X-551-70-212

PREPRINT

NASA TM X- 65314

THE CONVERSION OF GODDARD RANGE AND RANGE RATE TRACKING DATA TO METRIC UNITS

J. F. COOK

JUNE 1970

GSFC

GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND

X-551-70-212
PREPRINT

THE CONVERSION OF GODDARD RANGE AND RANGE RATE TRACKING DATA TO METRIC UNITS

by

J. F. Cook

June 1970

Goddard Space Flight Center
Greenbelt, Maryland

PRECEDING PAGE BLANK NOT FILMED.

CONTENTS

CONTENTS—Continued

ILLUSTRATIONS

TABLES

THE CONVERSION OF GODDARD RANGE AND RANGE RATE TRACKING DATA TO METRIC UNITS

ABSTRACT

The primary purpose of the tracking data conversion described in this report is to permit tracking system analysis prior to orbit computation. The fundamental tracking data recorded during the track of a space vehicle by a ground station seldom directly expresses the physical measurements involved. In order to directly interpret measured tracking data in terms of physical phenomena, prior to orbit or trajectory computation, a conversion of fundamental tracking data to metric form is required. The data conversion capability described in this report was implemented to provide a straightforward means of converting GRARR tracking data to metric form (i.e., range, range rate and angles) free from any data preprocessing such as mathematical smoothing or statistical editing. Such metric data is then used as basic information in post-flight tracking system analysis.

SUMMARY

Tracking data at the STADAN Goddard Range and Range Rate (GRARR) tracking stations is punched on paper tape in the standard Baudot 5-level teletype code. It is sent to the Goddard Space Flight Center where it is recorded on magnetic tape in a format not directly usable by trajectory computation engineers or scientists. The "STADAN" tracking data conversion program converts the GRARR data to MKS units of range, range rate and angles. The processed data is recorded on magnetic tape. The metric data is also printed out with a summary of the data points processed or rejected.

Data stretches often contain invalid or superfluous data which is of no direct interest. The STADAN tracking data conversion program thus also deletes unwanted data stretches; that is, data falling outside of desired timing and tracking observable limits.

The time tagging in this conversion is ground receive time and the desired translation to time at the tracked spacecraft is done in the orbit or trajectory computation program. The VHF and S-Band GRARR raw data formats are described in detail in reference 1. The overall Mission and Trajectory Analysis Division tracking data conversion and analysis scheme, of which the STADAN program is an integral part, is described in reference 2.

A thorough discussion of GRARR system sources of measurement error is presented in reference 3.

THE CONVERSION OF GODDARD RANGE AND RANGE RATE TRACKING DATA TO METRIC UNITS

1.0 INTRODUCTION

The "STADAN" program is designed to provide a means of converting Goddard Range and Range Rate (GRARR) data to metric form without such data processing as mathematical smoothing or statistical editing. The program converts, edits, and reformats tracking data into a standardized format.

2.0 PURPOSE

Tracking data from the STADAN network is provided to Goddard users on magnetic tape in a format not directly usable by trajectory computation engineers or scientists. In addition, it often contains invalid or superfluous data which is of no direct interest. The "STADAN" program converts the fundamental or "raw" tracking data to units of km, km per second, and degrees, and deletes unwanted data stretches, that is, data exceeding timing and tracking observable limits.

The "STADAN" program computes azimuth (AZ) and elevation (EL) format X- and Y-angles in degrees (both are provided to the user), calculates range in kilometers, and converts biased Doppler count to average range rate in kilometers per second. The processed data is recorded on magnetic tape in a standardized format. The rejected data is printed out with a summary of the data accepted or rejected.

3.0 INPUTS

Table 1

STADAN Program Input Tape Format

Character	Description	Format
1-2	Station number: 22-Tananarive (TAN) 26-Rosman (ROS) 27-Santiago (SAN) 28-Fairbanks (FBK) 52-Carnarvon (CRO)	I2

Table 1 (continued)

Character	Description	Format
3-5	GRARR system satellite numbers:	I3
	<u>S-Band</u>	
	OGO-A 454	
	OGO-B 649	
	OGO-C 581	
	OGO-D 673	
	OGO-E 814	
	OGO-F 951	
	GEOS-A 589	
	GEOS-B 802	
	<u>VHF</u>	
	IMP-A 346	
	IMP-B 460	
	IMP-C 542	
	IMP-D 658	
	IMP-E 770	
	IMP-F 751	
	IMP-G 953	
6-8	Day of year	I3
9-10	Hour	I2
11-12	Minute	I2
13-14	Second	I2
15-19	X-Angle	F5.2
20-24	Y-Angle	F5.2
25-32	First range measurement	F8.8
33-40	Second range measurement	F8.8
41-48	Third range measurement	F8.8
49-56	Fourth range measurement	F8.8
57-63	First range rate measurement	F7.0
64-70	Second range rate measurement	F7.0
71-77	Third range rate measurement	F7.0
78-84	Fourth range rate measurement	F7.0
85	Band indicator (C1):	I1
	0 = S-Band frequency (1799.200 MHz) for 3.2 MHz subcarrier	

Table 1 (continued)

Character	Description	Format
	1 = S-Band frequency (1800.000 MHz) for 2.4 MHz subcarrier 2 = S-Band frequency (1801.000 MHz) for 1.4 MHz subcarrier 3 - VHF frequency 4 - PLL operation	
86	Sampling rate and resolution tone indicator (C2): Resolution / Data Rate 0 = 20 kHz, 1/sec. 1 = 20 kHz, 2/sec. 2 = 20 kHz, 4/sec. 3 = 20 kHz, 6/min 4 = 100 kHz or 500 kHz, 1/sec. 5 = 100 kHz or 500 kHz, 2/sec. 6 = 100 kHz or 500 kHz, 4/sec. 7 = 100 kHz or 500 kHz, 6/min	I1
87-88	Range ambiguity	I2

3.1 Tape Input

The program reads one input tape file per program run. The format is described in table 1. This format is established from the raw data tracking tape by the STADAN pre-processor program. The STADAN pre-processor program is described in Appendix A.

3.2 Card Input

A data card is used to input the time span of data to be selected for processing. The card is read by the supervisory driver.

NOTE

The Santiago tracking station still supplies range-rate data that is transmitter frequency dependent. If Santiago data is to be processed, the transmitter frequency, in MHz, must be included on the data card.

4.0 CALCULATIONS

The program performs the following checks and calculations for each tracking data point (record) read from the input tape. Only the data points that pass all checks are included in the program output.

4.1 Time Check

Time from an input tape record is converted to forms suitable for processing and output and is checked for validity, or acceptable limits. Only data points occurring within the following time ranges are accepted for further processing:

$$1 \le \text{Day} \le 366$$
$$0 \le \text{Hour} < 24$$
$$0 \le \text{Minute} < 60$$
$$0 \le \text{Second} < 60$$

4.2 Azimuth and Elevation Computation

The following equations are used to compute azimuth and elevation from X- and Y-angles:

$$EL = \arcsin(\cos X \cos Y) \tag{1}$$

$$\begin{aligned} AZ &= \arctan(\sin X/\cos Y) \\ &= \arctan[(\sin X \cos Y/\cos EL)/(\sin Y/\cos EL)] \end{aligned} \tag{2}$$

Azimuth follows the usual convention running from 0° to 360° with 0° corresponding to true North and 90° to East. Elevation varies from 0° to 90°, 0° corresponding to the horizon.

4.3 Range Computation (All Stations Except Santiago)

The following equation is used to compute range in kilometers:

$$\text{Range} = \left(\text{Range}_{\text{Raw}} + \frac{\text{RA}}{8.}\right)\left(\frac{\text{C}}{2.}\right)(10^{-3}) \tag{3}$$

where:

Range = the computed range in kilometers,

$\text{Range}_{\text{Raw}}$ = the range measurement from the data record which has been corrected on-site for station delay. At VHF, transponder delay has been subtracted on-site. At S-Band, the "STADAN" program subtracts transponder delay.

RA = the number of 8 Hz ambiguities which have been added to the format time interval reading, characters 87-88 in Table I,

C = freespace speed of light of $2.997925(10^8)$ meters/second. (Reference 4)

For data before GRARR system modification, the range ambiguity is obtained from a separate integrated trajectory program based on simplified celestial mechanics which requires an input starting vector.

The GRARR system has been modified recently to provide unambiguous range in which case the integrated trajectory will no longer be required for recent data.

