LISTING FOR WRITE2

```
                 SUBROUTINE ABORT(A)
          C      HALT PROGRAM IN CASE OF ERROR
000003           DIMENSION IEOR(2)
000003           DATA IEOR/10HEND OF REE,1HL/
000003           WRITE (6,100) A
000011      100  FORMAT (10X,10HABORT CODE,1X,A10)
000011           CALL ENDF2
000012           WRITE(2) IEOR,(I,I=3,9)
000024           B=A/0.0+1.0
000030           STOP
000032           END
```


FIGURE A-22. LIST FOR SUBROUTINE ABORT

APPENDIX B

OUTPUT PROGRAM

I. INTRODUCTION

The output program (OUTPUT) lists groups of output files produced by the conversion program. External directives permit the user to select file types to be listed. This appendix describes the operation of OUTPUT. In the description below it will be helpful to refer to the logic diagram of Figure B-1 and the listing of Figure B-2.

II. DESIGN AND OPERATION

Upon entry into OUTPUT a control card is read. The format and content of the control card is treated in Section III C. The card content is printed and its validity is checked. An invalid command terminates the program. The tape on unit 2 is also rewound. The file label is read and compared with the validated directive. If the file label does not agree with the label, the file is bypassed and a bypass message is printed. The search continues until the appropriate file is found; if in the search the end of reel is encountered,a message to that effect is printed and another control card is read (statement 1).

Specified file types are listed 80 records at a time. Each record is identified with a record number. The listing continues until the file is exhausted (EOF) at which time another file of the same type is sought.

The variables employed in OUTPUT are defined in Table B-1.

Table B-1 Definition of Variables

MISSD	=	Directed Mission Number
FLTD	=	Directed Flight Number
LIND	=	Directed Line Number
RUND	=	Directed Run Number

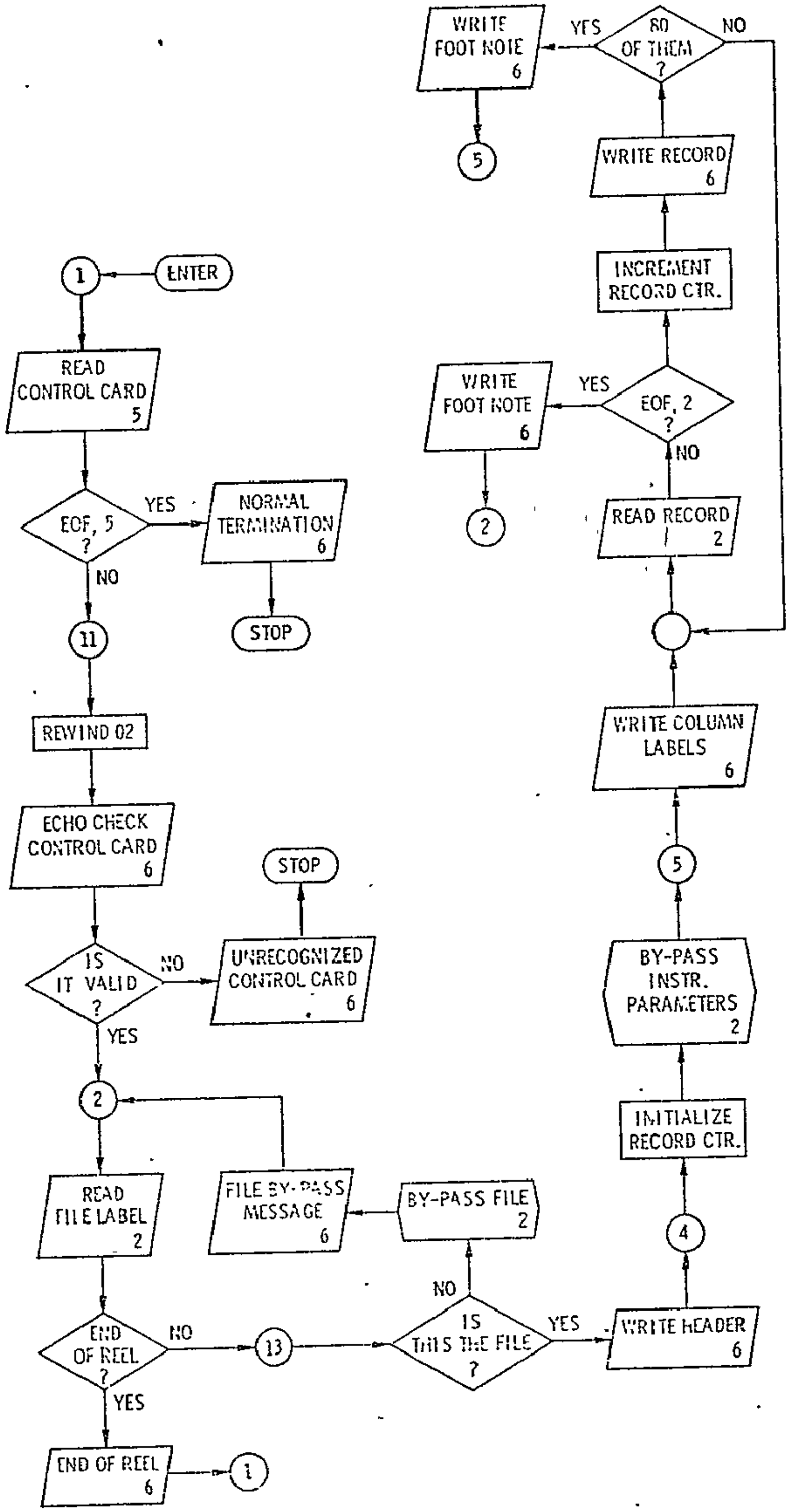


FIGURE B-1. DESCRIPTIVE LOGIC DIAGRAM FOR OUTPUT PROGRAM

```
                 PROGRAM OUTPUT(INPUT,OUTPUT,TAPE2,TAPE5=INPUT,TAPE6=OUTPUT)        00001295
          COUTPUT                            OUTPUT PROGRAM                           00001296
          C                                                                           00001297
          C               THIS PROGRAM WAS PREPARED BY                                00001298
          C                                                                           00001299
          C                        JOHN P. CLAASSEN                                   00001300
          C                        GLEN E. ELLIOTT                                    00001301
          C                                                                           00001302
          C                 UNIVERSITY OF KANSAS CENTER FOR RESEARCH                  00001303
          C                                                                           00001304
000003           INTEGER FLTD, RUND, FLT, RUN                                         00001305
000003           DIMENSION DATA(20),DATE(3)                                           00001306
000003           DIMENSION LSTYP(4), IMODE(4), IPL(2)                                 00001307
000003           DIMENSION IEOR(2)                                                    00001308
000003           LOGICAL EOF2                                                         00001309
000003           EQUIVALENCE (MODE,DATA(2)),                                          00001310
               *            (IP1,DATA(5)),(IP2,DATA(6))                               00001311
000003           DATA IEOR/10HEND OF REE,1HL/                                         00001312
000003           DATA IMODE/4HR.C.,4HS.S.,4HF.A.,4HA.A./, IPL/1HH,1HV/                00001313
000003           DATA LSTYP/6HFLTLIN,6HMISFLT,6HSPECIF,3HALL/                         00001314
          C                                 GET DIRECTIVE AND FILE ID.                00001315
000003     1     READ(5,100) LTYPE, MISSD,FLTD,LIND,RUND                              00001316
000021   100     FORMAT(A6,4X,4I5)                                                    00001317
000021           IF(EOF,5) 10,11                                                      00001318
000024    11     REWIND 2                                                             00001319
000026           WRITE(6,200) LTYPE, MISSD, FLTD, LIND, RUND
000044   200     FORMAT(1H1,///10X13HCONTROL CARD=,3X,A6,4X,4I5//)
000044           DO 20 I = 1,4
000046    20     IF(LTYPE.EQ.LSTYP(I)) GO TO 2
000052           WRITE(6,700)
000055   700     FORMAT(1H1,///10X22HUNRECOGNIZED DIRECTIVE)
000055           STOP
          C                                 FIND THE FILE.                            00001320
000057     2     CONTINUE                                                             00001321
000057           WRITE(6,300)
000063   300     FORMAT(1H1)
000063           READ(2) MISS, FLT, DATE, LIN, RUN, FREQ, FEED                        00001322
000104           IF(MISS.NE.IEOR(1)) GO TO 13                                         00001323
000106           IF(FLT.NE.IEOR(2)) GO TO 13                                          00001324
000110           WRITE(6,400)
000114   400     FORMAT(///10X10(1H*),11HEND OF REEL,10(1H*)///)
000114           GO TO 1                                                              00001325

000115    13     CONTINUE                                                             00001326
000115           IF(LTYPE.EQ.LSTYP(4)) GO TO 4                                        00001327
000117           IF(MISS.NE.MISSD) GO TO 3                                            00001328
000121           IF(FLT.NE.FLTD) GO TO 3                                              00001329
000123           IF(LTYPE.EQ.LSTYP(2)) GO TO 4                                        00001330
000125           IF(LIN.NE.LIND) GO TO 3                                              00001331
000127           IF(LTYPE.NE.LSTYP(3)) GO TO 4                                        00001332
000131           IF(RUN.EQ.RUND) GO TO 4                                              00001333
000133           WRITE(6,500) MISS, FLT, LIN, RUN
000146   500     FORMAT(//10X13HBYPASSED FILE,4I5)
000146     3     READ(2)                                                              00001334
000151           IF(EOF,2) 2,3                                                        00001335
          C                                 FOUND IT, OUTPUT HEADINGS.                00001336
000154     4     CONTINUE                                                             00001337
000154     9     WRITE(6,604)                                                         00001338
000160   604     FORMAT(1H1,53X,14HMICROWAVE DATA)                                    00001339
000160           LINEN=0                                                              00001340
          C                                 BYPASS ID RECORD.                         00001341
000161           READ(2) IDSET, ID1, ID2, ID3                                         00001342
000174     5     WRITE(6,600) MISS, FLT, DATE, LIN, RUN, FREQ, FEED,                  00001343
               *            IDSET, ID1, ID2, ID3                                      00001344
000226   600     FORMAT(/9H MISSION-,I3,9H  FLIGHT-,I3,                               00001345
               *            7H  DATE-,2I3,I5,2X,11H  FLT LINE-,1X,I6,6H  RUN-,I3,
               *            13H   FREQUENCY-,F5.1,7H  FEED-,2X,A6,8H   IPAR=,I2,3I3/)00001347
000226           WRITE(6,605)                                                         00001348
000232   605     FORMAT(/                                                             00001349
               * 62H RECORD  TIME     MODE  INCID  CROSS    XMIT  REC   S INT         00001350
               * 53HSCAT        SCAT      RAD    DEPOL    S/R     ALTITUDE,           00001351
               * 11H   FLT      /,                                                    00001352
               * 62H NO.                       ANGLE  ANGLE    POL    POL   TIME      00001353
               * 52H            DB                FACTOR   CODE          ,            00001354
               * 11H   DIRECT  /,                                                     00001355
               * 62H                          (DEG)  (DEG)              (SEC)         00001356
               * 53H                         (DEG K)                 (FEET),          00001357
               * 12H   (DEG)    //)                                                   00001358
          C                                 OUTPUT ONE DOUBLE COMPUTER PAGE.          00001359
000232           DO 6 I=1,60                                                          00001360
000234           CALL READ2(DATA,EOF2)                                                00001361
000236           IF(EOF2) GO TO 7                                                     00001362
000240    12     LINEN=LINEN+1                                                        00001363
000242           MODE=IMODE(MODE+1)                                                   00001364
000243           IP1=IPL(IP1+1)                                                       00001365
000245           IP2=IPL(IP2+1)                                                       00001366
```


