USER'S MANUAL

SSM/I ANTENNA TEMPERATURE TAPES

Revision 2

Remote Sensing Systems

1101 COLLEGE AVE., SUITE 220, SANTA ROSA, CA 95404

(707) 545-2904

RSS Technical Report 120193 Issued: December 1, 1993

USER'S MANUAL

SSM/I ANTENNA TEMPERATURE TAPES

Revision 2

By: Frank J. Wentz

Prepared by:

Remote Sensing Systems

1101 COLLEGE AVE., SUITE 220, SANTA ROSA, CA 95404 (707) 545-2904

TABLE OF CONTENTS

SECTION 1. INTRODUCTION

This document is the second revision of the Special Sensor Microwave/Imager (SSM/I) Antenna Temperature (T_A) Tapes User's Manual. The original User's Manual for the SSM/I T_A tapes was issued on March 25, 1988, and the first revision was issued on December 1, 1991. This second revision is a supplement to Revision 1 and provides the User with the following new features:

1. F10 SSM/I T_A's are now intercalibrated with the F08 observations.
2. Data inventory figures and erroneous data time windows are now available for F10 and F11.
3. More quality control procedures are now provided to the Users.

In addition, there are several other extensions that are discussed herein. The User is advised to pay particular attention to the underlined sentences in Section 2. They describe important changes that need to be considered when implementing this Revision-2 update.

A diskette is included with this manual and contains the following files:

DECODE2.FOR	-	The Revision-2 decoding Fortran routine
QUAL1.FOR	-	A quality control Fortran routine that flags erroneous observations
BADLOC.D08	-	A data file containing erroneous data periods for F08
BADLOC.D10	-	A data file containing erroneous data periods for F10
BADLOC.D11	-	A data file containing erroneous data periods for F11
BADCAL.D08	-	A data file containing erroneous calibration periods for F08
BADCAL.D10	-	A data file containing erroneous calibration periods for F10
BADCAL.D11	-	A data file containing erroneous calibration periods for F11

Routine QUAL1.FOR reads the BADLOC and BADCAL data files and uses this information to flag erroneous observations. Note that all files other than the BADCAL files are formatted ASCII files. The three BADCAL files are binary files described in Section 5.

SECTION 2. THE REVISION-2 DECODING ROUTINE: DECODE2

The Revision-2 T_A tape decoding routine is called DECODE2. The previous version (Revision-1), which was simply called DECODE, is thoroughly described in the Rev-1 SSM/I T_A User's Manual. For DECODE2, the following five changes have been made.

1. The ITB=0 option is the same as before. Unadjusted antenna temperatures are stored in the arrays TALO, ATAHI, and BTAHI. However, the ITB=1 option is different. In this case, the T_A's are corrected for along-scan biases, and the F10 T_A's are adjusted so as to agree with the F08 T_A's (see Section 6). These adjusted T_A's are stored in the arrays TALO, ATAHI, and BTAHI. Note that no adjustments are done to the 85 GHz T_A's. Brightness temperatures T_B are also computed for the ITB=1 option, and they are stored in a new common area called:

COMMON /TBDATA/ TBLO(5,64), ATBHI(2,128), BTBHI(2,128)

We now suggest the ITB=1 option always be used, since it provides both corrected T_A values and T_B values. Thus with DECODE2, the arrays TALO, ATAHI, and BTAHI always contain T_A's, and if the User wants T_B's, then they must access the new common /TBDATA/.

2. Along-scan biases for F10 and F11 are now implemented. In Rev-1 DECODE, only the F08 along-scan biases were implemented, and the F10 and F11 biases had been set to zero. When ITB=1 option is set, the along-scan biases are applied to the T_A's. The F10 along-scan biases were found by binning the T_A's according to scan position and averaging over one year of ocean observations. This is the same method that was used to find the F08 along-scan biases. We found that the F08 and F10 biases were very similar (±0.05 K). It thus appears that the along-scan bias is a stable characteristic of the SSM/I sensor and is probably due to the radiometer horn seeing the cold sky reflector at one edge of the Earth scan segment. We did not compute the along-scan biases for the F11. Rather, we assume that the along-scan bias is a stable characteristic of the SSM/I sensor, and DECODE2 simply sets the F11 along-scan biases to the F08 values.

3. The F10 T_A's are adjusted to correspond to the F08 T_A's. The adjustment coefficients were found by doing a cross-calibration between F08 and F10 over the year 1991. The F10-F08 correction is done when ITB is set to 1. This intercalibration is discussed in more detail in Section 6. No adjustment is made to the F11 T_A because we have not yet done the cross-calibration between F10 and F11. We plan to do the F11-F10 cross-calibration in the first quarter of 1994.

4. Starting in August 1992, an additional quality control procedure was implemented. This procedure detects sensor calibration errors and is discussed in Section 4. On the T_A tapes, quality flags are written to 64 bytes, which had been spare bytes. DECODE2 uses these quality flags to indicate calibration errors. When a calibration error occurs, DECODE2 appends a minus sign to the T_A and T_B value. Thus when using DECODE2 with the August 1992 data and thereafter, the User must check for negative T_A or T_B and discard these observations.

5. For a given frequency, if either the v-pol or h-pol uncorrected T_A value is outside the range 55 to 320 K, then for this pair of observations, no corrections or adjustments are done to the T_A's, and the brightness temperatures are set equal to the unadjusted T_A values. In this way, routine FDTATB in DECODE2 does not modify the anomalous T_A's (outside the 55-320 K range) nor the negative T_A's (calibration errors).

6. Previous to this update, we had been using a value of 44.75° for the F08 nadir angle. This value came from prelaunch measurements of the F08 SSM/I. Post-launch analyses now indicate that the true value of the F08 nadir angle is closer to 45.0°. This new value corrects the 10-km along-track geolocation error discussed in Section 6 of the Rev-1 SSM/I T_A User's Manual and improves the consistency between the F08 and F10 T_A's. This change in nadir angle corresponds to a 0.336° increase in the incidence angle. Accordingly DECODE2 now adds 0.336° to the old F08 incidence angle.

SECTION 3. DATA INVENTORY FIGURES AND ERRONEOUS DATA TIME WINDOWS

Enclosed are the data inventory figures and erroneous data time windows for F10 and F11 through March 1993. The method of identifying erroneous data periods for F08 is described in Section 7 of the Rev-1 SSM/I T_A User's Manual. For F10 and F11, there is one modification of the procedure. The T_A's that are outside the range of possible Earth values are excluded from the statistics shown in the figures. Thus, the F10 and F11 data inventory figures do not show these out-of-bounds (OOB) observations. The exclusion of OOB observations is done by the quality control routine QUAL1 described in Section 5.

Excluding the OOB T_A's from the figures results in a better discrimination of erroneous data periods. In the old procedure a few obviously bad observations, which could easily be identified and excluded, were producing small scatter groups in the data inventory figures that were difficult to interpret.

In producing the F10 figures prior to October 1991, the F08 data were used to specify the semimonthly T_A base map. Then, starting in October 1991, the F10 data from the previous semimonthly period were used to specify the T_A base map. There is less scatter when the F08 data are used because the time period for the base map is the same as that for the observations. The base maps for the F11 figures were generated from the F10 observations for the specified semimonthly period. The vertical bars in the figures show the time windows for erroneous data, as is discussed in the next paragraph.

We have visually inspected the F08, F10, and F11 data inventory figures and have identified the begin and end times for periods of erroneous data. Three small ASCII data files called BADLOC.D08, BADLOC.D10, and BADLOC.D11 containing these times are on the enclosed 3.5-inch diskette. BADLOC.D08 covers the period from the F08 launch in June 1987 through December 1991. BADLOC.D10 covers the period from the F10 launch in December 1990 through March 1993. BADLOC.D11 covers the period from the F11 launch in December 1991 through March 1993. The format of these files is given in Section 7 of the Rev-1 SSM/I T_A User's Manual.

In addition to the periods of erroneous data identified from the data inventory figures, BADLOC.D08 and BADLOC.D10 includes the following special time windows for data that should be excluded:

F08, day 190, 1987:	SSM/I turned on and cycling through AGC
F08, day 013, 1988:	SSM/I turned on and cycling through AGC
F10, day 342, 1990:	SSM/I turned on and cycling through AGC
F10, day 353-355, 1990:	Many observations outside possible T_A range
F10, day 5-6, 1991:	SSM/I turned on again and cycling through AGC
F10, day 353, 1991:	Many observations outside possible T_A range

SECTION 4. CALIBRATION ERRORS AND SCAN AVERAGING

Normally the SSM/I cold counts are between 200 and 2000, and the hot counts are between 1500 and 3400. Also the rms variation of the counts for a given channel and a given scan is only a few counts. Occasionally the cold and/or hot counts fall outside the specified ranges and/or show a large variation (greater than 9 counts). This is clearly erroneous data and should be excluded.

The calibration error problem is compounded by the fact that starting on October 9, 1990, Fleet Numerical Oceanography Center (FNOC) implemented new software that averages the cold and hot counts over ten pairs of A/B scans. Thus the erroneous OOB calibration data corrupts 10 pairs of scans rather than 1 pair. There are two other problems associated with the FNOC scan averaging procedure. First, the averaging is done over 10 pairs of scans without regard to data gaps. When there are data gaps exceeding a few minutes, the assumption of calibration stability over the averaging period may be violated. The second, less serious problem is that the averaging period is not centered on the scan time but is rather a forward average of 10 pairs of scans. This offset in the average time relative to the scan time produces small systematic errors of the order of 0.1 K which vary with orbit position.

Starting with the T_A data on August 29, 1992, we undo the FNOC scan averaging. Thus our F10 and F11 T_A data from August 29, 1992, and onward are computed from the calibration data for a single scan rather than 10 pairs of scans. Previous to this, our F10 and F11 T_A data are computed using the FNOC scan averaging procedure.

We are providing the User with three data files: BADCAL.D08, BADCAL.D10, and BADCAL.D11 (one for each SSM/I). These files contain the times for groups of scans that were corrupted by the calibration error problem between October 9, 1990 and August 29, 1992. The routine QUAL1 described in Section 5 reads the BADCAL file and excludes these groups of bad scans.