4.4 Range Rate Computation (Santiago only)

The range rate or spacecraft velocity relative to the tracking station is determined from the Doppler shift of a signal transmitted to the spacecraft and returned to the ground station. The range rate reading recorded in the data format is the number of cycles of reference frequency occurring during the time required to count "N" cycles of two-way Doppler plus bias frequency. This number, C_o, may be related to the doppler and the average range rate ($\dot{r}$) according to the following expression:

Table 2

Bias and Counter Reference Frequencies and Equations Depicting Relationship Between C_0 and Spacecraft Velocity. (Reference 1)

		Rosman, Tananarive, Carnarvan and Alaska			Santiago	
		S-Band		VHF	S-Band	VHF
		PLL*	Crystal	Crystal	Crystal	Crystal
Inserted Doppler Bias Frequency	f_b	$\frac{\rho f_t}{4500}$	$\frac{f_t}{3600}$	$\frac{f_t}{5000}$	500 kHz	30 kHz
Doppler Time Interval Reference Frequency	f_r	$\frac{\rho f_t}{225}$	$\frac{f_t}{180}$	$\frac{f_t}{15}$	10 MHz	10 MHz
Number of Reference Cycles Counted	C_0	$\left(\frac{\frac{N}{225}}{\frac{1}{4500}+\left(\frac{-2\bar{\dot{r}}}{C+\bar{\dot{r}}}\right)}\right)$	$\left(\frac{\frac{N}{180}}{\frac{1}{3600}+\left(\frac{-2\bar{\dot{r}}}{C+\bar{\dot{r}}}\right)}\right)$	$\frac{N/15}{\frac{1}{5000}+\left(\frac{-2\bar{\dot{r}}}{C+\bar{\dot{r}}}\right)}$	$\frac{10^7 N}{500{,}000+\left(\frac{-2\bar{\dot{r}}}{C+\bar{\dot{r}}}\right)f_t}$	$\frac{10^7 N}{30{,}000+\left(\frac{-2\bar{\dot{r}}}{C+\bar{\dot{r}}}\right)f_t}$

*ρ = phase-locked transponder multiplication constant

NOTE: See equation 5 for definition of various constants.

Table 3

Goddard Range and Range Rate
"N" Counter Numbers

Recording Rate	Carrier			
	S-Band ("N" Counter)*	S-Band Count Output (Zero Doppler)**	VHF ("N" Counter)*	VHF Count Output (Zero Doppler)**
4 per second	65,503	1,310,060	4,093	1,364,333
2 per second	131,007	2,620,140	8,187	2,729,000
1 per second	229,263	4,585,260	14,328	4,776,000
6 per minute	3,133,956	62,679,120	182,182	60,727,333

*Note: S-Band and VHF "N" counter numbers equal the number of cycles of Doppler plus bias counted by the "N" counter of the Range Rate Extractor.

**Note: $\dot{R}$ Data Format reading at Zero Doppler.

$$C_o = \frac{Nf_r}{f_b + f_d} = \frac{Nf_r}{f_b + \left(\frac{-2\bar{\dot{r}} f_t}{C + \bar{\dot{r}}}\right)} . \tag{4}$$

Therefore:

$$\bar{\dot{r}} = \frac{c\left(f_b - \frac{Nf_r}{C_o}\right)}{2f_t - \left(f_b - \frac{Nf_r}{C_o}\right)} \tag{5}$$

where

$\bar{\dot{r}}$ = average range rate,

C = speed of light ($2.997925(10^8)$ meters/second)

f_b = inserted bias frequency to permit $\bar{\dot{r}}$ sign determination (see table 2),

N = N-counter number (a constant, see table 3),

C_o = average range rate measurement on the input tape corresponding to the number of reference frequency, f_r, cycles counted in the time required to accumulate "N" Doppler plus bias cycles (a variable),

f_t = transmitted signal from which Doppler is extracted (input on the data card).

Table 4

STADAN Program Output Tape Format

Character	Description	Format
1-2	Station number	I2
3-5	Satellite number	I3
6-7	Blank	
8-10	Day of year	I3
11-12	Hour	I2
13-14	Minute	I2
15-23	Second	F9.6
24-30	X-angle	F7.3
31-37	Y-angle	F7.3
38-44	Azimuth	F7.3
45-50	Elevation	F6.3
51-62	Range	F12.5
63-71	Range rate	F9.6
72	System indicator	I1

The modified GRARR system (Rosman, Alaska, Tananarive, Carnarvon) provides a method of Doppler extraction which removes the dependency of the velocity measurement on the transmit frequency by deriving the reference and bias frequencies from the same source as the transmitted frequency. The expression for C_o for spacecraft using the GRARR S-Band crystal transponder illustrates this independence of the transmit frequency:

$$C_o = \frac{N f_r}{f_b + f_d} = \frac{N/180}{\frac{1}{3600} + \left(\frac{-2\bar{\dot{r}}}{C + \bar{\dot{r}}}\right)} . \tag{6}$$

Therefore:

$$\bar{\dot{r}} = \frac{-C\left(\frac{N/180}{C_o} - \frac{1}{3600}\right)}{\left(\frac{N/180}{C_o} - \frac{1}{3600}\right) + 2} . \tag{7}$$

The values of bias frequency, f_b, and counter reference frequency, f_r, are listed in table 2 along with the equations showing the relationship between the data format range rate format number, C_o, and the spacecraft velocity.

4.5 Time Tagging

Each of the data types of angles, range, and range rate is associated with a particular "time tag" (Reference 2). In the GRARR system there is a raw data frame time, T_F, followed by four frames of data at a data rate T_{DR}. The appropriate time tag for subsequent frames is, therefore given by:

$$T_S = T_F + kT_{DR} \text{ seconds} \tag{8}$$

where

T_S = frame time

T_F = basic frame time

T_{DR} = data rate

$k = 0, 1, 2, 3$

The ground measurement time for completion of the range measurement

$$T_{RM} = T_S + T_R + T_w \tag{9}$$

where

T_{RM} = range time tag (ground measurement time)

T_S = frame time

T_W = inserted WWV time delay correction

T_R = time increment corresponding to relative phase difference between transmitted and received sidetone frequency.

The ground measurement time for completion of the range rate measurement is given by:

$$T_{\dot{R}M} = T_S + \Delta T + T_W \qquad (10)$$

where

$T_{\dot{R}M}$ = range rate time tag

T_S = frame time

T_W = inserted WWV time delay correction

ΔT = measured Doppler count time interval required to accumulate N cycles of bias frequency plus cycles of carrier phase change.

Equations (4) and (6), it should be noted here, are stated in terms of spacecraft time, and suitable adjustment of the time tagging must be made in the orbit determination program.

The GRARR angles are time tagged at ground receive time.

That is:

$$T_a = T_F + T_w \text{ seconds} \qquad (11)$$

where

T_F = basic frame time

T_W = WWV propagation correction

T_a = angle time tag

Note that only one angle read-out is available in the raw data for every four frames of range and range rate data.

For tracking data recorded after November 1968, WWV correction has been made on-site, i.e., $T_W = 0$.

```
69181  8  0   0   69181  10  0   0       START TIME, STOP TIME

DLOW =  7121.3    MDAY =  7121.4


         STATION NUMBER HAS CHANGED
DATA RECORDED PRIOR TO GRARR SYSTEM MODIFICATION

28  951  181   4  25  29  -42.29  71.29  .00834004  .00836904  .00839826  .00842767  5279766.  5285109.  5290400.  5295653.  2  4  0
28  951  181   4  25  33  -40.46  72.29  .00845728  .00848706  .00851702  .00854717  5300857.  5306016.  5311131.  5316194   2  4  0
28  951  181   4  25  37  -38.57  73.29  .00857758  .00860799  .00863866  .00866953  5321219.  5326195.  5331126.  5336015.  2  4  0
28  951  181   4  25  41  -36.54  74.22  .00870058  .00873179  .00876314  .00879468  5340861.  5345657.  5350414.  5355323.  2  4  0

         STATION NUMBER HAS CHANGED

26 10   181  938  15.000000-75.495  58.835  329.651  7.447     0.0          0.0        2
26 10   181  938  15.012744  0.0     0.0       0.0    0.0   1910.27331     0.0        2
26 10   181  938  15.200104  0.0     0.0       0.0    0.0      0.0        -6.067524   2
26110   181  938  16.012704  0.0     0.0       0.0    0.0   1904.20851     0.0        2
26119   181  938  16.200140  0.0     0.0       0.0    0.0      0.0        -6.0588002  2
26210   181  938  17.012663  0.0     0.0       0.0    0.0   1898.15270     0.0        2
26210   181  938  17.200177  0.0     0.0       0.0    0.0      0.0        -6.049949   2
26310   181  938  18.012623  0.0     0.0       0.0    0.0   1892.10589     0.0        2
26310   181  938  18.200215  0.0     0.0       0.0    0.0      0.0        -6.041041   2

            TOTAL RECORDS READ =  5       ACCEPTED =  1       REJECTED =  4
```

Figure 1. Sample Printer Output

5.0 OUTPUTS

5.1 Tape Output

The program records the edited, processed data on an output magnetic tape in a standardized data format (see table 4). Tapes in this format may be recorded with any density, track level, recording technique, or blocking format that can be defined in the JCL of the IBM 360 Operating System. A file mark is placed at the end of the data.