FIGURE B-2. SOURCE LISTING FOR PROGRAM OUTPUT

```
C00246     6     WRITE(6,601) LINEN,(LATA(I), I = 1,14)
C00264   601     FORMAT
        C                                   MATCH HEADING WITH FORMATS.
        *        * LINE     TIME      MODE  INCID  CRCSS    XMIT  REC   S INT
        *        * NO.                      ANGLE  ANGLE    PCL   POL   TIME
        *        *                          (DEG)  (DEG)                (SEC)
             * (1X,I4, 2X,F8.1,3X, A6, F5.1,2X, F5.1,6X, A5, A1,4X, F5.3,2X,
        C
        *        *SCAT        SCAT       RAD  DEPOL  S/R     ALTITUDE*,
        *        *            DB              FACTOR CODE            *,
        *        *                     (DEG K)               (FEET)*,
             * E11.4, 2X,F6.1,2X, F7.1, F8.3,   I9,   F10.0,
        C
        *        *   FLT      */,
        *        *   DIREC    */,
        *        *   (DEG)     *)
             * F9.1)
        C
        C                                   FOOT PAGE WITH CODE KEY.
C00264           WRITE(6,602)
C00270   602     FORMAT(//14H KEY TC CODES-/,
             *              25X,32H   0 NO FLAG                         *,5X,
             *              21H  1000 POLAR REVERSAL/,
             *              25X,*   1 POSSIBLY OUTSIDE RANGE GATE*,5X,
             *              22H 10000 EXCESSIVE DEPOL/,
             *              25X,32H  10 OUTSIDE DYNAMIC RANGE       ,5X,
             *              24H100000 REC TEMP ABNORMAL/,
             *              25X,* 100 EXCESSIVE DOPPLER            *,4X,
             *               *1000000 WARM LOAD TEMP ESTIMATED*/,
             *              25X,37H     COMBINATIONS OF FLAGS CAN OCCUR./,1H1)
C00270           GO TO 5
        C                                   FOOT PARTIAL PAGE
C00271     7     WRITE(6,602)
C00275           IF(LTYPE.EQ.LSTYP(3)) GO TO 1
C00277           GO TO 2
C00300    10     STOP
C00302           END
```

ORIGINAL PAGE IS
OF POOR QUALITY

igure B-2. (Continued)

Table B-1 Definition of Variables (Continued)

MISS	=	File Mission Number
FLT	=	File Flight Number
LIN	=	File Line Number
RUN	=	File Run Number
LTYPE	=	Control Directive
LSTYPE	=	Program Recognized Directives
IEOR	=	10H End of Reel
DATA	=	Output Data
EOF2	=	EOF Indicator on File Code 02

III. SUBROUTINE READ2

READ2 performs the record retrieval function for OUTPUT. Output files are blocked in 20 record groups. READ2 retrieves these blocks and withdraws records as required by OUTPUT. The end of the file indicator EOF2 is generated by READ2. A listing for this routine appears in Figure B-3.

```
                  SUBROUTINE READ2(IDATA,EOF2)                                   00001404
              C
              C            THIS PROGRAM WAS PREPARED BY
              C                    GLEN E. ELLIOTT
              C            UNIVERSITY OF KANSAS CENTER FOR RESEARCH,INC.
              C
              C            THIS ROUTINE UNBLOCKS 20 RECORD BLOCKS AND PROVIDES
              C            AN END OF FILE INDICATOR, EOF2.
              C
000005            DIMENSION IDATA(20), IBLOCK(20,20)                             00001405
000005            LOGICAL EOF2                                                   00001406
000005            DATA NDEX/0/                                                   00001407
000005            EOF2=.FALSE.                                                   00001408
000005            IF(NDEX.NE.0) GO TO 10                                         00001409
000006            NDEX=1                                                         00001410
000007            READ(2) IBLOCK                                                 00001411
000014            IF(EOF,2) 100,10                                               00001412
000020        10  IF(IBLOCK(1,NDEX).EQ.0) GO TO 90                               00001413
000023            DO 20 I=1,20                                                   00001414
000025        20  IDATA(I)=IBLOCK(I,NDEX)                                        00001415
000035            NDEX=NDEX+1                                                    00001416
000036            IF(NDEX.GT.20) NDEX=0                                          00001417
000041            RETURN                                                         00001418
000042        90  READ(2)                                                        00001419
000045            IF(EOF,2) 100,90                                               00001420
000051       100  EOF2=.TRUE.                                                    00001421
000052            NDEX=0                                                         00001422
000053            RETURN                                                         00001423
000054            END                                                            00001424
```

ORIGINAL PAGE IS
OF POOR QUALITY

FIGURE B-3. SOURCE LISTING FOR SUBROUTINE READ2

APPENDIX C

INSTRUMENT CHARACTERISTICS MAINTENANCE PROGRAM (ICHAR)

I. INTRODUCTION

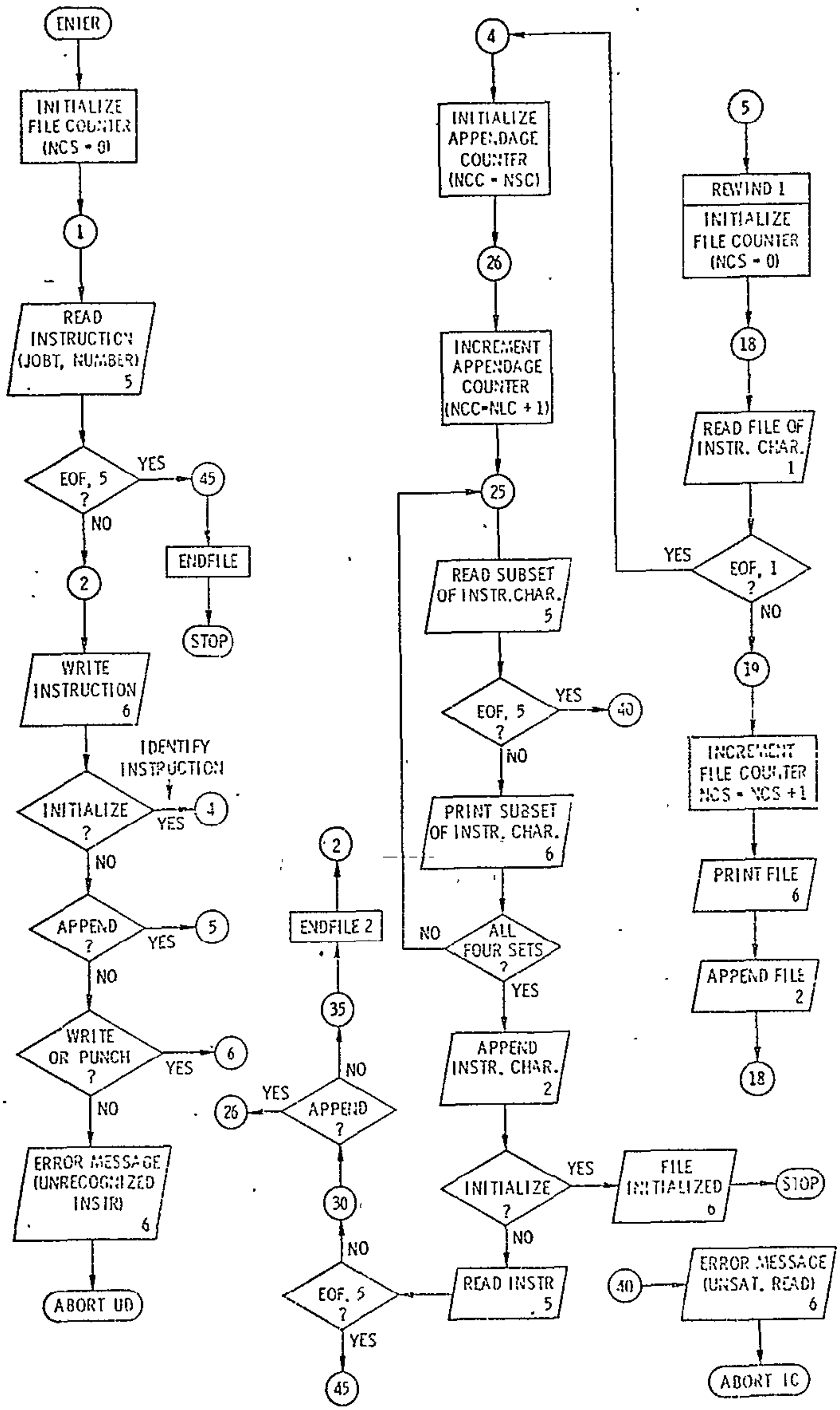
Program ICHAR maintains a historical file of RADSCAT instrument characteristics. New sets of instrument parameters may be submitted to ICHAR for storage; old sets may be printed or punched. The design and operation of ICHAR is treated in this appendix. The reader will find it helpful to refer to the logic diagram (Figure C-1) and listing (Figure C-2) for ICHAR.

II. DESIGN AND OPERATION

The files on the historical parameter tape are identified numerically. So upon entry into the program the file pointer is initialized to zero. An instruction is read and echo checked. The instruction is then identified as one of three types. If the instruction is INITIALIZE, the file appendage counter (NNC) is set equal to one and four subsets of instrument parameters are read and transferred to file code 02. The format and entry for the parameters were described in Section III D of the text. The program is then halted.

If the historical tape had been initialized already and the command was APPEND, the old historical tape would have been rewound (Statement 5) and the contents of that tape transferred to another one (Statements between 5 and 18). Once transferred, the new set would be appended to the new file (Statements 26 through 23). When insufficient parameters are presented to fill the new file, the program is aborted. Otherwise the next control card is read; if it is APPEND, the process is repeated until all APPEND commands have been executed.

If the (next) command had been PRINT or PUNCH, the program would have advanced to statement 6 and executed the instruction. The program continues until all control cards have been exhausted.