After August 29, 1992, there is no scan averaging for our T_A data, and the detection of calibration errors is a simpler, two-step process that does not require the BADCAL data. First, when producing the T_A tapes, we do consistency checks among the FNOC computed calibration coefficients, the T_A's, and the cold/hot counts. Occasionally anomalous FNOC values are found, and the best course of action is to exclude these observations. Accordingly, a new Q/C flag is included on the T_A tapes. The new Q/C flag is placed in 64 bytes in each T_A logical record (i.e., bytes 386, 396, ..., 1016). Previously, these bytes were set to zero and were not used. The new decoding routine DECODE2 accesses these bytes, and if the flag is set, DECODE2 appends a negative sign to the appropriate T_A or T_B value. It is then the Users' responsibility to check for negative T_A or T_B and discard these values.

The second step of detecting and removing calibration errors is to check the calibration data that is written to the T_A tapes. If the calibration values are out of range, then the entire scan should be excluded. These calibration checks are done by the Q/C routine QUAL1 discussed in Section 5.

For T_A data prior to October 9, 1990, there was no scan averaging and the BADCAL data are not need. The routine QUAL1 uses the scan time to determine if the BADCAL file needs to be read.

SECTION 5. THE QUALITY CONTROL ROUTINE: QUAL1

We are supplying the Users with a quality control routine call QUAL1.FOR. This Fortran routine automatically handles the data quality problems discussed in Sections 3 and 4. It interfaces with the DECODE2 common area /OUTDAT/. The User must call this routine once for each SSM/I A-B scan. When called, QUAL1 checks for the following conditions, and if the condition occurs, then a flag is set:

1. The scan time is within an erroneous data time period (see Section 3).
2. The calibration data for the scan is erroneous (see Section 4)
3. The scan is within a group of 10 scans that has erroneous calibration data (see Section 4)
4. The T_A value is outside the range of possible Earth values (see Section 3)

QUAL1 reads the data files containing erroneous data periods (BADLOC.D**) and the erroneous calibration periods (BADCAL.D**) that are listed in Section 1. The BADLOC files are formatted ASCII files, and the BADCAL files are binary files. The data in the BADCAL files are a simple series of 4-byte integers, with no header or record marks. Each integer is a time in units of seconds from the beginning of 1987. The integers are such that the first byte is the least-significant byte (i.e., the Microsoft™ convention).

There are two important requirements for the proper usage of QUAL1. First, the *routine must be called in chronological order.* That is to say, the time ITIME in common /OUTDAT/ must not decrease from one call to the next. This requirement for chronological calls is due to the sequential method of reading the data files BADLOC and BADCAL.

The second requirement is that the array TALO in common /OUTDAT/ must contain T_A's, not T_B's. The new Revision-2 DECODE2 guarantees that T_A's will be in TALO, regardless of the setting of the ITB option (see Section 2).

The routine QUAL1 is intended as an example of quality control for SSM/I. Depending on the application, the User does not necessarily need to implement the entire routine. However, as a minimum, we do suggest that the erroneous data time periods be excluded and that T_A's outside the range of possible Earth values not be used. With respect to calibration errors, the above condition 2 occurs for about 0.05% of the scans, and condition 3 occurs for about 0.5% of the scans. Condition 3 occurs ten times as frequently as condition 2 because the 10-scan averaging propagates a single scan error over 10 scans. If the highest quality data is required, then these two conditions should also be checked.

SECTION 6. CROSS-CALIBRATION OF F08 AND F10 SSM/I

Considerable effort was spent cross-calibrating F10 with F08. We first found that it was necessary to only use F08-F10 orbit crossovers in order to minimize the error due to the space-time mismatch. A crossover is defined as the overlap region of the F08 and F10 swaths. Two crossovers occur for each complete orbit. The typical time difference between F08 and F10 at a crossover is 30 minutes. Over the ocean, this time difference, on the average, has a negligible effect on the comparison statistics. However, over land for the morning crossovers, the land temperature is rapidly increasing by about +4 K/hour, thus making a time difference of 30 minutes significant. We therefore did not use morning land crossovers in the intercalibration. For the evening land crossovers, the cooling rate is about -1 K/hour. We applied a first-order correction to compensate for this cooling rate and then used the evening land crossovers in the intercalibration. For the ocean, no cooling or heating in the T_A's was detected, as was expected.

Before comparing the F08 and F10 T_A's, an adjustment is required to correct for the difference between the F08 and F10 incidence angle θ. The incidence angle difference is typically about 0.2°. Over land and ice, this θ difference has a very small effect (less than 0.1 K) on T_A, and no correction was made for land or ice. Over the oceans, the θ difference corresponds to a T_A difference of 0.2 K for horizontal polarization and 0.4 K for vertical polarization. For ocean observations, the F10 T_A's were first normalized to the F08 incidence angle before doing the intercalibration. The θ adjustment is based on the standard ocean brightness temperature model.

The F10 minus F08 T_A differences were regressed to the following linear expression.

$$T_{A10} - T_{A08} = A + B\,(T_{A10} + T_{A08})/2$$

The regression coefficients A and B were found for the five lower SSM/I channels:

CHANNEL	A (K)	B
19V	+0.08	0.00221
19H	+0.35	0.00079
22V	-0.33	0.00161
37V	-0.01	0.00335
37H	+0.44	0.00165

Over the range of Earth values from 120 K to 280 K, the F10 T_A's are about 0.5 to 0.8 K higher than the F08 T_A's, except for the 22V channel for which the difference is near zero. There is also a small tendency for the F10 minus F08 offset to increase with T_A.

To check this analysis, the F10 minus F08 offset was calculated in another, simpler way. Monthly, 1° latitude by 1° longitude ocean maps of the F08 and F10 T_A's were first computed. Then these monthly maps were differenced. The offsets found from this simpler approach agreed with the above offsets found from the crossover analysis to within 0.15 K.

Using the A and B coefficients, the routine DECODE2 adjusts the F10 T_A's to correspond to the F08 T_A's. The adjustment is the following:

$$T_{A10adj} = (1 - B)\, T_{A10} - A$$

T_{A10adj} can then be used in geophysical retrieval algorithms to generate products that are free of biases due to the sensor calibration difference between F08 and F10.

APPENDIX A. LIST OF ERRONEOUS DATA PERIODS

FOR F08 SATELLITE FROM JULY 1987 THROUGH DECEMBER 1991

(File BADLOC.D08 on enclosed 3.5" diskette)

1987	190	0.0	1987	190	15.7
1987	198	4.0	1987	198	5.0
1987	217	21.5	1987	217	23.5
1987	219	12.0	1987	219	14.0
1987	228	14.0	1987	228	16.0
1987	233	4.5	1987	233	6.5
1987	236	13.5	1987	239	15.0
1987	246	13.5	1987	246	14.5
1987	250	11.0	1987	250	12.0
1987	252	14.0	1987	252	16.0
1987	253	10.5	1987	253	12.5
1987	255	13.5	1987	255	15.0
1987	263	16.5	1987	263	18.0
1987	267	21.5	1987	267	23.0
1987	269	14.0	1987	269	16.0
1987	272	14.5	1987	272	16.0
1987	273	15.0	1987	273	15.5
1987	277	14.0	1987	277	15.5
1987	280	11.5	1987	280	13.0
1987	283	1.0	1987	283	3.0
1987	284	21.3	1987	284	23.0
1987	294	21.0	1987	294	22.0
1987	304	22.0	1987	304	23.7
1987	310	14.0	1987	310	15.0
1987	312	5.0	1987	312	5.7
1987	316	21.0	1987	316	23.5
1987	317	14.2	1987	317	16.0
1987	321	5.3	1987	321	7.8
1987	323	5.3	1987	323	6.0
1987	329	2.0	1987	329	3.7
1987	329	10.1	1987	329	10.8
1988	013	0.0	1988	013	19.0
1988	018	2.5	1988	018	4.5
1988	018	9.0	1988	018	10.3
1988	019	15.8	1988	019	17.5
1988	022	5.3	1988	022	8.6
1988	025	4.8	1988	025	7.2
1988	026	10.8	1988	026	12.6
1988	026	23.0	1988	027	1.0
1988	030	22.1	1988	030	23.5
1988	031	11.5	1988	031	13.3
1988	032	22.0	1988	032	23.3
1988	035	6.1	1988	035	6.8
1988	067	4.5	1988	067	5.1
1988	069	14.1	1988	069	15.8
1988	075	23.3	1988	076	0.2
1988	076	10.9	1988	076	13.1
1988	084	7.5	1988	084	9.8
1988	091	21.6	1988	092	1.1
1988	092	21.4	1988	092	23.5
1988	108	1.1	1988	108	2.0
1988	110	0.7	1988	110	3.3
1988	110	12.2	1988	110	16.0
1988	114	13.2	1988	114	15.2
1988	117	4.2	1988	117	7.5
1988	118	0.7	1988	118	2.3
1988	118	13.9	1988	118	16.0
1988	120	5.2	1988	120	6.9
1988	138	23.8	1988	139	1.0
1988	140	1.2	1988	140	3.6
1988	145	3.7	1988	145	5.6

1988	145	15.2	1988	145	18.0
1988	146	5.1	1988	146	7.5
1988	147	23.5	1988	148	2.0
1988	148	11.2	1988	148	12.5
1988	148	14.9	1988	148	16.4
1988	149	17.2	1988	149	21.1
1988	149	23.0	1988	150	0.5
1988	150	10.9	1988	150	11.8
1988	152	16.5	1988	152	17.9
1988	153	5.5	1988	153	9.1
1988	154	20.6	1988	154	22.4
1988	166	21.2	1988	166	22.9
1988	169	2.2	1988	169	4.2
1988	169	10.4	1988	169	11.0
1988	171	15.2	1988	171	15.4
1988	176	14.0	1988	176	15.7
1988	180	19.9	1988	180	21.0
1988	180	23.6	1988	181	0.6
1988	181	1.2	1988	181	1.6
1988	181	4.9	1988	181	6.1
1988	181	9.9	1988	181	11.2
1988	181	20.7	1988	181	21.0
1988	182	3.6	1988	182	4.0
1988	182	5.1	1988	182	5.3
1988	182	12.9	1988	182	13.5
1988	182	17.8	1988	182	19.7
1988	188	11.5	1988	188	13.2
1988	192	14.1	1988	192	16.0
1988	217	20.0	1988	217	21.0
1988	224	13.6	1988	224	15.3
1988	225	9.0	1988	225	11.4
1988	225	12.5	1988	225	13.1
1988	227	3.8	1988	227	5.2
1988	229	16.3	1988	229	16.8
1988	230	12.0	1988	230	13.7
1988	230	16.4	1988	230	19.5
1988	230	21.5	1988	230	22.8
1988	231	15.7	1988	231	16.1
1988	232	1.1	1988	232	3.3
1988	233	17.3	1988	233	21.3
1988	234	12.2	1988	234	12.8
1988	236	10.1	1988	236	11.2
1988	236	11.9	1988	236	12.6
1988	237	22.1	1988	237	22.8
1988	237	23.7	1988	238	0.9
1988	238	2.1	1988	238	2.6
1988	239	21.7	1988	239	23.6
1988	240	0.0	1988	240	0.5
1988	240	2.0	1988	240	4.0
1988	240	9.2	1988	240	10.0
1988	241	7.7	1988	241	8.0
1988	241	9.0	1988	241	12.2
1988	242	4.1	1988	242	6.2
1988	242	7.7	1988	242	8.0
1988	242	21.1	1988	242	22.4
1988	242	22.9	1988	242	23.3
1988	243	3.9	1988	243	4.6
1988	243	9.8	1988	243	10.0
1988	243	11.6	1988	243	12.6
1988	243	16.2	1988	243	19.1
1988	243	21.1	1988	243	21.4