5.2 Printer Output

The program provides a printout with each program run. The input data card is printed. Also printed are the converted time, all rejected records, and a summary of the records read, accepted, and rejected. Figure 1 shows a sample of the printer output.

6.0 ERROR COMMENTS

Except for the print of all rejected records and the rejected records count, no error comments are made.

7.0 IMPLEMENTATION STATEMENTS

The program is designed for use on the IBM 360 Computer, Models 75, 91, and 95 at Goddard Space Flight Center. It is written in FORTRAN IV for use with the FORTRAN H compiler, release 16.

The FORTRAN I/O logical unit assignments are given under OPERATING INSTRUCTIONS.

8.0 SUBROUTINES

The program consists of the STADAN (main program), ABNORM, SETCON, SETSAN, STACHK, TMECHK, and TRACE subroutines. Figure 2 describes the general program flow. The following paragraphs describe the STADAN program. Figures 3 through 9 are the FORTRAN listings.

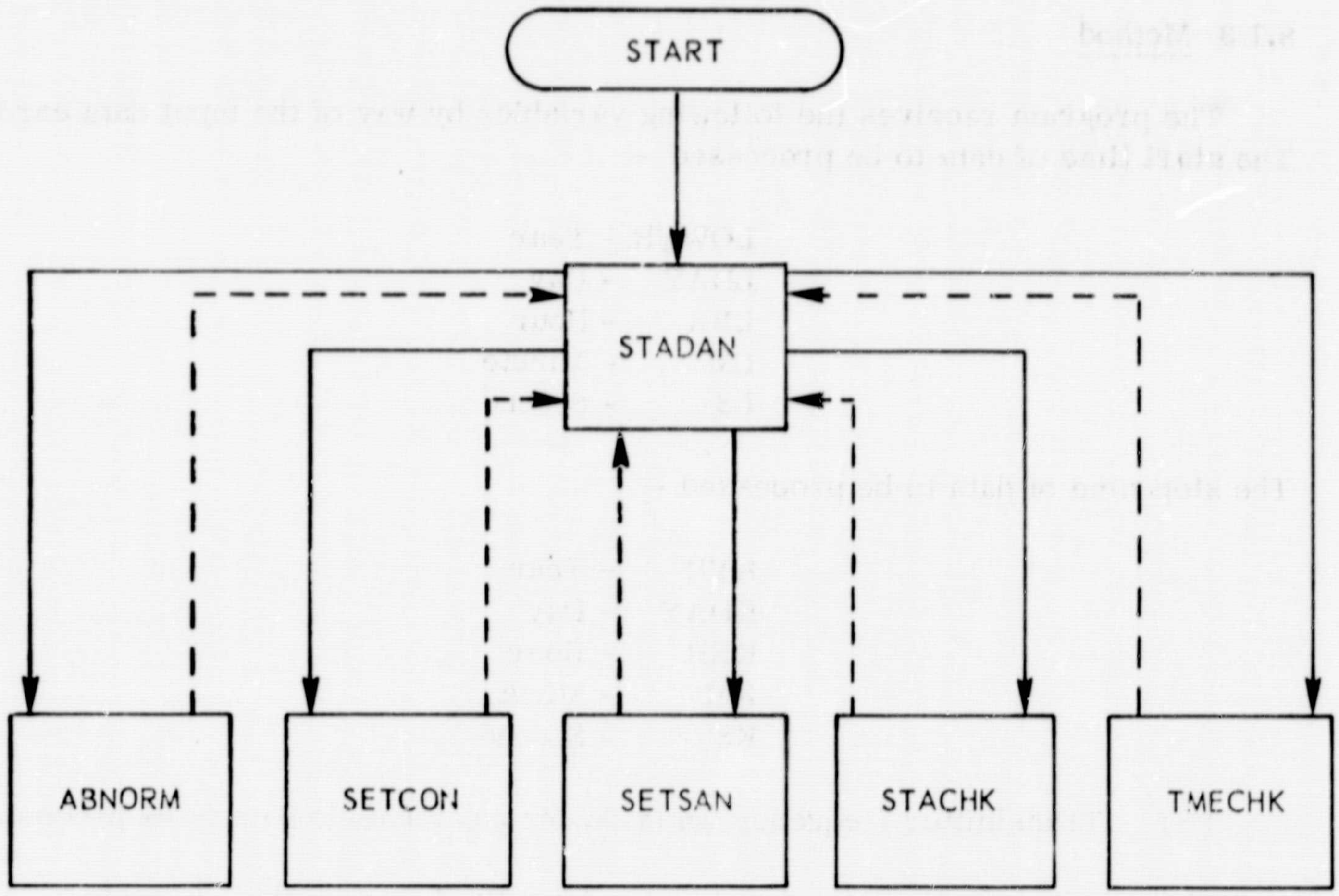


Figure 2. General Program Flow

8.1 STADAN Program

8.1.1 Purpose

The STADAN program edits and processes tracking data at VHF and S-Band frequencies and records the data on an output magnetic tape in a standardized format.

8.1.2 Restrictions

The tracking data input tape must be in the format described in table 1. This format is established by the STADAN pre-processor program described in Appendix A.

Printer output and tracking data output tape must be available. (Refer to OPERATING INSTRUCTIONS for logical unit assignments.)

The input data card must be available.

8.1.3 Method

The program receives the following variables by way of the input data card: The start time of data to be processed –

LOWYR	- Year
LDAY	- Day
LHR	- Hour
LM	- Minute
LS	- Second

The stop time of data to be processed –

KYR	- Year
KDAY	- Day
KHR	- Hour
KM	- Minute
KS	- Second

TXF - Transmitter frequency, in MHz, for any Santiago data to be processed.

The program reads data tape records formatted as described in table 1. Each record is tested for the start and stop time span limits. The day-time segment of the record is tested for acceptable limits in a call subroutine ABNORM with the following variables as arguments:

ID	- Day of year
IH	- Hour
M	- Minute
IS	- Second
IFLAG	- Data acceptable/not acceptable flag, returned by subroutine ABNORM.

Upon return from subroutine ABNORM, depending on the flag setting, the record will either be rejected and another read, or accepted and further processed.

An accepted record's angle measurements are processed and AZ/EL conversions are made.

A call to subroutine STACHK is made, with the following arguments, to determine if the data was recorded before or after the GRARR system modification date.

IST - Station number
IYR - Year
ID - Day
IGOTO - Before/after modification date flag.

Upon return from subroutine STACHK, where the flag IGOTO is set to reject or accept the record, processing either continues or the rejected record and the following message are printed before the next record is read:

DATA RECORDED PRIOR TO GRARR SYSTEM MODIFICATION

If the data is acceptable, a call either to subroutine SETSAN, for Santiago data only, or to subroutine SETCON, for all other stations, is made to set up constants for calculations of range rate.

As noted in the raw data format, there are four range and four range rate measurements contained in a record. Each of these measurements is processed and stored in arrays. The time element between two records is split into nine equal parts. The data is written on an output magnetic tape as follows:

(1) one record of time-angle measurements

(2) eight records alternating four time-range measurements with four time-range-rate measurements.

Processing continues until an end-of-file is detected on the input data tape. The data summary is then printed as follows:

TOTAL RECORDS READ = ACCEPTED = REJECTED =

The total records output should equal nine times the accepted records count.

8.2 ABNORM SUBROUTINE

8.2.1 Purpose

The ABNORM subroutine tests the time element of the input data record for normal limitations.