ORIGINAL PAGE IS OF POOR QUALITY

FIGURE C-1. DESCRIPTIVE FLOW CHART FOR PROGRAM ICHAR

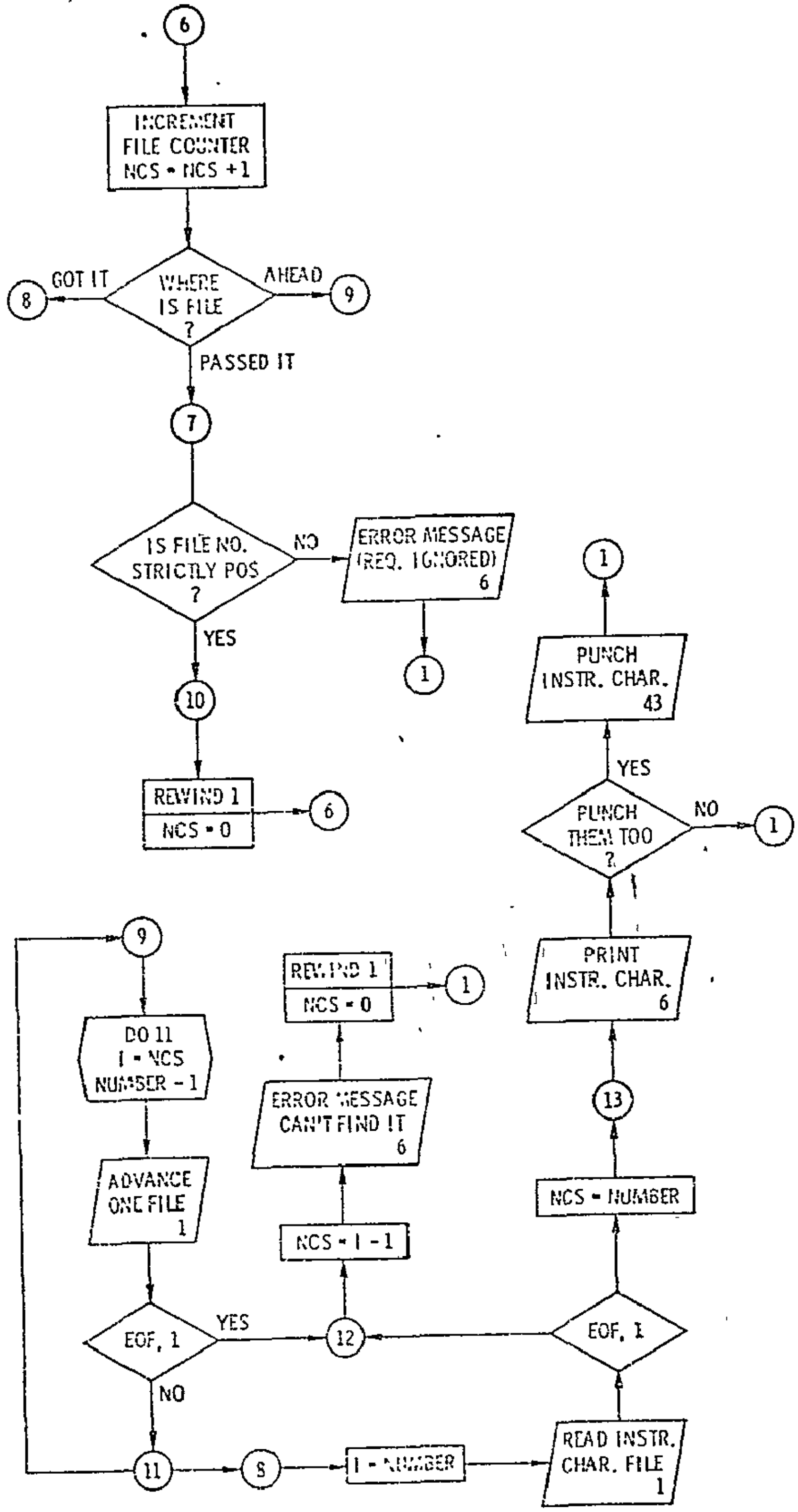


FIGURE C-1. (continued)

```
                PROGRAM ICHAR(INPUT,OUTPUT,PUNCH,TAPE1,TAPE2,
               * TAPE5=INPUT,TAPE6=OUTPUT,TAPE4 )=PUNCH)
          CICHAR        MAINTAIN ICHAR FILE.
          C
          C             THIS PROGRAM WAS PREPARED BY
          C
          C                  JOHN P. CLAASSEN
          C                  GLEN E. ELLIOTT
          C
          C             UNIVERSITY OF KANSAS CENTER FOR RESEARCH,INC.
          C
          C             THIS PROGRAM WILL STORE RADSCAT INSTRUMENT
          C             CHARACTERISTICS ON A HISTORICAL TAPE. THE HISTORICAL
          C             TAPE IS INITIALLY PREPARED BY USING THE DIRECTIVE
          C             *INITIALIZE* .  TO APPEND NEW SETS OF CHARACTERISTICS THE
          C             COMMAND *APPEND* WILL TRANSFER OLD SETS FROM TAPE UNIT
          C             1 TO TAPE UNIT 2 AND APPEND THE NEW SET .  NEW SETS ARE
          C             READ FROM CARDS.  PREVIOUSLY STORED SETS MAY
          C             BE PRINTED OR PUNCHED BY USING THE COMMANDS PRINT OR
          C             PUNCH RESPECTIVELY.  THE SET NO. MUST BE DESIGNATED .
          C             THE SETS WILL HAVE BEEN STORED NUMERICALLY IN ORDER ON
          C             THE TAPE.     TO MAINTAIN THE INSTRUMENT CHARACTERISTICS
          C             IT IS ADVISABLE TO RETAIN THE OLD TAPE WHEN A
          C             NEW ONE ONE IS PREPARED.  OLDER VERSIONS MAY BE DISCARDED.
          C             IT IS ADVISABLE ALSO TO USE A TAPE COMPARISON UTILITY
          C             PROGRAM TO VERIFY THE TRANSFER.
          C
          C             NCS = FILE COUNTER FOR THE OLD TAPE ALSO SERVES AS
          C             A FILE POINTER ON UNIT 1
          C             NCC = FILE COUNTER FOR THE APPENDED SETS
          C             JOBT = EXTERNAL DIRECTIVE. SEE DATA STATEMENT
          C             NUMBER = NUMBER OF THE REQUESTED FILE
          C             PAR = INSTRUMENT PARAMETERS
          C             IDAT = DATE
          C             NUM2 = SET NO. AS READ FROM TAPE
          C
000003          DIMENSION PAR(160,4), IDAT(3), IOPT(4)
000003          DATA IOPT/10HINITIALIZE,6HAPPEND,5HPRINT,5HPUNCH/, NCS/0/
          C                                   READ DIRECTIVE AND SET NUMBER.
000003     1    READ(5,500) JOPT, NUMBER
000013          IF (EOF,5) 45,2
000016   500    FORMAT(A10,I5)


000016     2    WRITE(6,600) JOPT,NUMBER
000026   600    FORMAT(1H1,///* DIRECTIVE IS *,A10,I5)
          C                                   CHECK DIRECTIVE
000026          DO 3 N=1,4
000030          IF(JOPT.EQ.IOPT(N)) GO TO(4,5,6,6),N
000041     3    CONTINUE
000043          WRITE(6,601)
000047   601    FORMAT(//* UNRECOGNIZED DIRECTIVE*//)
000047          CALL ABORT(2HUD)
          C                    FIND FILE
000051      6   NCS = NCS+1
000053          IF(NUMBER-NCS) 7,8,9
000055     7    IF(NUMBER.GE.1) GO TO 10
000060          WRITE(6,602) NUMBER
000065   602    FORMAT(//* IC SET NUMBER*,I5,* DOES NOT EXIST.*,
              * * REQUEST IGNORED.*//)
000065          GO TO 1
000066    10    REWIND 1
000070          NCS = 0
000071          GO TO 6
000072      12  NCS = I-1
000074          WRITE (6,608)  NCS,NUMBER
000104     608  FORMAT (//* THERE ARE ONLY*I5,* FILES.*
              * * CANT FIND FILE*I5//)
000104          REWIND 1
000106          NCS = 0
000107          GO TO 1
          C                                   FORWARD SPACE TO CORRECT SET.
000110     9    NUM=NUMBER-1
000112          DO 11 I=NCS,NUM
000114          READ(1)
000117          IF(EOF,1) 12,11
000122    11    CONTINUE
          C                                   READ INSTRUMENT CHARACTERISTICS
000125      8   I = NUMBER
000127          READ(1) NUM2,IDAT,PAR
000140          IF(EOF,1) 12,13
000143    13    NCS=NUMBER
          C                                   WRITE INSTRUMENT CHARACTERISTICS.
000145          WRITE (6,610)
000150     610  FORMAT (1H1,///)
000150          DO 15 I=1,4
000152          WRITE(6,603) NUM2  ,IDAT,PAR(1,I),PAR(2,I),PAR(3,I)
```

FIGURE C-2. SOURCE LISTING FOR PROGRAM ICHAR

```
000174     603   FORMAT(//* SET*,I5,*  DATE*,3I5,F15.4,5X,A10,E15.4)
000174           DO 14 J=4,75,7
000176           L=J+6
000200           IF(L.GT.75) L=75
000203     14    WRITE(6,604)(PAR(K,I),K=J,L)
000222     604   FORMAT(7E15.4)
000222           WRITE(6,603) NU(2   ,IDAT,PAR(76,I),PAR(77,I),PAR(78,I)
000245           DO 15 J=79,160,7
000247           L=J+6
000251           IF(L.GT.160) L=160
000254     15    WRITE(6,604)(PAR(K,I),K=J,L)
000275           IF(N.EQ.3) GO TO 1
          C                                PUNCH IC*S.
000277           DO 17 I=1,4
000301           LL=1
000302           WRITE(43,4303)    NUM2,IDAT        ,PAR(1,I),PAR(2,I),PAR(3,I),
               *        NUMBER, I, LL
000333     4303  FORMAT(4I5,10X,E10.3,A10,E10.3,12X,I2,I1,I5)
000333           DO 16 J=4,75,7
000335           LL=LL+1
000337           L=J+6
000341     16    WRITE(43,4304)(PAR(K,I),K=J,L),NUMBER,I,LL
000364     4304  FORMAT(7E10.3,2X,I2,I1,I5)
000364           LL=LL+1
000366           WRITE(43,4303)    NUM2,IDAT        ,PAR(1,I),PAR(2,I),PAR(3,I),
               *        NUMBER, I, LL
000416           DO 17 J=79,160,7
000420           LL=LL+1
000422           L=J+6
000424     17    WRITE(43,4304)(PAR(K,I),K=J,L),NUMBER,I,LL
000451           GO TO 1
          C
          C                                COPY OLD IC*S
          C
000452     5     REWIND 1
000454           NCS = 0
000455           WRITE(6,605)
000461     605   FORMAT(1H1,* THE FOLLOWING IC SETS WERE COPIED FROM *,
               * *THE OLD FILE*,/
               * * SET NUMBER     ID NUMBER       DATE*)
000461     18    READ(1) NUM2,IDAT,PAR
000472           IF(EOF,1) 4,19
000475      19   NCS = NCS+1


000477           WRITE(6,606) NCS,NUM2,IDAT
000510     606   FORMAT(3X,I5,9X,I5,3X,3I5)
000510           WRITE(2) NUM2,IDAT,PAR
000521           GO TO 18
          C                                READ NEW SETS.
000522       4   NCC = NCS
000524      26   NCC = NCC+1
000526           WRITE(6,607)
000531     607   FORMAT(1H1/* THE FOLLOWING IC SETS WERE APPENDED TO *
               * *THE OUTPUT FILE.*)
000531     25    DO 23 I=1,4
000533           READ(5,4303) NUMBER,IDAT,PAR(1,I),PAR(2,I),PAR(3,I)
000555           IF(EOF,5) 40,20
000560     20    WRITE(6,603) NCC,IDAT,PAR(1,I),PAR(2,I),PAR(3,I)
000603           DO 21 J=4,75,7
000605           L=J+6
000607           IF(L.GT.75) L=75
000612           READ(5,4304)(PAR(K,I),K=J,L)
000627           IF(EOF,5) 40,21
000632     21    WRITE(6,604)(PAR(K,I),K=J,L)
000651           READ(5,4303) NUMBER,IDAT,PAR(76,I),PAR(77,I),PAR(78,I)
000674           IF(EOF,5) 40,22
000677     22    WRITE(6,603) NCC,IDAT,PAR(76,I),PAR(77,I),PAR(78,I)
000722           DO 23 J=79,160,7
000724           L=J+6
000726           IF(L.GT.160) L=160
000731           READ(5,4304)(PAR(K,I),K=J,L)
000746           IF(EOF,5) 40,23
000751     23    WRITE(6,604)(PAR(K,I),K=J,L)
          C                                APPEND IC*S
000772           WRITE(2) NCC,IDAT,PAR
001003           IF(JOBT.NE.IOPT(1)) GO TO 29
001005           WRITE(6,800)
001011     800   FORMAT(//10X*FILE INITIALIZED*//)
001011           STOP
001013     29    READ(5,500)  JOBT, NUMBER
001023           IF (EOF,5)  45,30
001026      30   IF( JOBT .EQ. IOPT(2) )  26,35
001033      35   ENDFILE 2
001035           GO TO 2
001036      40   WRITE (6,609)
001042     609   FORMAT (//* UNSATISFIED READ.*,
               * * INCOMPLETE SET OF CHARACTERISTICS*//)
```