1988	244	0.1	1988	244	1.0
1988	244	2.4	1988	244	5.5
1988	244	9.0	1988	244	9.8
1988	244	15.3	1988	244	16.6
1988	246	6.9	1988	246	8.0
1988	246	11.7	1988	246	12.2
1988	246	14.9	1988	246	16.4
1988	249	10.2	1988	249	12.0
1988	249	12.7	1988	249	13.6
1988	249	17.6	1988	249	20.3
1988	250	4.1	1988	250	7.6
1988	250	8.9	1988	250	9.6
1988	250	12.4	1988	250	13.2
1988	250	15.6	1988	250	15.8
1988	251	15.6	1988	251	16.1
1988	254	9.8	1988	254	15.8
1988	254	17.4	1988	254	18.6
1988	255	3.1	1988	255	4.1
1988	255	14.8	1988	255	15.4
1988	255	18.0	1988	255	21.9
1988	255	23.3	1988	255	23.5
1988	256	3.8	1988	256	4.1
1988	256	11.1	1988	256	12.7
1988	257	7.6	1988	257	9.1
1988	257	10.9	1988	257	14.4
1988	258	0.8	1988	258	2.2
1988	258	8.9	1988	258	10.0
1988	258	10.8	1988	258	11.9
1988	258	15.5	1988	258	15.7
1988	258	16.7	1988	258	19.8
1988	258	20.9	1988	258	21.1
1988	259	22.7	1988	259	23.7
1988	260	2.2	1988	260	3.3
1988	260	10.3	1988	260	10.6
1988	260	13.8	1988	260	14.2
1988	260	14.7	1988	260	14.9
1988	261	22.8	1988	261	23.0
1988	262	22.9	1988	262	23.4
1988	263	2.6	1988	263	3.4
1988	263	9.7	1988	263	10.4
1988	263	10.9	1988	263	11.4
1988	263	23.0	1988	263	23.2
1988	264	2.6	1988	264	2.8
1988	264	4.7	1988	264	6.6
1988	265	7.7	1988	265	8.3
1988	265	8.6	1988	265	9.1
1988	265	11.3	1988	265	12.7
1988	265	23.1	1988	266	0.8
1988	284	15.7	1988	284	15.9
1988	284	16.9	1988	284	20.2
1988	284	20.7	1988	284	20.9
1988	286	8.3	1988	286	8.9
1988	286	15.0	1988	286	16.2
1988	286	16.6	1988	286	21.2
1988	286	22.2	1988	286	23.4
1988	289	7.7	1988	289	8.3
1988	289	9.3	1988	289	11.0
1988	289	11.7	1988	289	12.0
1988	289	14.5	1988	289	15.2
1988	290	12.7	1988	290	15.1
1988	290	15.4	1988	290	16.0
1988	290	16.8	1988	290	17.3
1988	290	17.8	1988	290	22.3
1988	290	23.0	1988	290	23.2
1988	291	0.8	1988	291	1.6
1988	291	8.9	1988	291	12.0
1988	291	12.6	1988	291	13.5
1988	291	21.2	1988	291	22.5
1988	292	0.5	1988	292	0.9
1988	292	7.5	1988	292	8.5
1988	292	8.8	1988	292	10.1
1988	292	10.4	1988	292	11.5
1988	292	13.2	1988	292	13.5
1988	292	14.5	1988	292	15.2
1988	292	17.0	1988	292	17.3
1988	292	17.7	1988	292	18.3
1988	292	19.0	1988	292	19.8
1988	292	21.0	1988	292	22.6
1988	293	5.6	1988	293	7.5
1988	293	16.8	1988	293	18.4
1988	299	7.3	1988	299	7.8
1988	301	16.9	1988	301	18.8
1988	301	20.8	1988	301	21.5
1988	301	22.9	1988	301	23.5
1988	303	9.9	1988	303	10.1
1988	303	15.0	1988	303	16.6
1988	307	18.0	1988	307	18.3
1988	308	8.8	1988	308	10.6
1988	311	2.0	1988	311	3.7
1988	311	5.6	1988	311	5.8
1988	311	23.7	1988	312	0.0
1988	312	3.1	1988	312	5.1
1988	312	5.9	1988	312	7.5
1988	312	9.7	1988	312	10.0
1988	312	11.0	1988	312	11.6
1988	312	16.4	1988	312	17.3
1988	313	1.3	1988	313	1.5
1988	313	4.7	1988	313	6.2
1988	313	6.5	1988	313	7.7
1988	313	9.5	1988	313	9.8
1988	314	15.5	1988	314	15.8
1988	315	14.0	1988	315	15.8
1988	320	17.6	1988	320	17.9
1988	321	3.2	1988	321	4.7
1988	321	8.9	1988	321	13.5
1988	321	14.3	1988	321	14.8
1988	322	9.3	1988	322	10.4
1988	322	12.8	1988	322	14.3
1988	325	16.9	1988	325	18.1
1988	326	2.0	1988	326	3.2
1988	326	4.9	1988	326	5.2
1988	326	22.4	1988	326	22.9
1988	327	0.0	1988	327	1.8
1988	327	5.3	1988	327	6.1
1988	327	11.8	1988	327	12.8
1988	327	20.7	1988	327	21.9
1988	328	2.0	1988	328	3.1
1988	328	18.3	1988	328	18.9
1988	329	17.9	1988	329	20.0
1988	330	23.1	1988	331	1.0
1988	331	9.1	1988	331	10.8
1988	332	12.4	1988	332	13.8
1988	332	14.1	1988	332	15.2
1988	332	18.9	1988	332	19.9
1988	334	5.6	1988	334	7.8
1988	335	20.7	1988	335	21.3
1988	336	5.1	1988	336	7.4
1988	336	8.0	1988	336	9.1
1988	336	16.5	1988	336	17.9
1988	337	11.8	1988	337	12.8
1988	338	7.7	1988	338	9.2
1988	338	11.1	1988	338	11.7
1988	338	18.1	1988	338	19.5
1988	338	20.0	1988	338	20.8
1988	338	21.9	1988	338	22.3
1988	339	0.4	1988	339	1.4
1988	339	2.7	1988	339	3.8
1988	339	4.3	1988	339	4.9
1988	339	14.3	1988	339	15.0
1988	349	3.9	1988	349	4.6

1988	349	9.4	1988	349	10.9
1988	349	12.3	1988	349	12.5
1988	349	12.8	1988	349	14.9
1988	349	15.6	1988	349	16.8
1988	349	17.0	1988	349	18.1
1988	349	18.5	1988	349	20.7
1988	349	21.0	1988	349	22.5
1988	349	22.7	1988	349	23.4
1988	350	3.9	1988	350	5.2
1988	350	5.4	1988	350	5.7
1988	350	6.0	1988	350	8.1
1988	350	8.6	1988	350	9.9
1988	350	10.2	1988	350	11.5
1988	350	12.0	1988	350	12.4
1988	350	16.8	1988	350	18.0
1988	351	16.6	1988	351	19.7
1988	351	20.0	1988	351	20.3
1988	351	21.6	1988	351	21.8
1988	351	23.8	1988	352	0.7
1988	352	1.6	1988	352	2.3
1988	352	2.9	1988	352	3.1
1988	352	4.8	1988	352	5.3
1988	352	5.6	1988	352	7.1
1988	352	8.1	1988	352	8.9
1988	352	9.8	1988	352	10.6
1988	354	9.8	1988	354	11.0
1988	354	11.2	1988	354	11.9
1988	354	16.2	1988	354	17.5
1988	356	6.0	1988	356	6.6
1988	357	2.7	1988	357	3.0
1989	7	5.8	1989	7	8.8
1989	30	14.1	1989	30	15.8
1989	46	14.2	1989	46	15.8
1989	56	3.2	1989	56	4.2
1989	56	4.8	1989	56	7.4
1989	92	1.4	1989	92	2.0
1989	130	23.5	1989	131	0.0
1989	155	23.7	1989	156	0.0
1989	156	23.4	1989	157	0.0
1989	163	23.6	1989	164	0.0
1989	164	23.3	1989	165	0.0
1989	169	22.2	1989	170	0.0
1989	173	23.1	1989	174	0.0
1989	179	23.6	1989	180	0.0
1989	182	22.9	1989	183	0.0
1989	186	23.7	1989	187	0.0
1989	187	23.5	1989	188	0.0
1989	195	23.6	1989	196	0.0
1989	207	22.7	1989	208	0.0
1989	234	23.6	1989	235	0.0
1989	271	20.8	1989	271	22.5
1989	276	23.0	1989	277	1.0
1990	019	16.6	1990	019	17.0
1990	065	23.9	1990	066	0.0
1990	169	3.1	1990	169	5.0
1990	216	14.2	1990	216	16.2
1990	243	9.9	1990	243	11.4
1990	272	22.1	1990	273	0.1
1990	282	9.7	1990	282	10.4
1990	294	0.0	1990	294	3.0
1990	300	20.5	1990	300	21.5
1990	301	6.0	1990	301	7.5
1990	307	18.7	1990	307	19.8
1990	308	0.0	1990	308	1.0
1990	309	0.0	1990	309	1.0
1991	74	0.0	1991	74	0.8
1991	98	18.5	1991	98	19.3
1991	134	15.0	1991	134	16.0
1991	181	22.1	1991	181	22.6
1991	184	8.0	1991	184	8.5
1991	201	21.9	1991	201	23.7
1991	204	16.1	1991	204	16.4
1991	219	12.7	1991	219	13.1
1991	228	0.0	1991	228	9.0
1991	229	2.1	1991	229	2.4
1991	335	3.5	1991	335	3.8