8.2.2 Restrictions

The tracking data input tape must be in the format described in table 1. This format is established by the STADAN pre-processor program described in Appendix A.

```
SUBPROGRAM MAIN

PROGRAM TYPE - MAIN

INPUT/OUTPUT

TYPE OPERATION        FORTRAN LOGICAL UNIT       TYPE ACCESS

   READ                        5                   SEQUENTIAL
   WRITE                       6                   SEQUENTIAL
   READ                       14                   SEQUENTIAL
   REWIND                     14
   WRITE                      16                   SEQUENTIAL
   REWIND                     16
   ENDFILE                    16

THIS SUBPROGRAM HAS  210 SOURCE CARDS.

ADDITIONAL ENTRY POINTS: NONE

COMMON REQUIRED: NONE

SUBPROGRAMS REQUIRED:  TRACE     DATAN2    DSIN      DARSIN
                       DCOS      ABNORM    IBCOM#    SETCON
                       SETSAN    STACHK    TMECHK

THE TOTAL PROGRAM LENGTH IS     6218 BYTES.
```

Figure 3. STADAN Program Automated Description and FORTRAN Listing

```
      IMPLICIT REAL*8 ( A-H,O-Z )                                       ST000010
C        S T A D A N                                                    ST000030
      DIMENSION R(4),RD(4),RDFIX(4),IC(2)
      DIMENSION TRM(4), TRDM(4), RAN(4), RAND(4)                        ST000080
      DIMENSION T(10), DATA(10), ICODE(10)                              ST000090
      DIMENSION ICOL(10)
      DATA C, IYR, BL, Q, ISYS/ 299792.5D0,66,0.D0,57.295779D0,2 /
      DATA SMIN,SHR,SDAY/60.D0,3600.D0,86400.D0/
      DATA DCON/.277777D-3/
      DATA MAN1/'MAN1'/
      DATA ITRACE/'MAIN'/
      DATA CMUS/-299792.5D0/
      DATA REFREQ/1.D7/
C                                                             *
C                  S T A D A N  DATA CARDS                    *
C                                                             *
C        1)   YYDDD_HHMM_SS______YYDDD_HHMM_SS__TTTT.TTT      *
C             START AND STOP TIME TO ACCEPT DATA              *
C             IF THERE IS ANY SANTIAGO DATA, THE TRANSMITTER  *
C             FREQUENCY MUST BE INCLUDED ON THE DATA CARD.    *
C                                                             *
      CALL TRACE( ITRACE )                                                        **S/R CALL**
      ITALLY=0
      IWRITE=0
      KC1=44
      KC2=44
      IREC = 0                                                          ST000130
      ISTOLD = 999                                                      ST000150
      ISTCHK=999
C        YYDDD HHMM SS  6X YYDDD HHMM SS                                ST000400
      READ(5,11)LOWYR,LDAY,LHR,LM,LS,KYR,KDAY,KHR,KM,KS,TXF                       *INPUT/OUTPUT
      WRITE(6,15)LOWYR,LDAY,LHR,LM,LS,KYR,KDAY,KHR,KM,KS,TXF                      *INPUT/OUTPUT
   11 FORMAT(I2,I3,1X,2I2,1X,I2,6X,I2,I3,1X,2I2,1X,I2,2X,F8.3)                    *INPUT/OUTPUT
   15 FORMAT(1X,I2,I3,1X,2I2,1X,I2,6X,I2,I3,1X,2I2,1X,I2,2X,F8.3)                 *INPUT/OUTPUT
      CALL TMECHK(DLOW,LOWYR,LDAY,LHR,LM,LS)                                      **S/R CALL**
      CALL TMECHK(HDAY,KYR,KDAY,KHR,KM,KS)                                        **S/R CALL**
      WRITE( 6, 12 ) DLOW, HDAY                                         ST000480  *INPUT/OUTPUT
   12 FORMAT( 1H0, 7HDLOW =  ,  F8.1  , 5X, 7HHDAY =  ,  F8.1   )       OT        *INPUT/OUTPUT
      IYR = LOWYR
      ZXF=TXF*1.D6
      IF( ISTOLD ) 9, 777, 9                                            ST000500
    9 CONTINUE                                                          ST000510
```

Figure 3. Continued

```
1000 READ(14,500,END=777,ERR=350)IST,ISAT,ID,IH,M,IS,X,Y,R,RD,IC,IRA                    *INPUT/OUTPUT
 500 FORMAT(I2,2(I3),3(I2),2(F5.2),4(F8.8),4(F7.0),2(I1),I2)                            *INPUT/OUTPUT
     ITALLY=ITALLY+1
C    WRITE(6,498)
C   1IST,ISAT,ID,IH,M,IS,X,Y,R(1),R(2),R(3),R(4),RD(1),RD(2),RD(3),
C   2RD(4),IC(1),IC(2),IRA
C 498 FORMAT(5X,I2,2(1X,I3),3(1X,I3),2(1X,F5.2),4(1X,F8.8),4(1X,F7.7),
C   12(1X,I1),1X,I2)
     CALL ABNORM( ID, IH, M, IS, IFLAG )                                 ST000720      **S/R CALL**
     IF( IFLAG .NE. 0 ) GO TO 1000                                       ST000730
     TE = ID*86400 + IH*3600 + M*60 + IS                                 ST000740
     CALL TMECHK(DTEST,IYR,ID,IH,M,IS)                                                 **S/R CALL**
     IF( DTEST .LE. DLOW .OR. DTEST .GE. HDAY ) GO TO 1002               ST000770
     IF( IREC .EQ. 0 ) GO TO 49
     IF( IST .NE. ISTOLD ) GO TO 49
     GO TO 49                                                            ST000830
 350 ITALLY=ITALLY+1
     IWRITE=IWRITE+1
     WRITE(6,351)                                                                      *INPUT/OUTPUT
 351 FORMAT(1H0,2X,'READ ERROR ENCOUNTERED'/)                                          *INPUT/OUTPUT
     GO TO 1000
1002 CONTINUE
     CALL TRACE( MAN1 )                                                                **S/R CALL**
     GO TO 1000
  49 IF(IST.NE.ISTOLD) WRITE(6,609)                                                    *INPUT/OUTPUT
 609 FORMAT(1H0,10X,26HSTATION NUMBER HAS CHANGED  )                     ST000930      *INPUT/OUTPUT
     ISTOLD=IST
     IF( X .GT. 0.0 )  X = X + .005                                      ST001040
     IF( X .LT. 0.0 )  X = X - .005                                      ST001050
     IF( Y .GT. 0.0 )  Y = Y + .005                                      ST001060
     IF( Y .LT. 0.0 )  Y = Y - .005                                      ST001070
     XR=X/Q
     YR=Y/Q
     W=DCOS(XR)*DCOS(YR)
     ELR=DARSIN(W)
     SAZ=DSIN(XR)*DCOS(YR)/DCOS(ELR)
     CAZ=DSIN(YR)/DCOS(ELR)
     AZR=DATAN2(SAZ,CAZ)
     AZ=AZR*Q
     EL=ELR*Q
     IF(AZ.LT.0.0) AZ=AZ+360.0D0
     CALL STACHK(IST,IYR,ID,IGOTO)                                                     **S/R CALL**
```

Figure 3. Continued

```
      IF(IGOTO) 1203,1203,300
 1203 IWRITE=IWRITE+1
      IF(IWRITE.GT.1) GO TO 1206
      WRITE(6,1204)                                                            *INPUT/OUTPUT
 1204 FORMAT(2X,'DATA RECORDED PRIOR TO GRARR SYSTEM MODIFICATION'/)           *INPUT/OUTPUT
 1206 WRITE(6,1205)IST,ISAT,ID,IH,M,IS,X,Y,R,RD,IC,IRA                          *INPUT/OUTPUT
 1205 FORMAT(5X,I2,2(1X,I3),3(1X,I2),2(1X,F6.2),4(1X,F9.8),4(1X,F8.0),         *INPUT/OUTPUT
     12(1X,I1),1X,I2)
      GO TO 1000
  300 CONTINUE
      IF(KC1.EQ.IC(1).AND.KC2.EQ.IC(2)) GO TO 244
  310 KC1=IC(1)
      KC2=IC(2)
      IF(IST.NE.27) GO TO 490
      CALL SETSAN(IC,XNF,BIASFR,SRATE,IBAND)                                   **S/R CALL**
      ISTCHK=IST
      GO TO 244
  490 CALL SETCON(IC,XNF,SRATE,IBAND)                                          **S/R CALL**
      ISTCHK=IST
  244 IF(IST.NE.ISTCHK) GO TO 310
      ICOL4 = 1. / SRATE + .1D-6
      ICOL5 = IBAND / 5
      DO  480 I = 1, 4
      XI = I - 1                                                       ST002130
      TRM(I)=XI*SRATE+R(I)
      RDFIX(I)=RD(I)*1.D-7
      TRDM(I)=TRM(I)+RDFIX(I)/2. -R(I)
      RA=IRA
      RAN(I)=(R(I)+RA/8.)*C/2.
      IF(IST.EQ.27) GO TO 482
      ALAMB=((XNF/180.)/RD(I))-DCON
      BLAMB=ALAMB*CMUS
      RAND(I)=BLAMB/(ALAMB+2.)
      GO TO 480
  482 DUKE=(XNF*REFREQ)/RD(I)
      CON=BIASFR-DUKE
      RAND(I)=C*CON/(2.*ZXF-CON)
  480 CONTINUE                                                         ST002210
C        *********************************************************
C        THIS BLOCK STORES AND SORTS THE NINE TIME POINTS. ******
      ICODE(1 ) = 1                                                    ST002280
      T(1) = 0.0D0
```