Figure C-2. (continued)

```
001042          CALL ABORT (2HIC)
001044      45  ENDFILE 2
001046          STOP                                                    00001622
001050          END                                                     00001623
```

```
                SUBROUTINE ABORT(A)
          C
          C     HALT PROGRAM IN CASE OF ERROR
          C
000003          WRITE (6,100) A
000011     100  FORMAT (10X,10HABORT CODE,1X,A10)
000011          ENDFILE 2
000013          B=A/0.0+1.0
000017          STOP
000021          END
```

Figure C-2. (Continued)

III. DEFINITION OF VARIABLES

The variables employed by ICHAR are defined in Table C-1.

Table C-1 Definition of Variables

NSC	=	File Pointer for File Code 01
NCC	=	File Pointer for File Code 02
JOBT	=	Job Instruction
NUMBER	=	File Number to be Printed or Punched
IOPT	=	Array of Recognized Instructions
IDAT	=	Date in Parameters
NUM2	=	Set Number as Read from File Code 01
PAR	=	Set of Instrument Parameters

APPENDIX D

PROGRAM TDUPE

I. INTRODUCTION

Program TDUPE is a translation and duplication program which prepares an integer version of the output tape. It is intended that copies of the integerized tape be sent to the investigators upon request for use on other computing machines. A utility program may be used to prepare additional copies. This appendix describes the design and operation of TDUPE. Helpful to the reader are the flow charts of Figure D-1 and the program listing of Figure D-2.

II. DESIGN AND OPERATION

Action in TDUPE is initiated by an external control card specifying the files, by mission and flight, which are to be copied. A file counter is initialized, file unit 1 is rewound, and the instruction is listed. The labels on the files are examined and compared with the instruction. Files not having the designated mission and flight are bypassed until one is found. At this time a record counter is initialized. Then the instrument parameters are read, scaled, and integerized. The file heading and instrument parameters are transferred to file unit 2. The file heading is also listed on the printer and the file counter is incremented.

Consecutive records are then read, scaled and integerized. In each instance the record counter is incremented. The records are accumulated in blocks by subroutine WRITE2 and transferred to tape unit 2 when the block is filled. The terminating block is filled with integer zeroes when there are insufficient records to fill it. When the file has been exhausted, an EOF is appended to the transferred file and a message is given to indicate the number of records which have been transferred.

The search for additional files continues until the end of the reel is encountered. At this point another instruction is sought. The program terminates