APPENDIX B. LIST OF ERRONEOUS DATA PERIODS

FOR F10 SATELLITE FROM DECEMBER 1990 THROUGH MARCH 1993

(File BADLOC.D10 on enclosed 3.5" diskette)

1990	342	15.7	1990	342	17.6
1990	346	20.1	1990	346	20.8
1990	353	13.9	1990	355	18.0
1991	005	16.0	1991	006	1.5
1991	223	9.0	1991	223	9.4
1991	267	6.3	1991	267	6.8
1991	293	0.0	1991	293	2.6
1991	353	13.6	1991	353	14.6
1992	008	2.8	1992	008	3.4
1992	078	1.6	1992	078	3.0
1992	095	19.6	1992	095	21.5
1992	185	19.5	1992	185	19.9
1992	200	0.0	1992	200	1.0
1992	203	1.6	1992	203	1.9
1993	011	21.5	1993	011	22.0
1993	025	14.0	1993	025	15.0
1993	025	20.7	1993	025	21.7
1993	035	21.0	1993	035	23.2
1993	053	4.4	1993	053	6.1

APPENDIX C. LIST OF ERRONEOUS DATA PERIODS

FOR F11 SATELLITE FROM DECEMBER 1991 THROUGH MARCH 1993

(File BADLOC.D11 on enclosed 3.5" diskette)

```
1991 337 18.6 1991 337 19.2
1991 343 17.4 1991 343 18.1
1991 361 10.1 1991 361 12.8
1991 362 18.0 1991 362 20.5
1992 008 11.5 1992 008 12.5
1992 021  8.3 1992 021 10.0
1992 117  0.5 1992 117  1.5
1992 208  7.2 1992 208  9.9
1992 247 12.7 1992 247 13.3
1993 053  7.5 1993 053  9.7
1993 065  2.9 1993 065  3.3
1993 072 10.2 1993 072 12.3
```