Figure 3. Continued

```
      J = 0                                                             ST002300
      DO 2010 I = 2,8,2                                                 ST002310
      J = J + 1                                                         ST002320
      ICODE(I) = 2                                                      ST002330
      DATA(I) = RAN(J)                                                  ST002340
 2010 T(I) = TRM(J)                                                     ST002350
      J = 0                                                             ST002360
      DO 2020 I = 3,9,2                                                 ST002370
      J = J + 1                                                         ST002380
      ICODE(I) = 3                                                      ST002390
      DATA(I) = RAND(J)                                                 ST002400
 2020 T(I) = TRDM(J)                                                    ST002410
      ICOL45 = ICOL4*10 + ICOL5
      ICOL(1) =       ICOL45
      ICOL(2) =       ICOL45
      ICOL(3) =       ICOL45
      ICOL(4) = 100 + ICOL45
      ICOL(5) = 100 + ICOL45
      ICOL(6) = 200 + ICOL45
      ICOL(7) = 200 + ICOL45
      ICOL(8) = 300 + ICOL45
      ICOL(9) = 300 + ICOL45
      K = 1                                                             ST002420
      L = 1                                                             ST002430
 2030 TM = T(K)                                                         ST002440
      DO 2040 I = K,9                                                   ST002450
      IF( T(I) .LT. TM ) GO TO 2050                                     ST002460
      GO TO 2040                                                        ST002470
 2050 TM = T(I)                                                         ST002480
      L = I                                                             ST002490
 2040 CONTINUE                                                          ST002500
      IF( L .EQ. K ) GO TO 2060                                         ST002510
      T(L) = T(K)                                                       ST002520
      T(K) = TM                                                         ST002530
      ICO = ICODE(L)                                                    ST002540
      ICODE(L) = ICODE(K)                                               ST002550
      ICODE(K) = ICO                                                    ST002560
      DAT = DATA(L)                                                     ST002570
      DATA(L) = DATA(K)                                                 ST002580
      DATA(K) = DAT                                                     ST002590
      ICO = ICOL(L)
      ICOL(L) = ICOL(K)
```

Figure 3. Continued

```
      ICOL(K) = ICO
 2060 K = K + 1                                                         ST002600
      L = K                                                             ST002610
      IF( K .GE. 9 ) GO TO 2070                                         ST002620
      GO TO 2030                                                        ST002630
C         ********************************************************
 2070  CONTINUE                                                         ST002640
      DO 2090 I = 1,9                                                   ST002650
      TEM=T(I)+TF                                                       ST002660
      ID=TEM/SDAY                                                       ST002670
      T(I)=(TF-ID*SDAY)+ T(I)                                           ST002680
      IH = T(I) / SHR                                                   ST002690
      T(I) = T(I) - IH*SHR                                              ST002700
      M = T(I) / SMIN                                                   ST002710
      T(I) = T(I) - M*SMIN                                              ST002720
      L = ICODE(I)                                                      ST002730
      ISAT = ICOL(I)
      GO TO ( 3010, 3020, 3030 ) , L                                    ST002740
 3010 WRITE(16,3600) IST,ISAT,ID,IH,M,T(I),X,Y,AZ,EL,BL,BL,ISYS                       *INPUT/OUTPUT
 3600 FORMAT(I2,I3,2X,I3,I2,I2,F9.6,3(F7.3),F6.3,F12.5,F9.6,I1)                        *INPUT/OUTPUT
C3010 WRITE(6,3601)IST,ISAT,ID,IH,M,T(I),X,Y,AZ,EL,BL,BL,ISYS
      GO TO 2090                                                        ST002790
 3020 WRITE(16,3600) IST,ISAT,ID,IH,M,T(I),BL,BL,BL,BL,DATA(I),BL,ISYS                 *INPUT/OUTPUT
C3020 WRITE( 6,3601) IST,ISAT,ID,IH,M,T(I),BL,BL,BL,BL,DATA(I),BL,ISYS
C3601 FORMAT(5X,I2,I3,2X,I3,I2,I2,F10.6,3(F8.3),F7.3,F13.5,F10.6, I1)
      GO TO 2090                                                        ST002820
 3030 WRITE(16,3600) IST,ISAT,ID,IH,M,T(I),BL,BL,BL,BL,BL,DATA(I),ISYS                 *INPUT/OUTPUT
C3030 WRITE( 6,3601) IST,ISAT,ID,IH,M,T(I),BL,BL,BL,BL,BL,DATA(I),ISYS
 2090 CONTINUE                                                          ST002850
      IREC = IREC + 1                                                   ST002860
      GO TO 1000                                                        ST002880
  777 CONTINUE                                                          ST002890
      WRITE(6,3603)ITALLY,IREC,IWRITE                                                  *INPUT/OUTPUT
 3603 FORMAT(1H0,10X,'TOTAL RECORDS READ = ',I5,10X,'ACCEPTED = ',I5,                  *INPUT/OUTPUT
     110X,'REJECTED = ',I5/)
      REWIND 14                                                                        *INPUT/OUTPUT
      WRITE(16,701)                                                                    *INPUT/OUTPUT
 701  FORMAT(2H-9)                                                                     *INPUT/OUTPUT
      END FILE 16                                                                      *INPUT/OUTPUT
      REWIND 16                                                                        *INPUT/OUTPUT
      STOP                                                              ST002910
      END                                                               ST002920
```

Figure 3. Continued

Printer output and tracking data output tapes must be available. (Refer to OPERATING INSTRUCTIONS for logical unit assignments.)

The variables passed as arguments by the STADAN program must be available.

8.2.3 Method

The subroutine receives the following variables as arguments from the calling routine:

ID	- Day of the year, loaded by STADAN
IH	- Hour, loaded by STADAN
M	- Minute, loaded by STADAN
IS	- Seconds
IFLAG	- Good/bad data flag, returned by ABNORM

Tests are made of the time elements to see if they fall within the normal ranges of the individual elements.

When one is not within the normal limits, the rejection flag is set, the record is printed out, and control is returned to the calling program.

8.2.4 Usage

The call to ABNORM is:

```
CALL ABNORM (ID, IH, M, IS, IFLAG)
```

8.3 SETCON SUBROUTINE

8.3.1 Purpose

The SETCON subroutine determines the constants to be used in calculations of range rate for all stations except Santiago.

8.3.2 Restrictions

The tracking data input tape must be in the format described in table 1. This format is established by the STADAN pre-processor program described in Appendix A.

SUBPROGRAM ABNORM

PROGRAM TYPE - SUBROUTINE

INPUT/OUTPUT

TYPE OPERATION	FORTRAN LOGICAL UNIT	TYPE ACCESS
WRITE	6	SEQUENTIAL

THIS SUBPROGRAM HAS 19 SOURCE CARDS.

ADDITIONAL ENTRY POINTS: NONE

COMMON REQUIRED: NONE

SUBPROGRAMS REQUIRED: TRACE IBCOM#

THE TOTAL PROGRAM LENGTH IS 566 BYTES.

```
      SUBROUTINE ABNORM( ID, IH, M, IS, IFLAG )                    00010
      IMPLICIT REAL*8 ( A-H, O-Z )                                  00020
C        IFLAG = 0 IMPLIES GOOD DATA                                00040
C        IFLAG .NE. 0 IMPLES DATA OUT OF NORMAL RANGE               00050
      DATA ITRACE/'ABNR'/
      CALL TRACE( ITRACE )                                                   **S/R CALL**
      IFLAG = 0                                                     00060
      IF( ID .LT. 1 .OR. ID .GT. 366 ) GO TO 77                     00070
      IF( IH .LT. 0 .OR. IH .GT. 24 ) GO TO 77                      00080
      IF( M  .LT. 0 .OR. M  .GT. 60 ) GO TO 77                      00090
      IF( IS .LT. 0 .OR. IS .GT. 60 ) GO TO 77                      00100
      GO TO 80                                                      00110
   77 IFLAG = 1                                                     00120
      WRITE( 6,600 ) ID, IH, M, IS                                  00130    *INPUT/OUTPUT
C     WRITE( 6,3601) IST,ISAT,ID,IH,M,T(I),X,Y,AZ,EL,BL,BL,ISYS
C3601 FORMAT(5X,I2,2X,2I3,2I2, F9.6,3(F7.3),F6.3,F12.5,F9.6,I1)
  600 FORMAT( 1H0, 10X, 20HTIME IS OUT OF RANGE ,10X,I3,3I2 )       00140    *INPUT/OUTPUT
   80 RETURN                                                        00150
      END                                                           00160
```

Figure 4. ABNORM Subroutine Automated Description and FORTRAN Listing

Printer output and tracking data output tapes must be available. (Refer to OPERATING INSTRUCTIONS for logical unit assignments.)

The variables passed as arguments by the STADAN program must be available.