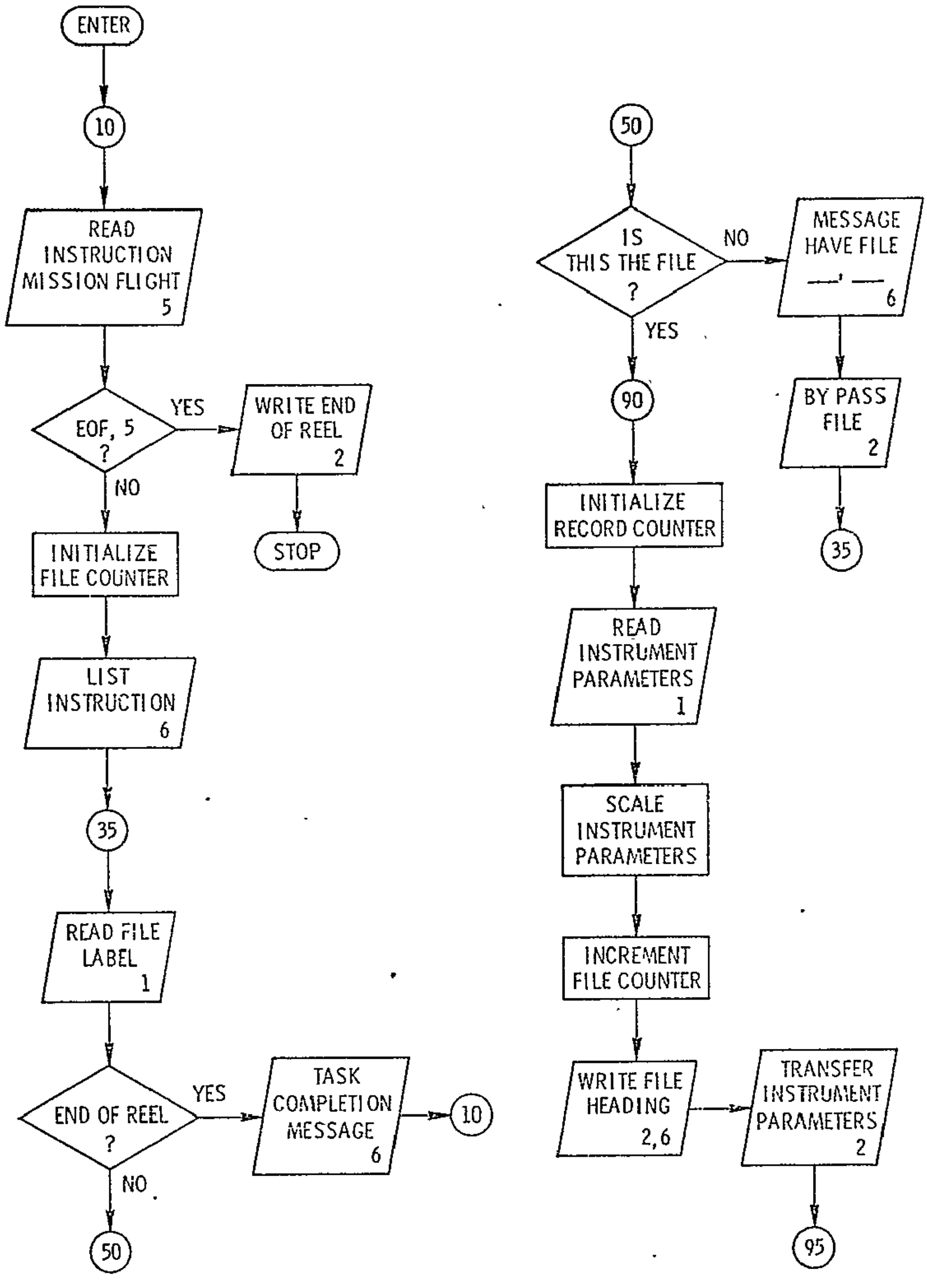


FIGURE D-1. DESCRIPTIVE LOGIC DIAGRAM FOR PROGRAM TDUPE

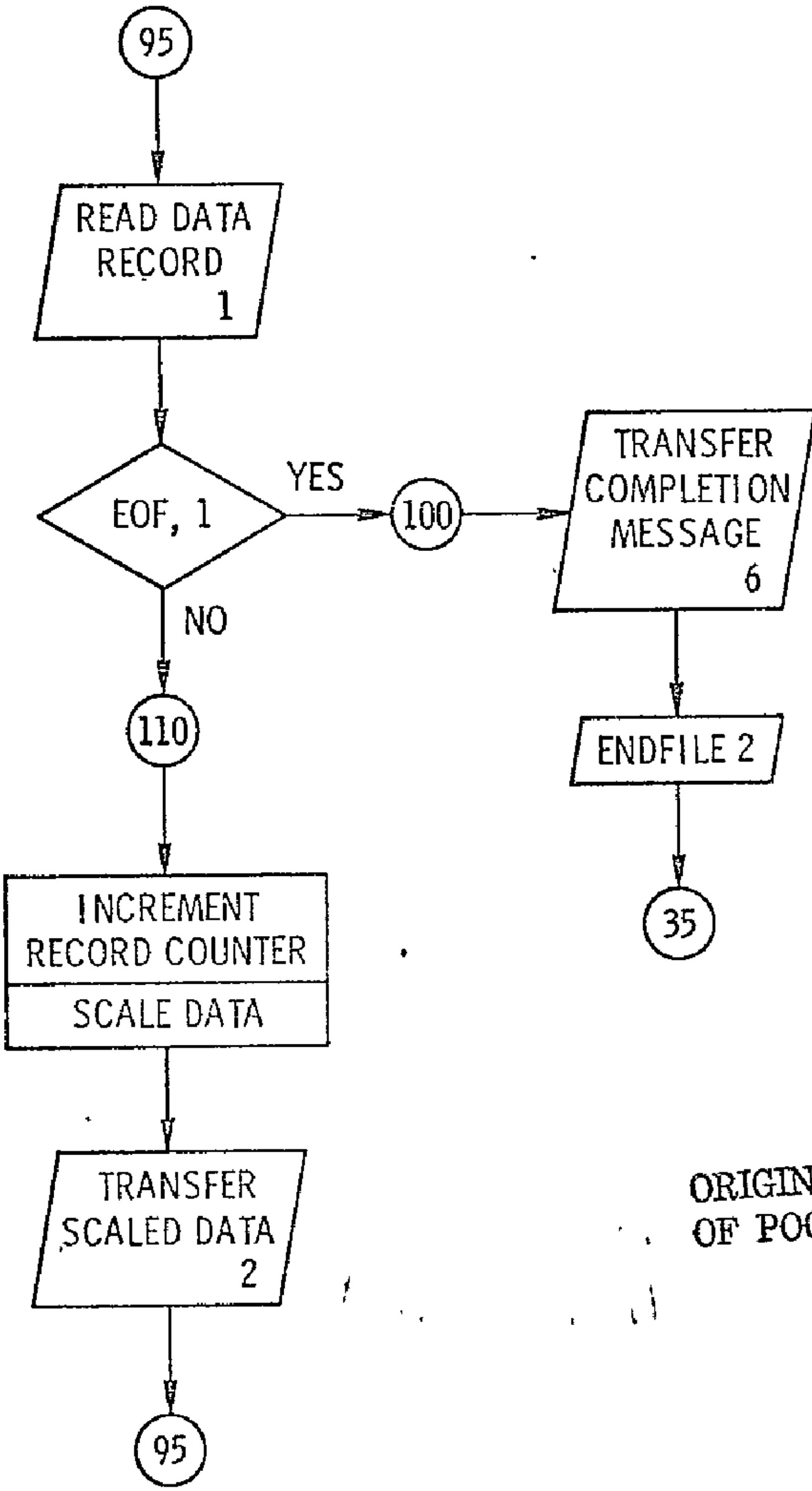


FIGURE D-1. (continued)

```
          PROGRAM TDUPE(INPUT,OUTPUT,TAPE1,TAPE2,TAPE5=INPUT,TAPE6=OUTPUT)
      C
      C           TAPE TRANSLATION AND DUPLICATION ROUTINE
      C
      C           THIS PROGRAM WAS PREPARED BY
      C
      C                JOHN P. CLAASSEN
      C
      C           UNIVERSITY OF KANSAS
      C           CENTER FOR RESEARCH
      C
      C
      C           THIS PROGRAM PREPARES A MASTER DUPLICATE TAPE OF THE
      C           RADSCAT OUTPUT TAPE FOR USE ON OTHER MACHINES.  OTHER
      C           COPIES OF THE MASTER DUPLICATE TAPE MAY BE PREPARED
      C           BY USING AN IN-HOUSE UTILITY PROGRAM.  THE OUTPUT DATA
      C           RECORDS ARE TRANSLATED TO INTEGER FORMAT AFTER
      C           MULTIPLICATION BY AN APPROPRIATE SCALE FACTOR.
      C           MESSAGES ARE LISTED TO INDICATE THE STATE OF THE
      C           TRANSFER.
      C
      C           CONTROL PARAMETERS
      C
      C           MISFLT  = MISSION AND FLT DATA TO BE DUPLICATED
      C
      C           INPUT/OUTPUT PARAMETERS
      C
      C           IHEAD = FILE HEADING
      C           LCHAR = INSTRUMENT PARAMETER LABEL
      C           ICHAR = INSTRUMENT CHARACTERISTICS
      C           DATA = INPUT RECORDS
      C           IDATA = INTEGER EQUIVALENT OF DATA
      C           IARRAY = OUTPUT RECORDS
      C
      C           WORKING PARAMETERS
      C
      C           NOF = NO. OF FILES
      C           IREC = NO. OF RECORDS IN A FILE
      C           IEOR = *END OF REEL*
      C
      C
000003      DIMENSION IHEAD(9), ICHAR(160), DATA(20), IDATA(20),

           * IARRAY(20), IEOR(2), MISFLT(2), LCHAR(4)
000003      DIMENSION RPAR(75), SPAR(85), SFACT(15)
000003      LOGICAL EOF1
000003      EQUIVALENCE (ICHAR(1),RPAR(1)),(ICHAR(76),SPAR(1))
000003      EQUIVALENCE (DATA(1), IDATA(1))
000003      DATA SFACT/1.0E10, 1.0, 1.0E5, 1.0E7, 1.0E9, 3*1.0, 4*10.0,
           * 1.0E3,2*100.0/
000003      DATA IEOR/10HEND OF REE,1HL/
      C
      C           READ THE DIRECTIVE
      C
000003   10 READ (5,1000)       MISFLT
000011 1000 FORMAT (2I10)
000011      IF (EOF,5) 20,30
      C
      C           WRITE *END OF REEL*
      C
000014   20 WRITE (2) IEOR, (I,I=3,9)
000026      STOP
      C
      C           INITIALIZE FILE COUNT
      C
000030   30 NOF = 0
000031      REWIND 1
000033      WRITE (6,2000) MISFLT
000041 2000 FORMAT (1H1,11X, *WILL PREPARE COPY OF DATA FROM MISSION*I5,* AND
           * FLT*I3///)
000041   35 READ (1) IHEAD
      C
      C           CHECK FOR END OF REEL
      C
000046      DO 40 I = 1,2
000050   40 IF(IHEAD(I) .NE. IEOR(I)) GO TO 50
000054      WRITE (6,3000) NOF
000061 3000 FORMAT (//11X,I5* FILES HAVE BEEN TRANSFERRED *///)
000061      GO TO 10
      C
      C           CHECK FOR FILE NAME
      C
000062   50 DO 60 I=1,2
000064   60 IF (IHEAD(I) .NE. MISFLT(I)) GO TO 70
000070      GO TO 90
000070   70 WRITE(6,4000) (IHEAD(I),I=1,2), MISFLT
```

FIGURE D-2. SOURCE LISTING FOR TDUPE

```
000104    4000 FORMAT (11X, *FILE HEADING * 2I5 * DOES NOT AGREE WITH DIRECTIVE
             * *2I5)
        C
        C               BY-PASS UNWANTED FILE
        C
000104      80 READ (1)
000107         IF (EOF,1) 35,80
        C
        C               INITIALIZE RECORD COUNT
        C
000112      90 IREC = 0
000113         READ (1) LCHAR, ICHAR
        C
        C               SCALE AND TRANSFER INSTR. PARAMETERS
        C
000122         ICHAR(1)=RPAR(1)*1.0E-6
000125         DO 91 I=3,20
000127     91  ICHAR(I)=RPAR(I)*1.0E4
000134         ICHAR(21)=RPAR(21)*10.0
000137         ICHAR(22)=RPAR(22)*10.0
000142         DO 92 I=23,75
000143     92  ICHAR(I)=RPAR(I)*1.0E4
000150         ICHAR(76)=SPAR(1)*1.0E-6
000153         DO 93 I=3,16
000155     93  ICHAR(I+75)=SPAR(I)*1.0E4
000162         DO 96 I=17,31
000164     96  ICHAR(I+75)=SPAR(I)*SFACT(I-16)
000171         DO 97 I=32,85
000173     97  ICHAR(I+75)=SPAR(I)*1.0E4
000200         WRITE (6,5000) (IHEAD(I),I=1,7)
000212    5000 FORMAT (/11X,*HAVE MISSION*I5,*  FLT*I4,* DATE*3I5*  FLT LINE*A6,
             * *  RUN*I3)
        C
        C               UPDATE FILE COUNTER.
        C
000212         NOF = NOF + 1
000214         WRITE(2) IHEAD
000221         WRITE(2) LCHAR, ICHAR
000230     95  CALL READ1(DATA,EOF1)
000232         IF(.NOT.EOF1) GO TO 110
000234     100 WRITE (6,6000) IREC,NOF
000244    6000 FORMAT (//11X,I4* RECORDS HAVE BEEN TRANSFERRED FROM FILE* I3///)
000244         CALL ENDF2


000245         GO TO 35
        C
        C               UPDATE RECORD COUNTER.
        C
000246     110 IREC = IREC + 1
        C
        C               SCALE AND TRANSFER RADSCAT MEASUREMENTS.
        C
000250         IARRAY(1)=DATA(1)*100.0
000253         IARRAY(2)=IDATA(2)
000254         IARRAY(3)=DATA(3)*100.0
000257         IARRAY(4)=DATA(4)*100.0
000261         IARRAY(5)=IDATA(5)
000263         IARRAY(6)=IDATA(6)
000264         IARRAY(7)=DATA(7)*10000.0
000267         IF (DATA(8) .LT. 1.0E-03) GO TO 120
000272         IARRAY(8) = DATA(8)*1.0 E06
000275         IARRAY(9)=1000000
000276         GO TO 130
000277     120 IARRAY(8)=DATA(8)*1.0E+09
000302         IARRAY(9)= 1000000000
000304     130 IARRAY(10)=DATA(9)*100.0
000307         IARRAY(11)=DATA(10)*100.0
000312         IARRAY(12) = DATA(11)*10000.0
000315         IARRAY(13)=IDATA(12)
000316         IARRAY(14)=DATA(13)*10.0
000321         IARRAY(15)=DATA(14)*100.0
000324         IARRAY(16)=DATA(15)*100.0
000327         CALL WRITE2(IARRAY)
000330         GO TO 95
000331         END
```

Figure D-2. (Continued)

```
                    SUBROUTINE READ1(IDATA,EOF1)
000005              DIMENSION IDATA(20), IBLOCK(20,20)
000005              LOGICAL EOF1
000005              DATA NDEX/0/
000005              EOF1=.FALSE.
000005              IF(NDEX.NE.0) GO TO 10
000006              NDEX=1
000007              READ(1) IBLOCK
000014              IF(EOF,1) 100,10
000020      10      IF(IBLOCK(1,NDEX).EQ.0) GO TO 90
000023              DO 20 I=1,20
000025      20      IDATA(I)=IBLOCK(I,NDEX)
000035              NDEX=NDEX+1
000036              IF(NDEX.GT.20) NDEX=0
000041              RETURN
000042      90      READ(1)
000045              IF(EOF,1) 100,90
000051      100     EOF1=.TRUE.
000052              NDEX=0
000053              RETURN
000054              END
```

```
                    SUBROUTINE WRITE2 (IARRAY)
000003              DIMENSION IBLOCK(20,20), IARRAY(1)
000003              DATA INDEX /1/
000003              DO 20 I = 1,20
000005        20    IBLOCK(I,INDEX) = IARRAY(I)
000015              INDEX = INDEX + 1
000016              IF (INDEX .LE. 20)  RETURN
000021              WRITE (2) IBLOCK
000026              INDEX = 1
000027              RETURN
000030              ENTRY ENDF 2
000036              IF (INDEX .LE. 1)   GO TO 30
000042              DO 25 I = INDEX,20
000043              DO 25 J= 1,20
000044        25    IBLOCK(J,I) = 0
000054              WRITE (2)  IBLOCK
000061              INDEX=1
000062      30      ENDFILE 2
000064              RETURN
000065              END
```


Figure D-2. (Continued)

when there are no additional instructions. However, before terminating an end of reel statement is written on file unit 2.

The variables are defined in Table D-1.

Table D-1 Definition of Variables

IHEAD	=	File Label
LCHAR	=	Instrument Parameters Label
ICHAR	=	Instrument Parameters
DATA	=	Input Records
IDATA	=	Integer Equivalent
IARRAY	=	Output Records
IEOR	=	11HEND of Reel
SFACT	=	Scale Factors

APPENDIX E

SPECIAL PROGRAMS

I. INTRODUCTION

Two special engineering routines WIDTH and GAIN prepare several antenna parameters essential to the reduction of the scatterometer data. WIDTH computes the equivalent beamwidth of the RADSCAT antenna; whereas GAIN computes the antenna directivity. Other parameters of special interest to antenna engineers are also computed in GAIN. These programs are provided with the realization that the antenna pattern may eventually change through an alteration of the antenna. In this case the equivalent beamwidth and gain should then be recomputed based on new pattern information and the instrument parameters changed accordingly.

Each of the programs are described below. Operating instructions are given in Section IV F of the main text.