APPENDIX D. LISTING OF SUBROUTINE DECODE2, REVISION 2

(File DECODE2.FOR on the enclosed 3.5" diskette)

```
      SUBROUTINE DECODE2(I85GHZ,ITB,IADJ,IREC,LREC)
C
C     REVISION-2 DECODE.  SAME AS REVISION-1 EXCEPT FOR THE FOLLOWING:
C     1.  CROSS-SCAN BIASES FOR F10 ARE IMPLEMENTED (IF ITB=1)
C     2.  F10 TA'S ARE ADJUSTED TO CORRESPOND TO F08 TA'S (IF ITB=1)
C     3.  0.336 DEG HAS BEEN ADDED TO F08 THT
C     4.  OPTION ITB=0 IS THE SAME AS BEFORE, I.E., ONLY TA'S ARE
C         ARE COMPUTED AND NO TA CORRECTIONS ARE APPLIED.
C         OPTION ITB=1 IS NOW DIFFERENT.  IN THIS CASE, THE TA'S
C         ARE CORRECTED FOR ALONG-SCAN BIASES AND THE F10 TA'S
C         ARE ADJUSTED SO AS TO AGREE WITH THE F08 TA'S.  THESES
C         ADJUSTED TA'S ARE STORED IN ARRAY TALO(5,64).  NO ADJUSTMENTS
C         ARE DONE FOR THE 85 GHZ CHANNELS.  ALSO, WHEN ITB=1, THE
C         TB'S ARE COMPUTED FROM THE ADJUSTED TA'S AND ARE STORED IN
C         A NEW COMMON CALLED:
C         COMMON /TBDATA/ TBLO(5,64),ATBHI(2,128),BTBHI(2,128)
C         THUS WHEN ITB=1, TA'S AND TB'S ARE BOTH AVAILABLE.
C         NOTE THAT IF THE V OR H-POL TA IS O.O.B.,THEN NO TA ADJUSTMENT
C         AND THE TB'S ARE SET TO THE UNADJUSTED TA VALUES.
C     5.  TA'S ARE SET NEGATIVE IF CALIBRATION DATA IS BAD
C
C ****************** DESCRIPTION OF ARGUMENTS **************************
C     I85GHZ=0 PROCESSES ONLY THE 19, 22, AND 37 GHZ OBS.
C     I85GHZ=1 PROCESSES ALL CHANNELS
C
C     ITB=0 COMPUTES UNADJUSTED TA'S
C     ITB=1 COMPUTES ADJUSTED TA'S AND TB'S
C
C     IADJ=0 DOES NO ALONGTRACK LOCATION CORRECTION
C     IADJ=1 DOES AN ALONGTRACK LOCATION CORRECTION
C
C     IREC DENOTES LOGICAL RECORD NUMBER (1 THROUGH 16)
C
C     LREC IS THE ARRAY THAT CONTAINS THE BYTES IN THE TAPE BLOCK
C     THE USER MUST SUPPLY THE TAPE READ ROUTINE FOR FILLING LREC
C     BEFORE CALLING THIS ROUTINE
```

```
C
C ************ DECLARATION OF VARIABLES, ARRAYS, AND COMMONS ***********
C
      CHARACTER*1 LREC(1784,16)
C
      INTEGER*2 IBUF
      COMMON /INDATA/ IBUF(1784)
C
      REAL*8 REV,XTIME
      INTEGER*4 ITIME,ITIMSC,IVOLT,IAGC,ICOLDA,IHOTA,ICOLDB,IHOTB,IASCTM
      INTEGER*4 IATOIL,IBTOIL,ISAT
      REAL*4 XLATSC,XLONSC,ALTSC,THT,HLTEMP,RFTEMP,FRTEMP
      REAL*4 PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,SPACER
      REAL*4 ALAT,ALON,BLAT,BLON,TALO,ATAHI,BTAHI
      COMMON/OUTDAT/ REV,XTIME,ITIME,ITIMSC,XLATSC,XLONSC,ALTSC,THT,
     1 HLTEMP(3),IVOLT(2),RFTEMP,FRTEMP,IAGC(6),
     2 IASCTM,PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,ISAT,SPACER(6),
     2 ICOLDA(5,7),IHOTA(5,7),ICOLDB(5,2),IHOTB(5,2),
     3 ALAT(128),ALON(128),BLAT(128),BLON(128),
     4 TALO(5,64),ATAHI(2,128),BTAHI(2,128),IATOIL(128),IBTOIL(128)
C
C     ********************** DATA INITIALIZATION **********************
C
      DATA N1,N2,N3/256,65536,16777216/
      DATA RAD/0.017453293/
C
C ********************** BEGIN EXECUTION ***************************
C
C     TRANSFER TAPE BYTES IN IREC LOGICAL RECORD TO ARRAY IBUF
C
      DO 100 I=1,1784
      IBUF(I)=ICHAR(LREC(I,IREC))
  100 CONTINUE
C
C     FIND SCAN TIME
C
      ITIME= N3*IBUF( 1)+N2*IBUF( 2)+N1*IBUF( 3)+IBUF( 4)
      IFRCTM=N3*IBUF(17)+N2*IBUF(18)+N1*IBUF(19)+IBUF(20)
      IF(IFRCTM.EQ.0) XTIME=ITIME
      IF(IFRCTM.NE.0) XTIME=ITIME+1.D-4*(IFRCTM-10000)
```

```
C
C       FIND SPACECRAFT TIME AND ORBIT NUMBER
C       IOLD=1 FOR ALL SCANS BEFORE 1989
C
        IF(ITIME.LT.63163966) THEN
        IOLD=1
        ITIMSC=     N3*IBUF( 9)+N2*IBUF(10)+N1*IBUF(11)+IBUF(12)
        ELSE
        IOLD=0
        ITIMSC=ITIME
        ENDIF
C
        IF(ITIME.GE.63163966.AND.ITIME.LT.84156110) THEN
        REV=1.D-4*(N3*IBUF( 9)+N2*IBUF(10)+N1*IBUF(11)+IBUF(12))
        ELSE
        REV=1.D-4*(N3*IBUF( 5)+N2*IBUF( 6)+N1*IBUF( 7)+IBUF( 8))
        ENDIF
C
C       FIND SPACECRAFT LATITUDE, LONGITUDE, AND ALTITUDE
C
        XLATSC=1.D-6*(N3*IBUF(13)+N2*IBUF(14)+N1*IBUF(15)+IBUF(16))-90.
        XLONSC=1.D-6*(N3*IBUF(21)+N2*IBUF(22)+N1*IBUF(23)+IBUF(24))
        ALTSC= 1.D-3*(N3*IBUF(25)+N2*IBUF(26)+N1*IBUF(27)+IBUF(28))
C
C       DETERMINE SATELLITE NUMBER AND INCIDENCE ANGLE
C       FACTOR OF .7040147 = SIN(44.75 DEG)
C       FACTOR OF .7116583 = SIN(45.37 DEG)
C
        IF(ITIME.GT.144554200) THEN
        ITERM=N3*IBUF( 9)+N2*IBUF(10)+N1*IBUF(11)+IBUF(12)
        THT=0.001*INT(ITERM/1000)
        ISAT=ITERM-1000*INT(ITERM/1000)
C
        ELSE
C       EARTHR=RADIUS OF CURVATURE OF EARTH'S SURFACE
        EARTHR=6345.7+55.0*SIN(RAD*XLATSC)**2
        DIFREV=DABS(REV-300-(XTIME-16530609)/6118.)
        IF(DIFREV.LT.100.) THEN
        ISAT=8
        THT=ASIN(0.7040147*(EARTHR+ALTSC)/EARTHR)/RAD
        ELSE
        ISAT=10
        THT=ASIN(0.7116583*(EARTHR+ALTSC)/EARTHR)/RAD
        ENDIF
C
        ENDIF
C
        IF(ISAT.EQ.8) THT=THT+0.336
```

```
C
C     FIND RADIOMETER CALIBRATION DATA
C
      HLTEMP(3)=0.01*(N1*IBUF(29)+IBUF(30))
      HLTEMP(2)=0.01*(N1*IBUF(31)+IBUF(32))
      HLTEMP(1)=0.01*(N1*IBUF(33)+IBUF(34))
      RFTEMP=   0.01*(N1*IBUF(39)+IBUF(40))
      FRTEMP=   0.01*(N1*IBUF(41)+IBUF(42))
      IVOLT(2)=       N1*IBUF(35)+IBUF(36)
      IVOLT(1)=       N1*IBUF(37)+IBUF(38)
      IAGC(3)=        N1*IBUF(43)+IBUF(44)
      IAGC(2)=        N1*IBUF(45)+IBUF(46)
      IAGC(1)=        N1*IBUF(47)+IBUF(48)
C
      N=-2
      DO 200 ICH=1,7
      IF(ICH.LE.6) SPACER(ICH)=0.
      DO 200 IP=1,5
      N=N+2
      ICOLDA(IP,ICH)=N1*IBUF(N + 77)+IBUF(N + 78)
      IHOTA(IP,ICH)= N1*IBUF(N +147)+IBUF(N +148)
  200 CONTINUE
C
      IAGC(6)=N1*IBUF(217)+IBUF(218)
      IAGC(5)=N1*IBUF(219)+IBUF(220)
      IAGC(4)=N1*IBUF(221)+IBUF(222)
C
      N=-2
      DO 300 ICH=1,2
      DO 300 IP=1,5
      N=N+2
      ICOLDB(IP,ICH)=N1*IBUF(N +223)+IBUF(N +224)
      IHOTB(IP,ICH)= N1*IBUF(N +243)+IBUF(N +244)
  300 CONTINUE
```

```
C
C     FIND ORBIT PARAMETERS
C
      IF(ITIME.LT.84156110) THEN
      IASCTM=0
      PERIOD=0.
      ASCLOC=0.
      ANGINC=0.
      AXIS=0.
      ECC=0.
      ANGPER=0.
C
      ELSE
      IASCTM=       N3*IBUF(49)+N2*IBUF(50)+N1*IBUF(51)+IBUF(52)
      PERIOD= 1.D-4*(N3*IBUF(53)+N2*IBUF(54)+N1*IBUF(55)+IBUF(56))
      ASCLOC= 1.D-3*(N3*IBUF(57)+N2*IBUF(58)+N1*IBUF(59)+IBUF(60))
      ANGINC= 1.D-6*(N3*IBUF(61)+N2*IBUF(62)+N1*IBUF(63)+IBUF(64))
      ANGINC=180.-ANGINC
      AXIS=   1.D-4*(N3*IBUF(65)+N2*IBUF(66)+N1*IBUF(67)+IBUF(68))
      ECC=   1.D-10*(N3*IBUF(69)+N2*IBUF(70)+N1*IBUF(71)+IBUF(72))
      ANGPER= 1.D-5*(N3*IBUF(73)+N2*IBUF(74)+N1*IBUF(75)+IBUF(76))
      ENDIF
C
C     DETERMINE IF ALONGTRACK AND YAW ADJUSTMENT IS NEEDED
C     TRKADJ=-1.2 CORRESPONDS TO A 15 KM ALONGTRACK ADJUSTMENT
C     YAW=.3125 CORRESPONDS TO A 0.5 DEG. YAW ROTATION
C
      TRKADJ=0.
      IF(IOLD.EQ.1.AND.IADJ.EQ.1) TRKADJ=-1.2
      YAW=0.
      IF(ISAT.EQ.8) YAW=.3125
C
C     FIND CELL LATITUDES AND LONGITUDES
C
      J85GHZ=I85GHZ
      IF(TRKADJ.NE.0.) J85GHZ=1
      CALL FDLTLN(J85GHZ,IOLD,YAW)
      IF(TRKADJ.NE.0.) CALL ADJLOC(TRKADJ)
C
C     FIND EITHER TA'S OR TB'S
C
      CALL FDTA(I85GHZ)
      IF(ITB.NE.0) CALL FDTATB(I85GHZ)
C
      RETURN
      END
C
```

```
      SUBROUTINE FDLTLN(J85GHZ,IOLD,YAW)
C
C     THIS SUBROUTINE FINDS THE LATS AND LONS FOR THE SSMI CELLS
C     J85GHZ=0 DOES NOT DO 85 GHZ LAT/LON, J85GHZ=1 DOES 85 GHZ LAT LON
C
      INTEGER*4 INDEX(19),JNDEX(3,109)
C
C     SPECIFY COMMON /INDATA/