8.3.3 Method

The subroutine receives the following variables as arguments from the calling routine:

IC	- two-dimensioned array loaded by STADAN
IC(1)	- contains the character which denotes either S-Band or VHF.
IC(2)	- contains the character which denotes the sampling rate and the highest resolution tone used.
XNF	- a variable containing the S-band or VHF "N" counter number returned by SETCON
SRATE	- a variable containing the recording or sampling rate, returned by SETCON
IBAND	- a variable indicating that the data is S-Band or VHF, returned by SETCON.

Upon entering subroutine SETCON, the values of IC determine the constants to be used by program STADAN in calculations of range rate.

After the constants are set, control is returned to the calling program.

8.3.4 Usage

The call to SETCON is:

```
CALL SETCON (IC, XNF, SRATE, IBAND)
```

8.4 SETSAN SUBROUTINE

8.4.1 Purpose

The SETSAN subroutine determines the constants to be used in the calculation of range rate for the Santiago tracking station.

SUBPROGRAM SETCON

PROGRAM TYPE - SUBROUTINE

NO INPUT/OUTPUT IS USED IN THIS SUBPROGRAM.

THIS SUBPROGRAM HAS 20 SOURCE CARDS.

ADDITIONAL ENTRY POINTS: NONE

COMMON REQUIRED: NONE

SUBPROGRAMS REQUIRED: TRACE

THE TOTAL PROGRAM LENGTH IS 714 BYTES.

```
SUBROUTINE SETCON(IC,XNF,SRATE,IBAND)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION IC(4),RECORD(8),SCOUNT(4),VCOUNT(4)
DATA RECORD/1.D0,.5D0,.25D0,10.D0,1.D0,.5D0,.25D0,10.D0/
DATA SCOUNT/.2292630D6,.131007D6,.0655030D6,3.133956D6/
DATA VCOUNT/.014328D6,.008187D6,.004093D6,.182182D6/
DATA ITRACE/'SETC'/
CALL TRACE(ITRACE)                                          **S/P CALL**
IBAND=0
J1=IC(1)+1
IF(J1.GE.4) IBAND=5
J2=IC(2)+1
SRATE=RECORD(J2)
IF(J2.GE.5) ISR=J2-4
IF(J2.LE.4) ISR=J2
LNF=ISR+IBAND-1
IF(LNF.LE.4) XNF=SCOUNT(ISR)
IF(LNF.GE.5) XNF=VCOUNT(ISR)
RETURN
END
```

Figure 5. SETCON Subroutine Automated Description and FORTRAN Listing

8.4.2 Restrictions

The tracking data input tape must be in the format described in table 1. This format is established by the STADAN pre-processor program described in Appendix A.

Printer output and tracking data output tapes must be available. (Refer to OPERATING INSTRUCTIONS for logical unit assignments.)

The variables passed as arguments by the STADAN program must be available.

8.4.3 Method

The subroutine receives the following variables as arguments from the calling routine:

IC	- two-dimensioned array loaded by STADAN
IC(1)	- contains the character which denotes either S-Band or VHF
IC(2)	- contains the character which denotes the sampling rate and the highest resolution tone used
XNF	- a variable containing the S-Band or VHF "N" counter number, returned by SETSAN
BIASFR	- a variable containing the bias frequency, returned by SETSAN
SRATE	- a variable containing the recording or sampling rate, returned by SETSAN
IBAND	- a variable indicating that the data is S-Band or VHF, returned by SETSAN.

Upon entering subroutine SETSAN, the values of IC determine the constants to be used by program STADAN in the calculation of range rate from Santiago.

After the constants are set, control is returned to the calling program.

8.4.4 Usage

The call to SETSAN is:

CALL SETSAN (IC,XNF, BIASFR, SRATE, IBAND)

SUBPROGRAM SETSAN

PROGRAM TYPE - SUBROUTINE

NO INPUT/OUTPUT IS USED IN THIS SUBPROGRAM.

THIS SUBPROGRAM HAS 26 SOURCE CARDS.

ADDITIONAL ENTRY POINTS: NONE

COMMON REQUIRED: NONE

SUBPROGRAMS REQUIRED: TRACE

THE TOTAL PROGRAM LENGTH IS 796 BYTES.

```
      SUBROUTINE SETSAN(IC,XNF,BIASFR,SRATE,IBAND)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION IC(2),RSPED(8),SRADAR(4),VRADIO(4),BFUGE(2)
      DATA RSPED/1.D0,.5D0,.25D0,10.D0,1.D0,.5D0,.25D0,10.D0/
      DATA SRADAR/.229263D6,.131007D6,.065503D6,3.133956D6/
      DATA VRADIO/.014328D6,.008187D6,.004093D6,.182182D6/
      DATA ITRACE/'SANT'/
      CALL TRACE(ITRACE)                                         ** S/R CALL **
      BIASFR=0.0D0
      IBAND=0
      J1=IC(1)+1
      IF(J1.GE.4) IBAND=5
      J2=IC(2)+1
      SRATE=RSPED(J2)
      IF(J2.GE.5) ISR=J2-4
      IF(J2.LE.4) ISR=J2
      LNF=ISR+IBAND-1
      IF(LNF.LE.4) GO TO 10
      IF(LNF.GE.5) GO TO 12
   10 XNF=SRADAR(ISR)
      BIASFR=500000.D0
      GO TO 15
   12 XNF=VRADIO(ISR)
      BIASFR=30000.D0
   15 RETURN
      END
```

Figure 6. SETSAN Subroutine Automated Description and FORTRAN Listing

8.5 STACHK SUBROUTINE

8.5.1 Purpose

The STACHK subroutine determines whether the data was recorded before or after the date of modification of the tracking stations to the GRARR system.

8.5.2 Restrictions

The tracking data input tape must be in the format described in table 1. This format is established by the STADAN pre-processor program described in Appendix A.

Printer output and tracking data output tapes must be available. (Refer to OPERATING INSTRUCTIONS for logical unit assignments.)

The variables passed as arguments by the STADAN program must be available.

8.5.3 Method

The subroutine receives the following variables as arguments from the calling routine:

ICT	- a variable containing the station number, loaded by STADAN
IYR	- a variable containing the year, loaded by STADAN
ID	- a variable containing the day, loaded by STADAN
IGOTO	- flag set depending on whether the data was recorded before or after the modification date

Upon entering subroutine STACHK, the modification date for each station is determined. The day of the year in the data record, passed as an argument, is tested against the particular stations modification date. The record is rejected in subroutine STADAN if the data was recorded prior to that stations modification date. (The modification year is set at 1969. All data prior to this will be rejected.)

NOTE

The Santiago station will not be modified. Therefore, all Santiago data will be accepted up to this point of flow of the program.

After testing is completed, control is returned to the calling program.

SUBPROGRAM STACHK

PROGRAM TYPE - SUBROUTINE

NO INPUT/OUTPUT IS USED IN THIS SUBPROGRAM.

THIS SUBPROGRAM HAS 16 SOURCE CARDS.

ADDITIONAL ENTRY POINTS: NONE

COMMON REQUIRED: NONE

SUBPROGRAMS REQUIRED: TRACE

THE TOTAL PROGRAM LENGTH IS 558 BYTES.

```
      SUBROUTINE STACHK(IST,IYR,ID,IGOTO)
      DATA ITRACE/'STCK'/
      CALL TRACE(ITRACE)                                               **S/R CALL**
      JYR=69
      IGOTO=1
      IDCHK=0
      IF(IST.EQ.22) IDCHK=145
      IF(IST.EQ.26) IDCHK=120
      IF(IST.EQ.28) IDCHK=298
      IF(IST.EQ.52) IDCHK=243
      IF(IST.EQ.27) IGOTO=1
      IF(IGOTO) 3,3,4
    4 IF(IYR.LT.JYR) IGOTO=0
      IF(ID.LT.IDCHK) IGOTO=0
    3 RETURN
      END
```

Figure 7. STACHK Subroutine Automated Description and FORTRAN Listing

8.5.4 Usage

The call to STACHK is:

CALL STACHK (IST, IYR, ID, IGOTO)

8.6 TMECHK SUBROUTINE

8.6.1 Purpose

The TMECHK subroutine converts the time element of each record plus the start and stop times of the data to be processed, (supplied by the data card) to more readily usable numbers.

8.6.2 Restrictions

The tracking data input tape must be in the format described in table 1. This format is established by the STADAN pre-processor program described in Appendix A.

Printer output and tracking data output tapes must be available. (Refer to OPERATING INSTRUCTIONS for logical unit assignments.)

The variables passed as arguments by the STADAN program must be available.

8.6.3 Method

The subroutine receives the following variables as arguments from the calling routine:

D	- a variable containing the converted time, returned by TMECHK
IYR	- a variable containing the year, loaded by STADAN
ID	- a variable containing the day of the year, loaded by STADAN
M	- a variable containing the minute, loaded by STADAN
IS	- a variable containing the second, loaded by STADAN.