II. PROGRAM WIDTH

The theory for this program was described in Appendix A, Section III B. In that development it is apparent that the equivalent beamwidth is given by

$$\theta_{eq} = \sqrt{\frac{4}{\pi} \int_0^{2\pi} \int_0^{\pi} P^2(\theta,\phi) \sin\theta \; d\theta \, d\phi} \tag{1}$$

where $P(\theta\,\phi)$ is the normalized power pattern function described in the standard spherical coordinate system (r , θ , ϕ). When the pattern is elliptically symmetric, the integration may be limited to one quadrant in which case

$$\theta_{eq} = \sqrt{\frac{16}{\pi} \int_0^{\pi} \int_0^{\pi/2} P^2(\theta,\phi) \; \sin\theta \; d\theta \, d\phi} \tag{2}$$

Here it is assumed that the main beam lies on the x-axis as shown in Figure E-1. It is convenient to work in terms of elevation angle ϵ rather than polar angle in which case

$$\theta_{eq} = \sqrt{\frac{16}{\pi} \int_0^{\pi} \int_0^{\pi/2} P^2(\theta,\phi) \cos\epsilon \; d\epsilon \; d\phi} \tag{3}$$

Since quadrature techniques usually require rectangular domains of integration, it is necessary to apply the transformation

$$\psi = \sin \epsilon \tag{4}$$

so that

$$\theta_{eq} = \sqrt{\frac{16}{\pi} \int_0^{\pi} \int_0^{1} P^2(\psi,\phi) \; d\psi \; d\phi} \tag{5}$$

The full limits of integration are not necessary since the integral converges rapidly over the main beam. Program WIDTH whose listing is shown in Figure E-2 use this property to advantage. As shown in the driver routine, the upper limits of integration are progressively increased by 1/2 degree intervals in ϵ. The convergence of the equivalent beamwidth can then be observed as the domain increases to include more of the beam.

A Gaussian-Legendre quadrature technique is employed to perform the double integration. This technique is embodied in RINTEG. The reader is referred to Klerer and Korn* or any other advanced text on numerically techniques for a discussion of this technique.

The main beam of the RADSCAT antenna pattern is functionally represented in the FUNCTION subroutine FUN. RINTEG calls FUN repeatedly to evalute the pattern function squared.

*Klerer, M. and Korn, G. A., "Digital Computer User's Handbook," McGraw-Hill Book Company, 1970.

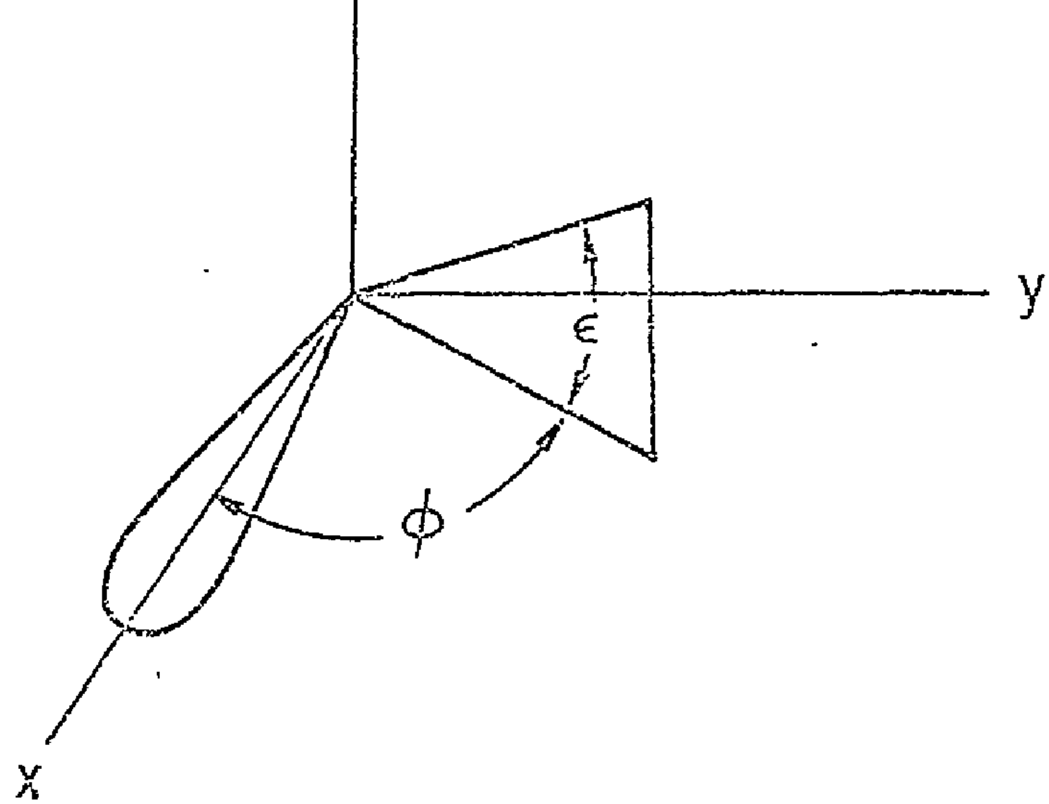


FIGURE E-1. ANTENNA PATTERN GEOMETRY FOR WIDTH

```
                    PROGRAM WIDTH(OUTPUT,TAPE6=OUTPUT)
            C
            C              THIS ROUTINE DETERMINES THE EQUIVALENT (PENCIL)
            C              BEAMWIDTH
            C              BEAMWIDTH FOR AN ANTENNA WHOSE PATTERN IS SPECIFIED
            C              IN SUBROUTINE FUN.  SUBROUTINE RINTEG INTEGRATES
            C              THE PATTERN BY A QUASSIAN (LEGENDRE) QUADRATURE METHOD.
            C              INTEGRATION ARE REPETIVELY PERFORMED AT ONE DEGREE
            C              INTERVALS OUT TO SIX DEGREES.  THE EQUIVALENT
            C              BEAMWIDTH IS CHOSEN FROM THE LISTING AT THE POINT AT
            C              WHICH THE VALUES BEGIN TO CONVERGE
            C
            C              THIS PROGRAM WAS PREPARED BY
            C
            C               JOHN P. CLAASSEN
            C
            C              UNIVERSITY OF KANSAS
            C              CENTER FOR RESEARCH.
            C
000003              EXTERNAL FUN
000003              DATA PI/3.14159265/
000003              WRITE (6,300)
000007         300  FORMAT (1H1//)
000007              WRITE (6,100)
000013         100 FORMAT(30X,5(1H*),*SEARCH FOR THE EQUIVALENT BEAMWIDTH*5(1H*)//)
000013              DO 20 I=1,6
000015              X2 = 0.0174532925*FLOAT(I)*0.50
000017              Y2 = X2
000021              X2 = SIN(X2)
000023              X1 = 0.0
000024              Y1 = 0.0
000025              IX = 10*I
000030              CALL RINTEG (FUN, Z, X2, X1, Y2, Y1, IX, IX, 4, 4)
000041              BEAM = SQRT (16.0*Z/PI)/0.0174532925
000047              WRITE (6,200)     I ,BEAM
000056         200  FORMAT (10X,*ACTIVE BEAMWIDTH IS *I3,* DEGREES, WHEREAS*,
                  * * THE EQUIVALENT BEAMWIDTH IS*F5.2,* DEGREES*)
000056          20  CONTINUE
000060              STOP
000062              END


000056              DO 30 I = 1,MP
000057              SY(I) = SAMPLE(MP,I)*HDELY
000066          30  CONTINUE
000070              DO 50 I = 1,NP
000071              DO 40 J = 1,MP
000072              C(I,J) = COEF(NP,I)*COEF(MP,J)
000106          40  CONTINUE
000110          50  CONTINUE
000112              DO 90 N=1,NX
000114              X1=X2
000116              X2=X1+DELX
000120              XM = X1 + HDELX
000121              Y2 = Y1P
000122              DO 80 M=1,MY
000124              Y1 = Y2
000126              Y2 = Y1 + DELY
000130              YM = Y1 + HDELY
000131              DO 70 I=1,MP
000133              X=SX(I)+XM
000136              DO 60 J=1,MP
000137              Y=SY(J)+YM
000142              SUM = SUM+C(I,J)*FUN(X,Y)
000160          60  CONTINUE
000162          70  CONTINUE
000164          80  CONTINUE
000167          90  CONTINUE
000171              ZZ  = SUM*DELX*DELY*0.250
000174              RETURN
000175              END
```

FIGURE E-2. SOURCE LISTING FOR WIDTH

```
                 SUBROUTINE RINTEG (FUN,ZZ,X2P,X1P,Y2P,Y1P,NNX,MMY,NNP,MMP)
       C               THIS ROUTINE EMPLOYS A QUASSIAN LEGENDER QUADRATURE
       C               INTEGRATION PROCEDURE.  INTEGRATION OVER X-Y SEGMENTS
       C               ARE PERFORMED AFTER TRANSLATION TO (-1,1)X(-1,1).
       C               FUN= INTEGRAND FUNCTION
       C               ZZ = EITHER A PARAMETER FOR FUN OR THE INTEGRATED RESULT
       C                   OR BOTH
       C               X2P,X1P = UPPER AND LOWER LIMITS ON X
       C               Y2P,Y1P = UPPER AND LOWER LIMITS ON Y
       C               NX,MY = SEGMENTS IN X AND Y, RESPECTIVELY
       C               NP,MP = DEGREE OF PRECISION IN X AND Y, RESPECTIVELY
000015           DIMENSION SAMPLE(5,5),COEF(5,5),SX(5),SY(5),C(5,5),SV(25),CV(25)
000015           EQUIVALENCE (SV(1),SAMPLE(1,1)), (CV(1),COEF(1,1))
000015           DATA SV/
               * 1.0, -0.577350269, -0.774596669, -0.861136312, -0.906179846,
               * 1.0, 0.577350269, 0.0, -0.339981044, -0.538469310,
               * 1.0, 0.0, .774596669, 0.339981044, 0.0,
               * 1.0, 0.0, 0.0, 0.861136312, 0.538469310,
               * 1.0, 0.0, 0.0, 0.0, 0.906179846/
000015           DATA CV/
               * 1.0, 1.0, 0.555555556, 0.347854845, 0.236926885,
               * 1.0, 1.0, 0.888888889, 0.652145155, 0.478628670,
               * 1.0, 0.0, 0.555555556, 0.652145155, 0.568888888,
               * 1.0, 0.0, 0.0,         0.347854845, 0.478628670,
               * 1.0, 0.0, 0.0, 0.0,                 0.236926885/
000015           SUM = 0.0
000016           X1 = X1P
000017           X2 = X2P
000020           Y1 = Y1P
000021           Y2 = Y2P
000022           NX = NNX
000023           MY = MMY
000025           NP = NNP
000026           MP = MMP
000030           DELX = (X2-X1)/FLOAT(NX)
000033           DELY =  (Y2-Y1)/FLOAT(MY)
000037           HDELX = DELX*0.50
000041           HDELY = DELX*0.50
000042           X2 = X1
000043           DO 20 I = 1,NP
000045           SX(I) = SAMPLE(NP,I)*HDELX
000054        20 CONTINUE
```

```
                 FUNCTION   FUN (XX,YY)
       C               THIS ROUTINE DEFINES THE SQUARE OF THE ANTENNA
       C               PATTERN OVER THE MAIN BEAM.
000005           ESP = XX
000005           ESP = ATAN(ESP,SQRT(1.0-ESP*ESP))
000015           PHIP =YY
000020           AMXEP = AMAX1 (ESP,PHIP)
000024           IF (AMXEP .LE. 0.0291 ) GO TO 10
000027           IF (AMXEP .LE. 0.0349 ) GO TO 20
000031           IF (AMXEP .LE. 0.0611 ) GO TO 30
000034           IF (AMXEP .LE. 0.105 ) GO TO 40
000036           IF (AMXEP .LE. 0.628) GO TO 50
000041           FUN = 10.0E-38
000042        10 FUN = EXP ( -4690.*ESP*ESP -3280.*PHIP*PHIP )
000050           FUN = FUN*FUN
000052           RETURN
000052        20 FUN = EXP(-121.1*ESP-3280.*PHIP*PHIP)
000060           FUN = FUN*FUN
000062           RETURN
000062        30 FUN = 0.0398*EXP(-821.*ESP*ESP-1072.*PHIP*PHIP)
000071           FUN = FUN*FUN
000073           RETURN
000073        40 FUN = 0.0159*EXP(-486.*ESP*ESP-53.6*PHIP)
000102           FUN = FUN*FUN
000104           RETURN
000104        50 FUN =0.000159*EXP(-8.59*SQRT(ESP*ESP+PHIP*PHIP))
000114           FUN = FUN*FUN
000116           RETURN
000116           END
```

Figure E-2. (Continued)

III. PROGRAM GAIN

A. Introduction

Program GAIN computes the antenna directivity, beam solid angle and, antenna efficiency. The theory and design of the program is described below.