C
      INTEGER*2 IBUF
      COMMON /INDATA/ IBUF(1784)
C
C     SPECIFY COMMON /OUTDAT/
C
      REAL*8 REV,XTIME
      INTEGER*4 ITIME,ITIMSC,IVOLT,IAGC,ICOLDA,IHOTA,ICOLDB,IHOTB,IASCTM
      INTEGER*4 IATOIL,IBTOIL,ISAT
      REAL*4 XLATSC,XLONSC,ALTSC,THT,HLTEMP,RFTEMP,FRTEMP
      REAL*4 PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,SPACER
      REAL*4 ALAT,ALON,BLAT,BLON,TALO,ATAHI,BTAHI
      COMMON/OUTDAT/ REV,XTIME,ITIME,ITIMSC,XLATSC,XLONSC,ALTSC,THT,
     1 HLTEMP(3),IVOLT(2),RFTEMP,FRTEMP,IAGC(6),
     2 IASCTM,PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,ISAT,SPACER(6),
     2 ICOLDA(5,7),IHOTA(5,7),ICOLDB(5,2),IHOTB(5,2),
     3 ALAT(128),ALON(128),BLAT(128),BLON(128),
     4 TALO(5,64),ATAHI(2,128),BTAHI(2,128),IATOIL(128),IBTOIL(128)
```

```
C
C      DATA INITIALIZATION
C
       DATA N1/256/
       DATA RAD/0.017453293/
       DATA INDEX/1,9,17,25,33,41,49,57,65,73,81,89,97,105,113,121,123,
     1 127,128/
       DATA JNDEX/
     1   5,  1,  9, 13,  9, 17, 21, 17, 25, 29, 25, 33, 37, 33, 41,
     1  45, 41, 49, 53, 49, 57, 61, 57, 65, 69, 65, 73, 77, 73, 81,
     1  85, 81, 89, 93, 89, 97,101, 97,105,109,105,113,117,113,121,
     1   3,  1,  5,  7,  5,  9, 11,  9, 13, 15, 13, 17, 19, 17, 21,
     1  23, 21, 25, 27, 25, 29, 31, 29, 33, 35, 33, 37, 39, 37, 41,
     1  43, 41, 45, 47, 45, 49, 51, 49, 53, 55, 53, 57, 59, 57, 61,
     1  63, 61, 65, 67, 65, 69, 71, 69, 73, 75, 73, 77, 79, 77, 81,
     1  83, 81, 85, 87, 85, 89, 91, 89, 93, 95, 93, 97, 99, 97,101,
     1 103,101,105,107,105,109,111,109,113,115,113,117,119,117,121,
     1 125,123,127,  2,  1,  3,  4,  3,  5,  6,  5,  7,  8,  7,  9,
     1  10,  9, 11, 12, 11, 13, 14, 13, 15, 16, 15, 17, 18, 17, 19,
     1  20, 19, 21, 22, 21, 23, 24, 23, 25, 26, 25, 27, 28, 27, 29,
     1  30, 29, 31, 32, 31, 33, 34, 33, 35, 36, 35, 37, 38, 37, 39,
     1  40, 39, 41, 42, 41, 43, 44, 43, 45, 46, 45, 47, 48, 47, 49,
     1  50, 49, 51, 52, 51, 53, 54, 53, 55, 56, 55, 57, 58, 57, 59,
     1  60, 59, 61, 62, 61, 63, 64, 63, 65, 66, 65, 67, 68, 67, 69,
     1  70, 69, 71, 72, 71, 73, 74, 73, 75, 76, 75, 77, 78, 77, 79,
     1  80, 79, 81, 82, 81, 83, 84, 83, 85, 86, 85, 87, 88, 87, 89,
     1  90, 89, 91, 92, 91, 93, 94, 93, 95, 96, 95, 97, 98, 97, 99,
     1 100, 99,101,102,101,103,104,103,105,106,105,107,108,107,109,
     1 110,109,111,112,111,113,114,113,115,116,115,117,118,117,119,
     1 120,119,121,122,121,123,124,123,125,126,125,127/
C
C      BEGIN EXECUTION
C
C      SET TABLE LAT/LON FOR A-SCAN
C
       N=-2
       DO 100 JCEL=1,19
       N=N+2
       ICEL=INDEX(JCEL)
       ALAT(ICEL)=0.01*(N1*IBUF(N +263)+IBUF(N +264)-9000)
       ALON(ICEL)=0.01*(N1*IBUF(N +301)+IBUF(N +302))
       IF(ALON(ICEL).GE.360.) ALON(ICEL)=ALON(ICEL)-360.
   100 CONTINUE
```

```
C
C     SET MID-POINTS FOR A-SCAN
C
      NCEL=46
      IF(J85GHZ.EQ.1) NCEL=109
      DO 200 JCEL=1,NCEL
      ICEL=JNDEX(1,JCEL)
      I1=JNDEX(2,JCEL)
      I2=JNDEX(3,JCEL)
      DIFLAT=ALAT(I2)-ALAT(I1)
      AVGLAT=0.5*(ALAT(I1)+ALAT(I2))
      DIFLON=ALON(I2)-ALON(I1)
      IF(DIFLON.GT. 180.) DIFLON=DIFLON-360.
      IF(DIFLON.LT.-180.) DIFLON=DIFLON+360.
      AVGLON=ALON(I1)+0.5*DIFLON
      XSQ=(2.*RAD*AVGLAT)**2
      XFAC=1.-0.16627142*XSQ+0.00807934*XSQ*XSQ-0.000151880*XSQ*XSQ*XSQ
      ALAT(ICEL)=AVGLAT*(1.+0.125*(RAD*DIFLON)**2*XFAC)
      X=RAD*(90.-ABS(AVGLAT))
      TANLAT=1./(X+X*X*X/3.)
      IF(AVGLAT.LT.0.) TANLAT=-TANLAT
      ALON(ICEL)=AVGLON-0.2500*RAD*DIFLAT*DIFLON*TANLAT
      IF(ALON(ICEL).LT.  0.) ALON(ICEL)=ALON(ICEL)+360.
      IF(ALON(ICEL).GE.360.) ALON(ICEL)=ALON(ICEL)-360.
  200 CONTINUE
C
C     DO YAW CORRECTION
C
      IF(J85GHZ.EQ.0) THEN
      IF(YAW.EQ.0.) RETURN
C
      DO 260 ICEL=1,127,2
      IF(ICEL.EQ.127) GO TO 250
      DIFLAT=ALAT(ICEL+2)-ALAT(ICEL)
      DIFLON=ALON(ICEL+2)-ALON(ICEL)
      IF(DIFLON.LT.-180.) DIFLON=DIFLON+360.
      IF(DIFLON.GT. 180.) DIFLON=DIFLON-360.
  250 CONTINUE
      ALAT(ICEL)=ALAT(ICEL)+YAW*DIFLAT
      ALON(ICEL)=ALON(ICEL)+YAW*DIFLON
      IF(ALON(ICEL).LT.  0.) ALON(ICEL)=ALON(ICEL)+360.
      IF(ALON(ICEL).GE.360.) ALON(ICEL)=ALON(ICEL)-360.
  260 CONTINUE
      RETURN
      ENDIF
```

```
C
C     SET TABLE LAT/LON FOR B-SCAN
C
      N=-2
      DO 300 JCEL=1,19
      N=N+2
      ICEL=INDEX(JCEL)
      IDEL=N1*IBUF(N +339)+IBUF(N +340)
      IF(IDEL.GT.32767) IDEL=IDEL-65536
      LATDEL=(IDEL+30000)/1000-30
      LONDEL=IDEL+29100-1000*(LATDEL+30)
      BLAT(ICEL)=ALAT(ICEL)+0.01*LATDEL
      BLON(ICEL)=ALON(ICEL)+0.01*LONDEL
      IF(BLON(ICEL).LT.  0.) BLON(ICEL)=BLON(ICEL)+360.
      IF(BLON(ICEL).GE.360.) BLON(ICEL)=BLON(ICEL)-360.
  300 CONTINUE
C
C     SET MID-POINTS FOR B-SCAN
C
      DO 400 JCEL=1,109
      ICEL=JNDEX(1,JCEL)
      I1=JNDEX(2,JCEL)
      I2=JNDEX(3,JCEL)
      DIFLAT=BLAT(I2)-BLAT(I1)
      AVGLAT=0.5*(BLAT(I1)+BLAT(I2))
      DIFLON=BLON(I2)-BLON(I1)
      IF(DIFLON.GT. 180.) DIFLON=DIFLON-360.
      IF(DIFLON.LT.-180.) DIFLON=DIFLON+360.
      AVGLON=BLON(I1)+0.5*DIFLON
      XSQ=(2.*RAD*AVGLAT)**2
      XFAC=1.-0.16627142*XSQ+0.00807934*XSQ*XSQ-0.000151880*XSQ*XSQ*XSQ
      BLAT(ICEL)=AVGLAT*(1.+0.125*(RAD*DIFLON)**2*XFAC)
      X=RAD*(90.-ABS(AVGLAT))
      TANLAT=1./(X+X*X*X/3.)
      IF(AVGLAT.LT.0.) TANLAT=-TANLAT
      BLON(ICEL)=AVGLON-0.2500*RAD*DIFLAT*DIFLON*TANLAT
      IF(BLON(ICEL).LT.  0.) BLON(ICEL)=BLON(ICEL)+360.
      IF(BLON(ICEL).GE.360.) BLON(ICEL)=BLON(ICEL)-360.
  400 CONTINUE
```

```
C
C     CORRECT CELL 128 FOR DATA BEFORE 1989
C
      IF(IOLD.EQ.1) THEN
      DIFLAT=ALAT(127)-ALAT(126)
      DIFLON=ALON(127)-ALON(126)
      IF(DIFLON.LT.-180.) DIFLON=DIFLON+360.
      IF(DIFLON.GT. 180.) DIFLON=DIFLON-360.
      ALAT(128)=ALAT(127)+DIFLAT
      ALON(128)=ALON(127)+DIFLON
      IF(ALON(128).LT.  0.) ALON(128)=ALON(128)+360.
      IF(ALON(128).GE.360.) ALON(128)=ALON(128)-360.
C
      DIFLAT=BLAT(127)-BLAT(126)
      DIFLON=BLON(127)-BLON(126)
      IF(DIFLON.LT.-180.) DIFLON=DIFLON+360.
      IF(DIFLON.GT. 180.) DIFLON=DIFLON-360.
      BLAT(128)=BLAT(127)+DIFLAT
      BLON(128)=BLON(127)+DIFLON
      IF(BLON(128).LT.  0.) BLON(128)=BLON(128)+360.
      IF(BLON(128).GE.360.) BLON(128)=BLON(128)-360.
      ENDIF
C
C     DO YAW CORRECTION
C
      IF(YAW.EQ.0.) RETURN
      DO 600 ICEL=1,128
      IF(ICEL.EQ.128) GO TO 580
      DIFLAT1=ALAT(ICEL+1)-ALAT(ICEL)
      DIFLON1=ALON(ICEL+1)-ALON(ICEL)
      IF(DIFLON1.LT.-180.) DIFLON1=DIFLON1+360.
      IF(DIFLON1.GT. 180.) DIFLON1=DIFLON1-360.
      DIFLAT2=BLAT(ICEL+1)-BLAT(ICEL)
      DIFLON2=BLON(ICEL+1)-BLON(ICEL)
      IF(DIFLON2.LT.-180.) DIFLON2=DIFLON2+360.
      IF(DIFLON2.GT. 180.) DIFLON2=DIFLON2-360.
  580 CONTINUE
      ALAT(ICEL)=ALAT(ICEL)+2.*YAW*DIFLAT1
      ALON(ICEL)=ALON(ICEL)+2.*YAW*DIFLON1
      IF(ALON(ICEL).LT.  0.) ALON(ICEL)=ALON(ICEL)+360.
      IF(ALON(ICEL).GE.360.) ALON(ICEL)=ALON(ICEL)-360.
      BLAT(ICEL)=BLAT(ICEL)+2.*YAW*DIFLAT2
      BLON(ICEL)=BLON(ICEL)+2.*YAW*DIFLON2
      IF(BLON(ICEL).LT.  0.) BLON(ICEL)=BLON(ICEL)+360.
      IF(BLON(ICEL).GE.360.) BLON(ICEL)=BLON(ICEL)-360.
  600 CONTINUE
      RETURN
      END
C
```

```
      SUBROUTINE ADJLOC(TRKADJ)
C
C     CORRECT FOR ALONGTRACK ERROR
C
C     SPECIFY COMMON /OUTDAT/
C
      REAL*8 REV,XTIME
      INTEGER*4 ITIME,ITIMSC,IVOLT,IAGC,ICOLDA,IHOTA,ICOLDB,IHOTB,IASCTM