Upon entering subroutine TMECHK, the time elements from the data records as well as the acceptable data time span from the data card are converted to more readily usable numbers. This makes the task of testing the data time against the time span of data to be processed easier to perform. The test is done in STADAN program.

```
SUBPROGRAM TMECHK

PROGRAM TYPE - SUBROUTINE

NO INPUT/OUTPUT IS USED IN THIS SUBPROGRAM.

THIS SUBPROGRAM HAS   17 SOURCE CARDS.

ADDITIONAL ENTRY POINTS: NONE

COMMON REQUIRED: NONE

SUBPROGRAMS REQUIRED: TRACE

THE TOTAL PROGRAM LENGTH IS      620 BYTES.
```

```
      SUBROUTINE TMECHK(D,IYR,ID,IH,M,IS)
      IMPLICIT REAL*8(A-H,O-Z)
      DATA ITRACE/'TMCK'/
      CALL TRACE(ITRACE)                                              ** S/R CALL **
      DF=IH*3600+M*60+IS
      DF=DF/86400.
      NYRS=IYR-50
      K=0
      J=52
   15 IF(IYR-J)50,50,40
   40 K=K+1
      J=J+4
      GO TO 15
   50 DD=NYRS*365+K+ID-1
      D=DD+DF
      RETURN
      END
```

Figure 8. TMECHK Subroutine Automated Description and FORTRAN Listing

After converting the time element, control is returned to the calling program.

8.6.4 Usage

The call to TMECHK is as follows:

For the start-stop time span –

```
CALL TMECHK (DLOW, LOWYR, LDAY, LH, LM, LS)

CALL TMECHK (HDAY, KYR, KDAY, KH, LM, KS)
```

For the data record time –

```
CALL TMECHK (DTEST, IYR, IH, M, S)
```

8.7 TRACE SUBROUTINE

8.7.1 Purpose

The TRACE subroutine is used primarily for debugging purposes by keeping track of the sequence of subroutine calls.

8.7.2 Restrictions

The variables passed as arguments by the calling subroutine must be available.

8.7.3 Method

The subroutine receives the following variables as an argument from the calling routine:

IA – a variable containing the literal data constant from a data statement in the calling routine.

The literal data constants are stored in an array dimensioned 90. When the array is completed, that is after 90 calls to subroutines have been made, it is dumped on unit 6. For normal runs, the unit is dummied. (Refer to OPERATING INSTRUCTIONS.)

SUBPROGRAM TRACE

PROGRAM TYPE - SUBROUTINE

INPUT/OUTPUT

TYPE OPERATION	FORTRAN LOGICAL UNIT	TYPE ACCESS
WRITE	6	SEQUENTIAL

THIS SUBPROGRAM HAS 19 SOURCE CARDS.

ADDITIONAL ENTRY POINTS: NONE

COMMON REQUIRED: NONE

SUBPROGRAMS REQUIRED: IBCOM#

THE TOTAL PROGRAM LENGTH IS 812 BYTES.

```
      SUBROUTINE TRACE( IA)
      DIMENSION IB( 90)
      DATA I/0/, IBLK/'    '/
      IF( I .EQ. 0 ) GO TO 111
100   CONTINUE
      IB(I) = IA
      I = I + 1
      IF( I .GT.  90 ) GO TO 200
      GO TO 777
111   DO 112 J = 1, 90
112   IB(J) = IBLK
      I = 1
      GO TO 100
200   I = 0
      WRITE( 6,600)(IB(J),J=1,90)                                    *INPUT/OUTPUT
600   FORMAT( 5X, 30A4 )                                             *INPUT/OUTPUT
777   CONTINUE
      RETURN
      END
```

AN END OF DATA SET HAS BEEN ENCOUNTERED.

7 SUBPROGRAMS HAVE BEEN PROCESSED.

10284 IS THE TOTAL LENGTH OF ALL THE ABOVE MODULES.

PROCESSING WILL NOW TERMINATE.

Figure 9. TRACE Subroutine Automated Description and FORTRAN Listing

8.7.4 Usage

The call to TRACE is:

CALL TRACE (ITRACE)

9.0 OPERATING INSTRUCTIONS

9.1 Logical Unit Assignments

The following FORTRAN logical units are assigned:

Logical unit 5 - card input
Logical unit 6 - printer output
Logical unit 14 - tracking data input
Logical unit 16 - tracking data output

9.2 Program Deck Setup

The program deck consists of the source deck, data card, fixed JCL cards that vary with the run parameters. A sample program deck is given below. Consult references 4 and 5 for a complete description of JCL cards needed for all possible run parameters.

Sample Deck Setup:

```
// JOBCARD
// EXEC FORTRANH
// SOURCE. SYSIN DD*
. . . Source Deck . . .
/*
// EXEC LINKGO
// GO.FT 14 F001 DD UNIT = 2400 - 9, DISP = (OLD, KEEP) LABEL = (1, BLP),
// DCB = (RECFM = FB, LRECL = 88, BLKSIZE = 8800, DEN = 3),
// VOL = SER = XXXXXX
// GO. FT 16 F001 DD UNIT =  2400 - 9, DISP = (NEW, KEEP), LABEL =
   (1, BLP),
// DCB = (RECFM = FB, LRECL = 72, BLK SIZE = 7200, DEN = 3),
// VOL = SER = XXXXXX
// GO. DATA 5 DD *
. . . Data Card . . .
/*
```

REFERENCES

1. Zillig, D. J., "Data Formats of the Goddard Range and Range Rate System and the Application Technological Satellite Range and Range Rate System," NASA-GSFC X-571-69-149, April 1969.

2. Schmid, P. E., "The Conversion of Fundamental Tracking Data to Metric Form," NASA-GSFC X-551-69-3, January 1969.

3. Grenchik, T. J. and B. H. Putney, "A Review of Goddard Range and Range Rate System Measurements and Data Processing Techniques," NASA-GSFC, 551-69-137, April 1969.

4. Vonbun, F. O. and J. P. Mayer, "Apollo Missions and Navigation Systems Characteristics," Apollo Navigation Working Group Technical Report AN 1.3, joint NASA GSFC-MSC publication as updated through January 2, 1970.

5. IBM System/360 Operating System, Job Control Language. IBM Corporation, Form C28-6539-5.

6. IBM System/360 Operating System, Supervisory and Data Management Macro-Instructions. IBM Corporation, Form C28-6647-0.

7. IBM System/360 Operating System, Utilities. IBM Corporation, Form C28-6586-9.

APPENDIX A

STADAN PRE-PROCESSOR PROGRAM

INTRODUCTION

The STADAN pre-processor program is used to reformat the tracking data into an acceptable format for processing by the STADAN program.

PROBLEM STATEMENT

Tracking data from the STADAN network is provided to Goddard users on magnetic tape in a format not usable by the STADAN program. See table A-1 for the teletype transmitted format.

The STADAN pre-processor program eliminates special characters, deletes records not in proper sequence, and reformats the data into one record.

INPUTS

Tape Input

The program reads one input tape file per program run.

Card Input

There are no data cards required.

OUTPUTS

Tape Output

The program records the edited, reformatted data on a standard label output magnetic tape. See table A-2 for the output format. A file mark is placed at the end of the data.

Printer Output

The program provides a printout with each program run. The first two input records are printed plus an output record count.

Method

Contained in the input data records are two special characters which denote sequence. The first line of the message should have a period preceding the time of year and the second line should have a slash preceding the identification data. When this sequence is detected, the special characters are deleted and the data is written on the output magnetic tape in the single record format described in table A-2.