B. Theory

The antenna beam solid angle is given by

$$\Omega = \int_0^{2\pi}\int_0^{\pi} P(\theta,\phi)\ \sin\theta\ d\theta\ d\phi \qquad (6)$$

where

$P(\theta,\phi)$ = normalized antenna pattern

θ = polar angle

ϕ = azimuthal angle

$p(\theta,\phi)$ represents the normalized power in the antenna pattern. It can be shown that the antenna directivity Γ is related to Ω by the following expression

$$\Gamma = \frac{4\pi}{\Omega} \qquad (7)$$

The antenna efficiency, on the otherhand, is the ratio of the power in the main beam to that in the entire pattern. Clearly then, the efficiency is given by

$$\eta_i = \int_0^{2\pi}\int_0^{\theta_i} P(\theta,\phi)\ \sin\theta\ d\theta\ d\phi \qquad (8)$$

where θ_i is a polar angle which subdivides the main beam. In the above expression it is tacitly assumed that the main beam lies on the positive z axis as illustrated in Figure E-3. θ_i is chosen somewhat arbitrarily. Commonly θ_i is chosen at the -3 dB point or at the first null of the antenna pattern. For narrow beam antenna, the latter is most appropriate for radiometric work.

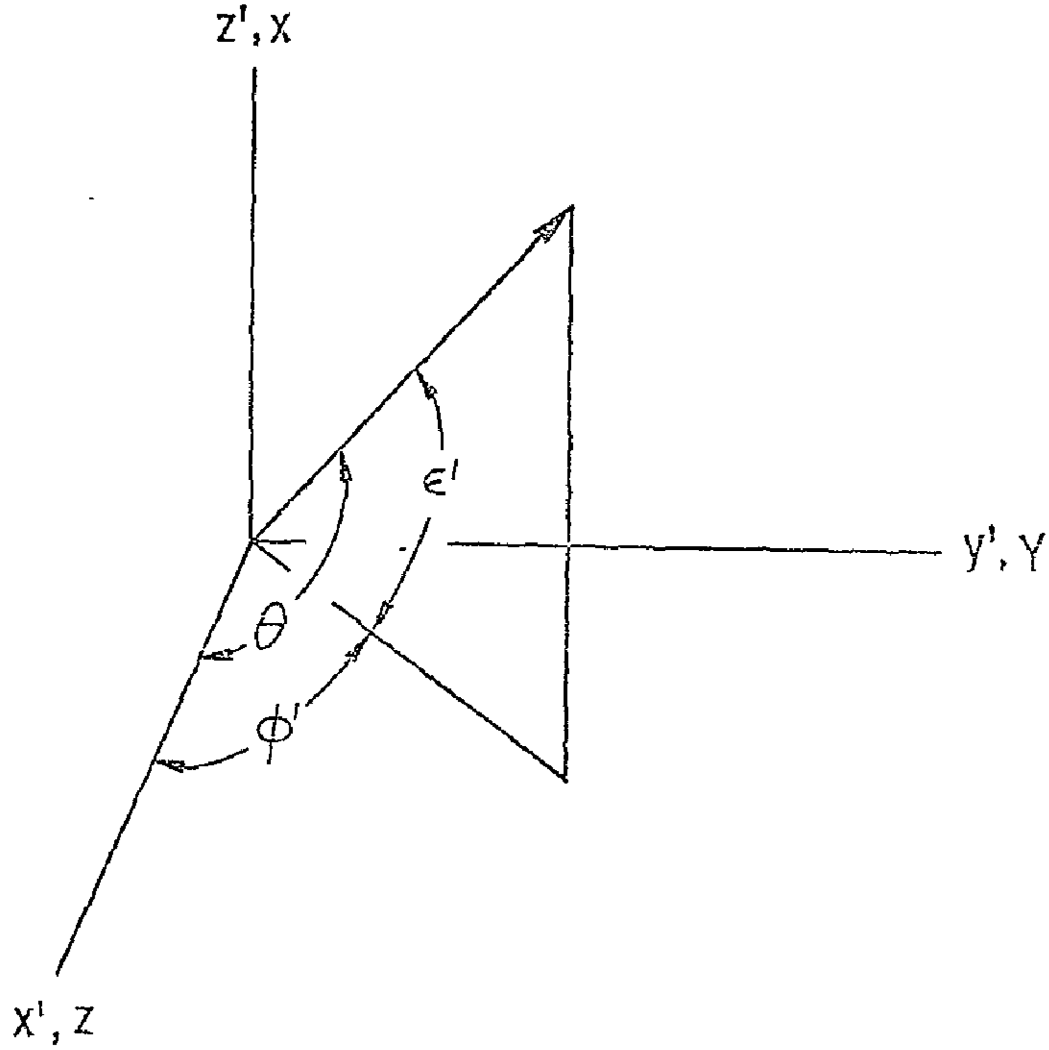


FIGURE E-3. ANTENNA PATTERN GEOMETRY FOR GAIN

C. Design and Operation of GAIN

In the discussion which follows it will be helpful to examine the FORTRAN listing for GAIN as shown in Figure E-4. Upon entry into GAIN, an array BEAM in which various polarized beam solid angle factors will be stored, is cleared. The main beam solid angles are then computed for various θ_i spaced at 1/2 degree intervals. The integration of the antenna pattern is performed by RINTEG which in turn calls for polarized pattern information from FUN1. The results are stored in BEAM. The accumulated solid beam angle is stored in TOTAL. The remainder of the polarized beam solid angle is then computed in the following statements.

At this point the antenna gain is computed and the gain and beam solid angle are printed.

In the remainder of the program the power content in the cross-polarized pattern (FUN2) is computed and added to TOTAL. A DO loop terminating on 30 then computes the antenna efficiency at one degree intervals out to 8 degrees. The result is listed.

Finally, the total beam solid angle, including polarized and cross-polarized contributions, is printed.

D. Subroutines FUN1 and FUN2

The antenna patterns described in FUN1 and FUN2 are actually represented in terms of another coordinate system. The relationship between the two coordinate systems is illustrated in Figure E-3. In the primed system the pattern is described in terms of the elevation angle ϵ and azimuth ϕ as shown in Figure E-1. The necessary transformations between the two are performed in the opening statements of FUN1 and FUN2. The latter statements simply evaluate the pattern over one of six domains in FUN1 and over five domains in FUN2.

```
                 PROGRAM GAIN (OUTPUT,TAPE6=OUTPUT)
           C
           C     THIS PROGRAM COMPUTES THE ANTENNA GAIN AND EFFECIENCY FROM
           C     ANTENNA (POWER) PATTERNS DESCRIBED IN FUN1 AND FUN2.
           C     FUN1 DESCRIBES THE POLARIZED PATTERN WHEREAS FUN2 DESCRIBES THE
           C     CROSS-POLARIZED PATTERN.
           C
           C             THIS PROGRAM WAS PREPARED BY
           C
           C              JOHN P. CLAASSEN
           C
           C             UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.
           C
000003           DIMENSION BEAM(10), EFF(10)
000003           EXTERNAL FUN1
000003           EXTERNAL FUN2
000003           DATA PI/3.14159265/
000003           DATA FACTOR/0.0174532925/
000003           WRITE (6,500)
000007       500 FORMAT (1H1//)
000007           TOTAL = 0.0
000010           DO 10 I = 1,10
000012           BEAM(I) = 0.0
000013        10 CONTINUE
000015           Y2 = PI/2.0
000016           Y1 = 0.0
000017           WRITE (6,100)
000023       100 FORMAT(30X,*ANTENNA GAIN AND EFFICIENCY FACTORS*//)
           C
           C             COMPUTE THE MAIN BEAM SOLID ANGLE
           C
000023           DO 20 I = 1,8
000025           X1 = COS(FLOAT(I)*FACTOR*0.50)
000031           X2 = COS(FLOAT(I-1)*FACTOR*0.50)
000040           CALL RINTEG (FUN1, Z, X2, X1, Y2, Y1, 2, 90, 3, 3)
000051           BEAM (I+1) = BEAM (I) + 4.0*Z
000055           TOTAL = TOTAL + 4.0*Z
000057        20 CONTINUE
           C
           C             COMPUTE THE REMAINDER BEAM SOLID ANGLE
           C
000061           X2 = COS(4.0*FACTOR)
000064           X1 = COS(120.0*FACTOR)
000071           CALL RINTEG (FUN1, Z, X2, X1, Y2, Y1, 115, 90, 3, 3)
000102           TOTAL = TOTAL + 4.0*Z
           C
           C             COMPUTE GAIN
           C
000105           GAIN1 = 4.0*PI/TOTAL
000107           GAIN2 = 10.0*ALOG10(GAIN1)
000113           WRITE ( 6,200) TOTAL, GAIN1, GAIN2
000124       200 FORMAT ( 5X,*POLARIZED BEAM SOLID ANGLE=*E14.6/
                *5X,*POLARIZED GAIN=*E14.6,* OR*F6.1,* DB*//)
           C
           C             COMPUTE THE CONTRIBTION FROM THE CROSS PATTERN
           C
000124           Y2 = 1.0
000125           X1 = COS (FACTOR*5.0)
000132           CALL RINTEG(FUN2,Z,X2,X1,Y2,Y1,5,90,4,4)
000143           TOTAL = TOTAL + 4.0*Z
           C
           C             COMPUTE BEAM EFFICIENCY
           C
000146           DO 30 I = 1,8
000150           EFF(I) = BEAM(I+1)/TOTAL
000152           WRITE (6,300) I, EFF(I)
000162       300 FORMAT(5X,*ACTIVE BEAMWIDTH IS *I3,* DEGREES.*,
                * *  CORRESPONDING EFFICIENCY IS*F6.3*   .*)
000162        30 CONTINUE
           C
           C             WRITE THE BEAM SOLID ANGLE
           C
000164           WRITE (6,400)  TOTAL
000172       400 FORMAT (/5X,*TOTAL BEAM SOLID ANGLE=*E14.6)
000172           STOP
000174           END
```