      INTEGER*4 IATOIL,IBTOIL,ISAT
      REAL*4 XLATSC,XLONSC,ALTSC,THT,HLTEMP,RFTEMP,FRTEMP
      REAL*4 PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,SPACER
      REAL*4 ALAT,ALON,BLAT,BLON,TALO,ATAHI,BTAHI
      COMMON/OUTDAT/ REV,XTIME,ITIME,ITIMSC,XLATSC,XLONSC,ALTSC,THT,
     1 HLTEMP(3),IVOLT(2),RFTEMP,FRTEMP,IAGC(6),
     2 IASCTM,PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,ISAT,SPACER(6),
     2 ICOLDA(5,7),IHOTA(5,7),ICOLDB(5,2),IHOTB(5,2),
     3 ALAT(128),ALON(128),BLAT(128),BLON(128),
     4 TALO(5,64),ATAHI(2,128),BTAHI(2,128),IATOIL(128),IBTOIL(128)
C
C     BEGIN EXECUTION
C
      DO 100 ICEL=1,128
      DIFLAT=BLAT(ICEL)-ALAT(ICEL)
      ALAT(ICEL)=ALAT(ICEL)+TRKADJ*DIFLAT
      BLAT(ICEL)=BLAT(ICEL)+TRKADJ*DIFLAT
C
      DIFLON=BLON(ICEL)-ALON(ICEL)
      IF(DIFLON.LT.-180.) DIFLON=DIFLON+360.
      IF(DIFLON.GT. 180.) DIFLON=DIFLON-360.
C
      ALON(ICEL)=ALON(ICEL)+TRKADJ*DIFLON
      IF(ALON(ICEL).LT.  0.) ALON(ICEL)=ALON(ICEL)+360.
      IF(ALON(ICEL).GE.360.) ALON(ICEL)=ALON(ICEL)-360.
C
      BLON(ICEL)=BLON(ICEL)+TRKADJ*DIFLON
      IF(BLON(ICEL).LT.  0.) BLON(ICEL)=BLON(ICEL)+360.
      IF(BLON(ICEL).GE.360.) BLON(ICEL)=BLON(ICEL)-360.
  100 CONTINUE
      RETURN
      END
C
```

```
      SUBROUTINE FDTA(I85GHZ)
C
C     THIS SUBROUTINE FINDS THE ANTENNA TEMPERATURES AND SURFACE TYPES
C
      INTEGER*4 ITALO(5),ITAHI(8),NBIT(7)
C
C     SPECIFY COMMON /INDATA/
C
      INTEGER*2 IBUF
      COMMON /INDATA/ IBUF(1784)
C
C     SPECIFY COMMON /OUTDAT/
C
      REAL*8 REV,XTIME
      INTEGER*4 ITIME,ITIMSC,IVOLT,IAGC,ICOLDA,IHOTA,ICOLDB,IHOTB,IASCTM
      INTEGER*4 IATOIL,IBTOIL,ISAT
      REAL*4 XLATSC,XLONSC,ALTSC,THT,HLTEMP,RFTEMP,FRTEMP
      REAL*4 PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,SPACER
      REAL*4 ALAT,ALON,BLAT,BLON,TALO,ATAHI,BTAHI
      COMMON/OUTDAT/ REV,XTIME,ITIME,ITIMSC,XLATSC,XLONSC,ALTSC,THT,
     1 HLTEMP(3),IVOLT(2),RFTEMP,FRTEMP,IAGC(6),
     2 IASCTM,PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,ISAT,SPACER(6),
     2 ICOLDA(5,7),IHOTA(5,7),ICOLDB(5,2),IHOTB(5,2),
     3 ALAT(128),ALON(128),BLAT(128),BLON(128),
     4 TALO(5,64),ATAHI(2,128),BTAHI(2,128),IATOIL(128),IBTOIL(128)
C
      DATA N1,N2/256,65536/
      DATA NBIT/1,2,4,8,16,32,64/
C
C     BEGIN EXECUTION
C
      N=-10
      M=-12
C
```

```
      DO 100 ICEL=1,64
      JCEL=2*ICEL-1
C
C     FIND THE TA'S FOR THE 3 LOWER FREQUENCIES
C
      N=N+10
C
      IWORK4=N2*IBUF(N +377)+N1*IBUF(N+378)+IBUF(N+379)
      ITAV=INT(IWORK4/4096)
      ITALO(1)=ITAV
      ITALO(2)=IWORK4-4096*ITAV
C
      IWORK4=N2*IBUF(N +380)+N1*IBUF(N+381)+IBUF(N+382)
      ITAV=INT(IWORK4/4096)
      ITALO(4)=ITAV
      ITALO(5)=IWORK4-4096*ITAV
C
      IWORK4=N2*IBUF(N +383)+N1*IBUF(N+384)+IBUF(N+385)
      ITAV=INT(IWORK4/4096)
      ITALO(3)=ITAV
C
      IQUAL=IBUF(N+386)
C
C     SCALE THE TA'S
C
      DO 20 ICH=1,5
      IF(ITALO(ICH).LE.3800) THEN
      TALO(ICH,ICEL)=0.1*ITALO(ICH)
      ELSE
      TALO(ICH,ICEL)=ITALO(ICH)-3420
      ENDIF
      IF(IAND(NBIT(ICH),IQUAL).NE.0) TALO(ICH,ICEL)=-TALO(ICH,ICEL)
   20 CONTINUE
C
C     FIND THE TOIL FLAGS
C
      IRES=IWORK4-4096*ITAV
      ITOIL1=INT(IRES/512)
      IATOIL(JCEL)=ITOIL1
C
      IF(I85GHZ.EQ.0) GO TO 100
C
C     FIND 85 GHZ TOILS'S AND TA'S
C
      M=M+12
C
      IRES=IRES-ITOIL1*512
      ITOIL2=INT(IRES/64)
      IRES=IRES-ITOIL2*64
      ITOIL3=INT(IRES/8)
      ITOIL4=IRES-ITOIL3*8
      IBTOIL(JCEL)=ITOIL2
      IATOIL(JCEL+1)=ITOIL3
      IBTOIL(JCEL+1)=ITOIL4
```

```
C
      IWORK4=N2*IBUF(M+1017)+N1*IBUF(M+1018)+IBUF(M+1019)
      ITAV=INT(IWORK4/4096)
      ITAHI(1)=ITAV
      ITAHI(3)=IWORK4-4096*ITAV
C
      IWORK4=N2*IBUF(M+1020)+N1*IBUF(M+1021)+IBUF(M+1022)
      ITAV=INT(IWORK4/4096)
      ITAHI(2)=ITAV
      ITAHI(4)=IWORK4-4096*ITAV
C
      IWORK4=N2*IBUF(M+1023)+N1*IBUF(M+1024)+IBUF(M+1025)
      ITAV=INT(IWORK4/4096)
      ITAHI(5)=ITAV
      ITAHI(7)=IWORK4-4096*ITAV
C
      IWORK4=N2*IBUF(M+1026)+N1*IBUF(M+1027)+IBUF(M+1028)
      ITAV=INT(IWORK4/4096)
      ITAHI(6)=ITAV
      ITAHI(8)=IWORK4-4096*ITAV
C
C     SCALE THE TA'S
C
      K=0
      DO 40 KCEL=JCEL,JCEL+1
      DO 40 ICH=1,2
      K=K+1
      IF(ITAHI(K).LE.3800) THEN
      ATAHI(ICH,KCEL)=0.1*ITAHI(K)
      ELSE
      ATAHI(ICH,KCEL)=ITAHI(K)-3420
      ENDIF
C
      K=K+1
C
      IF(ITAHI(K).LE.3800) THEN
      BTAHI(ICH,KCEL)=0.1*ITAHI(K)
      ELSE
      BTAHI(ICH,KCEL)=ITAHI(K)-3420
      ENDIF
C
      IF(IAND(NBIT(ICH+5),IQUAL).NE.0) THEN
      ATAHI(ICH,KCEL)=-ATAHI(ICH,KCEL)
      BTAHI(ICH,KCEL)=-BTAHI(ICH,KCEL)
      ENDIF
C
   40 CONTINUE
C
  100 CONTINUE
      RETURN
      END
C
```

```
      SUBROUTINE FDTATB(I85GHZ)
C
C     THIS SUBROUTINE ADJUST THE ANTENNA TEMPS AND FIND BRIGHTNESS TEMPS
C
      REAL*4 DELTA(4),CHI(2,4),AVGTA(5),GCOEF(5),HCOEF(5)
      REAL*4 SBIAS1(64),SBIAS2(64),SBIAS3(64),SBIAS4(64),SBIAS5(64)
      REAL*4 TBIAS1(64),TBIAS2(64),TBIAS3(64),TBIAS4(64),TBIAS5(64)
      REAL*4 AVV(4),AHV(4),AOV(4),AHH(4),AVH(4),AOH(4)
C
C     SPECIFY COMMON /OUTDAT/
C
      REAL*8 REV,XTIME
      INTEGER*4 ITIME,ITIMSC,IVOLT,IAGC,ICOLDA,IHOTA,ICOLDB,IHOTB,IASCTM
      INTEGER*4 IATOIL,IBTOIL,ISAT
      REAL*4 XLATSC,XLONSC,ALTSC,THT,HLTEMP,RFTEMP,FRTEMP
      REAL*4 PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,SPACER
      REAL*4 ALAT,ALON,BLAT,BLON,TALO,ATAHI,BTAHI
      COMMON/OUTDAT/ REV,XTIME,ITIME,ITIMSC,XLATSC,XLONSC,ALTSC,THT,
     1 HLTEMP(3),IVOLT(2),RFTEMP,FRTEMP,IAGC(6),
     2 IASCTM,PERIOD,ASCLOC,ANGINC,AXIS,ECC,ANGPER,ISAT,SPACER(6),
     2 ICOLDA(5,7),IHOTA(5,7),ICOLDB(5,2),IHOTB(5,2),
     3 ALAT(128),ALON(128),BLAT(128),BLON(128),
     4 TALO(5,64),ATAHI(2,128),BTAHI(2,128),IATOIL(128),IBTOIL(128)
C
      COMMON /TBDATA/ TBLO(5,64),ATBHI(2,128),BTBHI(2,128)
C
C     DATA INITIALIZATION
C
      DATA ISTART/1/
      DATA DELTA/0.03199,0.02685,0.01434,0.01186/
      DATA CHI/.00379,.00525,.00983,0.0,.02136,.02664,.01387,.01967/
      DATA AVGTA/190.93,130.14,215.42,211.39,158.16/
C
C     GCOEF,HCOEF ARE F10-F08 INTERCALIBRATION COEFS.
C
      DATA GCOEF/0.08, 0.35,-0.33,-0.01, 0.44/
      DATA HCOEF/.221132E-02,.786968E-03,.161037E-02,.335131E-02,
     1 .165331E-02/
```

```
C
C     F08 X-SCAN BIASES
C
      DATA SBIAS1/                               -.03,  .01,  .04,  .06,
     +  .09,  .10,  .12,  .12,  .13,  .13,  .13,  .13,  .13,  .14,
     +  .14,  .14,  .14,  .14,  .15,  .15,  .14,  .14,  .13,  .12,
     +  .12,  .11,  .11,  .11,  .11,  .11,  .12,  .11,  .10,  .10,
     +  .09,  .07,  .06,  .05,  .05,  .04,  .03,  .03,  .02,  .01,
     + -.01, -.02, -.04, -.05, -.06, -.08, -.09, -.09, -.11, -.12,
     + -.14, -.18, -.21, -.26, -.30, -.36, -.43, -.52, -.63, -.75/
      DATA SBIAS2/                               -.03, -.01,  .00,  .01,
     +  .01,  .01,  .01,  .01,  .01,  .01,  .01,  .02,  .03,  .04,
     +  .04,  .05,  .05,  .05,  .05,  .05,  .04,  .04,  .04,  .04,
     +  .04,  .04,  .04,  .04,  .04,  .05,  .06,  .07,  .07,  .08,
     +  .07,  .07,  .06,  .07,  .07,  .06,  .06,  .06,  .06,  .06,
     +  .05,  .04,  .04,  .04,  .04,  .02,  .00, -.01, -.02, -.03,
     + -.04, -.06, -.09, -.12, -.16, -.20, -.26, -.32, -.39, -.48/
      DATA SBIAS3/                               -.05,  .00,  .04,  .07,
     +  .08,  .09,  .10,  .11,  .11,  .12,  .12,  .13,  .14,  .14,
     +  .15,  .15,  .17,  .17,  .16,  .16,  .15,  .14,  .14,  .14,