OPERATING INSTRUCTIONS

Logical Unit Assignments

The following FORTRAN logical units are assigned:

Logical unit 5 - card input
Logical unit 6 - printer output
Logical unit 8 - tracking data input
Logical unit 9 - reformatted data output

Program Deck Setup

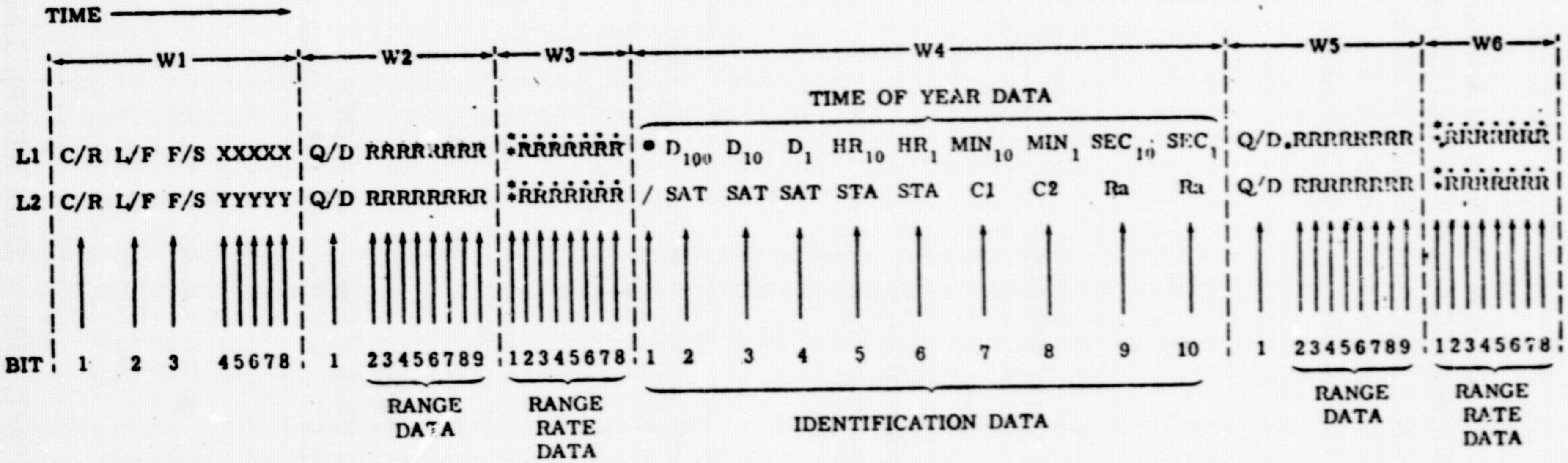
The program deck consists of the source deck, fixed JCL cards, and JCL cards that may vary with the run parameters. A sample program deck is given below. Consult references 1, 2 and 3 for a complete description of utilities programs and JCL cards needed for possible run parameters.

Sample Deck Setup:

```
// JOB CARD
// EXEC PGM = IEHINITT
// SYSPRINT DD SYSOUT = A
// LABEL DD DCB = (DEN = 3), UNIT = (2400 - 9, 1, DEFER)
INITT SER = XXXXXX, NUMBTAPE = 1, OWNER = 'XXXXXXXX', DISP =
  REWIND
/*
// EXEC FORTRANH
// SOURCE.SYSIN DD*
. . . Source Deck . . .
/*
// EXEC LINKGO
// GO. FT08F001 DD UNIT = 2400-7, DISP = (OLD, KEEP), LABEL =
  (1, BLP),
```

Table A-1

STADAN Pre-Processor Program Input Tape Format (GRARR Data Format)



C/R	CARRIAGE RETURN
L/F	LINE FEED
F/S	FIGURE SHIFT
X	ANTENNA POSITION (SIGN AND FOUR DECIMAL DIGITS)
Q/D	QUALITY DATA (SPACE INDICATES ALL LOOPS LOCKED, RANGE TONES ON, ANTENNA AUTO TRACK)
$\dot{R}$	RANGE RATE (0000000-9999999)
SEC	SECONDS (00-59)
MIN	MINUTES (00-59)
R	RANGE (00000000-99999999)
HR	HOURS (00-23)
D	DAYS (000-365)
Y	ANTENNA POSITION (SIGN AND FOUR DECIMAL DIGITS)
SAT	SATELLITE IDENTIFICATION (000-999)
STA	STATION IDENTIFICATION (00-99)
C1	SPACE CRAFT FREQUENCY
C2	SAMPLE RATE (0-4) AND RESOLUTION
Ra	RANGE AMBIGUITY • (00-63)

NOTE: DATA IS PUNCHED IN STANDARD BAUDOT 5-LEVEL CODE.

THERE ARE 52 CHARACTERS IN EACH LINE, MAKING A TOTAL OF 104 CHARACTERS TO BE TRANSMITTED SERIALLY.

•SPACE = ARC NOT IN USE; ? = ARC NOT ACQUIRED

‡0 = SAMPLE RATE AT 6/MIN; SPACE = SAMPLE RATE AT 4, 2, OR 1/SEC

```
// DCB = (RECFM = U, LRECL = 50, BLKSIZE = 50, DEN = 1, TRTCH = ET),
// VOL = SER = XXXX
// GO.FT09F001 DD UNIT = 2400-9, DISP = (NEW, KEEP), LABEL = (1, SL),
// DCB = (RECFM = FB, LRECL = 88, BLKSIZE = 8800, DEN = 3),
// VOL = SER = XXXXXX, DSN = SDATA 1
/*
```

Table A-2

STADAN Pre-Processor Program Output Tape Format

Column	Description	Designator
1-2	Station number	STA
3-5	Satellite number	SAT
6-14	Time (day, hour, minute, second)	T
15-19	X-angle	X
20-24	Y-angle	Y
25-32	First range measurement	R1
33-40	Second range measurement	R11
41-48	Third range measurement	R2
49-56	Fourth range measurement	R22
57-63	First range rate measurement	RR1
64-70	Second range rate measurement	RR11
71-77	Third range rate measurement	RR2
78-84	Fourth range rate measurement	RR22
85	Band indicator	C1
86	Sample rate and resolution tone indicator	C2
87-88	Range ambiguity	RA

```
SUBPROGRAM MAIN

PROGRAM TYPE - MAIN


INPUT/OUTPUT

TYPE OPERATION     FORTRAN LOGICAL UNIT     TYPE ACCESS


  WRITE                      6                 SEQUENTIAL
  READ                       8                 SEQUENTIAL
  WRITE                      9                 SEQUENTIAL
  ENDFILE                    9


THIS SUBPROGRAM HAS   42 SOURCE CARDS.
```

```
      IMPLICIT INTEGER*2(A-Z)
      DIMENSION LINE1(49),LINE2(49),STA(2),SAT(3),T(9),XX(5),C12R(4),
     1YY(5),R1(8),R11(8),R2(8),R22(8),RR1(7),RR11(7),RR2(7),RR22(7)
      DATA QUOTE,SLASH,BLANK/'" ','/ ',' '/
      EQUIVALENCE(LINE1(6),Q1),(LINE1(23),SP1),(LINE1(33),Q11)
      EQUIVALENCE(LINE2(6),Q2),(LINE2(23),SP2),(LINE2(33),Q22)
      EQUIVALENCE(LINE1(1),XX(1)),(LINE1(7),R1(1)),(LINE1(16),RR1(1))
      EQUIVALENCE(LINE1(24),T(1)),(LINE1(34),R11(1)),(LINE1(43),RR11(1))
      EQUIVALENCE(LINE2(1),YY(1)),(LINE2(7),R2(1)),(LINE2(16),RR2(1))
      EQUIVALENCE(LINE2(24),SAT(1)),(LINE2(27),STA(1))
      EQUIVALENCE(LINE2(29),C12R(1))
      EQUIVALENCE(LINE2(34),R22(1)),(LINE2(43),RR22(1))
      IREC=0
    6 READ(8,10,END=90)LINE1                                            *INPUT/OUTPUT
   10 FORMAT(49A1)                                                      *INPUT/OUTPUT
      IF(SP1.EQ.SLASH) GO TO 6
      IF(SP1.EQ.BLANK) GO TO 6
      GO TO 14
   14 READ(8,10,END=90)LINE2                                            *INPUT/OUTPUT
      IF(SP2.EQ.SLASH) GO TO 24
      IF(SP2.EQ.BLANK) GO TO 6
      DO 16 I=1,49
   16 LINE1(I)=LINE2(I)
      GO TO 14
   24 IF(Q1.NE.BLANK) GO TO 6
      IF(Q11.NE.BLANK) GO TO 6
      IF(Q2.NE.BLANK) GO TO 6
      IF(Q22.NE.BLANK) GO TO 6
      IF(IREC.GT.0) GO TO 19
      WRITE(6,27) LINE1                                                 *INPUT/OUTPUT
      WRITE(6,27) LINE2                                                 *INPUT/OUTPUT
   27 FORMAT(5X,49A1/)                                                  *INPUT/OUTPUT
   19 WRITE(9,15)STA,SAT,T,XX,YY,R1,R11,R2,R22,RR1,RR11,RR2,RR22,C12R   *INPUT/OUTPUT
   15 FORMAT(88A1)                                                      *INPUT/OUTPUT
      IREC=IREC+1
      GO TO 6
   90 WRITE(6,26)IREC                                                   *INPUT/OUTPUT
   26 FORMAT(10X,'AFTER EDITING AND CONSOLIDATING, THERE WERE ',I5,'RECO *INPUT/OUTPUT
     1RDS WRITTEN.')
      END FILE 9                                                        *INPUT/OUTPUT
      STOP
      END
```

Figure A-1. STADAN Pre-Processor Program Automatic Description and FORTRAN Listing