FIGURE E-4. SOURCE LISTING FOR GAIN

```
                SUBROUTINE RINTEG (FUN, ZZ,X2P,X1P,Y2P,Y1P,NNX,MMY,NNP,MMP)
        C              THIS ROUTINE EMPLOYS A QUASSIAN LEGENDRE QUADRATURE
        C              INTEGRATION PROCEDURE.  INTEGRATION OVER X-Y SEGMENTS
        C              ARE PERFORMED AFTER TRANSLATION TO (-1,1)X(-1,1).
        C              FUN= INTEGRAND FUNCTION
        C              ZZ = EITHER A PARAMETER FOR FUN OR THE INTEGRATED RESULT
        C                  OR BOTH
        C              X2P,X1P = UPPER AND LOWER LIMITS ON X
        C              Y2P,Y1P = UPPER AND LOWER LIMITS ON Y
        C              NX,MY = SEGMENTS IN X AND Y, RESPECTIVELY
        C              NP,MP = DEGREE OF PRECISION IN X AND Y, RESPECTIVELY
000015          DIMENSION SAMPLE(5,5),COEF(5,5),SX(5),SY(5),C(5,5),SV(25),CV(25)
000015          EQUIVALENCE (SV(1),SAMPLE(1,1)), (CV(1),COEF(1,1))
000015          DATA SV/
               * 1.0, -0.577350269, -0.774596669, -0.861136312, -0.906179846,
               * 1.0, 0.577350269, 0.0, -0.339981044, -0.538469310,
               * 1.0, 0.0, .774596669, 0.339981044, 0.0,
               * 1.0, 0.0, 0.0, 0.861136312, 0.538469310,
               * 1.0, 0.0, 0.0, 0.0, 0.906179846/
000015          DATA CV/
               * 1.0, 1.0, 0.555555556, 0.347854845, 0.236926885,
               * 1.0, 1.0, 0.888888889, 0.652145155, 0.478628670,
               * 1.0, 0.0, 0.555555556, 0.652145155, 0.568888888,
               * 1.0, 0.0, 0.0,           0.347854845, 0.478628670,
               * 1.0, 0.0, 0.0, 0.0,                   0.236926885/
        C
        C              CLEAR SUMMING VARIABLE
        C
000015          SUM = 0.0
        C
        C              RE-ASSIGN INPUT ARGUMENTS
        C
000016          X1 = X1P
000017          X2 = X2P
000020          Y1 = Y1P
000021          Y2 = Y2P
000022          NX = NNX
000023          MY = MMY
000025          NP = NNP
000026          MP = MMP
        C
        C              COMPUTE LENGTH OF CELL SIDES
        C
000030          DELX = (X2-X1)/FLOAT(NX)
000033          DELY =  (Y2-Y1)/FLOAT(MY)
000037          HDELX = DELX*0.50
000041          HDELY = DELX*0.50
        C
        C              FORM SAMPLE FACTOR FOR X
        C
000042          DO 20 I = 1,NP
000043          SX(I) = SAMPLE(NP,I)*HDELX
000052       20 CONTINUE
        C
        C              FORM SAMPLE FACTOR FOR Y
        C
000054          DO 30 I = 1,MP
000055          SY(I) = SAMPLE(MP,I)*HDELY
000064       30 CONTINUE
        C
        C              FORM GAUSSIAN WEIGHTS
        C
000066          DO 50 I = 1,NP
000067          DO 40 J = 1,MP
000070          C(I,J) = COEF(NP,I)*COEF(MP,J)
000104       40 CONTINUE
000106       50 CONTINUE
        C
        C              INTEGRATE IN STRIPS OF DELX
        C
000110          X2 = X1
000112          DO 90 N=1,NX
000113          X1=X2
000115          X2=X1+DELX
000117          XM = X1 + HDELX
        C
        C              INTEGRATE ALONG Y
        C
000120          Y2 = Y1P
000121          DO 80 M=1,MY
000123          Y1 = Y2
000125          Y2 = Y1 + DELY
000127          YM = Y1 + HDELY
        C
        C              TRANSFORM TO CELL (-1,1)X(-1,1)
```

Figure E-4. (Continued)

```
       C
000130       DO 70 I=1,NP
000132       X=SX(I)*XM
000135       DO 60 J=1,MP
       C
       C            FORM PARTIAL SUM
       C
000136       Y=SY(J)*YM
000141       SUM = SUM+C(I,J)*FUN(X,Y)
000157    60 CONTINUE
000161    70 CONTINUE
000163    80 CONTINUE
000166    90 CONTINUE
       C
       C            APPLY JACOBIAN
       C
000170       ZZ  = SUM*DELX*DELY*0.250
000173       RETURN
000174       END


             FUNCTION FUN1 (X,Y)
       C
       C                    POLARIZED PATTERN
       C            THIS FUNCTION SUBROUTINE EVALUATES THE PATTERN INTENSITY
       C            IN THE PRIMED COORDINATE SYSTEM.  THE PATTERN IS
       C            EXPRESSED IN TERMS OF ELEVATION AND AZIMTH IN THAT
       C            COORDINATE SYSTEM. THE INPUT ARGUMENTS, HOWEVER, ARE
       C            DESCRIBED IN AN UNPRIMED COORDINATE SYSTEM RELATED
       C            TO THE PRIMED SYSTEM BY A ROTATION ABOUT THE
       C            Y AXIS.  THE BORE SIGHT IS ON THE X-PRIME AXIS IN
       C            THE PRIMED SYSTEM AND ON THE Z AXIS IN THE UNPRIMED
       C            SYSTEM.
       C
       C
       C            TRANSFORM TO PATTERN COORDINATES
       C
000005       SINX = SQRT(1.0 - X*X)
000013       COSY = COS(Y)
000016       SINY=SQRT(1.0-COSY*COSY)
000022       COSTP =-SINX*COSY
000024       SINTP = SQRT(1.0-COSTP*COSTP)
000031       THETAP=ATAN2(SINTP,COSTP)
000034       PHIP=  ATAN2(SINX*SINY,X)
000045       APHIP = ABS(PHIP)
000046       AESPP = ABS(1.57079633-THETAP)
000051       ESQ = AESPP*AESPP
000052       PSQ = PHIP*PHIP
       C
       C            EVALUATE THE PATTERN FUNCTIONS
       C
000054       AMXEP = AMAX1(AESPP,PHIP)
000057       IF (AMXEP .LE. 0.0291 ) GO TO 100
000062       IF (AMXEP .LE. 0.0349 ) GO TO 200
000064       IF (AMXEP .LE. 0.0611 ) GO TO 300
000067       IF (AMXEP .LE. 0.105 ) GO TO 400
000071       IF (AMXEP .LE. 0.628 ) GO TO 500
000074       XI = -SINTP*COS(PHIP)
000077       XI = ATAN2(SQRT(1.0-XI*XI),XI)
000106       XID = XI*57.2957796
000110       IF (XID .LT. 100. .AND. XID .GT. 60. ) GO TO 600
000123       FUN1 = 0.0
000124       RETURN
000124   100 FUN1 = EXP ( -4690.*ESQ -3280.*PSQ )
000133       RETURN
000133   200 FUN1 =EXP(- 121.1*AESPP-3280.*PSQ)
000142       RETURN
000142   300 FUN1 = 0.0398*EXP(-820.7*ESQ-1072.*PSQ)
000152       RETURN
000152   400 FUN1 = 0.0159*EXP(-486.*ESQ-53.6*APHIP)
000162       RETURN
000162   500 FUN1 = 0.000159*EXP(-8.59*SQRT(ESQ+PSQ))
000173       RETURN
000173   600 TANPSI = SINTP/COSTP*SIN(PHIP)
000177       PSI = ATAN(TANPSI)
000201       COSPS = COS(PSI)**2
000204       SINPS = SIN(PSI)**2
000207       FUN1 = (3.16E-05*COSPS + 6.31E-06*SINPS)*
            * EXP(-(XID-80.)**2/(106.*COSPS + 36.*SINPS))
000224       RETURN
000225       END
```

Figure E-4. (Continued)

```
              FUNCTION FUN2 (X,Y)
         C
         C                  CROSS-POLARIZED PATTERN
         C           THIS FUNCTION SUBROUTINE EVALUATES THE PATTERN INTENSITY
         C           IN THE PRIMED COORDINATE SYSTEM.  THE PATTERN IS
         C           EXPRESSED IN TERMS OF ELEVATION AND AZIMTH IN THAT
         C           COORDINATE SYSTEM. THE INPUT ARGUMENTS, HOWEVER, ARE
         C           DESCRIBED IN AN UNPRIMED COORDINATE SYSTEM RELATED
         C           TO THE PRIMED SYSTEM BY A ROTATION ABOUT THE
         C           Y AXIS.  THE BORE SIGHT IS ON THE X-PRIME AXIS IN
         C           THE PRIMED SYSTEM AND ON THE Z AXIS IN THE UNPRIMED
         C           SYSTEM.
         C
         C
         C           TRANSFORM TO PATTERN COORDINATES
         C
000005        SINX = SQRT(1.0 - X*X)
000013        COSY = COS(Y)
000016        SINY=SQRT(1.0-COSY*COSY)
000022        COSTP =-SINX*COSY
000024        SINTP = SQRT(1.0-COSTP*COSTP)
000031        THETAP=ATAN2(SINTP,COSTP)
000034        PHIP=  ATAN2(SINX*SINY,X)
000045        APHIP = ABS(PHIP)
000046        AESPP = ABS(1.57079633-THETAP)
000051        ESQ = AESPP*AESPP
000052        PSQ = PHIP*PHIP
         C
         C           EVALUATE THE PATTERN FUNCTIONS
         C
000054        AMXEP = AMAX1(AESPP,PHIP)
000057        IF (AMXEP .LE. 0.0291 ) GO TO 100
000062        IF (AMXEP .LE. 0.0349 ) GO TO 200
000064        IF (AMXEP .LE. 0.0611 ) GO TO 300
000067        IF (AMXEP .LE. 0.105 ) GO TO 400
000071        FUN2 = 0.0
000072        RETURN
000073    100 FUN2 = 0.0063*EXP (-4690.*ESQ-3280.*PSQ)
000103        RETURN
000103    200 FUN2 = 0.0063*EXP ( -4690.*ESQ -3280.*PSQ )
000113        RETURN
000113    300 FUN2 = 0.00025*EXP(-821.*PSQ-821.*ESQ)

000123        RETURN
000123    400 FUN2 = 0.00025*EXP(-821.*PSQ-821.*ESQ)
000133        RETURN
000133        END
```


Figure E-4. (Continued)

CRINC LABORATORIES

Chemical Engineering Low Temperature Laboratory

Remote Sensing Laboratory

Flight Research Laboratory

Chemical Engineering Heat Transfer Laboratory

Nuclear Engineering Laboratory

Environmental Health Engineering Laboratory

Information Processing Laboratory

Water Resources Institute

Technology Transfer Laboratory

CRINC