     +  .13,  .13,  .13,  .13,  .13,  .13,  .14,  .13,  .14,  .13,
     +  .12,  .11,  .10,  .09,  .09,  .08,  .07,  .05,  .04,  .03,
     +  .02,  .00, -.01, -.02, -.04, -.05, -.07, -.10, -.12, -.14,
     + -.17, -.20, -.25, -.31, -.36, -.42, -.50, -.63, -.79, -.97/
      DATA SBIAS4/                               -.06, -.03,  .00,  .01,
     +  .03,  .04,  .05,  .06,  .06,  .07,  .09,  .10,  .11,  .13,
     +  .14,  .15,  .15,  .16,  .16,  .16,  .16,  .16,  .16,  .15,
     +  .14,  .14,  .14,  .13,  .13,  .13,  .14,  .13,  .13,  .13,
     +  .12,  .11,  .10,  .10,  .09,  .08,  .08,  .06,  .06,  .05,
     +  .04,  .03,  .02,  .01,  .00, -.02, -.05, -.07, -.09, -.12,
     + -.16, -.18, -.22, -.27, -.33, -.42, -.50, -.60, -.74, -.90/
      DATA SBIAS5/                               -.23, -.20, -.17, -.15,
     + -.14, -.13, -.12, -.10, -.09, -.10, -.09, -.07, -.05, -.04,
     + -.03,  .00,  .01,  .03,  .04,  .04,  .04,  .05,  .05,  .05,
     +  .04,  .04,  .04,  .05,  .05,  .07,  .08,  .09,  .10,  .11,
     +  .10,  .10,  .09,  .10,  .10,  .09,  .09,  .09,  .09,  .08,
     +  .07,  .07,  .07,  .09,  .09,  .08,  .08,  .08,  .08,  .08,
     +  .06,  .04,  .00, -.03, -.07, -.11, -.16, -.22, -.29, -.41/
```

```
C
C     F10 XSCAN BIASES
C
      DATA TBIAS1/                          -.01,  .03,  .05,  .07,
     +  .09,  .10,  .11,  .12,  .12,  .14,  .14,  .14,  .14,  .15,
     +  .16,  .15,  .15,  .14,  .14,  .14,  .14,  .14,  .13,  .13,
     +  .14,  .14,  .14,  .14,  .14,  .14,  .13,  .13,  .12,  .12,
     +  .11,  .10,  .08,  .07,  .07,  .07,  .06,  .05,  .04,  .02,
     +  .00, -.01, -.03, -.04, -.05, -.06, -.08, -.09, -.11, -.14,
     + -.15, -.18, -.21, -.26, -.31, -.37, -.46, -.57, -.69, -.82/
      DATA TBIAS2/                           .00,  .02,  .02,  .02,
     +  .01,  .01,  .01,  .01,  .01,  .00,  .00,  .00,  .00,  .00,
     +  .00,  .00,  .00, -.01, -.02, -.02, -.02, -.02, -.02, -.01,
     +  .00,  .01,  .02,  .03,  .04,  .05,  .06,  .06,  .06,  .06,
     +  .05,  .05,  .05,  .06,  .07,  .08,  .08,  .08,  .07,  .08,
     +  .08,  .08,  .08,  .08,  .08,  .07,  .06,  .04,  .03,  .02,
     +  .01, -.01, -.04, -.07, -.11, -.15, -.21, -.27, -.36, -.45/
      DATA TBIAS3/                          -.04,  .01,  .04,  .06,
     +  .08,  .09,  .11,  .11,  .12,  .13,  .14,  .15,  .15,  .15,
     +  .15,  .15,  .15,  .14,  .13,  .14,  .14,  .14,  .14,  .14,
     +  .14,  .15,  .15,  .16,  .16,  .18,  .18,  .16,  .15,  .14,
     +  .14,  .14,  .14,  .13,  .11,  .10,  .09,  .08,  .07,  .05,
     +  .03,  .01, -.01, -.03, -.05, -.07, -.09, -.12, -.14, -.15,
     + -.17, -.20, -.24, -.28, -.34, -.42, -.52, -.64, -.77, -.95/
      DATA TBIAS4/                          -.09, -.06, -.03, -.02,
     + -.01,  .01,  .02,  .03,  .05,  .06,  .07,  .08,  .09,  .10,
     +  .11,  .12,  .12,  .12,  .13,  .13,  .13,  .14,  .14,  .14,
     +  .15,  .16,  .16,  .16,  .16,  .17,  .16,  .16,  .15,  .15,
     +  .14,  .13,  .12,  .12,  .12,  .11,  .11,  .09,  .08,  .07,
     +  .06,  .05,  .04,  .03,  .01, -.01, -.04, -.05, -.07, -.09,
     + -.12, -.16, -.21, -.25, -.31, -.38, -.46, -.56, -.69, -.84/
      DATA TBIAS5/                          -.20, -.18, -.16, -.15,
     + -.15, -.15, -.14, -.13, -.11, -.11, -.10, -.09, -.08, -.07,
     + -.07, -.05, -.04, -.05, -.05, -.04, -.04, -.03, -.02, -.01,
     +  .01,  .02,  .04,  .05,  .06,  .08,  .09,  .09,  .10,  .10,
     +  .09,  .09,  .10,  .11,  .12,  .12,  .13,  .13,  .13,  .13,
     +  .12,  .13,  .12,  .13,  .13,  .13,  .12,  .12,  .12,  .11,
     +  .10,  .08,  .05,  .01, -.02, -.05, -.11, -.18, -.27, -.39/
```

```
C
C     BEGIN EXECUTION
C
      IF(ISTART.EQ.1) THEN
      ISTART=0
      DO 10 IFREQ=1,4
      IF(IFREQ.EQ.2) GO TO 10
      XFAC=(1.-CHI(1,IFREQ)*CHI(2,IFREQ))*(1.-DELTA(IFREQ))
      AVV(IFREQ)=(1.+CHI(1,IFREQ))/XFAC
      AHV(IFREQ)=-CHI(1,IFREQ)*(1.+CHI(2,IFREQ))/XFAC
      AOV(IFREQ)=(1.-AVV(IFREQ)-AHV(IFREQ))*2.7
      AHH(IFREQ)=(1.+CHI(2,IFREQ))/XFAC
      AVH(IFREQ)=-CHI(2,IFREQ)*(1.+CHI(1,IFREQ))/XFAC
      AOH(IFREQ)=(1.-AHH(IFREQ)-AVH(IFREQ))*2.7
   10 CONTINUE
C
      DO 20 ICEL=1,64
      SBIAS1(ICEL)=1.-SBIAS1(ICEL)/AVGTA(1)
      SBIAS2(ICEL)=1.-SBIAS2(ICEL)/AVGTA(2)
      SBIAS3(ICEL)=1.-SBIAS3(ICEL)/AVGTA(3)
      SBIAS4(ICEL)=1.-SBIAS4(ICEL)/AVGTA(4)
      SBIAS5(ICEL)=1.-SBIAS5(ICEL)/AVGTA(5)
C
      TBIAS1(ICEL)=(1.-TBIAS1(ICEL)/AVGTA(1))*(1.-HCOEF(1))
      TBIAS2(ICEL)=(1.-TBIAS2(ICEL)/AVGTA(2))*(1.-HCOEF(2))
      TBIAS3(ICEL)=(1.-TBIAS3(ICEL)/AVGTA(3))*(1.-HCOEF(3))
      TBIAS4(ICEL)=(1.-TBIAS4(ICEL)/AVGTA(4))*(1.-HCOEF(4))
      TBIAS5(ICEL)=(1.-TBIAS5(ICEL)/AVGTA(5))*(1.-HCOEF(5))
   20 CONTINUE
      ENDIF
C
```

```
      DO 100 ICEL=1,64
C
      IBAD1=0
      IBAD2=0
      IBAD3=0
      IF(TALO(1,ICEL).LT.55 .OR. TALO(1,ICEL).GT.320) IBAD1=1
      IF(TALO(2,ICEL).LT.55 .OR. TALO(2,ICEL).GT.320) IBAD1=1
      IF(TALO(3,ICEL).LT.55 .OR. TALO(3,ICEL).GT.320) IBAD2=1
      IF(TALO(4,ICEL).LT.55 .OR. TALO(4,ICEL).GT.320) IBAD3=1
      IF(TALO(5,ICEL).LT.55 .OR. TALO(5,ICEL).GT.320) IBAD3=1
C
      IF(ISAT.EQ.8) THEN
      TA19V=TALO(1,ICEL)*SBIAS1(ICEL)
      TA19H=TALO(2,ICEL)*SBIAS2(ICEL)
      TA22V=TALO(3,ICEL)*SBIAS3(ICEL)
      TA37V=TALO(4,ICEL)*SBIAS4(ICEL)
      TA37H=TALO(5,ICEL)*SBIAS5(ICEL)
      ENDIF
C
      IF(ISAT.EQ.10) THEN
      TA19V=TALO(1,ICEL)*TBIAS1(ICEL)-GCOEF(1)
      TA19H=TALO(2,ICEL)*TBIAS2(ICEL)-GCOEF(2)
      TA22V=TALO(3,ICEL)*TBIAS3(ICEL)-GCOEF(3)
      TA37V=TALO(4,ICEL)*TBIAS4(ICEL)-GCOEF(4)
      TA37H=TALO(5,ICEL)*TBIAS5(ICEL)-GCOEF(5)
      ENDIF
C
      IF(ISAT.NE.8 .AND. ISAT.NE.10) THEN
      TA19V=TALO(1,ICEL)*SBIAS1(ICEL)
      TA19H=TALO(2,ICEL)*SBIAS2(ICEL)
      TA22V=TALO(3,ICEL)*SBIAS3(ICEL)
      TA37V=TALO(4,ICEL)*SBIAS4(ICEL)
      TA37H=TALO(5,ICEL)*SBIAS5(ICEL)
      ENDIF
C
      IF(IBAD1.EQ.0) THEN
      TALO(1,ICEL)=TA19V
      TALO(2,ICEL)=TA19H
      TBLO(1,ICEL)=AVV(1)*TA19V+AHV(1)*TA19H+AOV(1)
      TBLO(2,ICEL)=AHH(1)*TA19H+AVH(1)*TA19V+AOH(1)
      ELSE
      TBLO(1,ICEL)=TALO(1,ICEL)
      TBLO(2,ICEL)=TALO(2,ICEL)
      ENDIF
```

```
C
      IF(IBAD2.EQ.0) THEN
      TALO(3,ICEL)=TA22V
      TBLO(3,ICEL)=1.01993*TA22V+1.994
      ELSE
      TBLO(3,ICEL)=TALO(3,ICEL)
      ENDIF
C
      IF(IBAD3.EQ.0) THEN
      TALO(4,ICEL)=TA37V
      TALO(5,ICEL)=TA37H
      TBLO(4,ICEL)=AVV(3)*TA37V+AHV(3)*TA37H+AOV(3)
      TBLO(5,ICEL)=AHH(3)*TA37H+AVH(3)*TA37V+AOH(3)
      ELSE
      TBLO(4,ICEL)=TALO(4,ICEL)
      TBLO(5,ICEL)=TALO(5,ICEL)
      ENDIF
C
  100 CONTINUE
C
      IF(I85GHZ.EQ.0) RETURN
C
      DO 200 ICEL=1,128
      IF(ATAHI(1,ICEL).GE.55 .AND. ATAHI(1,ICEL).LE.320 .AND.
     1  ATAHI(2,ICEL).GE.55 .AND. ATAHI(2,ICEL).LE.320) THEN
      ATBHI(1,ICEL)=AVV(4)*ATAHI(1,ICEL)+AHV(4)*ATAHI(2,ICEL)+AOV(4)
      ATBHI(2,ICEL)=AHH(4)*ATAHI(2,ICEL)+AVH(4)*ATAHI(1,ICEL)+AOH(4)
      ELSE
      ATBHI(1,ICEL)=ATAHI(1,ICEL)
      ATBHI(2,ICEL)=ATAHI(2,ICEL)
      ENDIF
C
      IF(BTAHI(1,ICEL).GE.55 .AND. BTAHI(1,ICEL).LE.320 .AND.
     1  BTAHI(2,ICEL).GE.55 .AND. BTAHI(2,ICEL).LE.320) THEN
      BTBHI(1,ICEL)=AVV(4)*BTAHI(1,ICEL)+AHV(4)*BTAHI(2,ICEL)+AOV(4)
      BTBHI(2,ICEL)=AHH(4)*BTAHI(2,ICEL)+AVH(4)*BTAHI(1,ICEL)+AOH(4)
      ELSE
      BTBHI(1,ICEL)=BTAHI(1,ICEL)
      BTBHI(2,ICEL)=BTAHI(2,ICEL)
      ENDIF
  200 CONTINUE
      RETURN
      END
```